Alma Mater Studiorum · Università di Bologna

FACOLTÀ DI SCIENZE MATEMATICHE, FISICHE E NATURALI
Corso di Laurea Magistrale in Informatica

Una semantica distribuita per il Multi-CCS utilizzando reti di Petri

Tesi di Laurea in Modelli e Sistemi Concorrenti

Relatore:
Chiar.mo Prof.
ROBERTO GORRIERI

Presentata da:
MASSIMO MORARA

Sessione II
Anno Accademico 2010-2011

dedico questa mia tesi all'indimenticabile professor Antonio Pignedoli, ringraziandolo per aver concretamente mostrato, a generazioni di studenti e a me tra questi, che la curiosità di un uomo di cultura non ha confini

Indice

Elenco delle figure

Elenco delle tabelle

Introduzione

Le reti di Petri – o reti P/T (Posti/Transizioni) – sono un gruppo di modelli di calcolo, inventati da Carl Adam Petri nel 1962, usati per descrivere l'evoluzione e l'interazione di processi che operano parallelamente.

In prima approssimazione, possono essere viste come grafi orientati in cui i nodi sono rigidamente divisi in due gruppi – i "posti" e le "transizioni" – e in cui gli archi collegano un elemento di un gruppo a un elemento dell'altro ma mai due elementi del medesimo gruppo. I posti possono essere visti come gli stati dei processi e sono anche visti come contenitori di gettoni, o token, ognuno dei quali rappresenta un processo in quel determinato stato. Le transizioni rappresentano le evoluzioni dei processi dove gli archi dai posti alle transizioni rappresentano i processi (in un determinato stato) necessari ad attivare la (e consumati dalla) transizione e, al contrario, gli archi dalle transizioni ai posti rappresentano i processi (in un determinato stato) generati dalla transizione. Agli archi sono associati dei numeri interi positivi che indicano il numero di token necessari per attivare (e consumati), in un senso, o prodotti, nell'altro senso.

È evidente che una transizione in cui il numero di token (sommando tutti gli archi in entrata) necessari ad attivarla è maggiore di uno, permette di rappresentare una trasformazione che richiede a più processi di interagire; le reti P/T sono quindi in grado di descrivere interazioni tra un numero, anche elevato, di processi.

Delle reti di Petri esistono molte varianti e, in seguito, tratterò le reti P/T non limitate (*unbounded*), ovvero senza limiti al numero di token nei singoli posti. Per una trattazione formale e più approfondita sulle reti di Petri, rimando ai testi di Peterson ([10]) e di Reisig ([11]).

Le algebre di processo (o *Process Calculi*) hanno ugualmente lo scopo di descrivere i sistemi distribuiti ma utilizzando un diverso approccio, basato sull'uso di termini, operatori e leggi algebriche per descrivere l'evoluzione e l'interazione tra i processi. Basate su un formalismo tendenzialmente semplice, grazie alla capacità di combinare i singoli termini, sono in grado di rappresentare sistemi anche notevolmente complessi e di studiarne con maggiore facilità varie proprietà.

La prima, e probabilmente più famosa, algebra di processo è il CCS (*Calculus of Communicating Systems*). Introdotto da Robin Milner nel 1980, permette di descrivere l'interazione di esattamente due processi paralleli.

La semantica (interpretazione) di riferimento per il CCS è data dagli LTS (*Labelled Transition Sysntem*), ovvero da grafi orientati nei quali i nodi rappresentano stati (ma non di singoli processi: uno stato visto come combinazione degli stati di tutti i processi attivi) e gli archi (etichettati) rappresentano le transizioni.

Gli LTS sono formalmente più semplici delle reti P/T ma meno adeguati a rappresentare sistemi concorrenti. Se, per fare un semplice esempio, dobbiamo descrivere 10 processi identici che, indipendentemente l'uno dall'altro, possono essere in due stati diversi e passare liberamente dall'uno all'altro, mentre con le reti P/T avremo bisogno di due posti (uno per ogni stato), due transizioni e quattro archi, con gli LTS avremo bisogno di non meno di 11 stati – se l'identità dei singoli processi in un determinato stato non è significativa, 1.024 (ovvero 2^{10}) altrimenti – e di non meno di 20 archi.

In [8] Milner dimostrò che tramite un sottoinsieme del CCS – costituito da processo nullo, azione prefissa, somma e costanti – è possibile esprimere ogni LTS finito. L'idea di fondo è semplice: si associa una costante a ogni stato dell'LTS e si definisce la costante come somma di termini $a_i.C_i$, dove i nomi a_i sono le etichette delle transizioni sostenute dallo stato e le costanti C_i corrispondono agli stati raggiungibili dalle singole transizioni.

Grazie a questo risultato sappiamo che, salvo isomorfismo, è possibile esprimere ogni LTS finito tramite ogni algebra di processo che contiene il sottoinsieme del CCS di cui sopra e quindi, ovviamente, lo stesso CCS.

Dal momento che il CCS permette di rappresentare l'interazione di esattamente due processi, è evidente che non può essere in grado – con lo stesso meccanismo – di esprimere ogni rete P/T dal momento che in queste le interazioni possono avvenire tra un numero di processi anche molto elevato.

Una semantica sulle reti P/T, piuttosto che sugli LTS, sarebbe comunque preferibile poiché, con una rappresentazione che non dipende dal numero dei processi in un determinato stato – che diventa un parametro variabile, e non fisso, della stessa – permetterebbe una rappresentazione più compatta dei medesimi sistemi; in alcuni casi, permetterebbe rappresentazioni finite laddove una semantica sugli LTS genererebbe rappresentazioni infinite. La maggiore compattezza, inoltre, permetterebbe di determinare con maggiore facilità proprietà come *reachability*, *liveness* e *deadlock* e permetterebbe di focalizzare maggiormente l'attenzione sugli eventi che modificano il sistema ovvero sulle relative cause e conseguenze.

In passato sono state costruite delle semantiche su reti di Petri a partire da algebre di processo, ma con significative limitazioni; senza pretesa di completezza, cito in particolare le seguenti:

- Degano, De Nicola e Montanari, in [2], e Olderog, in [9], hanno sviluppato semantiche, basate sul CCS, che si applicano solo a reti P/T sicure, ovvero reti P/T che possono contenere al massimo un token in ogni posto; generano reti P/T finite solamente per processi CCS regolari – ovvero processi nei quali restrizione e composizione parallela non possono essere presenti ricorsivamente – ovvero gli stessi processi che generano LTS finiti;

- Golzt, in [3] e in [4], ha sviluppato una semantica basata su un sottoinsieme del CCS – e quindi non in grado di modellare interazioni tra più di due processi – che non comprende restrizione;

- Busi e Gorrieri, in [1], hanno sviluppato una semantica basata sul π-calcolo – ancora col limite dell'iterazione tra due processi – che però è basata su reti P/T non limitate con archi inibitori, ovvero su una variante particolare delle reti di Petri.

Gorrieri e Versari, in [6] e con maggiori dettagli in [7], hanno quindi introdotto il Multi-CCS: una nuova algebra di processo che (a quanto mi risulta) è la prima a supportare una semantica sulle reti P/T non limitate e a contenente l'intero CCS come sottoinsieme e quindi in grado, tra l'altro, di esprimere ogni LTS finito.

Definita quindi una semantica che associa, tramite un'opportuna funzione di decomposizione e opportune regole, processi del Multi-CCS in marcature e transizioni di una specifica rete P/T non limitata e dopo aver mostrato che tali marcature sono bisimili ai corrispondenti stati nella semantica LTS, mostrano che i processi espressi tramite un sottoinsieme di questo linguaggio generano sistemi P/T non limitati e finiti e, viceversa, che ogni sistema P/T non limitato e finito può essere espresso – al netto di un opportuno isomorfismo – tramite un processo di questo sottolinguaggio.

In questa tesi ho ulteriormente approfondito alcuni aspetti del Multi-CCS, della sua semantica sugli LTS e, sopratutto, della sua semantica sulle reti P/T non limitate.

Più in dettaglio:

- in [7] la semantica P/T non è espressa nei minimi dettagli; in particolare si presuppone l'esistenza di un meccanismo (una sorta di oracolo) che, di volta in volta, sceglie gli opportuni nomi aggiuntivi da utilizzare nelle sostituzioni innescate dalle restrizioni; ho quindi definito un opportuno insieme di nomi ristretti sostitutivi (previsti in [7], ma non definiti nel dettaglio) e ho ridefinito la funzione di decomposizione – aggiungendovi anche un paio di parametri – per selezionare in maniera deterministica, quando necessario, i nuovi nomi ristretti;

- seguendo un suggerimento – presente in [7], a indicazione di un possibile sviluppo – ho introdotto una coppia di nomi ristretti speciali (che ho definito "nomi ristretti banali") da utilizzare nei casi in cui i nomi sostituendi non sono in grado (tenendo conto della restrizione) di agire; ho quindi modificato la funzione di decomposizione per utilizzarli quando opportuno, ottenendo una variante della semantica di partenza che, in alcune circostanze, permette di esprimere con sistemi P/T finiti processi Multi-CCS che altrimenti avrebbero generato (con la semantica originaria) sistemi P/T infiniti;

- seguendo l'esempio di Gorrieri, che in [5] esplora altri aspetti del Multi-CCS, ho modificato (con ulteriori modifiche, finalizzate a snellire la casistica presente in [5]) le regole di sincronizzazione – sia nella semantica LTS che in quella P/T – per dare alle stesse una migliore simmetria e per eliminare dalle etichette di transizione elementi superflui;

- ho rilevato che le regole di congruenza strutturale riportate in [7] erano insufficienti; ho infatti individuato due casi di processi Multi-CCS che determinavano la medesima marcatura nella semantica P/T ma stati diversi, e non bisimili tra loro, nella semantica LTS; ho corretto il problema aggiungendo un paio di nuove regole di congruenza nella semantica LTS; ho anche rimosso una delle regole di congruenza e una regola di transizione originarie che, grazie alle regole di congruenza aggiunte, sono risultate superflue;

- ho individuato un errore nell'originaria semantica P/T che, per particolari processi, portava a un'indesiderata unificazione di posti nella marcatura ottenuta e alla conseguente generazione di transizioni senza riscontro nel corrispondente stato della semantica LTS; ho corretto il problema, a costo di una rilevante complicazione formale, per mantenere distinti tali posti;

- ho modificato il meccanismo di generazione di un processo Multi-CCS, a partire da un sistema P/T finito, per tener conto delle modifiche che avevo apportato nella semantica (in particolare, con l'introduzione dei nomi ristretti banali);

- ho corretto una piccola svista, nel meccanismo di cui al punto precedente, che portava alla generazione di un processo Multi-CCS con un eccesso di transizioni rispetto al sistema P/T di partenza.

Oltre a quanto elencato, ho cercato di dare una puntuale e rigorosa dimostrazione dei risultati relativi alla correttezza delle semantica definita sulle reti P/T, ovvero del fatto che la marcatura ottenuta da un processo Multi-CCS è bisimile al corrispondente stato nella semantica LTS. Tale dimostrazione – che corrisponde al capitolo 4, ovvero alla parte più corposa della tesi, circa metà della stessa – è risultata alquanto impegnativa ed elaborata per diversi motivi:

- l'introduzione di un preciso meccanismo di generazione dei nomi ristretti ha comportato la necessità di dimostrare molti risultati alla luce di questo, con complicazioni che a volte non erano presenti in [7] in quanto ci si poteva affidare alla maggiore flessibilità di una modalità non definita nei dettagli;

- l'introduzione della variante relativa ai nomi ristretti banali ha semplificato le reti P/T generate ma ha significativamente appesantito la casistica nelle dimostrazioni che coinvolgevano la restrizione; ho anche introdotto delle definizioni e delle proposizioni, probabilmente privi di interesse autonomo, che non hanno altro scopo se non quello di semplificare tali casistiche;

- una delle assunzioni di partenza di [7], secondo la quale la bisimulazione è una relazione di equivalenza utilizzabile per dimostrare molti dei risultati intermedi, è risultata fallace – come abbiamo indipendentemente rilevato Gorrieri ed io – e ho quindi individuato (dopo vari tentativi infruttuosi) una nuova relazione di equivalenza – che ha finito per essere ritagliata sulla semantica P/T adottata e sulle necessità relative alla dimostrazione, e quindi probabilmente priva di interesse autonomo – che è "più forte" della (implica la) bisimulazione;

- le particolari caratteristiche della sincronizzazioni Multi-CCS, particolarmente nella semantica P/T, hanno complicato alcune dimostrazioni per i casi di composizione parallela e mi hanno portato a introdurre definizioni e risultati relativi a transizioni contemporaneamente supportate; riportando tali definizioni sulla semantica LTS, mi sono trovato a confrontarmi con la "step semantic" che nel frattempo Gorrieri aveva sviluppato in [5]; dalle differenze dei due approcci sono scaturite le osservazioni che hanno portato a individuare e introdurre le regole di congruenza strutturale di cui accennavo in precedenza;

- ultimo aspetto, ma (temo) non per importanza: la mia inesperienza nel campo può avermi portato a complicare inutilmente certi aspetti o a non individuare potenziali soluzioni più semplici ed efficaci.

Capitolo 1

Reti di Petri P/T non limitate

Le Reti di Petri sono un gruppo di modelli di calcolo ampiamente noto e diffusamente trattato. Tra le tanti possibili, nel presente capitolo intendo dare (sull'esempio di [6] e di [7]) una rapida descrizione formale - tendenzialmente limitata alle esigenze dei capitoli seguenti - di una specifica classe, tra le più studiate, di questi strumenti: le reti P/T, o reti Posti/Transizioni. Mi concentrerò poi, in particolare, sulle reti P/T senza limiti al numero di token nei singoli posti e con transizioni che possono consumare e/o produrre anche più di un token nei posti che coinvolgono. Per esigenze legate alle dimostrazioni del capitolo 4, introduco anche la definizione di multiset di transizioni contemporaneamente supportate e alcune osservazioni a questi associate.

Tenendo conto che intendo metterle a confronto con i *Labelled Transition System* (anche dei quali darò quindi una essenzialissima descrizione formale), ancora sull'esempio di [6] e di [7], darò una definizione formale non-standard delle reti P/T ma che meglio si adatta alle mie esigenze.

1.1 Labelled Transition System e bisimulazione

Definizioni 1.1.1. *Un* Labelled Transition System *– o anche* LTS*, in breve – è una tripla $TS = (St, A, \longrightarrow)$ dove*

- *St è un insieme di* stati*,*
- *A è un insieme di* etichette *e*
- *$\longrightarrow \subseteq St \times A \times St$ è la* relazione di transizione*.*

Con $s \xrightarrow{a} s'$ intendiamo $(s, a, s') \in \longrightarrow$; data una transizione *$s \xrightarrow{a} s'$, diciamo che s è l'*origine*, s' la* destinazione *e a l'*etichetta della transizione*.*

Diciamo che un LTS è finito *quando sono finiti sia St che $\longrightarrow$.*

Possiamo osservare che, in base alla definizione precedente, un LTS è "finito" anche quando l'insieme A delle sue etichette è infinito. È però ovvio che, dato un LTS finito $TS = \{St, A, \longrightarrow\}$ in cui l'insieme A è infinito, possiamo sempre costruire un LTS sostanzialmente identico (e sicuramente isomorfo, secondo una definizione che daremo in seguito) filtrando da A le etichette che non vengono usate nelle transizioni, ovvero $TS_{fin} = \{St, A_{fin}, \longrightarrow\}$ dove

$$A_{fin} = \{a \mid a \in A \wedge \exists s, s' \in T.(s, a, s') \in \longrightarrow\}$$

Definizioni 1.1.2. *Un* LTS radicato *– o* LTS con radice *– è una quadrupla $TS(s_0) = (S, A, \longrightarrow, s_0)$ dove $(St, A, \longrightarrow)$ costituisce un LTS e $s_0 \in S$; diciamo che s_0 è lo* stato iniziale*.*

Un cammino*, o* percorso*, dallo stato s_1 allo stato s_{n+1}, è una sequenza di transizioni $s_1 \xrightarrow{a_1} s_2 \xrightarrow{a_2} s_3 \cdots s_n \xrightarrow{a_n} s_{n+1}$; diciamo che uno stato s' è raggiungibile da s quando esiste un percorso che posta da s ad s'.*

Definizioni 1.1.3. *Diciamo che una funzione biunivoca $f : St_1 \to St_2$ è un* isomorfismo *tra TS_1 e TS_2, dove $TS_1 = (St_1, A_1, \longrightarrow_1)$ e $TS_2 = (St_2, A_2, \longrightarrow_2)$ sono due LTS, quando*

$$s \xrightarrow{a}_1 s' \iff f(s) \xrightarrow{a}_2 f(s')$$

Se esiste un isomorfismo tra TS_1 e TS_2, diciamo che i due LTS sono isomorfi e lo rappresentiamo con $TS_1 \cong TS_2$.

Diciamo che due LTS radicati, $TS_1(s_1) = (St_1, A_1, \longrightarrow_1, s_1)$ e $TS_2(s_2) = (St_2, A_2, \longrightarrow_2, s_2)$, sono isomorfi *se esiste un isomorfismo $f : St_1 \to St_2$ tra* *i due LTS non radicati tale che* $f(s_1) = s_2$*; lo rappresentiamo con* $TS_1(s_1) \cong TS_2(s_2)$

Si può facilmente dimostrare che l'isomorfismo è riflessivo, simmetrico e transitivo e che quindi stabilisce delle classi di equivalenza, tra gli LTS e tra gli LTS radicati. Queste classi di equivalenza sono però, per così dire, molto piccole; o molto discriminanti.

Vediamo ora la definizione di un'altra relazione di equivalenza – non dimostreremo che è tale, ma è semplice farlo – che è comunque significativa e che suddivide gli LTS in classi più ampie.

Definizioni 1.1.4. *Definiamo* bisimulazione forte *– o anche semplicemente* bisimulazione *– tra due LTS,* $TS_1 = (St_1, A_1, \longrightarrow_1)$ *e* $TS_2 = (St_2, A_2, \longrightarrow_2)$*, una relazione* $R \subseteq St_1 \times St_2$ *tale che, se* $(s_1, s_2) \in R$*, allora per ogni* $a \in A_1 \cup A_2$ *deve valere*

- $\forall s_1' \in St_1.(s_1 \xrightarrow{a}_1 s_1' \implies \exists s_2' \in St_2.(s_2 \xrightarrow{a}_2 s_2' \wedge (s_1', s_2') \in R))$*;*
- $\forall s_2' \in St_2.(s_2 \xrightarrow{a}_2 s_2' \implies \exists s_1' \in St_1.(s_1 \xrightarrow{a}_1 s_1' \wedge (s_1', s_2') \in R))$*.*

Diciamo che due stati $s_1 \in St_1$ *e* $s_2 \in St_2$ *sono* fortemente bisimili *– o anche, semplicemente,* bisimili *– quando esiste una bisimulazione* $R \subseteq St_1 \times St_2$ *tale che* $(s_1, s_2) \in R$*.*

Quando $TS_1 = TS_2$*, parliamo di* bisimulazione su TS_1.

1.2 Reti di Petri Posti/Transizioni

Definizioni 1.2.1. *Dato un insieme* S*, un* multiset finito *– o, più semplicemente,* multiset *– su* S *è una funzione* $m : S \to \mathbb{N}$ *(dove* $\mathbb{N}$ *rappresenta l'insieme dei numeri naturali) con dominio finito, dove per* dominio del multiset m*, che indichiamo con* $dom(m)$*, intendiamo il sottoinsieme di* S *definito come segue*

$$dom(m) = \{s \in S \mid m(s) > 0\}$$

ovvero m è un multiset finito di S quando $dom(m) \in \mathcal{P}_{fin}(S)$ (dove $\mathcal{P}_{fin}(S)$ rappresenta l'insieme dei sottoinsiemi finiti di S).

Per molteplicità *di un elemento $s \in S$ nel multiset finito m intendiamo il valore $m(s)$.*

Con $\mathcal{M}_{fin}(S)$ indichiamo l'insieme di tutti i possibili multiset finiti definiti su S; generalmente indichiamo con m e sue variazioni (m', m'', m_1, m_2, ecc.) i suoi generici elementi. Definiamo vuoto *un multiset m quando $dom(m) = \emptyset$; con abuso notazionale lo indicheremo proprio con $\emptyset$.*

Notazioni 1.2.2. *Scriviamo $m_1 \subseteq m_2$ intendendo che $m_1(s) \leqslant m_2(s)$ per ogni $s \in S$. Scriviamo anche $m_1 \subset m_2$ quando $m_1 \subseteq m_2$ ed esiste almeno un $s \in S$ tale che $m_1(s) < m_2(s)$.*

Con gli operatori $\oplus$ e $\setminus$ indichiamo, rispettivamente, l'unione a la differenza di multiset, ovvero

$$(m_1 \oplus m_2)(s) = m_1(s) + m_2(s)$$

$$(m_1 \setminus m_2)(s) = \begin{cases} m_1(s) - m_2(s) & \textit{quando } m_1(s) \geqslant m_2(s) \\ 0 & \textit{altrimenti} \end{cases}$$

Il prodotto scalare tra un intero j e un multiset finito m è definito come

$$(j \cdot m)(s) = j \cdot m(s)$$

Dato un insieme $S = \{s_1, s_2, \cdots\}$ e un multiset finito m su S tale che $dom(m) = \{s_{i_1}, s_{i_2}, \cdots, s_{i_n}\}$, possiamo rappresentare m come $k_1 s_{i_1} \oplus k_2 s_{i_2} \oplus \cdots \oplus k_n s_{i_n}$ quando $m(s_{i_j}) = k_j$, con $j = 1, 2, \cdots, n$.

Date le precedenti definizioni, possiamo osservare le seguenti ovvietà

Osservazione 1.2.1. *Fissato un insieme S, per ogni $m \in \mathcal{M}_{fin}(S)$ e per ogni intero $i \in \mathbb{N}$ abbiamo che*

$$\emptyset \subseteq m \qquad m \oplus \emptyset = m \qquad m \setminus \emptyset = m \qquad \emptyset \setminus m = \emptyset$$

$$i \cdot \emptyset = \emptyset \qquad 0 \cdot m = \emptyset \qquad 1 \cdot m = m$$

Definizioni 1.2.3. *Una* rete Posti/Transizioni etichettata *– o anche* rete P/T etichettata*, o semplicemente* rete P/T *– è una tripla $N = (S, A, T)$ costituita da:*

- *un insieme numerabile S di* posti*,*
- *un insieme numerabile A di* etichette *e*
- *un insieme $T \subseteq (\mathcal{M}_{fin}(S) \setminus \emptyset) \times A \times \mathcal{M}_{fin}(S)$ di* transizioni*.*

Diciamo che un multiset finito su S è una marcatura*; fissata una marcatura m, e dato un posto s, diciamo che il posto contiene $m(s)$* token*.*

Diciamo che una rete P/T è finita *se sono finiti sia S che T.*

Mettendo a confronto la precedente definizione di reti P/T con la precedente di Labelled Transition System (1.1.1, pag. 9), possiamo osservare quanto segue

Osservazione 1.2.2. *Gli LTS possono essere visti come specializzazioni delle reti P/T.*

Infatti, l'unica sostanziale differenza è sulla definizione dell'insieme delle transizioni. Nel caso degli LTS è definito sull'insieme degli stati, che possiamo considerare equivalenti ai posti delle reti P/T; nel caso delle reti P/T sono definite su insiemi di parti del medesimo insieme. Data la relazione $\longrightarrow \subseteq St \times A \times St$ di un LTS, costituita da triple (s, a, s'), possiamo costruire banalmente un insieme T di transizioni $(1 \cdot s, a, 1 \cdot s')$, conformemente alla definizione di rete P/T. Possiamo quindi ricondurre un qualunque LTS a una rete P/T con una banale trasformazione; il contrario, ovviamente, non è generalmente vero ed è quindi evidente che le reti P/T hanno un potere espressivo maggiore degli LTS.

Notazione 1.2.4. *Data una transizione $t = (m, a, m')$ in una rete P/T, usiamo ${}^{\bullet}t$ per indicare m (che chiamiamo* preset della transizione*), usiamo $t^{\bullet}$ per indicare m' (che chiamiamo* postset della transizione*) e usiamo $l(t)$ per indicare l'etichetta a. Possiamo anche rappresentarla come ${}^{\bullet}t \xrightarrow{l(t)} t^{\bullet}$, ovvero $m \xrightarrow{a} m'$.*

Per come ho definito le transizioni, è del tutto lecito che il postset di una transizione sia la marcatura nulla $\emptyset$; ho invece imposto che il preset non possa mai essere vuoto. Questa scelta è motivata da una visione dei token come di processi – o anche di risorse – che possono terminare (o essere consumate) ma che non possono essere attivati (o create) dal nulla. Questo non toglie che è possibile definire una transizione $t = (m, a, m \oplus m')$ che, di fatto, non consuma nulla ma che crea nuovi token; con la definizione che ho dato, ho però imposto che questo avvenga col supporto di un preset non vuoto che abbia – idealmente, se non formalmente – una qualche funzione attiva.

Si osservi che, avendo definito le transizioni T come un insieme di triple, otteniamo che se, date due transizioni t_1 e t_2, abbiamo ${}^\bullet t_1 = {}^\bullet t_2$, $l(t_1) = l(t_2)$ e $t_1^\bullet = t_2^\bullet$, allora abbiamo anche $t_1 = t_2$.

Definizioni 1.2.5. *Data una rete P/T, $N = (S, A, T)$, diciamo che la transizione $t \in T$ è* abilitata*, o anche che è* supportata*, dalla marcatura $m \in \mathcal{M}_{fin}(S)$ quando ${}^\bullet t \subseteq m$. L'esecuzione di t, supportata da m, produce la marcatura $m' = (m \setminus {}^\bullet t) \oplus t^\bullet$.*

Data una rete P/T, $N = (S, A, T)$, diciamo che il multiset di transizioni $k_1t_1 \oplus k_2t_2, \oplus \ldots \oplus k_nt_n \in \mathcal{M}_{fin}(T)$ è contemporaneamente abilitato*, o anche che è* contemporaneamente supportato*, dalla marcatura m quando*

$$\left(\bigoplus_{i=1}^{n} k_i \cdot {}^\bullet t_i \right) \subseteq m$$

L'esecuzione, vista come contemporanea, di tutte le transizioni (con le loro molteplicità) $t_1, t_2, \ldots, t_n$, supportate contemporaneamente da m, produce la marcatura

$$m' = \left(m \setminus \left(\bigoplus_{i=1}^{n} k_i \cdot {}^\bullet t_i \right) \right) \oplus \left(\bigoplus_{i=1}^{n} k_i \cdot t_i^\bullet \right)$$

Notazioni 1.2.6. *Indichiamo che la marcatura m supporta una transizione t producendo la marcatura m', sinteticamente, con $m[t\rangle m'$; quando non siamo interessati alla marcatura prodotta ma solo a evidenziare che la marcatura m supporta la transizione t, useremo $m[t\rangle$.*

Per distinguere i multiset di transizioni dai multiset di posti delle reti di Petri, non indicheremo i primi con la consueta lettera m, e relative variazioni, ma con la lettera u e relative variazioni (u', u'', u_1, ecc.).

Sia quindi $u = k_1t_1 \oplus k_2t_2 \oplus \ldots \oplus k_nt_n$, un multiset di transizioni; useremo $m[[u\rangle\rangle m'$, o anche $m[[k_1t_1 \oplus k_2t_2 \oplus \ldots \oplus k_nt_n\rangle\rangle m'$, per indicare che la marcatura m supporta contemporaneamente il multiset di transizioni u producendo la marcatura m'; quando non saremo interessati alla marcatura prodotta ma solamente a evidenziare che la marcatura m supporta contemporaneamente il multiset di transizioni u, useremo la notazione $m[[u\rangle\rangle$, o anche $m[[k_1t_1 \oplus k_2t_2 \oplus \ldots \oplus k_nt_n\rangle\rangle$.

Alla consueta definizione di "transazione supportata" (o "abilitata") da una marcatura, ho aggiunto la definizione (e una relativa nuova notazione) di "multiset di transizioni contemporaneamente supportate" (o "abilitate") da una marcatura. Questo sarà utile ai fini della dimostrazione della correttezza della semantica, per il Multi-CCS, che presenterò nel capitolo 4, ed è giustificata dal fatto che alcune transizioni possono combinarsi (sincronizzarsi) per dare origine a un'unica transizione la cui etichetta è ottenuta, come vedremo, sincronizzando e mescolando le etichette delle sub-transizioni e il cui effetto sulle marcature è, sostanzialmente, quello appena descritto.

Date le precedenti definizioni, sono sostanzialmente ovvie le seguenti

Osservazioni 1.2.3. *Per ogni marcatura m, vale $m[[\emptyset\rangle\rangle m$.*

Se $u = 1 \cdot t_1$ è un multiset di transizioni costituito da una sola transizione con molteplicità uno (e tutte le altre con molteplicità zero) – ovvero: se u è un singoletto – è vero che

$$m[[t_1\rangle\rangle m' \iff m[t_1\rangle m'$$

Vediamo ora definizioni relative a sistema P/T, sequenze di transizioni e marcature conseguentemente raggiungibili.

Definizioni 1.2.7. *Un* sistema Posti/Transizioni etichettato *– o anche* sistema Posti/Transizioni, *o semplicemente* sistema P/T *– è una quadrupla*

$N(m_0) = (S, A, T, m_0)$ *dove* (S, A, T) *è una rete P/T e* m_0 *è una marcatura non vuota, definita su* S*, che chiamiamo* marcatura iniziale.

Definiamo l'insieme delle marcature raggiungibili *dalla marcatura* m*, che indichiamo con* $[m\rangle$*, come il più piccolo insieme di marcature tale che*

- $m \in [m\rangle$
- $(m_1 \in [m\rangle \wedge \exists t \in T.m_1[t\rangle m_2) \implies m_2 \in [m\rangle$

Diciamo che una marcatura m *è* raggiungibile *se è raggiungibile dalla marcatura iniziale* m_0*.*

Definiamo sequenza di scatto*, con inizio nella marcatura* m*, come segue, in maniera induttiva*

- m *è una sequenza di scatto;*
- *se* $m[t_1\rangle m_1[t_2\rangle \cdots m_{n-2}[t_{n-1}\rangle m_{n-1}$ *è una sequenza di scatto, con* $n \geqslant 1$*, e se* $m_{n-1}[t_n\rangle m_n$*, allora* $m[t_1\rangle m_1[t_2\rangle \cdots m_{n-1}[t_n\rangle m_n$ *è una sequenza di scatto.*

Data una sequenza di scatto $m[t_1\rangle m_1[t_2\rangle \cdots m_{n-1}[t_n\rangle m_n$*, chiamiamo* sequenza di transizione *la sequenza* $t_1\ t_2 \cdots t_n$ *e* traiettoria di scatto *– o, semplicemente,* traiettoria *– la sequenza* $m\ m_1 \cdots m_n$*.*

Osservo esplicitamente che la consueta definizione di "sequenza di scatto" – e relative "sequenza di transizione" e "traiettoria di scatto" – esprimono un concetto significativamente diverso da quello, che ho introdotto in precedenza, di "multiset di transizioni contemporaneamente supportate". Infatti, mentre ogni transizione di un multiset di transizioni contemporaneamente supportate è necessariamente supportata, anche se vista singolarmente, dalla marcatura di riferimento, in una sequenza di scatto le transizioni successive alla prima non sono necessariamente supportate fin dall'inizio dal momento che il postset della prima transizione potrebbe produrre token necessari alla seconda transizione, che potrebbe produrre token necessari alla terza, e via di seguito.

Notazioni 1.2.8. *Quando non siamo interessati alle marcature intermedie della traiettoria, ma solo alla marcatura iniziale e alla marcatura finale, possiamo rappresentare la sequenza di scatto, in maniera semplificata, come* $m[t_1\ t_2 \cdots t_n\rangle m_n$*; quando non siamo neppure interessati alla marcatura finale, ma solo a esplicitare che la marcatura* m *supporta una sequenza di scatto con sequenza di transizione* $t_1\ t_2 \cdots t_n$*, possiamo usare la notazione* $m[t_1\ t_2 \cdots t_n\rangle$*.*

Date le precedenti definizioni, è ovvia la seguente

Osservazione 1.2.4. *Date due marcature* m *e* m' *e dato un multiset di transizioni* $u = k_1t_1 \oplus k_2t_2 \oplus \ldots \oplus k_nt_n$*, è vero che*

$$m[[k_1t_1 \oplus k_2t_2 \oplus \ldots \oplus k_nt_n\rangle\rangle m' \implies m[t_{\pi(1)}\ t_{\pi(2)}\ \ldots\ t_{\pi(l)}\rangle m'$$

dove

$$l = \sum_{i=1}^{n} k_i$$

e dove π *è una qualunque permutazione degli* l *numeri*

$$\underbrace{1,\ldots,1}_{k_1\ volte}, \underbrace{2,\ldots,2}_{k_2\ volte},\ \ldots,\ \underbrace{n,\ldots,n}_{k_n\ volte}$$

ovvero degli indici delle transizioni di u*, contati con la loro molteplicità; in particolare abbiamo*

$$m[[k_1t_1 \oplus k_2t_2 \oplus \ldots \oplus k_nt_n\rangle\rangle m' \implies m[\underbrace{t_1\ldots t_1}_{k_1\ volte}\ \underbrace{t_2\ldots t_2}_{k_2\ volte}\ \ldots\ \underbrace{t_n\ldots t_n}_{k_n\ volte}\rangle m'$$

L'osservazione di cui sopra è conseguenza del fatto che i postset possono solo aggiungere token a una marcatura e non toglierli; l'esecuzione sequenziale di un qualunque sottoinsieme di un multiset di transizioni contemporaneamente supportate non può quindi annullare il supporto alle transizioni rimanenti; è quindi ovvio che le transizioni contemporaneamente supportate possono essere eseguite in qualsiasi ordine.

Dato un sistema P/T $N(m_0)$, non è detto che tutti i posti siano raggiungibili partendo dalla marcatura iniziale m_0 e non è detto che tutte le transizioni siano effettivamente attivabili, sempre partendo dalla marcatura iniziale m_0. Ha quindi senso definire la sottorete raggiungibile come segue

Definizioni 1.2.9. *Dato un sistema P/T $N(m_0) = (S, A, T, m_0)$, la corrispondente* sottorete raggiungibile, *che indichiamo con $Net(m_0)$, è costituita da $Net(m_0) = (S', A', T', m_0)$ ed è definita in base a*

$$S' = \{s \in S \mid \exists m \in [m_0\rangle . m(s) \geqslant 1\}$$
$$T' = \{t \in T \mid \exists m \in [m_0\rangle . m[t\rangle\}$$
$$A' = \{a \in A \mid \exists t \in T' . a = l(t)\}$$

Diciamo che un sistema P/T $N(m_0)$ è ridotto *quando $Net(m_0) = N(m_0)$ ovvero quando coincide con la propria sottorete raggiungibile.*

Vediamo ora alcune classificazioni per i sistemi P/T.

Definizioni 1.2.10. *Dato un sistema P/T $N(m_0) = (S, A, T, m_0)$, lo definiamo*

- finito, *se sia S che T sono finiti;*
- sequenziale, *se m_0 è un* singoletto *– ovvero: se corrisponde a un unico token in un unico posto – e se, per ogni transizione $t \in T$, ${}^\bullet t$ è un singoletto e $t^\bullet$ è o un singoletto o la marcatura nulla $\emptyset$;*
- sicuro, *se per ogni marcatura raggiungibile dalla marcatura iniziale m_0, ogni posto contiene al più un token, ovvero se*

$$\forall m \in [m_0\rangle . \forall s \in S . m(s) \leqslant 1$$

- limitato, *se esiste un numero naturale k tale che ogni posto contiene al più k token in ogni marcatura raggiungibile, ovvero se*

$$\exists k \in \mathbb{N} . \forall m \in [m_0\rangle . \forall s \in S . m(s) \leqslant k$$

- *una* rete BPP *(Basic Parallel Processes), se ogni transizione ha esattamente un posto nel preset, ovvero se per ogni* $t \in T$, ${}^{\bullet}t$ *è un singoletto.*

Date le precedenti definizioni, possiamo concludere quanto segue

Proposizione 1.2.5. *Dato un sistema P/T* $N(m_0)$*, è vero che*

- *se* $N(m_0)$ *è sequenziale, è anche sicuro;*
- *se* $N(m_0)$ *è sequenziale, è anche una rete BPP;*
- *se* $N(m_0)$ *è sicuro, è anche limitato;*
- *se* $N(m_0)$ *è finito e limitato, l'insieme* $[m_0\rangle$ *delle marcature raggiungibili da* m_0 *è finito.*

Dimostrazione. Tutti i punti, tranne l'ultimo, sono banali conseguenze delle ultime definizioni.

Per quanto riguarda l'ultimo punto, osserviamo che dalla finitezza abbiamo che la cardinalità di S è pari a un preciso numero intero c; dalla limitatezza abbiamo che un numero intero k limita il numero di token in ogni posto, per ogni marcatura. Ragionando sulle possibili combinazioni, otteniamo un limite superiore al numero marcature compatibili con questi vincoli con non possono essere più di $(k+1)^c$. □

Vediamo anche delle definizioni di isomorfismo, sia tra reti P/T che tra sistemi P/T, che corrispondono alle analoghe viste, in 1.1.3 (pag. 10), tra LTS e LTS radicati.

Definizioni 1.2.11. *Definiamo* isomorfismo *tra due reti P/T* N_1 *e* N_2*, dove* $N_1 = (S_1, A_1, T_1)$ *e* $N_2 = (S_2, A_2, T_2)$*, una funzione biunivoca* $f : S_1 \to S_2$ *tale che*

$$(m, a, m') \in T_1 \iff (f(m), a, f(m')) \in T_2$$

Se esiste un isomorfismo tra N_1 *e* N_2*, diciamo che le due reti P/T sono isomorfe e lo rappresentiamo con* $N_1 \cong N_2$*.*

Diciamo che due sistemi P/T $N_1(m_1)$ e $N_2(m_2)$ sono isomorfi*, dove $N_1(m_1) = (S_1, A_1, T_1, m_1)$ e $N_2(m_2) = (S_2, A_2, T_2, m_2)$, se esiste un isomorfismo $f : S_1 \to S_2$ tra le due reti P/T tale che $f(m_1) = m_2$; lo rappresentiamo con $N_1(m_1) \cong N_2(m_2)$*

È possibile definire anche la bisimulazione ma lo faremo, nella prossima sezione, dopo aver definito – e tramite – gli interleaving marking graph.

1.3 Interleaving Marking Graph e bisimulazione

Visti anche i ragionamenti della sezione precedente, viene abbastanza spontaneo pensare alle marcature come agli stati di un LTS e, di conseguenza, alle etichette delle transizioni di una rete P/T come alle etichette delle transizioni di tale LTS. Pensando poi alla marcatura iniziale di un sistema P/T come allo stato iniziale dell'LTS, possiamo intuire che un sistema P/T può essere ricondotto a un LTS con radice.

Sfruttando tale conversione, si può quindi pensare di definire la bisimilitudine tra marcature di reti P/T e persino tra interi sistemi P/T.

Vediamo la cosa da un punto di vista formale.

Definizione 1.3.1. *Data una rete P/T $N = (S, A, T)$, il corrispondente* interleaving marking graph *è l'LTS $IMG(N) = (\mathcal{M}_{fin}(S), A, \longrightarrow)$ dove, per quanto riguarda la relazione di transizione $\longrightarrow \subseteq \mathcal{M}_{fin}(S) \times A \times \mathcal{M}_{fin}(S)$, abbiamo*

$$m \xrightarrow{a} m' \iff \exists t \in T.(m[t\rangle m' \wedge l(t) = a)$$

Ho quindi definito gli "interleaving marking graph" associandoli a una rete P/T e non – come abituale – ai sistemi P/T. Questa definizione – più flessibile e sostanzialmente retro-compatibile con quella classica – mi sarà più utile nelle dimostrazioni dei capitoli seguenti.

Possiamo comunque osservare che – escludendo i casi degeneri in cui l'insieme S è pari all'insieme vuoto – l'interleaving marking graph associato a una rete N è un grafo infinito. Sia infatti $s \in S$ in generico posto di N; tutte le marcature $m_k = k \cdot s$, con $k \in \mathbb{N}$, sono contenute in $\mathcal{M}_{fin}(S)$ ed è quindi evidente che l'insieme delle marcature in $IMG(N)$ è infinito. Analogamente, è banale osservare che anche il numero delle transizioni è (salvo casi degeneri) infinito: è sufficiente individuare una marcatura che supporta una transizione e, aggiungendo k token a un qualsiasi posto del suo preset, variando k possiamo ottenere un'infinità di elementi di $\longrightarrow$.

Introduciamo ora una definizione più "classica", relativa ai sistemi P/T.

Definizione 1.3.2. *Dato un sistema P/T $N(m_0) = (S, A, T, m_0)$, il corrispondente* interleaving marking graph radicato *– o* interleaving marking graph con radice *o anche, più semplicemente,* interleaving marking graph, *quando questo non genera ambiguità – è l'LTS radicato $IMGR(N(m_0)) = ([m_0\rangle, A, \longrightarrow, m_0)$ dove, per quanto riguarda la relazione di transizione $\longrightarrow \subseteq [m_0\rangle \times A \times [m_0\rangle$, abbiamo*

$$m \xrightarrow{a} m' \iff \exists t \in T.(m[t\rangle m' \wedge l(t) = a)$$

Si osservi che gli interleaving marking graph radicati non sono, contrariamente a quelli non radicati, necessariamente infiniti. Si osservi, per fare un esempio banale, $IMGR(N(1 \cdot s_1))$, dove $S = \{s_1\}$, $A = \{a\}$ e $\longrightarrow = \{(1 \cdot s_1, a, \mathbf{0})\}$: le uniche marcature raggiungibili sono quella iniziale e la marcatura nulla e l'unico arco è quello tra queste due marcature.

Avendo ricondotto le reti P/T agli LTS e i sistemi P/T agli LTS radicati, possiamo definire la bisimulazione tra le marcature di due reti P/T, o di due sistemi P/T, e tra i sistemi P/T stessi, come riflesso della bisimulazione dei corrispondenti interleaving marking graph.

Definizioni 1.3.3. *Date le reti P/T $N_1 = (S_1, A_1, T_1)$ e $N_2 = (S_2, A_2, T_2)$, diciamo che due marcature $m_1 \in \mathcal{M}_{fin}(S_1)$ e $m_2 \in \mathcal{M}_{fin}(S_2)$ sono* fortemente interleaving bisimili *– o* interleaving bisimili*, o* fortemente bisimili *o anche semplicemente* bisimili *–, e lo indichiamo con $m_1 \sim m_2$, quando esiste una bisimulazione forte $R \subseteq \mathcal{M}_{fin}(S_1) \times \mathcal{M}_{fin}(S_2)$, sugli LTS $IMG(N_1)$ e $IMG(N_2)$, tale che $(m_1, m_2) \in R$.*

Diciamo che in due sistemi P/T $N_1(m_1)$ e $N_2(m_2)$ sono fortemente interleaving bisimili *– o* interleaving bisimili*, o* fortemente bisimili *o anche semplicemente* bisimili *–, e lo indichiamo con $N_1(m_1) \sim N_2(m_2)$, quando $m_1 \sim m_2$.*

La definizione precedente ci dice che due marcature sono bisimili relativamente alle proprie reti P/T. Questo vale anche se le marcature fanno parte di due sistemi P/T; in questo caso, possiamo osservare che vale la seguente

Proposizione 1.3.1. *Dati i sistemi P/T $N_m(m_0) = (S_m, A_m, T_m, m_0)$, con la marcatura $m \in [m_0\rangle$, e $N_n(n_0) = (S_n, A_n, T_n, n_0)$, con la marcatura $n \in [n_0\rangle$, abbiamo che $m \sim n$, nelle rispettive reti P/T, se e solo se esiste una bisimulazione forte $R \subseteq [m_0\rangle \times [n_0\rangle$ tale che $(m, n) \in R$.*

Dimostrazione. L'esistenza di una bisimulazione R come da enunciato implica, in maniera quasi banale, che $m \sim n$. Sia infatti $R \subseteq [m_0\rangle \times [n_0\rangle$, con $(m, n) \in R$, una bisimulazione forte; dal momento che $[m_0\rangle \subseteq \mathcal{M}_{fin}(S_m)$ e che $[n_0\rangle \subseteq \mathcal{M}_{fin}(S_n)$, è evidente che

$$(m, n) \in R \subseteq [m_0\rangle \times [n_0\rangle \subseteq \mathcal{M}_{fin}(S_m) \times \mathcal{M}_{fin}(S_n)$$

La bisimulazione R, immersa negli LTS non radicati derivati dalle reti P/T, giustifica quindi il fatto che $m \sim n$.

Vediamo ora che $m \sim n$ implica l'esistenza di una bisimulazione R come da enunciato. Sia $R' \subseteq \mathcal{M}_{fin}(S_m) \times \mathcal{M}_{fin}(S_n)$, con $(m, n) \in R'$, una generica bisimulazione forte che giustifica il fatto che $m \sim n$. Definiamo

$$R = \{(m', n') \mid (m', n') \in R' \wedge m' \in [m_0\rangle \wedge n' \in [n_0\rangle\}$$

Dalle ipotesi sappiamo che $(m, n) \in R'$, $m \in [m_0\rangle$ e $n \in [n_0\rangle$; abbiamo quindi $(m, n) \in R$. Dalla sua definizione abbiamo banalmente che $R \subseteq [m_0\rangle \times [n_0\rangle$; se riusciamo a dimostrare anche che R è una bisimulazione forte, la tesi è dimostrata.

Sia (m', n') un generico elemento di R; sia a un generico elemento di $A_m \cup A_n$; sia m'' un generico elemento di $\mathcal{M}_{fin}(S_m)$ tale che $m' \xrightarrow{a} m''$, in corrispondenza di una transizione $t_m \in T_m$ tale che $l(t_m) = a$; dal momento che, per costruzione di R, $(m', n') \in R'$ e che R' e' una bisimulazione, abbiamo che esiste un $n'' \in \mathcal{M}_{fin}(S_n)$ tale che $n' \xrightarrow{a} n''$, in corrispondenza di una transizione $t_n \in T_n$ tale che $l(t_n) = a$, e $(m'', n'') \in R'$; osserviamo che $m' \in [m_0\rangle$ e $m'[t_m\rangle m''$ implicano $m'' \in [m_0\rangle$; analogamente abbiamo anche che $n'' \in [n_0\rangle$; possiamo quindi concludere che, per un generico $a \in A_m \cup A_n$ e per una generica marcatura $m'' \in \mathcal{M}_{fin}(S_m)$,

$$m' \xrightarrow{a} m'' \implies \exists n'' \in \mathcal{M}_{fin}(S_n).(n' \xrightarrow{a} n'' \wedge (m'', n'') \in R)$$

Analogamente possiamo provare che, per un generico $a \in A_m \cup A_n$ e per una generica marcatura $n'' \in \mathcal{M}_{fin}(S_n)$,

$$n' \xrightarrow{a} n'' \implies \exists m'' \in \mathcal{M}_{fin}(S_m).(m' \xrightarrow{a} m'' \wedge (m'', n'') \in R)$$

Insieme alla genericità di $a \in A_m \cup A_n$, di $m'' \in \mathcal{M}_{fin}(S_m)$ e di $n'' \in \mathcal{M}_{fin}(S_n)$, le due implicazioni appena viste dimostrano che R è una bisimulazione. $\square$

Sfruttando lo stesso meccanismo, possiamo anche definire la bisimulazione tra una rete P/T e un LTS.

Definizioni 1.3.4. *Data una rete P/T $N = (S, A_R, T)$ e un LTS $TS = (St, A_L, \longrightarrow)$, diciamo che la marcatura $m \in \mathcal{M}_{fin}(S)$ e lo stato $q \in St$ sono* fortemente interleaving bisimili *– o* interleaving bisimili*, o* fortemente bisimili *o anche semplicemente* bisimili *–, e lo indichiamo con $m \sim q$, quando esiste una bisimulazione forte $R \subseteq \mathcal{M}_{fin}(S) \times St$, tra gli LTS $IMG(N)$ e TS, tale che $(m, q) \in R$.*

Diciamo che un sistema P/T $N(m_0)$ e un LTS radicato $TS(q_0)$ sono fortemente interleaving bisimili *– o* interleaving bisimili*, o* fortemente bisimili *o anche semplicemente* bisimili *–, e lo indichiamo con $N(m_0) \sim TS(q_0)$, quando $m_0 \sim q_0$.*

Analogamente al caso della bisimilitudine tra due marcature di due sistemi P/T, abbiamo

Proposizione 1.3.2. *Dati un sistema P/T $N(m_0) = (S, A_S, T, m_0)$, con la marcatura $m \in [m_0\rangle$, e l'LTS radicato $TS(q_0) = (St, A_L, \longrightarrow, q_0)$, con la marcatura $q \in St$, abbiamo che $m \sim q$ se e solo se esiste una bisimulazione forte $R \in [m_0\rangle \times St$ tale che $(m, q) \in R$.*

Dimostrazione. Si può dimostrare in maniera analoga alla dimostrazione della proposizione 1.3.1 (pag. 22). □

Vediamo ora una caratterizzazione della bisimulazione tra marcature in un sistema P/T che, in un certo senso, astrae dai corrispondenti LTS.

Proposizione 1.3.3. *Siano $N_m = (S_m, A_m, T_m)$ e $N_n = (S_n, A_n, T_n)$ due reti P/T; siano $m \in \mathcal{M}_{fin}(S_m)$ e $n \in \mathcal{M}_{fin}(S_n)$; abbiamo che $m \sim n$ se e solo se (1) esiste una relazione $R \subseteq \mathcal{M}_{fin}(S_m) \times \mathcal{M}_{fin}(S_n)$ tale che $(m, n) \in R$ e (2) tale che, per ogni coppia $(m', n') \in R$, sono vere entrambe le implicazioni seguenti*

$$\forall t_m \in T_m.(m'[t_m\rangle m'' \implies \exists t_n \in T_n.(n'[t_n\rangle n'' \wedge l(t_n) = l(t_m) \wedge (m'', n'') \in R)$$
$$\forall t_n \in T_n.(n'[t_n\rangle n'' \implies \exists t_m \in T_m.(m'[t_m\rangle m'' \wedge l(t_m) = l(t_n) \wedge (m'', n'') \in R)$$

Dimostrazione. Vediamo, per prima cosa, che $m \sim n$ implica che valgono le due condizioni dell'enunciato.

Supponiamo che $m \sim n$; sia $R \subseteq \mathcal{M}_{fin}(S_m) \times \mathcal{M}_{fin}(S_n)$, con $(m, n) \in R$, una generica bisimulazione che giustifica $m \sim n$; sia (m', n') un generico elemento di R; supponiamo che, per una generica transizione $t_m \in T_m$, dove $a = l(t_m)$, sia $m'[t_m\rangle m''$; avremo quindi, dalla definizione 1.3.1 (pag. 20) di interleaving marking graph, che in $IMG(N_m)$ abbiamo $m' \xrightarrow{a} m''$; dalla definizione di bisimulazione abbiamo che esiste una marcatura $n'' \in \mathcal{M}_{fin}(S_n)$ tale che $(m'', n'') \in R$ e tale che $n' \xrightarrow{a} n''$, ovvero tale che $n'[t_n\rangle n''$ per una opportuna transizione $t_n \in T_n$ con $l(t_n) = a = l(t_m)$. Dalla genericità della transizione $t_m \in T_m$ abbiamo la validità della prima delle due implicazioni per $(m', n') \in R$; la seconda implicazione si può provare, in maniera del tutto analoga, partendo da una generica transizione $t_n \in T_n$. Dalla genericità di $(m', n') \in R$ concludiamo che R è la relazione cercata.

Vediamo ora che la validità delle due condizioni della proposizione implica che vale $m \sim n$.

Sia $R \subseteq \mathcal{M}_{fin}(S_m) \times \mathcal{M}_{fin}(S_n)$ la relazione tale per cui $(m, n) \in R$ e per cui, per ogni suo elemento, valgono le due implicazioni dell'enunciato. Sia (m', n') un generico elemento di R. Supponiamo che, per un generico $a \in A_m \cup A_n$ e per un generico $m'' \in \mathcal{M}_{fin}(S_m)$, in $IMG(N_m)$ valga $m' \xrightarrow{a} m''$; allora, dalla definizione 1.3.1 di Interleaving Marking Graph, abbiamo che esiste una transizione $t_m \in T_m$ tale che $m'[t_m\rangle m''$ e tale che $l(t_m) = a$; dalle prima delle due implicazioni dell'enunciato abbiamo che esiste una transizione $t_n \in T_n$ tale che $n'[t_n\rangle n''$, tale che $l(t_n) = l(t_m) = a$ e tale che $(m'', n'') \in R$; applicando in senso inverso la definizione 1.3.1 abbiamo che, in $IMG(N_n)$, vale $n' \xrightarrow{a} n''$; abbiamo quindi concluso che, dati un generico $(m', n') \in R$, un generico $m'' \in \mathcal{M}_{fin}(S_m)$ e una generica $a \in A_m \cup A_n$, vale

$$m' \xrightarrow{a} m'' \implies \exists n'' \in \mathcal{M}_{fin}(S_n).(n' \xrightarrow{a} n'' \wedge (m'', n'') \in R)$$

In maniera del tutto analoga – partendo da una generica transazione $n' \xrightarrow{a} n''$ in $IMG(N_n)$ e sfruttando la seconda implicazione dell'enunciato – si può provare che, dati un generico $(m', n') \in R$, un generico $n'' \in \mathcal{M}_{fin}(S_n)$ e una generica $a \in A_m \cup A_n$, vale

$$n' \xrightarrow{a} n'' \implies \exists m'' \in \mathcal{M}_{fin}(S_m).(m' \xrightarrow{a} m'' \wedge (m'', n'') \in R)$$

Abbiamo quindi dimostrato, ai sensi della definizione 1.1.4 (pag. 11), che R è una bisimulazione e quindi – ricordando che, per ipotesi, $(m, n) \in R$ – che $m \sim n$. □

Abbiamo quindi sostanzialmente trovato una definizione alternativa – ma equivalente – di bisimilitudine tra marcature di reti P/T che non ha bisogno di ricondursi ai corrispondenti interleaving marking graph.

Analogamente, per le marcature bisimili di sistemi P/T, abbiamo il seguente

Corollario 1.3.4. *Dati i sistemi P/T* $N_m(m_0) = (S_m, A_m, T_m, m_0)$, *con* $m \in [m_0\rangle$, *e* $N_n(n_0) = (S_n, A_n, T_n, n_0)$, *con* $n \in [n_0\rangle$, *abbiamo che* $m \sim n$ *se e solo se (1) esiste una relazione* $R \subseteq [m_0\rangle \times [n_0\rangle$ *tale che* $(m, n) \in R$ *e (2) tale che, per ogni coppia* $(m', n') \in R$, *sono vere entrambe le implicazioni seguenti*

$$\begin{aligned} &\forall t_m \in T_m.(m'[t_m\rangle m'' \\ &\qquad \Longrightarrow \exists t_n \in T_n.(n'[t_n\rangle n'' \wedge l(t_n) = l(t_m) \wedge (m'', n'') \in R)) \\ &\forall t_n \in T_n.(n'[t_n\rangle n'' \\ &\qquad \Longrightarrow \exists t_m \in T_m.(m'[t_m\rangle m'' \wedge l(t_m) = l(t_n) \wedge (m'', n'') \in R)) \end{aligned}$$

Dimostrazione. Vediamo, per prima cosa, che $m \sim n$ implica che valgono le due condizioni dell'enunciato.

Supponiamo che $m \sim n$; dalla proposizione 1.3.1 (pag. 22) abbiamo che esiste una bisimulazione $R \subseteq [m_0\rangle \times [n_0\rangle$ con $(m, n) \in R$. In maniera del tutto analoga a quanto fatto nella dimostrazione della proposizione 1.3.3 (pag. 24) si può dimostrare che per R valgono le due implicazioni dell'enunciato e, quindi, che R è la bisimulazione cercata.

Vediamo ora che la validità delle due condizioni della proposizione implica che vale $m \sim n$.

Sia $R \subseteq [m_0\rangle \times [n_0\rangle$ la relazione tale per cui $(m, n) \in R$ e per cui, per ogni suo elemento, valgono le due implicazioni dell'enunciato. Dal momento che $[m_0\rangle \subseteq \mathcal{M}_{fin}(S_m)$ e che $[n_0\rangle \subseteq \mathcal{M}_{fin}(S_n)$ abbiamo banalmente che

$$(m, n) \in R \subseteq [m_0\rangle \times [n_0\rangle \subseteq \mathcal{M}_{fin}(S_m) \times \mathcal{M}_{fin}(S_n)$$

Dal momento che le implicazioni dell'enunciato del presente corollario sono esattamente quelle dell'enunciato della proposizione 1.3.3, possiamo applicare a R tale proposizione concludendo che $m \sim n$. $\square$

Capitolo 2

Multi-CCS

In questo capitolo riporto una definizione formale del Multi-CCS quasi identica a quella introdotta in [6], e meglio precisata in [7], con alcune modifiche ispirate a [5]. Aggiungo anche alcune lievi modifiche nella gestione delle sostituzioni e un paio di regole di congruenza strutturale nella semantica LTS, non previste originariamente, che ho scoperto essere necessarie per mantenere coerenza con la semantica sulle reti P/T. Rimuovo anche una regola di congruenza strutturale e una regola di transizione originarie che, grazie alle nuove regole di congruenza strutturale, sono risultate superflue.

Dal momento che non posso presumente che il Multi-CCS sia altrettanto noto, in seguito riporto maggiori dettagli e commenti rispetto a quanto fatto per gli LTS e le reti di Petri.

La semantica relativa alle reti P/T non limitate – sulla quale sono intervenuto, in maniera significativa, per migliorare alcuni aspetti e precisarne altri – sarà aggetto del capitolo 3.

2.1 Definizione del Multi-CCS

Definizioni 2.1.1. *Sia $\mathcal{L}$ un insieme numerabile di* nomi di canali *– ai quali, più semplicemente, ci riferiremo come* nomi *– che rappresenteremo, in senso generale, come $a, b, c, \ldots$; sia $\overline{\mathcal{L}}$ il corrispondente insieme dei* nomi complementari – *o* co-nomi – *che rappresenteremo, in senso generale, come* $\overline{a}, \overline{b}, \overline{c}, \ldots$ *e per i quali deve valere*

$$a \in \mathcal{L} \iff \overline{a} \in \overline{\mathcal{L}}$$

Diciamo che l'insieme $\mathcal{V} = \mathcal{L} \cup \overline{\mathcal{L}}$ *è l'insieme delle* azioni visibili *e rappresenteremo i suoi generici elementi come* $\alpha, \beta, \gamma, \ldots$. *Con* $\overline{\alpha}$ *indicheremo l'*azione complementare – *o* co-azione – *di* α, *dove intendiamo che sia* $\overline{\overline{\alpha}} = \alpha$. *Sia* $Act = \mathcal{V} \cup \{\tau\}$, *con* $\tau \notin \mathcal{V}$, *l'insieme delle* azioni, *dove* τ – *che chiamiamo* azione invisibile – *rappresenta un'attività interna al sistema; rappresenteremo con* μ *e variazioni* (μ', μ'', μ_1, μ_2, *ecc.) le generiche azioni.*

Sia $\mathcal{C}$ *un insieme numerabile di* costanti di processo – *che, in assenza di ambiguità, chiameremo semplicemente* costanti –, *i cui elementi (che generalmente rappresenteremo come* $A, B, C, \ldots$*) devono essere distinti da quelli di Act, ovvero tale che*

$$\mathcal{C} \cap Act = \emptyset$$

Osservo che le definizioni appena date corrispondono pienamente alle corrispondenti del CCS.

Definizioni 2.1.2. *Con la seguente grammatica definiamo due categorie di processi: i* processi sequenziali *(che indichiamo con* p*) – e la categoria più ampia dei* processi *(che indichiamo con* q*):*

$$p ::= \mathbf{0} \quad \Big| \quad \mu.q \quad \Big| \quad \underline{\mu}.q \quad \Big| \quad p+p \qquad (\textit{processi sequenziali})$$
$$q ::= p \quad \Big| \quad q \mid q \quad \Big| \quad (\nu a)q \quad \Big| \quad C \qquad (\textit{processi})$$

Nella grammatica appena definita:

- *definiamo* $\mathbf{0}$ processo nullo *o* processo terminato*;*
- *definiamo* $\mu.q$ processo normalmente prefisso *e chiamiamo la corrispondente azione* μ azione normalmente prefissa*;*
- *definiamo* $\underline{\mu}.q$ processo fortemente prefisso *e chiamiamo la corrispondente azione* $\underline{\mu}$ azione fortemente prefissa*;*

- *definiamo* $p + p$ somma protetta di processi sequenziali *o, più semplicemente,* somma di processi*;*

- *definiamo* $q \mid q$ composizione parallela di processi*;*

- *definiamo* $(\nu a)q$ processo ristretto *o, più semplicemente,* restrizione*;*

- *C rappresenta una costante, in $\mathcal{C}$.*

*Con $\mathcal{P}$ indicheremo l'*insieme dei processi Multi-CCS*, ovvero l'insieme dei termini definiti dalla grammatica precedente a patto che siano* chiusi *(ovvero: che ogni costante usata sia definita, nella forma $C \stackrel{def}{=} q$, anche ricorsivamente, dove q è un processo)* e protetti *(ovvero: in ogni definizione di costante, ogni costante usata deve essere preceduta da un'azione normalmente prefissa). Indicheremo i suoi termini generici con q, e in subordine con r, e variazioni (q', q'', q_1, q_2, q_a, r', ecc.).*

*Con $\mathcal{P}_{seq}$ indicheremo l'*insieme dei processi sequenziali Multi-CCS*, ottenuti dalla grammatica precedente con l'ulteriore vincolo di essere chiusi e protetti. Indicheremo i suoi termini generici con p e variazioni (p', p'', p_1, ecc.).*

Oltre alla definizione, vediamo le intenzioni:

- il processo nullo $\mathbf{0}$ è inteso a rappresentare un processo terminato o, comunque, che non può compiere transizioni;

- il processo normalmente prefisso $\mu.q$ è inteso a rappresentare un processo che compie l'azione μ (visibile o non visibile), trasformandosi nel processo q;

- il processo fortemente prefisso $\underline{\mu}.q$ è inteso a rappresentare un processo che inizia, o continua, una transizione atomica con l'azione μ, presupponendo che la transizione continui immediatamente con un'azione di q (e, quindi, che q sia in grado di muovere);

- la somma di processi $p + p$ è intesa a rappresentare, come nel CCS, un processo che, in una sua prima azione, può comportarsi come due processi distinti ma che, in seguito, sarà tenuto a comportarsi come il processo corrispondente alla prima azione; è importante notare che, nel caso del Multi-CCS, la somma è definita su processi sequenziali;

- la composizione parallela di processi $q \mid q$ è intesa a rappresentare due distinti processi che possono evolvere separatamente ma anche – come vedremo in seguito e analogamente al corrispondente costrutto del CCS – "sincronizzarsi" operando una transizione che coinvolge entrambi;

- la restrizione $(\nu a)q$ è intesa a rappresentare una modifica al processo q nel quale il nome visibile a e il suo co-nome $\overline{a}$ sono resi privati, non possono agire direttamente ma solo sincronizzandosi reciprocamente all'interno di q (e non con altre a e $\overline{a}$ all'esterno di q).

Osserviamo esplicitamente che, data la grammatica appena vista – e l'ulteriore vincolo sulle costanti esplicitato nella definizione –, i termini

$$(a.\mathbf{0} \mid b.\mathbf{0}) + c.\mathbf{0} \qquad\qquad A \stackrel{def}{=} \underline{a}.A$$

non sono processi conformi; infatti

- nel primo termine abbiamo che il primo addendo non è un processo sequenziale ma un processo ordinario mentre

- nel secondo termine la costante A è presente nella definizione di se stessa (il che, di per sé, sarebbe lecito) preceduta da un'azione fortemente prefissa, non da un'azione normalmente prefissa, come richiesto dalle regole appena introdotte.

Per chi già conosce il CCS, gli unici elementi veramente nuovi sono il processo fortemente prefisso e quello che ne consegue relativamente alle transizioni. Mentre nel CCS le transizioni sono determinate direttamente dalle singole azioni (o dalla sincronia di esattamente due azioni che originano un'azione invisibile), con l'idea dei processi fortemente prefissi introduciamo – come vedremo meglio nella prossima sezione – transizioni che, pur essendo atomicamente indivisibili, sono caratterizzate da più azioni. Le etichette delle transizioni non possono quindi più essere limitate a elementi dell'insieme Act, ma – in prima approssimazione – possono essere sequenze, di uno o più termini, di tale insieme. Questa complicazione è necessaria poiché con le sincronizzazioni del CCS tradizionale non saremmo in grado, nelle reti P/T, di esprimere transizioni che richiedono più di due token nel preset.

2.2 Definizioni accessorie

Prima di procedere oltre, è opportuno dare alcune definizioni.

2.2.1 Nomi liberi, nomi legati, nomi e constanti di un processo

Per prima cosa, definiamo i "nomi liberi" e i "nomi legati" di un processo.

Definizione 2.2.1. *Sia $q \in \mathcal{P}$; l'insieme $fn(q) \subseteq \mathcal{P}(\mathcal{A}ct)$ dei* nomi liberi *(*free names*) in q è definito come $fn(q) = F(q, \emptyset)$ dove*

$$F : \mathcal{P} \times \mathcal{P}(\mathcal{C}) \to \mathcal{P}(\mathcal{A}ct)$$

è definita induttivamente come segue:

$$\begin{aligned}
F(\mathbf{0}, C) &= \emptyset \\
F(\mu.q, C) &= \{\mu\} \cup F(q, C) \\
F(\underline{\mu}.q, C) &= \{\mu\} \cup F(q, C) \\
F(p_1 + p_2, C) &= F(p_1, C) \cup F(p_2, C) \\
F(q_1 \mid q_2, C) &= F(q_1, C) \cup F(q_2, C) \\
F((\nu a)q, C) &= F(q, C) \setminus \{a, \overline{a}\} \\
F(A, C) &= \begin{cases} \emptyset & \textit{quando } A \in C \\ F(q, C \cup \{A\}) & \textit{quando } A \notin C \wedge A \stackrel{def}{=} q \end{cases}
\end{aligned}$$

Definizione 2.2.2. *Sia $q \in \mathcal{P}$; l'insieme $bn(q) \subseteq \mathcal{P}(\mathcal{A}ct)$ dei* nomi legati *(*bound names*) in q è definito come $bn(q) = B(q, \emptyset)$ dove*

$$B : \mathcal{P} \times \mathcal{P}(\mathcal{C}) \to \mathcal{P}(\mathcal{A}ct)$$

è definita induttivamente come segue:

$$\begin{aligned}
B(\mathbf{0}, C) &= \emptyset \\
B(\mu.q, C) &= B(q, C) \\
B(\underline{\mu}.q, C) &= B(q, C) \\
B(p_1 + p_2, C) &= B(p_1, C) \cup B(p_2, C) \\
B(q_1 \mid q_2, C) &= B(q_1, C) \cup B(q_2, C) \\
B((\nu a)q, C) &= \{a, \overline{a}\} \cup B(q, C) \\
B(A, C) &= \begin{cases} \emptyset & \textit{quando } A \in C \\ B(q, C \cup \{A\}) & \textit{quando } A \notin C \wedge A \stackrel{def}{=} q \end{cases}
\end{aligned}$$

Poi i "nomi" di un processo, banalmente come unione dei nomi liberi e dei nomi legati.

Definizione 2.2.3. *Sia $q \in \mathcal{P}$; l'insieme $n(q) \subseteq \mathcal{P}(\mathcal{A}ct)$ dei* nomi *(*names*) in q è definito come $n(q) = fn(q) \cup bn(q)$.*

Infine l'insieme delle costanti di un processo.

Definizione 2.2.4. *Sia $q \in \mathcal{P}$; l'insieme $c(q) \subseteq \mathcal{P}(\mathcal{A}ct)$ delle* costanti *in q è definito come $c(q) = C(q, \emptyset)$ dove*

$$C : \mathcal{P} \times \mathcal{P}(\mathcal{C}) \to \mathcal{P}(\mathcal{C})$$

è definita induttivamente come segue:

$$\begin{aligned}
C(\mathbf{0}, A) &= \emptyset \\
C(\mu.q, A) &= C(q, A) \\
C(\underline{\mu}.q, A) &= C(q, A)
\end{aligned}$$

$$
\begin{aligned}
C(p_1 + p_2, A) &= C(p_1, A) \cup C(p_2, A) \\
C(q_1 \mid q_2, A) &= C(q_1, A) \cup C(q_2, A) \\
C((\nu a)q, A) &= C(q, A) \\
C(B, A) &= \begin{cases} \emptyset & \textit{quando } B \in A \\ \{B\} \cup C(q, A \cup \{B\}) & \textit{quando } B \notin A \wedge B \overset{def}{=} q \end{cases}
\end{aligned}
$$

Date le precedenti definizioni, è banale osservare quanto segue.

Osservazione 2.2.1. *Siano $q, q_1, q_2 \in \mathcal{P}$ generici processi Multi-CCS; siano $p_1, p_2 \in \mathcal{P}_{seq}$ generici processi Multi-CCS sequenziali; sia $\mu \in \mathcal{A}ct$ una generica azione; abbiamo*

$$
\begin{array}{lll}
fn(q) \subseteq fn(\mu.q) & bn(q) = bn(\mu.q) & c(q) = c(\mu.q) \\
fn(q) \subseteq fn(\underline{\mu}.q) & bn(q) = bn(\underline{\mu}.q) & c(q) = c(\underline{\mu}.q) \\
fn(p_1) \subseteq fn(p_1 + p_2) & bn(p_1) \subseteq bn(p_1 + p_2) & c(p_1) \subseteq c(p_1 + p_2) \\
fn(p_2) \subseteq fn(p_1 + p_2) & bn(p_2) \subseteq bn(p_1 + p_2) & c(p_2) \subseteq c(p_1 + p_2) \\
fn(q_1) \subseteq fn(q_1 \mid q_2) & bn(q_1) \subseteq bn(q_1 \mid q_2) & c(q_1) \subseteq c(q_1 \mid q_2) \\
fn(q_2) \subseteq fn(q_1 \mid q_2) & bn(q_2) \subseteq bn(q_1 \mid q_2) & c(q_2) \subseteq c(q_1 \mid q_2)
\end{array}
$$

2.2.2 Nomi di un'etichetta

Più avanti avremo bisogno di trattare elementi di $\mathcal{V}^+ \cup \{\tau\}$[1] e avremo bisogno di sapere se una determinata azione è presente in un preciso elemento di tale insieme. Definiamo quindi

Definizione 2.2.5. *L'*insieme dei nomi di un'etichetta $\sigma \in \mathcal{V}^+ \cup \{\tau\}$ *– che, sinteticamente indicheremo con $n(\sigma)$ – è pari a*

$$
n(\sigma) = \begin{cases} \{\tau\} & \textit{quando } \sigma = \tau \\ \{v_1, v_2, \ldots, v_n\} & \textit{quando } \sigma = v_1 v_2 \ldots v_n \textit{ con } v_1, v_2, \ldots, v_n \in \mathcal{V} \end{cases}
$$

[1]dove per A^+ intendiamo le sequenze di lunghezza n, con $n \geqslant 1$, di elementi di A, ovvero $a_1 a_2 a_3 \ldots a_n$, con $\forall i \in [1..n].a_i \in A$

2.2.3 Sostituzioni

Definizione 2.2.6. *Con* sostituzione di a con b in q *– che sinteticamente indicheremo con la notazione* $q\{b/a\}$ *–, dove* $q \in \mathcal{P}$ *è un generico processo Multi-CCS e* $a, b \in \mathcal{L}$ *sono due generici nomi, intendiamo il termine ottenuto da* q *sostituendo* b *e* $\overline{b}$ *alle occorrenze libere di* a *e, rispettivamente,* $\overline{a}$ *in* q*. È definita induttivamente dalle seguenti regole:*

$$\mathbf{0}\{b/a\} = \mathbf{0} \tag{2.1a}$$

$$(a.q)\{b/a\} = b.(q\{b/a\}) \tag{2.1b}$$

$$(\overline{a}.q)\{b/a\} = \overline{b}.(q\{b/a\}) \tag{2.1c}$$

$$(\underline{a}.q)\{b/a\} = \underline{b}.(q\{b/a\}) \tag{2.1d}$$

$$(\underline{\overline{a}}.q)\{b/a\} = \underline{\overline{b}}.(q\{b/a\}) \tag{2.1e}$$

$$(\mu.q)\{b/a\} = \mu.(q\{b/a\}) \qquad \textit{quando } \mu \notin \{a, \overline{a}\} \tag{2.1f}$$

$$(\underline{\mu}.q)\{b/a\} = \underline{\mu}.(q\{b/a\}) \qquad \textit{quando } \mu \notin \{a, \overline{a}\} \tag{2.1g}$$

$$(p_1 + p_2)\{b/a\} = p_1\{b/a\} + p_2\{b/a\} \tag{2.1h}$$

$$(q_1 \mid q_2)\{b/a\} = q_1\{b/a\} \mid q_2\{b/a\} \tag{2.1i}$$

$$((\nu c)q)\{b/a\} = (\nu c)(q\{b/a\}) \qquad \textit{quando } c \notin \{a, b\} \tag{2.1j}$$

$$((\nu b)q)\{b/a\} = (\nu c)((q\{c/b\})\{b/a\}) \qquad \textit{con } c, \overline{c} \notin n(q) \tag{2.1k}$$

$$((\nu a)q)\{b/a\} = (\nu a)q \tag{2.1l}$$

Osserviamo che in [7], al posto dell'unica equazione 2.1l, sono presenti le due seguenti

$$((\nu a)q)\{b/a\} = (\nu b)q\{b/a\} \qquad \text{quando } b, \overline{b} \notin fn(q) \tag{2.2a}$$

$$((\nu a)q)\{b/a\} = (\nu a)q \qquad \text{quando } b, \overline{b} \in fn(q) \tag{2.2b}$$

La sostituzione fittizia (in quanto priva di reale efficacia) di 2.2a è motivata dalla necessità di introdurre degli identificativi aggiuntivi per distinguere, nella semantica definita sulle reti P/T (che vedremo nel prossimo capitolo), posti che, altrimenti, sarebbero stati indistinguibili causando comportamenti non coerenti con la corrispondente semantica LTS. Le modifiche che introduco nella semantica delle reti P/T – che, tra le altre cose, prevedono l'aggiunta, quando necessario, di un opportuno identificativo extra ai posti – annulla tale necessità.

Riguardo alle sostituzioni applicate alle costanti, in [7] si procede a generare nuove costanti di processo. Più formalmente: se $A \stackrel{def}{=} q$ è una costante definita pari al processo $q \in \mathcal{P}$, la sostituzione $\{b/a\}$ applicata ad A genera una nuova costante $A_{\{b/a\}} \stackrel{def}{=} q\{b/a\}$. Per semplificare: quando a una costante si applicano più sostituzioni, la nuova costante è definita in base alla combinazione delle sostituzioni; citando l'esempio di [7], quando applichiamo le sostituzioni, nell'ordine, $\{a'/a\}$ e $\{a''/a'\}$, dove a' e a'' sono nuovi nomi ristretti, la costante risultante è $A_{\{a''/a\}}$.

Per semplificare ulteriormente, ho scelto di adottare una lieve modifica: considero significative le sostituzioni solo quando il nome sostituito e/o il corrispondente co-nome sono liberi in A. Ovvero: la sostituzione $\{b/a\}$ sulla costante $A \stackrel{def}{=} q$:

- genera la nuova costante $A_{\{b/a\}} \stackrel{def}{=} q\{b/a\}$ quando $\{a, \overline{a}\} \cap fn(q) \neq \emptyset$;
- è banalmente pari ad A altrimenti (quando $\{a, \overline{a}\} \cap fn(q) = \emptyset$).

Questa scelta evita di moltiplicare inutilmente le costanti – e i posti nelle reti P/T – in alcuni casi come, ad esempio, in

$$B \stackrel{def}{=} (\nu a)(a.B \mid \overline{a}.B)$$

2.2.4 Congruenza strutturale

Definiamo ora la "Congruenza strutturale" del Multi-CCS; una relazione tra processi in $\mathcal{P}$ che verrà usata nelle regole di transizione per la semantica definita sugli LTS.

Definizione 2.2.7. *La* congruenza strutturale *tra due generici processi Multi-CCS* $q_1, q_2 \in \mathcal{P}$ *– che diciamo* strutturalmente congruenti *e lo indichiamo, in maniera sintetica con* $q_1 \equiv q_2$ *– è la congruenza strutturale indotta dalle seguenti condizioni di base*

- $q_1 = (r_1 \mid r_2) \mid r_3$ *e* $q_2 = r_1 \mid (r_2 \mid r_3)$*, per opportuni processi* $r_1, r_2, r_3 \in \mathcal{P}$*;*

- $q_1 = (\nu a)(r_1 \mid r_2)$ *e* $q_2 = r_1 \mid (\nu a)r_2$*, per un opportuno nome* $a \in \mathcal{L}$ *e per opportuni processi* $r_1, r_2 \in \mathcal{P}$ *per il primo dei quali sia vero che* $a, \overline{a} \notin fn(r_1)$*;*

- $q_1 = (\nu a)r$ *e* $q_2 = (\nu b)(r\{b/a\})$*, per opportuni nomi* $a, b \in \mathcal{L}$ *e per un opportuno processo* $r \in \mathcal{P}$ *per il quale sia vero che* $b, \overline{b} \notin fn(r)$ *(detta anche* regola di alpha-conversione*);*

- $q_1 = r_1 \mid r_2$ *e* $q_2 = r_2 \mid r_1$*, per opportuni processi* $r_1, r_2 \in \mathcal{P}$ *(*commutatività*);*

- $q_1 = A$ *e* $q_2 = r$*, con* $A \stackrel{def}{=} r$ *per un opportuno processo* $r \in \mathcal{P}$ *(*regola della costante*).*

Le ultime due condizioni di base non erano originariamente previste in [6] e in [7]. Più avanti spiego il motivo per cui ho ritenuto necessario introdurle.

In [7] era inoltre prevista una condizione simmetrica rispetto alla seconda, ovvero (ipotizzando ovviamente $b, \overline{b} \notin fn(r_2)$)

$$(\nu a)(r_1 \mid r_2) \equiv ((\nu a)r_1) \mid r_2$$

Introducendo la regola della commutatività, quest'ultima regola è diventata superflua in quanto conseguenza della seconda e – appunto – della quarta; infatti

$$(\nu a)(r_1 \mid r_2) \equiv (\nu a)(r_2 \mid r_1) \equiv r_2 \mid ((\nu a)r_1) \equiv ((\nu a)r_1) \mid r_2$$

2.3 Semantica operazionale sugli LTS

Vediamo ora la semantica operazionale, definita sugli LTS, del Multi-CCS.

2.3.1 Regole di transizione

Nella tradizione dell'analoga operazione effettuata sul CCS, vediamo ora la definizione di una semantica sugli LTS per il Multi-CCS.

$$\text{(Pref)}\ \mu.q \xrightarrow{\mu} q \qquad \text{(S-pref)}\ \frac{q \xrightarrow{\sigma'} q'}{\underline{\mu}.q \xrightarrow{\sigma} q'}\ \ Conc(\mu,\sigma',\sigma)$$

$$\text{(Sum-1)}\ \frac{p_1 \xrightarrow{\sigma} p_1'}{p_1 + p_2 \xrightarrow{\sigma} p_1'} \qquad \text{(Sum-2)}\ \frac{p_2 \xrightarrow{\sigma} p_2'}{p_1 + p_2 \xrightarrow{\sigma} p_2'}$$

$$\text{(Par-1)}\ \frac{q_1 \xrightarrow{\sigma} q_1'}{q_1 \mid q_2 \xrightarrow{\sigma} q_1' \mid q_2} \qquad \text{(Par-2)}\ \frac{q_2 \xrightarrow{\sigma} q_2'}{q_1 \mid q_2 \xrightarrow{\sigma} q_1 \mid q_2'}$$

$$\text{(Com)}\ \frac{q_1 \xrightarrow{\sigma_1} q_1' \quad q_2 \xrightarrow{\sigma_2} q_2'}{q_1 \mid q_2 \xrightarrow{\sigma} q_1' \mid q_2'}\ \ Sync(\sigma_1,\sigma_2,\sigma)$$

$$\text{(Res)}\ \frac{q \xrightarrow{\sigma} q'}{(\nu a)q \xrightarrow{\sigma} (\nu a)q'}\ \ a,\overline{a} \notin n(\sigma)$$

$$\text{(Cong)}\ \frac{q \equiv q_1 \quad q_1 \xrightarrow{\sigma} q_1' \quad q_1' \equiv q'}{q \xrightarrow{\sigma} q'}$$

Tabella 2.1: Semantica operazionale sugli LTS per il Multi-CCS

Definizioni 2.3.1. *La* semantica operazionale *per i processi del Multi-CCS è data dall'LTS costituito da* $(\mathcal{P}, \mathcal{A}, \longrightarrow)$ *dove:*

- $\mathcal{P}$ *è l'insieme dei processi Multi-CCS come da definizione 2.1.2 (pag. 28);*

- $\mathcal{A} = \mathcal{V}^+ \cup \{\tau\}$ *è l'insieme delle etichette delle transizioni, che rappresentiamo genericamente con* σ *e relative variazioni* (σ', σ'', σ_1, σ_2, *ecc.);*

- $\longrightarrow \subseteq \mathcal{P} \times \mathcal{A} \times \mathcal{P}$ *è il più piccolo insieme di transizioni generato dalle regole di tabella 2.1 (pagina precedente) col supporto delle regole di sincronizzazione di tabella 2.2 (pagina a fronte) – nelle quali, preciso, i primi due argomenti di* $Merge$ *sono elementi di* $\mathcal{A} \cup \{\epsilon\}$*, dove con* ϵ *intendiamo una sequenza vuota di azioni, mentre gli argomenti di* $Sync$ *e il terzo argomento di* $Merge$ *sono elementi di* $\mathcal{A}$ *(ovvero: non prevedono sequenze vuote di azioni) – e delle regole di concatenazione* $Conc$*, definita in tabella 2.3 (pagina a fronte), nella quale il primo argomento è un elemento di* $\mathcal{A}ct$ *e i due seguenti sono elementi di* $\mathcal{A}$*.*

In realtà, la definizione che ho dato è differente da quella originaria, di [6] e di [7], e si ispira – semplificandola in qualche dettaglio tecnico ma rimanendo aderente alla sostanza delle modifiche – a quanto Gorrieri propone in [5]. Vado ora a descrivere le differenze e a motivare le modifiche.

Per prima cosa segnalo che, grazie all'introduzione (la cui necessità motiverò più avanti) della regola delle costanti, ho soppresso l'originaria regola (Cons) – riportata in tabella 2.4 (pag. 40) – in quanto la sua funzione è stata assorbita dalla regola (Cong).

Segnalo poi che l'originario insieme $\mathcal{A}$ delle etichette delle transizioni non era $\mathcal{V}^+ \cup \{\tau\}$ bensì $\mathcal{A}ct^*$[2] (anche se, di fatto, le regole di transizione generano etichette in $\mathcal{A}ct^+$). Il punto è che le originarie regole di transizione non escludevano transizioni etichettate $\tau\tau$ o $\alpha\tau$, ovvero etichette costituite da più di un τ oppure con uno o più τ insieme ad azioni visibili. Si osservi infatti l'originaria regola (S-pref), in tabella 2.5 (pag. 40), e la si applichi ai termini $\underline{\tau}.\underline{\tau}.\tau.\mathbf{0}$, $\underline{a}.\tau.\mathbf{0}$ e $\underline{\tau}.a.\mathbf{0}$; dall'originaria (S-pref) otterremmo

$$\underline{\tau}.\underline{\tau}.\tau.\mathbf{0} \xrightarrow{\tau\tau\tau} \mathbf{0}$$

[2]dove con A^* si intende l'insieme delle sequenze di elementi di A di lunghezza n, ove $n \in \mathbb{N}$ può anche essere zero; ovvero ϵ (sequenza nulla) oppure (quando $n \geqslant 1$) le sequenze $a_1 a_2 \ldots a_n$, con $\forall i \in [1..n] \in A$

$$Sync(\alpha, \overline{\alpha}, \tau) \qquad \frac{Merge(\sigma_1, \sigma_2, \sigma)}{Sync(\alpha\sigma_1, \overline{\alpha}\sigma_2, \sigma)}$$

$$\frac{Sync(\sigma_1, \sigma_2, \sigma')}{Sync(\mu\sigma_1, \sigma_2, \sigma)} \quad Conc(\mu, \sigma', \sigma)$$

$$\frac{Sync(\sigma_1, \sigma_2, \sigma')}{Sync(\sigma_1, \mu\sigma_2, \sigma)} \quad Conc(\mu, \sigma', \sigma)$$

$$Merge(\alpha, \overline{\alpha}, \tau) \qquad \frac{Merge(\sigma_1, \sigma_2, \sigma)}{Merge(\alpha\sigma_1, \overline{\alpha}\sigma_2, \sigma)}$$

$$Merge(\mu, \epsilon, \mu) \qquad Merge(\epsilon, \mu, \mu)$$

$$\frac{Merge(\sigma_1, \sigma_2, \sigma')}{Merge(\mu\sigma_1, \sigma_2, \sigma)} \quad Conc(\mu, \sigma', \sigma)$$

$$\frac{Merge(\sigma_1, \sigma_2, \sigma')}{Merge(\sigma_1, \mu\sigma_2, \sigma)} \quad Conc(\mu, \sigma', \sigma)$$

Tabella 2.2: Regole di sincronizzazione per Multi-CCS

$$Conc(\mu, \tau, \mu) \qquad Conc(\tau, \sigma, \sigma) \qquad \frac{\mu \neq \tau \neq \sigma}{Conc(\mu, \sigma, \mu\sigma)}$$

Tabella 2.3: Regole di concatenazione per azioni del Multi-CCS

$$(\text{Cons})\ \frac{q \xrightarrow{\sigma} q'}{C \xrightarrow{\sigma} q'} \quad C \stackrel{def}{=} q$$

Tabella 2.4: Originaria regola (Cons)

$$\underline{a}.\tau.\mathbf{0} \xrightarrow{a\tau} \mathbf{0}$$
$$\underline{\tau}.a.\mathbf{0} \xrightarrow{\tau a} \mathbf{0}$$

Tenendo conto che mi sembra inopportuno che possano esistere transizioni con tali etichette – e che, per di più, τ sia considerata diversa da $\tau\tau$, ed entrambe considerate diverse da $\tau\tau\tau$, e così via, e, analogamente, che α sia considerata diversa da $\alpha\tau$ ed entrambe considerate diverse da $\tau\alpha$, e così via –, non fosse altro perché avrebbe complicato l'eventuale definizione di bisimulazione debole, ho deciso di seguire l'esempio di [5] e di far "collassare" le τ in una sola, quando più di una, o di farle sparire, quando associate a una o più azioni visibili.

$$(\text{S-pref})\ \frac{q \xrightarrow{\sigma} q'}{\underline{\mu}.q \xrightarrow{\mu\sigma} q'}$$

Tabella 2.5: Originaria regola (S-pref)

Per semplificare le regole descritte in [5] – appesantite da una dettagliata casistica – ho introdotto la funzione di concatenazione *Conc*, definita in tabella 2.3 (pagina precedente), ottenendo l'opportuno collasso delle τ (equivalente a quello ottenibile dalle regole in [5]) e l'ho applicata, oltre che alla regola (S-pref) (unica regola modificata direttamente), dove opportuno nelle regole di sincronizzazione di tabella 2.2 (pagina precedente) applicandola quindi, indirettamente, anche alla regola (Com). Grazie all'introduzione di

$Conc$, le transizioni di prima diventano

$$\underline{\tau}.\underline{\tau}.\tau.\mathbf{0} \xrightarrow{\tau} \mathbf{0}$$
$$\underline{a}.\tau.\mathbf{0} \xrightarrow{a} \mathbf{0}$$
$$\underline{\tau}.a.\mathbf{0} \xrightarrow{a} \mathbf{0}$$

eliminando le τ superflue dalle etichette.

Quanto alle regole di sincronizzazione, si osservino le originarie di [7] – riportate in tabella 2.6 – e le si applichi ai termini

$$q_1 = \underline{b}.a.\mathbf{0} \mid \underline{c}.\overline{a}.\mathbf{0}$$
$$q_2 = \underline{a}.b.\mathbf{0} \mid \underline{\overline{a}}.c.\mathbf{0}$$
$$q_3 = a.\mathbf{0} \mid \underline{\overline{a}}.\underline{b}.c.\mathbf{0}$$

$$Sync(\alpha, \overline{\alpha}, \tau) \qquad \frac{Sync(\sigma_1, \sigma_2, \sigma)}{Sync(\alpha\sigma_1, \overline{\alpha}\sigma_2, \sigma)}$$

$$\frac{\sigma \neq \epsilon}{Sync(\alpha\sigma, \overline{\alpha}, \sigma)} \qquad \frac{\sigma \neq \epsilon}{Sync(\alpha, \overline{\alpha}\sigma, \sigma)}$$

$$\frac{Sync(\sigma_1, \sigma_2, \sigma)}{Sync(\mu\sigma_1, \sigma_2, \mu\sigma)} \qquad \frac{Sync(\sigma_1, \sigma_2, \sigma)}{Sync(\sigma_1, \mu\sigma_2, \mu, \sigma)}$$

Tabella 2.6: Originarie regole di sincronizzazione per Multi-CCS

Mentre a q_1 è consentita la seguente transizione

$$q_1 = \underline{b}.a.\mathbf{0} \mid \underline{c}.\overline{a}.\mathbf{0} \xrightarrow{bc} \mathbf{0} \mid \mathbf{0}$$

e pure la medesima transizione con etichetta cb, dal momento che le regole (le ultime due di $Sync$) consentono il "consumo" dei termini che non si sincronizzano direttamente (b e c, in questo caso) prima della sincronizzazione vera e propria (su a e il suo co-nome), la corrispondente transizione su q_2

$$q_2 = \underline{a}.b.\mathbf{0} \mid \underline{\overline{a}}.c.\mathbf{0} \xrightarrow{bc} \mathbf{0} \mid \mathbf{0}$$

non è consentita dalle regole originarie. È invece consentita la transizione

$$q_3 = a.\mathbf{0} \mid \underline{\overline{a}}.\underline{b}.c.\mathbf{0} \xrightarrow{bc} \mathbf{0} \mid \mathbf{0}$$

Per superare questa dicotomia, ispirandomi a (e semplificando) [5], ho modificato le regole – come da tabella 2.2 (pag. 39) – mantenendo, sostanzialmente, il comportamento della *Sync* originaria fino alla prima sincronizzazione di due azioni, passando poi il controllo alla nuova regola *Merge* che si comporta come *Sync* ma che prevede anche delle regole di terminazione. Grazie alle modifiche effettuate, anche la transizione per q_2 è ora ammessa.

Si osservi che non è possibile utilizzare solo *Merge*, al posto di *Sync*, poiché *Merge* non impone che vi siano effettive sincronizzazioni; la sola *Merge* consentirebbe infatti

$$\underline{a}.b.\mathbf{0} \mid \underline{c}.d.\mathbf{0} \xrightarrow{acbd} \mathbf{0} \mid \mathbf{0}$$

perché si limiterebbe a consumare e miscelare i termini delle due transizioni di partenza (ab e cd) senza imporre alcuna sincronizzazione effettiva. Tale sincronizzazione è invece imposta dalla *Sync*.

2.3.2 Necessità della nuove regole di congruenza

Come accennavo nella sottosezione 2.2.4, ho aggiunto alle regole di congruenza strutturale originarie una regola per la commutatività e una per la costante.

La necessità della regola di commutatività può essere evidenziata dal confronto tra i due seguenti processi:

$$q_1 = (\underline{a}.c.\mathbf{0} \mid b.\mathbf{0}) \mid (\overline{a}.\mathbf{0} \mid \underline{\overline{b}}.\overline{c}.\mathbf{0})$$
$$q_2 = (b.\mathbf{0} \mid \underline{a}.c.\mathbf{0}) \mid (\overline{a}.\mathbf{0} \mid \underline{\overline{b}}.\overline{c}.\mathbf{0})$$

Osserviamo che q_2 è in grado di effettuare, tra le altre, la transizione $q_2 \xrightarrow{\tau} \mathbf{0} \mid ((\mathbf{0} \mid \mathbf{0}) \mid \mathbf{0})$; infatti, utilizzando esclusivamente la prima regola di congruenza, abbiamo

$$q_2 = (b.\mathbf{0} \mid \underline{a}.c.\mathbf{0}) \mid (\overline{a}.\mathbf{0} \mid \underline{\overline{b}}.\overline{c}.\mathbf{0}) \equiv b.\mathbf{0} \mid (\underline{a}.c.\mathbf{0} \mid (\overline{a}.\mathbf{0} \mid \underline{\overline{b}}.\overline{c}.\mathbf{0}))$$

$$\equiv b.\mathbf{0} \mid ((\underline{a}.c.\mathbf{0} \mid \overline{a}.\mathbf{0}) \mid \underline{\overline{b}}.\overline{c}.\mathbf{0})$$

e quindi, applicando più volte la regola (Com), abbiamo

$$\dfrac{b.\mathbf{0} \xrightarrow{b} \mathbf{0} \qquad \dfrac{\dfrac{\underline{a}.c.\mathbf{0} \xrightarrow{ac} \mathbf{0} \quad \overline{a}.\mathbf{0} \xrightarrow{\overline{a}} \mathbf{0}}{\underline{a}.c.\mathbf{0} \mid \overline{a}.\mathbf{0} \xrightarrow{c} \mathbf{0} \mid \mathbf{0}} \quad \underline{\overline{b}}.\overline{c}.\mathbf{0} \xrightarrow{\overline{b}\overline{c}} \mathbf{0}}{(\underline{a}.c.\mathbf{0} \mid \overline{a}.\mathbf{0}) \mid \underline{\overline{b}}.\overline{c}.\mathbf{0} \xrightarrow{\overline{b}} (\mathbf{0} \mid \mathbf{0}) \mid \mathbf{0}}}{b.\mathbf{0} \mid ((\underline{a}.c.\mathbf{0} \mid \overline{a}.\mathbf{0}) \mid \underline{\overline{b}}.\overline{c}.\mathbf{0}) \xrightarrow{\tau} \mathbf{0} \mid ((\mathbf{0} \mid \mathbf{0}) \mid \mathbf{0})}$$

Applicando poi la regola (Cong) alla transizione appena mostrata, si ottiene $q_2 \xrightarrow{\tau} \mathbf{0} \mid ((\mathbf{0} \mid \mathbf{0}) \mid \mathbf{0})$.

Non riusciamo però, utilizzando solo le prime quattro regole di congruenza strutturale, a dimostrare che $q_1 \xrightarrow{\tau} \mathbf{0} \mid ((\mathbf{0} \mid \mathbf{0}) \mid \mathbf{0})$; infatti non riusciamo a mettere "a contatto" due processi sequenziali in grado di sincronizzarsi; riusciamo ugualmente, grazie anche alle regole (Par-1) e/o (Par-2) a ottenere delle sincronizzazioni, ma l'applicazione di queste regole ci fa perdere (in un certo senso) un livello di sincronizzazione, impedendoci di sincronizzare, in un'unica transizione, tutti e quattro i processi sequenziali.

La necessità della regola della costante (o almeno di una regola che abbia un effetto analogo) può, ancor più semplicemente, essere evidenziata dal confronto tra i due seguenti processi

$$q_1 = \underline{a}.b.\mathbf{0} \mid (\overline{a}.\mathbf{0} \mid \overline{b}.\mathbf{0})$$
$$q_2 = \underline{a}.b.\mathbf{0} \mid C$$

dove $C \stackrel{def}{=} \overline{a}.\mathbf{0} \mid \overline{b}.\mathbf{0}$.

Osserviamo che q_1 è in grado di effettuare, tra le altre, la transizione $q_1 \xrightarrow{\tau} (\mathbf{0} \mid \mathbf{0}) \mid \mathbf{0}$; infatti, utilizzando esclusivamente la prima regola di congruenza, abbiamo

$$q_1 = \underline{a}.b.\mathbf{0} \mid (\overline{a}.\mathbf{0} \mid \overline{b}.\mathbf{0}) \equiv (\underline{a}.b.\mathbf{0} \mid \overline{a}.\mathbf{0}) \mid \overline{b}.\mathbf{0}$$

e quindi, applicando due volte la regola (Com), abbiamo

$$\dfrac{\dfrac{\underline{a}.b.\mathbf{0} \xrightarrow{ab} \mathbf{0} \quad \overline{a}.\mathbf{0} \xrightarrow{\overline{a}} \mathbf{0}}{\underline{a}.b.\mathbf{0} \mid \overline{a}.\mathbf{0} \xrightarrow{b} \mathbf{0} \mid \mathbf{0}} \quad \overline{b}.\mathbf{0} \xrightarrow{\overline{b}} \mathbf{0}}{(\underline{a}.b.\mathbf{0} \mid \overline{a}.\mathbf{0}) \mid \overline{b}.\mathbf{0} \xrightarrow{\tau} (\mathbf{0} \mid \mathbf{0}) \mid \mathbf{0}}$$

Applicando poi la regola (Cong) alla transizione appena mostrata, si ottiene $q_1 \xrightarrow{\tau} (\mathbf{0} \mid \mathbf{0}) \mid \mathbf{0}$.

Non riusciamo però, utilizzando solo le prime quattro regole di congruenza strutturale e l'originaria regola (Cons) (vedasi tabella 2.4 (pag. 40)), a dimostrare che $q_2 \xrightarrow{\tau} (\mathbf{0} \mid \mathbf{0}) \mid \mathbf{0}$; infatti, le regole di transizione consentono a una costante una sola transizione (per volta) mentre per ottenere la transizione τ abbiamo bisogno che la costante supporti contemporaneamente (come definiremo meglio in seguito) più di una transizione.

Quelli evidenziati non sono problemi secondari poiché, come vedremo nel capitolo 3, nella semantica definita sulle reti P/T, i termini q_1 e q_2 (in entrambi gli esempi) generano due marcature coincidenti. Se vogliamo conservare la correttezza tra le due semantiche – ovvero, se vogliamo che i processi Multi-CCS, visti nella semantica LTS e nella semantica P/T, abbiano un comportamento coerente, in termini di etichette di transizioni possibili – è ovviamente necessario che q_1 e q_2 possano effettuare le medesime transizioni anche nella semantica LTS. Ho quindi introdotto la quarta e la quinta regola di congruenza strutturale, grazie alla quale (in entrambi gli esempi) abbiamo $q_1 \equiv q_2$ e quindi, applicando la regola (Cong), che $q_1 \xrightarrow{\tau} \mathbf{0} \mid ((\mathbf{0} \mid \mathbf{0}) \mid \mathbf{0})$, nel primo esempio, e che $q_2 \xrightarrow{\tau} (\mathbf{0} \mid \mathbf{0}) \mid \mathbf{0}$, nel secondo.

2.3.3 Peso di una transizione

Sulla base delle definizioni delle transizioni sugli LTS, possiamo definire il "peso" di una transizione come segue

Definizione 2.3.2. *Sia $t \in \longrightarrow$ una transizione generata dalle regole di tabella 2.1 (pag. 37), definiamo induttivamente il* peso *della transizione – e lo indichiamo sinteticamente come $\gamma(t)$ – come segue:*

- *$\gamma(t) = 1$, se t è giustificata dalla regola (Pref);*
- *$\gamma(t) = \gamma(t')+1$, se t è giustificata dalla regola (S-pref), (Sum-1), (Sum-2), (Par-1), (Par-2), (Res), (Cong) o (Cons), dove t' è la transizione della premessa;*

- $\gamma(t) = \gamma(t_1) + \gamma(t_2) + 1$, *se t è giustificata dalla regola (Com), dove t_1 e t_2 sono le transizioni della premessa.*

Nel caso la transizione t possa essere giustificata da diverse regole o alberi di prova, il peso $\gamma(t)$ è il minore dei pesi ottenibili dall'applicazione delle regole indicate.

Grazie al peso delle transizioni potremo, in alcune delle dimostrazioni per induzione, disporre di un valore numerico decrescente che giustificherà l'induzione stessa.

2.3.4 Transizioni contemporaneamente supportate

Nelle definizioni 1.2.5 (pag. 14) ho introdotto la definizione di multiset di transizioni contemporaneamente supportare per le reti di Petri. Può aver senso (e tornerà utile nelle definizioni che seguiranno) definire qualcosa di analogo per i processi Multi-CCS. L'osservazione da cui si può partire è che i due processi seguenti

$$a.\mathbf{0} \mid b.\mathbf{0} \qquad\qquad a.b.\mathbf{0} + b.a.\mathbf{0}$$

sono bisimili (entrambi possono supportare una transizione con etichetta a e una con etichetta b, in qualsiasi ordine, poi più nulla) ma – almeno intuitivamente – sono diversi poiché il primo è costituito da processi paralleli e quindi le sue azioni sono (in un certo senso) eseguibili contemporaneamente mentre il secondo è costituito da un'alternativa (somma) tra processi sequenziali e quindi l'esecuzione contemporanea non è proponibile. Questa differenza diventa evidente quando sostituiamo, in entrambi i processi, l'azione b con la complementare di a

$$a.\mathbf{0} \mid \overline{a}.\mathbf{0} \qquad\qquad a.\overline{a}.\mathbf{0} + \overline{a}.a.\mathbf{0}$$

ottenendo un primo processo che è in grado di sincronizzare a con $\overline{a}$ ottenendo una transizione con etichetta τ e un secondo processo che non è in grado di effettuare questa sincronizzazione.

Rimanendo al processo parallelo $a.\mathbf{0} \mid \overline{a}.\mathbf{0}$, ancora a un livello intuitivo, possiamo vederlo come un processo che è in grado di effettuare una transizione con etichetta a, oppure una transizione con etichetta $\overline{a}$, oppure due transizioni, in contemporanea, con etichette a e $\overline{a}$, oppure una transizione con etichetta τ. Queste due ultime alternative hanno poi una destinazione comune (il processo simil-nullo $\mathbf{0} \mid \mathbf{0}$).

Per formalizzare queste intuizioni, mi sono ispirato alle regole delle step semantics proposta da Gorrieri in [5]; propongo però una versione più semplice, meno ambiziosa ma sufficiente ai miei scopi (dimostrare la correttezza della semantica sulle reti P/T che proporrò nel capitolo 4).

$$\text{(Null-s)} \qquad q[[\emptyset\rangle\rangle q$$

$$\text{(Seq-s)} \qquad \frac{p \xrightarrow{\sigma} q}{p[[1 \cdot \sigma\rangle\rangle q} \qquad \text{con } p \text{ sequenziale}$$

$$\text{(S-Com-s)} \qquad \frac{q_1[[u_1\rangle\rangle q_1' \quad q_2[[u_2\rangle\rangle q_2'}{q_1 \mid q_2[[u\rangle\rangle q_1' \mid q_2'} \qquad MSync(u_1, u_2, \emptyset, u)$$

$$\text{(Res-s)} \qquad \frac{q[[u\rangle\rangle q'}{(\nu a)q[[u\rangle\rangle(\nu a)q'} \qquad \forall k\sigma \in u.a, \overline{a} \notin n(\sigma)$$

$$\text{(Cong-s)} \qquad \frac{q \equiv q_1 \quad q_1[[u\rangle\rangle q_1' \quad q_1' \equiv q'}{q[[u\rangle\rangle q'}$$

Tabella 2.7: Step semantics sugli LTS per il Multi-CCS

Definizioni 2.3.3. *Diciamo che un processo Multi-CCS $q \in \mathcal{P}$ supporta contemporaneamente – sempre – il multiset nullo $\emptyset$, come da regola (Null-s) di tabella 2.7; indichiamo la cosa con $q[[\emptyset\rangle\rangle q$[3].*

[3]l'introduzione di questa regola ci serve per mantenere una sorta di compatibilità con

Diciamo, come da regola (Seq-s) di tabella 2.7, che un processo sequenziale Multi-CCS $p \in \mathcal{P}_{seq}$ *supporta contemporaneamente un multiset di etichette di transizione costituito dal singoletto* $1 \cdot \sigma$ *verso il processo* $q' \in \mathcal{P}$ *– e lo indichiamo con* $p[[1 \cdot \sigma\rangle\rangle q'$ *(o anche, più semplicemente, con* $p[[\sigma\rangle\rangle q'$*, quando è chiaro dal contesto che* σ *rappresenta un multiset singoletto di etichette e non un'etichetta), quando esiste in* $\longrightarrow$ *una transizione tale che* $p \xrightarrow{\sigma} q'$*.*

Diciamo che un processo non sequenziale Multi-CCS $q \in (\mathcal{P} \setminus \mathcal{P}_{seq})$ *supporta contemporaneamente un multiset di etichette di transizione – che indichiamo anche con la lettera* u *o sue variazioni (*u'*,* u''*,* u_1*,* u_a*, ecc.) – verso il processo* $q' \in \mathcal{P}$ *– e lo indichiamo con* $q[[u\rangle\rangle q'$*, quando sono ricorsivamente soddisfatte le regole di tabella 2.7 (ricordando che la regola (Seq-s) è riservata ai processi sequenziali e quindi utilizzabile solo nelle premesse) col supporto delle regole di sincronizzazione di tabella 2.2 (pag. 39) – nelle quali, preciso anche in questo caso, i primi due argomenti di* $Merge$ *sono elementi di* $\mathcal{A} \cup \{\epsilon\}$*, dove con* ϵ *intendiamo una sequenza vuota di azioni, mentre gli argomenti di* $Sync$ *e il terzo argomento di* $Merge$ *sono elementi di* $\mathcal{A}$ *(ovvero: non prevedono sequenze vuote di azioni) – e dalle regole di multi-sincronizzazione definite dalla tabella 2.8.*

$$MSync(u_1, u_2, u_s, u_1 \oplus u_2 \oplus u_s)$$

$$\frac{Sync(\sigma_1, \sigma_2, \sigma) \qquad MSync(u_1, u_2, u_s \oplus \sigma, u)}{MSync(u_1 \oplus \sigma_1, u_2 \oplus \sigma_2, u_s, u)}$$

Tabella 2.8: Regole di sincronizzazione multipla

l'analoga definizione, 1.2.5 (pag. 14), che ho dato sulle reti di Petri; ci torna comunque comoda per semplificare le regole poiché consente di eliminare la necessità delle regole (Par-1-s) e (Par-2-s) dal momento che la loro funzione viene assorbita dalla regola (S-com-s)

Tra le regole riportate in [5] e quelle che propongo, le differenze che credo siano più evidenti sono che

- la regola *MSync* che propongo è più complessa e non consente più di una sincronizzazione per ogni etichetta supportata dalle componenti parallele;
- ho soppresso le regole (Par-1-s) e (Par-2-s) poiché, avendo definito il supporto sulla marcatura nulla, la regola (S-com-s) ne assorbe le funzioni;
- ho aggiunto una regola (Cong-s) che non era originariamente prevista;
- ho soppresso la regola (Cons-s) poiché, con l'introduzione della regola di congruenza strutturale della costante, le sue funzioni sono assorbite dalla regola (Cong-s).

La *MSync* più complessa (e prudente) e l'introduzione della (Cong-s) mi permettono di rispecchiare più fedelmente le regole di transizione di tabella 2.1 (pag. 37) e quindi, in alcune circostanze, si semplificano le dimostrazioni per induzione strutturale come, ad esempio, quella della seguente proposizione. Sono comunque convinto che le regole che qui propongo siano sostanzialmente equivalenti a quelle di [5] e la seguente proposizione 2.3.2 (pag. 53), se non proprio una dimostrazione conclusiva, è un forte indizio in questa direzione.

Vediamo ora una proposizione che ci conferma – analogamente a quanto banalmente osservato sulle reti P/T – che il supporto contemporaneo di un multiset di etichette di transizioni costituito da un singoletto, è equivalente al supporto di una transizione con tale etichetta.

Proposizione 2.3.1. *Siano $q, q' \in \mathcal{P}$ due generici processi Multi-CCS e $\sigma \in \mathcal{A}$ una generica etichetta di transizione; è vero che*

$$q \xrightarrow{\sigma} q' \iff q[[1 \cdot \sigma\rangle\rangle q'$$

Dimostrazione. Per prima cosa ricordiamo che, se q è un processo sequenziale, la tesi corrisponde alla definizione di $q[[1 \cdot \sigma\rangle\rangle q'$, ovvero alla regola (Seq-s) di tabella 2.7 (pag. 46).

Sia $q \xrightarrow{\sigma} q'$ una transizione in $\longrightarrow$; dimostriamo, per casistica sulle transizioni e induzione sulle premesse, che vale $q[[1 \cdot \sigma\rangle\rangle q'$:

- se la transizione con etichetta σ è giustificata dalla regola (Pref), (S-pref), (Sum-1) o (Sum-2), abbiamo che q è un processo sequenziale e, come osservato poc'anzi, $q[[1 \cdot \sigma\rangle\rangle q'$ grazie alla regola (Seq-s);

- se la transizione con etichetta σ è giustificata dalla regola (Par-1), abbiamo che $q = q_1 \mid q_2$ e $q' = q_1' \mid q_2$, per opportuni processi $q_1, q_1', q_2 \in \mathcal{P}$, e che esiste una transizione $q_1 \xrightarrow{\sigma} q_1'$; per ipotesi induttiva su quest'ultima transizione abbiamo che $q_1[[1 \cdot \sigma\rangle\rangle q_1'$ e quindi, tenendo conto che è vero per definizione che $q_2[[\emptyset\rangle\rangle q_2$, applicando la regola (S-com-s) – con $MSync(1 \cdot \sigma, \emptyset, \emptyset, 1 \cdot \sigma)$ – possiamo concludere che $q = q_1 \mid q_2[[1 \cdot \sigma\rangle\rangle q_1' \mid q_2 = q'$;

- se la transizione con etichetta σ è giustificata dalla regola (Par-2), la dimostrazione è analoga a quella del caso precedente ma operando sulla seconda componente parallela;

- se la transizione con etichetta σ è giustificata dalla regola (Com), abbiamo che $q = q_1 \mid q_2$ e $q' = q_1' \mid q_2'$, per opportuni processi $q_1, q_1', q_2, q_2' \in \mathcal{P}$, e che esistono due transizioni $q_1 \xrightarrow{\sigma_1} q_1'$ e $q_2 \xrightarrow{\sigma_2} q_2'$, per opportune etichette $\sigma_1, \sigma_2 \in \mathcal{A}$ tali che $Sync(\sigma_1, \sigma_2, \sigma)$; per ipotesi induttiva su queste ultime due transizioni abbiamo che $q_1[[1 \cdot \sigma_1\rangle\rangle q_1'$ e che $q_2[[1 \cdot \sigma_2\rangle\rangle q_2'$; possiamo quindi applicare la regola (S-com-s) con
$$\frac{Sync(\sigma_1, \sigma_2, \sigma) \qquad MSync(\emptyset, \emptyset, \sigma, \sigma)}{MSync(\sigma_1, \sigma_2, \emptyset, \sigma)}$$
concludendo che $q = q_1 \mid q_2[[1 \cdot \sigma\rangle\rangle q_1' \mid q_2' = q'$;

- se la transizione con etichetta σ è giustificata dalla regola (Res), abbiamo che $q = (\nu a)q_1$ e $q' = (\nu a)q_1'$, per opportuni processi $q_1, q_1' \in \mathcal{P}$

e per un opportuno nome $a \in \mathcal{L}$ tale che $a, \overline{a} \notin n(\sigma)$, e che esiste una transizione $q_1 \xrightarrow{\sigma} q_1'$; per ipotesi induttiva su quest'ultima transizione abbiamo che $q_1[[1 \cdot \sigma\rangle\rangle q_1'$ e, applicando la regola (Res-s), possiamo concludere che $q = (\nu a)q_1[[1 \cdot \sigma\rangle\rangle(\nu a)q_1' = q'$;

- se la transizione con etichetta σ è giustificata dalla regola (Cong), abbiamo che $q \equiv q_1$ e $q_1' \equiv q'$, per opportuni processi $q_1, q_1' \in \mathcal{P}$, e che esiste una transizione $q_1 \xrightarrow{\sigma} q_1'$; per ipotesi induttiva su quest'ultima transizione abbiamo che $q_1[[1 \cdot \sigma\rangle\rangle q_1'$ e, applicando la regola (Cong-s), possiamo concludere che $q[[1 \cdot \sigma\rangle\rangle q'$.

Supponiamo ora che $q[[1 \cdot \sigma\rangle\rangle q'$; dimostriamo l'assunto per casistica sull'ultima regola usata per costruire l'ipotesi che $q \xrightarrow{\sigma} q'$ e per induzione sulle relative premesse:

- non può essere che $q[[1 \cdot \sigma\rangle\rangle q'$ sia giustificato dalla regola (Null-s), poiché $1 \cdot \sigma \neq \emptyset$; la negazione della premessa conferma, in questo caso, l'assunto;

- se $q[[1 \cdot \sigma\rangle\rangle q'$ è giustificato dalla regola (Seq-s), q è necessariamente sequenziale e, dalla premessa della regola, abbiamo proprio $q \xrightarrow{\sigma} q'$;

- se $q[[1 \cdot \sigma\rangle\rangle q'$ è giustificato dalla regola (S-com-s), abbiamo che $q = q_1 \mid q_2$ e $q' = q_1' \mid q_2'$, per opportuni processi $q_1, q_1', q_2, q_2' \in \mathcal{P}$, e che $q_1[[u_1\rangle\rangle q_1'$ e $q_2[[u_2\rangle\rangle q_2'$, per opportuni multiset di etichette di transizioni $u_1, u_2 \in \mathcal{M}_{fin}(\mathcal{A})$ tali che $MSync(u_1, u_2, \emptyset, \sigma)$; dal momento che (come è facile rilevare osservando le regole $MSync$) ogni sincronizzazione porta inevitabilmente nel risultato (il quarto parametro di $MSync$) un singoletto che non può essere rimosso dalle seguenti sincronizzazioni, dal momento che il risultato è costituito da un unico singoletto, abbiamo solo tre casi possibili:

  - quando $u_1 = 1 \cdot \sigma$ e $u_2 = \emptyset$ (ovvero $q_2' = q_2$), dall'ipotesi induttiva su $q_1[[1 \cdot \sigma\rangle\rangle q_1'$ abbiamo che $q_1 \xrightarrow{\sigma} q_1'$ e quindi, applicando la regola (Par-1), abbiamo anche che $q = q_1 \mid q_2 \xrightarrow{\sigma} q_1' \mid q_2 = q_1' \mid q_2' = q'$;

  - quando $u_1 = \emptyset$ (ovvero $q'_1 = q_1$) e $u_2 = 1 \cdot \sigma$, la dimostrazione è analoga a quella del sottocaso precedente ma operando per induzione su q_2;
  - quando $u_1 = 1 \cdot \sigma_1$ e $u_2 = 1 \cdot \sigma_2$, per opportune etichette $\sigma_1, \sigma_2 \in \mathcal{A}$ tali che $Sync(\sigma_1, \sigma_2, \sigma)$, applicando l'ipotesi induttiva a $q_1[[1 \cdot \sigma_1\rangle\rangle q'_1$ e a $q_2[[1 \cdot \sigma_2\rangle\rangle q'_2$ otteniamo $q_1 \xrightarrow{\sigma_1} q'_1$ e $q_2 \xrightarrow{\sigma_2} q'_2$; ricordando che $Sync(\sigma_1, \sigma_2, \sigma)$, possiamo applicare alle ultime due transizioni la regola (Com) ottenendo $q = q_1 \mid q_2 \xrightarrow{\sigma} q'_1 \mid q'_2 = q'$;

- se $q[[1 \cdot \sigma\rangle\rangle q'$ è giustificato dalla regola (Res-s), abbiamo che $q = (\nu a)q_1$ e $q' = (\nu a)q'_1$, per un opportuno nome $a \in \mathcal{L}$, tale che $a, \overline{a} \notin n(\sigma)$, e per opportuni processi $q_1, q'_1 \in \mathcal{P}$, e che $q_1[[1 \cdot \sigma\rangle\rangle q'_1$; applicando a quest'ultimo supporto l'ipotesi induttiva, otteniamo $q_1 \xrightarrow{\sigma} q'_1$; ricordando che $a, \overline{a} \notin n(\sigma)$, possiamo applicare a quest'ultima transizione la regola (Res) ottenendo $q = (\nu a)q_1 \xrightarrow{\sigma} (\nu a)q'_1 = q'$;

- se $q[[1 \cdot \sigma\rangle\rangle q'$ è giustificato dalla regola (Cong-s), abbiamo che $q \equiv q_1$ e $q'_1 \equiv q'$, per opportuni processi $q_1, q'_1 \in \mathcal{P}$, e che $q_1[[1 \cdot \sigma\rangle\rangle q'_1$; applicando a quest'ultimo supporto l'ipotesi induttiva, otteniamo $q_1 \xrightarrow{\sigma} q'_1$ e, applicando a quest'ultima transizione la regola (Cong), otteniamo $q \xrightarrow{\sigma} q'$.

$\square$

$$MSync(u, u) \qquad \frac{Sync(\sigma_1, \sigma_2, \sigma) \quad MSync(u \oplus 1 \cdot \sigma, u')}{MSync(u \oplus 1 \cdot \sigma_1 \oplus 1 \cdot \sigma_2, u')}$$

Tabella 2.9: Regole di sincronizzazione multipla originarie

Ricordiamo che la regola (S-com-s) consente una sola sincronizzazione per ogni etichetta supportata da ogni singola componente e osserviamo che questo è fondamentale – nella precedente dimostrazione – per applicare l'ipotesi induttiva. Se infatti avessimo avuto la regola $MSync$ originaria di [5], ovvero quella di tabella 2.9 (pagina precedente), in un processo come

$$q = (a.\mathbf{0} \mid b.\mathbf{0}) \mid \underline{a}.\overline{b}.\mathbf{0}$$

non avremmo potuto dimostrare, nello stesso modo, che

$$q[[\tau\rangle\rangle(\mathbf{0} \mid \mathbf{0}) \mid \mathbf{0} \implies q \xrightarrow{\tau} (\mathbf{0} \mid \mathbf{0}) \mid \mathbf{0}$$

Infatti, sulle componenti sequenziali abbiamo che $a.\mathbf{0}[[a\rangle\rangle\mathbf{0}$, che $b.\mathbf{0}[[b\rangle\rangle\mathbf{0}$ e che $\underline{a}.\overline{b}.\mathbf{0}[[\overline{a}\overline{b}\rangle\rangle\mathbf{0}$; applicando la regola (S-com-s), con $MSync(a \oplus b, a \oplus b)$, sui primi due supporti avremmo ottenuto $(a.\mathbf{0} \mid b.\mathbf{0})[[a \oplus b\rangle\rangle\mathbf{0} \mid \mathbf{0}$; applicando la regola (S-com-s) a questo risultato e alla terza componente, con

$$\frac{Sync(a, \overline{a}\overline{b}, \overline{b}) \qquad \dfrac{Sync(b, \overline{b}, \tau) \qquad MSync(\tau, \tau)}{MSync(b \oplus \overline{b}, \tau)}}{MSync(a \oplus b \oplus \overline{a}\overline{b}, \tau)}$$

avremmo dimostrato $q[[\tau\rangle\rangle(0 \mid 0) \mid 0$.

Ma, pur in presenza di $q[[\tau\rangle\rangle(0 \mid 0) \mid 0$, dalla prima componente avremmo ottenuto un $(a.\mathbf{0} \mid b.\mathbf{0})[[a \oplus b\rangle\rangle\mathbf{0} \mid \mathbf{0}$ su cui, non essendo $a \oplus b$ un singoletto, non avremmo potuto applicare l'ipotesi induttiva.

Si osservi però che, con le regole che propongo, riusciamo ugualmente a dimostrare che $q[[\tau\rangle\rangle(0 \mid 0) \mid 0$; abbiamo infatti che $q \equiv q'$, dove

$$q' = a.\mathbf{0} \mid (b.\mathbf{0} \mid \underline{a}.\overline{b}.\mathbf{0})$$

e che $(0 \mid 0) \mid 0 \equiv 0 \mid (0 \mid 0)$; è facile vedere che applicando due volte la regola (S-com-s) – prima a $b.\mathbf{0} \mid \underline{a}.\overline{b}.\mathbf{0}$ poi all'intero q' – possiamo ottenere $b.\mathbf{0} \mid \underline{a}.\overline{b}.\mathbf{0}[[\overline{a}\rangle\rangle\mathbf{0} \mid \mathbf{0}$ e $q'[[\tau\rangle\rangle\mathbf{0} \mid (\mathbf{0} \mid \mathbf{0})$; applicando infine la regola (Cong-s), possiamo concludere che $q[[\tau\rangle\rangle(\mathbf{0} \mid \mathbf{0}) \mid \mathbf{0}$.

Quest'ultima osservazione ci può portare a sospettare che la più complessa – e più prudente – $MSync$ che ho introdotto sia equivalente all'originaria presente in [5]; la seguente proposizione lo conferma.

Proposizione 2.3.2. *Siano $q, q' \in \mathcal{P}$ due generici processi Multi-CCS; sia $u \in \mathcal{M}_{fin}(\mathcal{A})$ un generico multiset di etichette di transizioni; siano $\sigma_1, \sigma_2, \sigma \in \mathcal{A}$ tre generiche etichette di transizione; allora è vero che*

$$(Sync(\sigma_1, \sigma_2, \sigma) \wedge q[[u \oplus 1 \cdot \sigma_1 \oplus 1 \cdot \sigma_2\rangle\rangle q') \implies q[[u \oplus 1 \cdot \sigma\rangle\rangle q'$$

Dimostrazione. Non vedremo una dimostrazione dettagliata – anche perché la casistica è decisamente complessa – ma ne vedremo gli elementi principali e le motivazioni che ne stanno alla base.

Quando il supporto non è motivato dalla regola (S-com-s), l'implicazione di dimostra facilmente o per inapplicabilità delle premesse (quando q è sequenziale, e quindi non è in grado di supportare contemporaneamente σ_1 e σ_2) o per induzione sulle premesse delle regole applicate.

Il caso (S-com-s) – ovvero $q = q_a \mid q_b$, con $q_a[[u_a\rangle\rangle q'_a$, con $q_b[[u_b\rangle\rangle q'_b$ e con $q' = q'_a \mid q'_b$ – è più complesso e genera la complessa casistica a cui accennavo; alcuni casi sono semplici e facilmente affrontabili:

- se u_a (o anche u_b) contiene sia σ_1 che σ_2, l'implicazione si dimostra per induzione sulla premessa relativa;

- se u_a contiene σ_1 e u_b contiene σ_2, ovvero $u_a = u'_a \oplus 1 \cdot \sigma_1$ e $u_b = u'_b \oplus 1 \cdot \sigma_2$, (o viceversa), l'implicazione si dimostra banalmente grazie al seguente albero di prova per $MSync$:
$$\frac{Sync(\sigma_1, \sigma_2, \sigma) \qquad MSync(u'_a, u'_b, 1 \cdot \sigma, u'_a \oplus u'_b \oplus 1 \cdot \sigma)}{MSync(u'_a \oplus 1 \cdot \sigma_1, u'_b \oplus 1 \cdot \sigma_2, \emptyset, u'_a \oplus u'_b \oplus 1 \cdot \sigma)}$$

La dimostrazione si complica quando σ_1 o σ_2 o entrambi sono ottenuti dalla sincronizzazione di due etichette presenti in u_a e/o in u_b.

Per prima cosa osserviamo che se q_a e q_b supportano contemporaneamente più di una etichetta di transizione, non possono essere sequenziali ma, al netto di possibili restrizioni o sostituzioni di costanti, sono a loro volta processi paralleli; più formalmente, abbiamo che $q_a, q_b \in \mathcal{P}ar$, dove $\mathcal{P}ar$ è il più piccolo insieme di processi definito ricorsivamente dalle seguenti regole:

- $q \in \mathcal{P}ar$ se $q = q_x \mid q_y$, con $q_x, q_y \in \mathcal{P}$;

- $q \in \mathcal{P}ar$ se $q = (\nu a)q'$, con $a \in \mathcal{L}$ e $q' \in \mathcal{P}ar$;
- $q \in \mathcal{P}ar$ se $q = B$, con $B \overset{def}{=} q'$ e $q' \in \mathcal{P}ar$.

Osserviamo che se $q_a = (\nu c)q'_a$, con $q'_a \in \mathcal{P}ar$:

- quando $c, \overline{c} \notin fn(q_b)$, abbiamo $q = q_a \mid q_b = ((\nu c)q'_a) \mid q_b \equiv (\nu c)(q'_a \mid q_b)$;
- quando $\{c, \overline{c}\} \cap fn(q_b) \cap \emptyset$, e $d \in \mathcal{L}$ è un nome tale che $d, \overline{d} \notin fn(q_a) \cup fn(q_b)$ abbiamo $q = q_a \mid q_b = ((\nu c)q'_a) \mid q_b \equiv ((\nu d)q'_a\{d/c\}) \mid q_b \equiv (\nu d)(q'_a\{d/c\} \mid q_b)$.

Ovvero: quando $q_a = (\nu c)q'_a$, facendo eventualmente uso dell'alpha conversione, possiamo trovare un processo congruente con q nella forma in cui la restrizione abbraccia entrambe le componenti. Al netto di successive applicazioni della regola (Res-s), possiamo quindi ignorare le restrizioni sulla prima (e, ragionando in maniera speculare, anche sulla seconda) componente di q.

Analogamente, possono essere ignorate le costanti, al netto di applicazioni successive della regola (Cong-s), grazie anche all'introduzione delle regola di congruenza delle costanti.

Al netto di successive e ripetute applicazioni delle regole (Res-s) e (Cong-s), possiamo quindi ricondurci al caso in cui q_a e/o q_b (se supportano più transizioni contemporaneamente) sono pari a $q_a = q_{aa} \mid q_{ab}$ e/o $q_b = q_{ba} \mid q_{bb}$.

Osserviamo ora che qualunque etichetta di transizione σ, presente in un multiset di etichette di transizioni contemporaneamente supportate, è giustificata da un albero binario di transizioni le cui foglie sono processi sequenziali; l'unica regola (per così dire) che "genera" una transizione è infatti la definizione del supporto contemporaneo per processi sequenziali; le regole per processi non sequenziali si limitano a trasmetterle o combinarle.

Combinando le ultime osservazioni abbiamo che a ogni transizione σ, contemporaneamente supportata da un processo q, possiamo affermare che (salvo congruenze) q può essere visto come combinazione parallela di più processi, ognuno dei quali supporta una delle transizioni delle foglie dell'albero generato da σ che, ricombinato al contrario, ci porta al risultato voluto.

La casistica è complessa, ma vediamo un caso esplicativo: supponiamo che sia $u_a = u'_a \oplus 1 \cdot \sigma_{a1} \oplus 1 \cdot \sigma_{a2}$ e $u_b = u'_b \oplus 1 \cdot \sigma_{b1} \oplus 1 \cdot \sigma_{b2}$ con $Sync(\sigma_{a1}, \sigma_{b1}, \sigma_1)$ e $Sync(\sigma_{a2}, \sigma_{b2}, \sigma_2)$. Dal momento che abbiamo anche $Sync(\sigma_1, \sigma_2, \sigma)$, dobbiamo supporre che sia presente tra σ_{a1} e σ_{b1} (e analogamente tra σ_{a2} e σ_{b2}) una coppia nome/co-nome (almeno) che permetta la sincronizzazione.

Supponiamo quindi, per fissare le idee, che c e $\overline{c}$ siano la coppia di azioni che permette la sincronizzazione tra σ_{a1} e σ_{b1} in σ_1; analogamente supponiamo che d e $\overline{d}$ siano la coppia di azioni che permette la sincronizzazione tra σ_{a2} e σ_{b2} in σ_2 e che e ed $\overline{e}$ siano la coppia di azioni che permette la sincronizzazione tra σ_1 e σ_2 in σ.

Supponiamo, ancora per fissare le idee, che sia $\sigma_{a1} = c$, $\sigma_{a2} = de$, $\sigma_{b1} = \overline{ce}$, $\sigma_{b2} = \overline{d}$. Ovvero $\sigma_1 = \overline{e}$, $\sigma_2 = e$ e $\sigma = \tau$.

Come abbiamo visto possiamo ricondurci al caso in cui $q_a = q_{aa} \mid q_{ab}$ e $q_b = q_{ba} \mid q_{bb}$

Vediamo ora che possiamo supporre che σ_{a1} sia supportato da q_{aa} e σ_{a2} da q_{ab} (o il contrario).

Se così non fosse, potremmo ricondurci, applicando le regole prima e quarta di congruenza strutturale, al caso appena descritto; supponiamo ad esempio che sia σ_{a1} che σ_{a2} siano supportate da q_{ab}; abbiamo:

$$q = q_a \mid q_b = (q_{aa} \mid q_{ab}) \mid (q_{ba} \mid q_{bb}) \equiv q_{aa} \mid (q_{ab} \mid (q_{ba} \mid q_{bb}))$$

ovvero σ_1 e σ_2 supportati da $q_{ab} \mid q_b$; possiamo quindi applicare il ragionamento (q_{ab} pari (salvo applicazione di altre regole) congruente a due processi paralleli); nel caso invece sia σ_{a1} che σ_{b1} sono supportate da q_{ab} abbiamo:

$$\begin{aligned} q = q_a \mid q_b = (q_{aa} \mid q_{ab}) \mid (q_{ba} \mid q_{bb}) &\equiv q_{aa} \mid (q_{ab} \mid (q_{ba} \mid q_{bb})) \\ &\equiv q_{aa} \mid ((q_{ab} \mid q_{ba}) \mid q_{bb}) \end{aligned}$$

e possiamo ricondurci al caso $q_{aa} \mid q'_b$, con $q'_b = (q_{ab} \mid q_{ba}) \mid q_{bb}$ che ha assorbito l'innocuo (ai fini della presente dimostrazione) termine q_{ab}.

Analogamente si può supporre che che σ_{a2} sia supportato da q_{ba} e σ_{b2} da q_{bb}, salvo congruenze.

Abbiamo quindi

$$
\begin{aligned}
q = q_a \mid q_b = (q_{aa} \mid q_{ab}) \mid (q_{ba} \mid q_{bb}) &\equiv q_{aa} \mid (q_{ab} \mid (q_{ba} \mid q_{bb})) \\
&\equiv q_{aa} \mid ((q_{ab} \mid q_{ba}) \mid q_{bb})
\end{aligned}
$$

Possiamo quindi operare la sincronizzazione tra $\sigma_{b1} = de$ e $\sigma_{a2} = \overline{ce}$ in $q_{ab} \mid q_{ba}$ ottenendo questo termine supporta σ_{ab1} pari a $\bar{c}d$ (o pari a $d\bar{c}$: la sostanza non cambia).

Sincronizzando poi $\sigma_{ab1} = \bar{c}d$ con $\sigma_{b2} = \overline{d}$ in $(q_{ab} \mid q_{ba}) \mid q_{bb}$ otteniamo che questo termine supporta $\sigma_{ab2} = \bar{c}$.

Sincronizzando infine $\sigma_{a1} = c$ con $\sigma_{ab2} = \bar{c}$ in $q_{aa} \mid ((q_{ab} \mid q_{ba}) \mid q_{bb})$ otteniamo che questo termine (congruente al q originario) supporta $\sigma = \tau$. Ovvero: il risultato ipotizzato.

Gli altri casi si trattano in maniera analoga, facendo l'opportuno uso della congruenza strutturale per costruire un opportuno termine congruente che sia in grado di sincronizzare le etichette di partenza nel termine σ desiderato. $\square$

Grazie alla proposizione appena dimostrata, possiamo concludere che le regole (S-com-s), (Res-s) e $MSync$ che ho introdotto in questo capitolo, sono equivalenti alle regole (S-com-s) e $MSync$ di [5].

2.4 Esempi

Vediamo ora alcuni esempi concreti e significativi di processi Multi-CCS.

2.4.1 Sincronizzazione di più di due processi

Con le regole di transizione descritte in tabella 2.1 (pag. 37) è possibile sincronizzare, con un'unica transizione, non solo due processi – come già con il CCS – ma un generico numero $n \geqslant 2$ di processi.

Vediamo un semplice esempio di sincronizzazione per tre processi; sia $a \in \mathcal{L}$ il nome di un'azione e siano $q_1, q_2 \in \mathcal{P}$ due processi Multi-CCS definiti come segue

$$q_1 = \underline{a}.a.\mathbf{0}$$
$$q_2 = \overline{a}.\mathbf{0}$$

Osserviamo che il Multi-CCS consente (unicamente, al netto delle possibili congruenze) la seguente transizione

$$(\nu a)((q_1 \mid q_2) \mid q_2) \xrightarrow{\tau} (\mathbf{0} \mid \mathbf{0}) \mid \mathbf{0}$$

Infatti il termine q_1, può effettuare – applicando le regole (S-pref) e (Pref) – una transizione con etichetta aa; i termini q_2 possono effettuare transizioni – applicando la regola (Pref) – con etichetta $\overline{a}$; applicando la regola (Com), il termine q_1 e il termine q_2 più a sinistra possono sincronizzarsi ottenendo una transizione che etichetta a; quest'ultima transizione può sincronizzarsi con la transizione del secondo q_2 ottenendo la transizione con etichetta τ di cui sopra. La restrizione impone che la transizione τ sia l'unica possibile, occupandosi di filtrare le restrizioni con etichette basate su a.

Questo esempio può facilmente essere generalizzato per mostrare che un generico numero n (con $n \geqslant 2$) di processi può sincronizzarsi in un'unica transizione atomica: è sufficiente che q_1 sia una sequenza di $n-2$ azioni a fortemente prefisse seguite da un'unica azione a normalmente prefissa; il processo q_2 rimane inalterato ma q_1 viene messo in parallelo con $n-1$ processi q_2. In questo modo q_1 può sincronizzare ognuna delle proprie $n-1$ azioni a con l'azione $\overline{a}$ di uno degli $n-1$ processi q_2 ottenendo un'azione τ.

Osserviamo anche che per la transizione

$$(\nu a)(q_1 \mid (q_2 \mid q_2)) \xrightarrow{\tau} \mathbf{0} \mid (\mathbf{0} \mid \mathbf{0})$$

non sarebbe possibile in assenza della regola (Cong); infatti i due termini q_2 non sarebbero in grado di sincronizzarsi e non sarebbero quindi in grado di sincronizzarsi con q_1; la regola (Cong) consente la transizione di cui sopra osservando che

$$(q_1 \mid q_2) \mid q_2 \equiv q_1 \mid (q_2 \mid q_2) \qquad (\mathbf{0} \mid \mathbf{0}) \mid \mathbf{0} \equiv \mathbf{0} \mid (\mathbf{0} \mid \mathbf{0})$$

2.4.2 Sincronizzazione multipla tra due processi

Nell'esempio precedente abbiamo visto che $n \geqslant 2$ processi possono sincronizzarsi in un'unica transizione atomica. Vediamo ora che due processi possono sincronizzarsi su $n \geqslant 1$ azioni, sempre in un'unica transizione atomica.

L'esempio è banale; sia $a \in \mathcal{L}$ il nome di un'azione e siano $q_1, q_2 \in \mathcal{P}$ due processi Multi-CCS definiti come segue

$$q_1 = \underline{a}.\underline{a}.a.\mathbf{0}$$
$$q_2 = \underline{\overline{a}}.\underline{\overline{a}}.\overline{a}.\mathbf{0}$$

Osserviamo che il Multi-CCS consente (unicamente) la seguente transizione

$$(\nu a)(q_1 \mid q_2) \xrightarrow{\tau} \mathbf{0} \mid \mathbf{0}$$

È infatti evidente che q_1 e q_2 – applicando due volte la regola (S-pref) e una volta la (Pref) – possono effettuare, rispettivamente, transizioni con etichetta aaa e $\overline{aaa}$; come mostra il seguente albero di prova

$$\cfrac{\cfrac{Sync(a, \overline{a}, \tau)}{Sync(aa, \overline{aa}, \tau)}}{Sync(aaa, \overline{aaa}, \tau)}$$

le due transizioni si possono sincronizzare in un'unica transizione con etichetta τ; la restrizione si occupa di "filtrare" le due transizioni con etichette basate su a.

Anche in questo caso, modificare l'esempio per permettere un generico numero $n \geqslant 1$ di transizioni è molto semplice: è sufficiente che q_1 sia costituita da $n-1$ azioni a fortemente prefisse seguite da un'unica azione a normalmente prefissa; q_2 deve corrispondere a q_1 ma con le azioni complementari.

Capitolo 3

Una variante della semantica sulle reti P/T per il Multi-CCS

Già in [6] e in [7] è presente una semantica che permette di associare una rete P/T non limitata all'insieme di tutti i termini Multi-CCS e, usando la marcatura associata a un singolo termine Multi-CCS come marcatura iniziale, permette di costruire dei sistemi P/T. Tale semantica – che riporto brevemente e commento nella prima sezione – oltre a presentare alcuni problemi non descrive nel dettaglio un aspetto importante, ovvero non descrive come vengono determinate le azioni ristrette sostitutive ma si affida a una sorta di "oracolo" nella funzione di decomposizione. Questa libertà viene sfruttata in alcune dimostrazioni che però lasciando il dubbio se, in presenza di un algoritmo "rigido" per la determinazioni delle azioni ristrette, gli stessi risultati continuino a valere o meno.

Presento quindi, in questo capitolo, una nuova semantica – una variante di quella descritta in [6] e in [7] – in cui esplicito un algoritmo per determinare le azioni ristrette.

Ho inoltre introdotto, a costo di una complicazione che però ricade principalmente sulle dimostrazioni dei risultati, quello che ritengo sia un miglioramento – per altro conseguito approfondendo un'ipotesi suggerita, per futuri sviluppi, proprio in [7] – che consente di ottenere sistemi P/T finiti in corrispondenza di termini Multi-CCS che, con la semantica originale, avrebbero generato sistemi P/T infiniti.

3.1 Versione originaria

Introdotti un (non meglio definito) insieme numerabile $\mathcal{N}$ di nuovi nomi per azioni ristrette e il relativo insieme di co-nomi $\overline{\mathcal{N}}$, definito l'insieme $\mathcal{P}^{\mathcal{N}}$ dei "processi estesi" Multi-CCS – in cui le azioni possono appartenere non ad $\mathcal{A}ct$ ma ad $\mathcal{A}ct'$ (ovvero $\mathcal{A}ct \cup \mathcal{N} \cup \overline{\mathcal{N}}$) e definito S_{MCCS} come pari all'insieme $\mathcal{P}^{\mathcal{N}}_{seq}$, ovvero al sottoinsieme di $\mathcal{P}^{\mathcal{N}}$ costituito dai soli processi sequenziali[1], in [6] e in [7] si costruisce l'originaria semantica per reti P/T basandosi su una funzione di decomposizione *dec* che associa a un termine Multi-CCS (esteso) una marcatura in $\mathcal{M}_{fin}(S_{MCCS})$.

Vediamo ora la funzione *dec* originaria e le corrispondenti regole di transizione. Vedremo poi alcuni limiti e problemi che hanno motivato la costruzione della nuova variante.

3.1.1 Definizione di *dec* ed esempi

L'originaria funzione di decomposizione $dec : \mathcal{P}^{\mathcal{N}} \to \mathcal{M}_{fin}(S_{MCCS})$ è definita induttivamente come segue

$$dec(\mathbf{0}) = \emptyset \tag{3.1a}$$

$$dec(\mu.q) = \{\mu.q\} \tag{3.1b}$$

$$dec(\underline{\mu}.q) = \{\underline{\mu}.q\} \tag{3.1c}$$

$$dec(p + p') = \{p + p'\} \tag{3.1d}$$

$$dec(q \mid q') = dec(q) \oplus dec(q') \tag{3.1e}$$

$$dec((\nu a)q) = dec(q\{a'/a\}) \qquad \text{dove } a' \in \mathcal{N} \text{ è una nuova azione ristretta} \tag{3.1f}$$

[1]tornerò su queste definizioni, in maniera più formale, nelle prossime sezioni

$$dec(C) = dec(p) \qquad\qquad \text{quando } C \stackrel{def}{=} p \tag{3.1g}$$

L'equazione più problematica è la (3.1f) in quanto introduce un nuovo elemento, un nuovo nome ristretto $a' \in \mathcal{N}$ (dove, come accennavo prima, $\mathcal{N} \cup \overline{\mathcal{N}}$ è un nuovo insieme numerabile, non meglio definito, di azioni – nomi più co-nomi – ristrette), che rischia (in presenza di definizioni ricorsive), di generare un numero infinito di nuove azioni e di nuovi posti.

Non sempre questa generazione di infinite nuove azioni ristrette (e conseguenti nuovi posti) è necessaria; in [7] è riportato il seguente esempio

$$A \stackrel{def}{=} (\nu b)(a.A \mid b.A) \tag{3.2}$$

Le infinite sostituzioni di b (a causa della ricorsione della definizione di A) si applicano sempre e solo al nome e mai al suo co-nome; trattandosi di azioni ristrette, siamo sicuri che non verranno mai eseguite. Non è quindi necessario utilizzare infiniti nuovi simboli: sarebbe sufficiente utilizzare un unico simbolo - b', in ipotesi - per sostituire b ad ogni iterazione della sostituzione. In questo modo potremmo ottenere una $Net(A)$ con numero finito di posti mentre con l'applicazione della funzione dec descritta sopra la $Net(A)$ è infinita.

La semplificazione ipotizzata non può però essere applicata sempre; ancora da [7] abbiamo un controesempio:

$$C \stackrel{def}{=} (\nu b)(a.C \mid (b.c.\mathbf{0} + \bar{b}.\mathbf{0})) \tag{3.3}$$

In questo caso, l'azione sostitutiva di b è presente sia in maniera diretta che complementare e ad ogni iterazione della definizione ricorsiva è necessario introdurre un nuovo nome ristretto per sostituire b. Se, erroneamente, ad ogni iterazione si sostituisse b con un unico nome ristretto b', le componenti $b'.c.\mathbf{0}+\overline{b'}.\mathbf{0}$ di diversi livelli di ricorsione potrebbero sincronizzarsi, abilitando una transizione con etichetta τ, e abilitando in seguito una transizione con etichetta c, che invece non devono essere abilitate.

Si deve anche tener presente che i nomi ristretti che sono potenzialmente riciclabili per sostituire nomi che hanno uno stessa tipologia di utilizzo, non possono essere più riciclati per sostituire nomi di tipologia differente. Per fare un esempio, se alla definizione di cui sopra di A aggiungiamo la seguente

$$B \stackrel{def}{=} (\nu b)(a.B \mid \bar{b}.B) \tag{3.4}$$

e se li mettiamo in parallelo

$$D \stackrel{def}{=} A \mid B \tag{3.5}$$

rimarrà vero che sia A che B possono utilizzare un unico nome ristretto nelle proprie sostituzioni di b, ma il nome utilizzato da A, per evitare un'indesiderata sincronizzazione, dovrà essere diverso da quello utilizzato da B.

3.1.2 Regole di transizione

$$\text{(pref)}\ \{\mu.q\} \xrightarrow{\mu} dec(q) \qquad \text{(sum-1)}\ \frac{\{p\} \xrightarrow{\sigma} m'}{\{p + p'\} \xrightarrow{\sigma} m'}$$

$$\text{(s-pref)}\ \frac{m_1 \xrightarrow{\sigma} m_1'}{\{\underline{\mu}.q\} \xrightarrow{\mu\sigma} m_1' \oplus m_2} \quad m_1 \oplus m_2 = dec(q)$$

$$\text{(com)}\ \frac{m_1 \xrightarrow{\sigma_1} m_1' \quad m_2 \xrightarrow{\sigma_2} m_2'}{m_1 \oplus m_2 \xrightarrow{\sigma} m_1' \oplus m_2'} \quad sync(\sigma_1, \sigma_2, \sigma)$$

Tabella 3.1: Regole originarie per le transizioni in $\mathcal{M}_{fin}(S_{MCCS})$

In tabella 3.1 abbiamo le originarie regole di transizione riportate in [6] e in [7][2]. La regola (sum-2) è intuibile, data la regola (sum-1); le regole di sincronizzazione *sync* sono le medesime del Multi-CCS con le opportune integrazioni dovute all'aggiunta dei nomi ristretti in $\mathcal{N}$.

[2]per analogia con [6] e [7], indico le regole di transizione nella semantica LTS del Multi-CCS con iniziale maiuscola mentre (sia ora che in seguito) indico con iniziale minuscola le regole di transizione definite nella semantica P/T; quindi, per fare un esempio, (Com)

3.1.3 Limiti e problemi

Un limite della precedente definizione della funzione *dec* è che non fornisce un algoritmo concreto per determinare un nuovo nome ristretto in caso di restrizione. L'equazione 3.1f si limita riferirsi a "*a new restricted action*". Questa estrema libertà viene anche usata in [7] per dimostrare alcuni risultati interessanti con ragionamenti del tipo "*it is possible to choose new restricted names during decomposition so that [...]*"[3], "*under the assumption that the choice of the new name is not fixed by a rule, but can be made as necessary, function dec induces some structural axioms on Multi-CCS processes*" o "*commutativity holds because it is possible to choose the new names as needed*"[4]. C'è da domandarsi se gli stessi risultati si possono dimostrare in presenza di una regola precisa per la scelta dei nuovi nomi.

Il seguente esempio mostra però un problema concreto, relativo alla funzione *dec* e alle conseguenti regole di transizione. Sia $q \in \mathcal{P}$ come segue

$$\begin{aligned} q &= a.B \mid a.B \\ B &\stackrel{def}{=} (\nu b)(b.\mathbf{0} + \overline{b}.\mathbf{0}) \end{aligned} \tag{3.6}$$

Applicando la *dec* a q otterremo

$$\begin{aligned} dec(q) &= dec(a.B \mid a.B) = dec(a.B) \oplus dec(a.B) \\ &= \{a.B\} \oplus \{a.B\} = 2 \cdot \{a.B\} \end{aligned}$$

ovvero un unico posto $\{a.B\}$ con due token.

La regola (pref) di tabella 3.1 ci indica la transizione

$$\{a.B\} \xrightarrow{a} dec(B)$$

dove

$$dec(B) = dec((\nu b)(b.\mathbf{0} + \overline{b}.\mathbf{0})) = dec(b'.\mathbf{0} + \overline{b'}.\mathbf{0}) = \{b'.\mathbf{0} + \overline{b'}.\mathbf{0}\}$$

indica una regola di transizione nella semantica LTS mentre (com) indica la corrispondente regola nella semantica P/T; analogamente, con *Sync* indico le regole di sincronizzazione nella semantica LTS mentre con *sync* indico le corrispondenti regole nella semantica P/T

[3] dalla dimostrazione del Lemma 2, Appendice B

[4] le ultime due citazioni dalla sottosezione 4.1

dove b' è una nuova azione ristretta.

Ma se inizialmente abbiamo 2 token in $\{a.B\}$, dopo due transizioni con etichetta a avremo due token in $\{b'.\mathbf{0}+\overline{b'}.\mathbf{0}\}$ e questi – grazie alle regole (sum-1), (sum-2) e (com) – potranno sincronizzarsi generando un'azione τ che non è ottenibile nella semantica LTS. Abbiamo quindi perso la bisimilitudine tra q (visto nella semantica di default, ovvero quella LTS) e $dec(q)$. Il problema (l'errore delle regole originarie) è quello di considerare uguali i due termini in parallelo (i due $a.B$) senza rilevare che nella costante B del termine di sinistra è di fatto presente un'azione ristretta da sostituire che deve essere diversa dall'azione ristretta da sostituire nella costante B del termine di destra. In pratica, i due termini in parallelo $a.B$ devono essere considerati diversi poiché – di fatto – il contenuto della costante B li rende diversi.

È quindi opportuno costruire una dec che possa distinguere i due $a.B$ (ma che non li distingua quando non è necessario, come quando B non fa riferimento a restrizioni) e delle regole di transizione in grado di tener conto di questa distinzione.

3.2 Nuova versione

Come abbiamo visto nella sezione precedente, è necessario correggere la funzione dec perché distingua (ma solo quando necessario) posti che la precedente dec considerava erroneamente identici.

Abbiamo anche visto che sarebbe opportuno modificare dec anche per ottenere sistemi P/T finiti in circostanze in cui la dec originaria avrebbe originato sistemi infiniti.

Ho cercato di ottenere questi risultati contestualmente alla precisazione della composizione dell'insieme dei nomi ristretti $\mathcal{N}$ e alla costruzione di una regola esplicita per la generazione dei nuovi nomi. Questo ha richiesto – nella soluzione che ho adottato – l'introduzione di due ulteriori parametri per la dec e la conseguente modifica delle regole di transizione. Questi due nuovi parametri hanno introdotto una ovvia complicazione – evidente nella dimostrazione dei risultati analoghi a quelli di [7] – che però ritengo fosse solo apparentemente assente in origine, mascherata dall'estrema libertà di scelta dei nuovi nomi.

Presento quindi la formalizzazione di una nuova *dec* e delle, conseguenti, nuove regole di transizione.

3.2.1 Nuova definizione delle azioni ristrette $\mathcal{N}$

In [7], come già in [6], l'insieme $\mathcal{N}$ delle possibili azioni ristrette non è definito se non come "*denumerable*" (numerabile). Provvediamo ora a dare una definizione più puntuale.

Definizioni 3.2.1. *Sia η un simbolo tale che sia vera la condizione seguente*

$$(\eta_d \notin \mathcal{L}) \wedge (\eta_c \notin \mathcal{L}) \wedge (\forall i \in \mathbb{N}.\eta_i, \widehat{\eta_i} \notin \mathcal{L}) \tag{3.7}$$

Definiamo primariamente l'insieme dei nomi ristretti significativi

$$\mathcal{N}_{\mathbb{N}} = \{\eta_i \mid i \in \mathbb{N}\}$$

e l'insieme dei supporti ristretti

$$\widehat{\mathcal{N}_{\mathbb{N}}} = \{\widehat{\eta_i} \mid i \in \mathbb{N}\}$$

Definiamo poi i nomi η_c e η_d come, rispettivamente, nome ristretto banale diretto *e* nome ristretto banale complementato*; insieme li chiamiamo* nomi ristretti banali.

Definiamo quindi l'insieme delle nomi ristretti $\mathcal{N}$ *come*

$$\mathcal{N} = \{\eta_d\} \cup \{\eta_c\} \cup \mathcal{N}_{\mathbb{N}}$$

Definiamo l'insieme dei co-nomi ristretti *come*

$$\overline{\mathcal{N}} = \{\overline{\eta_d}\} \cup \{\overline{\eta_c}\} \cup \{\overline{\eta_i} \mid i \in \mathbb{N}\}$$

Definiamo gli elementi dell'insieme $\mathcal{N} \cup \overline{\mathcal{N}}$ azioni ristrette *e indichiamo i suoi generici elementi con δ e variazioni (δ', δ'', δ_1, δ_2, ecc.); definiamo le azioni η_c, η_d, $\overline{\eta_c}$ e $\overline{\eta_d}$* azioni ristrette banali*; definiamo le azioni η_i e $\overline{\eta_i}$, con $i \in \mathbb{N}$,* azioni ristrette non banali.

L'insieme $\mathcal{N} \cup \overline{\mathcal{N}}$ così definito è, ovviamente, numerabile.

3.2.2 Processi estesi, azioni estese e posti

Come in [6] e in [7], definiamo gli insiemi dei processi estesi, dei processi estesi sequenziali e delle azioni estese. Definiamo anche – ma in maniera differente da [6] e da [7] – l'insieme S_{MCCS} dei posti, che corrisponderanno ai posti della rete e dei sistemi P/T associati al Multi-CCS.

Definizioni 3.2.2. *Definiamo l'insieme delle* azioni estese *come* $\mathcal{A}ct' = \mathcal{A}ct \cup \mathcal{N} \cup \overline{\mathcal{N}}$*; cosi come per l'insieme delle azioni* $\mathcal{A}ct$*, indichiamo i suoi generici elementi con* μ *e sue variazioni.*

Definiamo l'insieme dei processi estesi $\mathcal{P}^{\mathcal{N}}$ *come l'insieme dei processi ottenuti dalla stessa grammatica e dagli stessi vincoli, di definizione 2.1.2 (pag. 28), che generano l'insieme dei processi* $\mathcal{P}$ *ma con la variante che i processi normalmente e fortemente prefissi sono basati su azioni estese* $\mathcal{A}ct'$ *e non solo su azioni* $\mathcal{A}ct$*.*

Definiamo $\mathcal{P}^{\mathcal{N}}_{seq}$ *come l'insieme dei* processi sequenziali estesi *ottenuti dalla stessa grammatica e dagli stessi vincoli di definizione 2.1.2 usando azioni estese* $\mathcal{A}ct'$*.*

Chiamiamo posti *gli elementi di* $S_{MCCS} = \mathcal{P}^{\mathcal{N}}_{seq} \times ((\widehat{\mathcal{N}_{\mathbb{N}}} \times \mathcal{N}_{\mathbb{N}}) \cup \{\epsilon\})$*, dove* ϵ *è un valore speciale che chiamiamo* identificativo extra nullo.

Osserviamo che la definizione dei posti – e del relativo insieme S_{MCCS} – si è complicato: non sono più semplicemente processi estesi sequenziali (come nella versione originaria di [6] e [7]) ma fanno coppia con un termine aggiuntivo costituito o da ϵ o da una coppia di valori ristretti (un supporto ristretto significativo o un nome ristretto significativo).

Notazioni 3.2.3. *Indicheremo i valori* $(\widehat{\eta_i}, \eta_j) \in \widehat{\mathcal{N}_{\mathbb{N}}} \times \mathcal{N}_{\mathbb{N}}$ *con* $[\widehat{\eta_i}, \eta_j]$*.*

Indicheremo gli elementi di $S_{MCCS} = \mathcal{P}^{\mathcal{N}}_{seq} \times ((\widehat{\mathcal{N}_{\mathbb{N}}} \times \mathcal{N}_{\mathbb{N}}) \cup \{\epsilon\})$ *come*

$$\{p\}$$

quando abbiamo a che fare con $(p, \epsilon) \in \mathcal{P}^{\mathcal{N}}_{seq} \times \{\epsilon\}$*, o come*

$$\{p\}[\widehat{\eta_i}, \eta_j]$$

quando abbiamo a che fare con $(p, (\widehat{\eta_i}, \eta_j)) \in \mathcal{P}^{\mathcal{N}}_{seq} \times (\widehat{\mathcal{N}_{\mathbb{N}}} \times \mathcal{N}_{\mathbb{N}})$*.*

3.2.3 Nomi liberi, nomi legati, nomi e costanti di un processo esteso

L'esempio 3.2 (pag. 61) ci porta a domandarci quando è possibile riutilizzare un nome ristretto già utilizzato.

Il punto è capire se il nome da sostituire ha un utilizzo solo diretto, solo complementato, entrambi o nessuno; nel primi due casi, il nome sostituente può essere sempre lo stesso (uno per gli usi esclusivamente diretti e uno per quelli esclusivamente complementati); nel caso di entrambi gli utilizzi, invece, il nome sostituente non potrà (in generale) essere riutilizzato. Se il nome non è utilizzato per nulla, la sostituzione può persino essere ignorata.

Per prima cosa dobbiamo quindi riuscire a capire quali usi ha il nome da sostituire. A questo scopo – così come, per i termini $q \in \mathcal{P}$ del Multi-CCS – definiamo i "nomi liberi" di un processo esteso $q \in \mathcal{P}^{\mathcal{N}}$ e, per completezza, i "nomi legati" e i "nomi". Definiamo anche le "costanti" di un processo esteso.

Definizione 3.2.4. *Sia $q \in \mathcal{P}^{\mathcal{N}}$; l'insieme $fn(q)$ dei* nomi liberi *(*free names*) in q è definito come $fn(q) = F(q, \emptyset)$ dove*

$$F : \mathcal{P}^{\mathcal{N}} \times \mathcal{P}(\mathcal{C}) \to \mathcal{P}(\mathcal{A}ct')$$

è la definita induttivamente in maniera identica all'analoga F della definizione 2.2.1 (pag. 31).

Definizione 3.2.5. *Sia $q \in \mathcal{P}^{\mathcal{N}}$; l'insieme $bn(q)$ dei* nomi legati *(*bound names*) in q è definito come $bn(q) = B(q, \emptyset)$ dove*

$$B : \mathcal{P}^{\mathcal{N}} \times \mathcal{P}(\mathcal{C}) \to \mathcal{P}(\mathcal{A}ct')$$

è la definita induttivamente in maniera identica all'analoga B della definizione 2.2.2 (pag. 32).

Definizione 3.2.6. *Sia $q \in \mathcal{P}^{\mathcal{N}}$; l'insieme $n(q)$ dei* nomi *(*names*) in q è definito come $n(q) = fn(q) \cup bn(q)$.*

Definizione 3.2.7. *Sia* $q \in \mathcal{P}^{\mathcal{N}}$*; l'insieme* $c(q)$ *delle* costanti *di* q *è definito come* $c(q) = C(q, \emptyset)$ *dove*

$$C : \mathcal{P}^{\mathcal{N}} \times \mathcal{P}(\mathcal{C}) \to \mathcal{P}(\mathcal{C})$$

è la definita induttivamente in maniera identica all'analoga C *della definizione 2.2.4 (pag. 32).*

Osserviamo esplicitamente che, anche per i processi estesi, continuano a valere le relazioni rilevate nell'osservazione 2.2.1 (pag. 33).

Data la definizione di "nomi liberi", prima di sostituire un nome $a \in \mathcal{L}$ in un processo q con un nome ristretto, sarà necessario stabilire il valore di $\{a, \overline{a}\} \cap fn(q)$:

- se sarà pari a $\{a, \overline{a}\}$, sapremo che il nome sostituente verrà introdotto sia per un uso diretto che per un uso complementato;
- se sarà pari a $\{a\}$, sapremo che l'uso del nome sostituente sarà solo diretto;
- se sarà pari a $\{\overline{a}\}$, sapremo che l'uso del nome sostituente sarà solo complementato;
- se sarà pari a $\emptyset$, sapremo di avere di fronte una sostituzione degenere in cui le azioni sostituenti non andranno mai a sostituire il nome a o il suo complementare.

Si noti che il meccanismo descritto potrebbe essere migliorato: si basa sul considerare eseguibili le azioni fortemente prefisse di tipo $\underline{\mu}.p$ senza domandarsi se possono o meno essere completate da un'azione in p. Quindi restituisce, per fare un esempio,

$$\{a, \overline{a}\} \cap fn(\underline{a}.\mathbf{0}) = \{a\}$$

quando sarebbe invece preferibile che restituisse $\emptyset$, dal momento che $\underline{a}.\mathbf{0}$ non potrà mai eseguire l'azione a. Rischiamo però solo di avere dei falsi positivi che ci porteranno a considerare come non riciclabili delle azioni che invece potrebbero essere considerate tali, ovvero avremo un risultato sub-ottimale ma questo difetto non introduce nessun errore.

3.2.4 Identificativo extra per i posti

Prima di ridefinire *dec* è opportuno definire una funzione di supporto *eid* (*extra identifier*) il cui scopo è – dato un termine $\mathcal{P}^{\mathcal{N}}$ e i medesimi valori numerici usati per la *dec* – restituire l'opportuno valore in $(\widehat{\mathcal{N}_{\mathbb{N}}} \times \mathcal{N}_{\mathbb{N}}) \cup \{\epsilon\}$, ovvero la componente extra (rispetto alle originarie definizioni) dei posti della nostra rete P/T.

Questo ci sarà utile per distinguere, quando necessario[5], termini apparentemente identici come, per esempio, i due sottotermini $a.B$ nell'esempio 3.6 (pag. 63).

Definizione 3.2.8. *Definiamo la funzione*

$$eid : (\mathcal{P}^{\mathcal{N}} \times \mathbb{N} \times \mathbb{N} \times \mathcal{P}(\mathcal{C})) \to ((\widehat{\mathcal{N}_{\mathbb{N}}} \times \mathcal{N}_{\mathbb{N}}) \cup \{\epsilon\})$$

induttivamente come segue

$$\begin{aligned}
eid(\mathbf{0}, e, i, A) &= \epsilon \\
eid(\mu.q, e, i, A) &= eid(q, e, i, A) \\
eid(\underline{\mu}.q, e, i, A) &= eid(q, e, i, A) \\
eid(p_1 + p_2, e, i, A) &= \begin{cases} eid(p_1, e+1, i, A) & \textit{se } eid(p_1, e+1, i, A) \neq \epsilon \\ eid(p_2, e+1, i+2^e, A) & \textit{altrimenti} \end{cases} \\
eid(q_1 \mid q_2, e, i, A) &= \begin{cases} eid(q_1, e+1, i, A) & \textit{se } eid(q_1, e+1, i, A) \neq \epsilon \\ eid(q_2, e+1, i+2^e, A) & \textit{altrimenti} \end{cases} \\
eid((\nu a)q, e, i, A) &= \begin{cases} [\widehat{\eta}_e, \eta_i] & \textit{se } a, \overline{a} \in fn(q) \\ \epsilon & \textit{altrimenti} \end{cases}
\end{aligned}$$

[5] ovvero, quando il valore risultante sarà diverso da ϵ

$$eid(C, e, i, A) = \begin{cases} \epsilon & \textit{se } C \in A \\ eid(q, e, i, A \cup \{C\}) & \textit{altrimenti, con } C \stackrel{def}{=} q \end{cases}$$

L'idea è quindi quella di analizzare in termine, con il limite della doppia sostituzione delle costanti, per verificare se verrà effettuata almeno una sostituzione con un nome ristretto non banale. Se la risposta è no (nessuna sostituzione significativa) la funzione restituisce l'identificativo nullo (ϵ), altrimenti restituisce un identificativo definito dai parametri di supporto, e ed i, al momento della sostituzione significativa: $[\widehat{\eta_e}, \eta_i]$[6].

Osserviamo che gli identificativi non nulli ottenuti da *eid* contengono un nome ristretto (il secondo elemento) che corrisponde a una coppia di azioni ristrette reciprocamente complementari in qualche modo associate al termine; contengono però anche un supporto ristretto (il primo elemento) che è solamente un'etichetta, per distinguere tra loro termini diversi, ma non corrisponde a un'azione ristretta effettivamente usata. Ovvero: se l'extra identifier corrisponde, per fare un esempio, a $[\widehat{\eta_1}, \eta_7]$, il termine associato farà uso (direttamente o indirettamente) di entrambe le azioni ristrette η_7 e $\overline{\eta_7}$ ma non è detto che faccia uso di η_1 o di $\overline{\eta_1}$ in quanto (come vedremo, fra poco, in corrispondenza della ridefinizione della funzione *dec*) il valore 1 è semplicemente il valore dell'esponente del passo nel momento della sostituzione di un nome visibile con η_7.

In presenza di più sostituzioni significative, l'algoritmo ne sceglie deterministicamente una ma ognuna di queste potrebbe essere utilizzata per costituire un valido identificativo univoco.

Vediamo ora alcuni risultati, relativi a *eid*, che ci saranno utili in seguito.

Per prima cosa dimostriamo una cosa intuitivamente evidente: che l'aggiunta, al quarto parametro di *eid*, di un insieme che ha intersezione vuota con l'insieme delle costanti del primo è ininfluente.

[6] la più intuitiva sostituzione con $[e, i]$ avrebbe potuto portare ad ambiguità, nel caso e o i fossero anche etichette di azioni visibili, in caso di sostituzione del tipo $dec(p, e, i)\{\eta/a\}$, con a generica azione visibile; con la sostituzione descritta, grazie alla condizione di equazione 3.7 (pag. 65) nella definizione delle azioni ristrette, questo rischio è scongiurato

Proposizione 3.2.1. *Per ogni processo esteso* $q \in \mathcal{P}^{\mathcal{N}}$*, per ogni coppia di numeri naturali* $e, i \in \mathbb{N}$ *e per ogni coppia di insiemi di costanti di processo* $A, B \subseteq \mathcal{P}(\mathcal{C})$*, abbiamo*

$$c(q) \cap B = \emptyset \implies eid(q, e, i, A) = eid(q, e, i, A \cup B)$$

Dimostrazione. Dimostriamo la tesi ragionando per induzione strutturale su q:

- se $q = \mathbf{0}$, abbiamo banalmente

$$\begin{aligned} eid(q, e, i, A) &= eid(\mathbf{0}, e, i, A) = \epsilon = eid(\mathbf{0}, e, i, A \cup B) \\ &= eid(q, e, i, A \cup B) \end{aligned}$$

- se $q = \mu.q'$, per un'opportuna azione estesa $\mu \in \mathcal{A}ct'$ e per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, ricordando l'osservazione 2.2.1 (pag. 33) abbiamo

$$(c(q') = c(\mu.q') \wedge c(\mu.q') \cap B = \emptyset) \implies c(q') \cap B = \emptyset$$

  ovvero possiamo applicare l'induzione strutturale a q' ottenendo

$$eid(q', e, i, A) = eid(q', e, i, A \cup B)$$

  e quindi

$$\begin{aligned} eid(q, e, i, A) &= eid(\mu.q', e, i, A) = eid(q', e, i, A) = eid(q', e, i, A \cup B) \\ &= eid(\mu.q', e, i, A \cup B) = eid(q, e, i, A \cup B) \end{aligned}$$

- se $q = \underline{\mu}.q'$, per un'opportuna azione estesa $\mu \in \mathcal{A}ct'$ e per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, la prova è analoga alla precedente;

- se $q = p_1 + p_2$, per opportuni processi estesi sequenziali $p_1, p_2 \in \mathcal{P}^{\mathcal{N}}_{seq}$, ricordando l'osservazione 2.2.1 abbiamo

$$(c(p_1) \subseteq c(p_1 + p_2) = c(q) \wedge c(q) \cap B = \emptyset) \implies c(p_1) \cap B = \emptyset$$

$$(c(p_2) \subseteq c(p_1 + p_2) = c(q) \wedge c(q) \cap B = \emptyset) \implies c(p_2) \cap B = \emptyset$$

ovvero possiamo applicare l'induzione strutturale a p_1 e a p_2 ottenendo

$$eid(p_1, e+1, i, A) = eid(p_1, e+1, i, A \cup B)$$
$$eid(p_2, e+1, i+2^e, A) = eid(p_2, e+1, i+2^e, A \cup B)$$

e quindi, tenendo anche conto della casistica nella definizione di eid nel caso $p_1 + p_2$, quando è vero che

$$eid(p_1, e+1, i, A) \neq \epsilon \neq eid(p_1, e+1, i, A \cup B)$$

abbiamo

$$\begin{aligned} eid(q, e, i, A) &= eid(p_1 + p_2, e, i, A) = eid(p_1, e+1, i, A) \\ &= eid(p_1, e+1, i, A \cup B) = eid(p_1 + p_2, e, i, A \cup B) \\ &= eid(q, e, i, A \cup B) \end{aligned}$$

Quando invece è vero che

$$eid(p_1, e+1, i, A) = \epsilon = eid(p_1, e+1, i, A \cup B)$$

la dimostrazione è analoga ma passando da p_2;

- se $q = q_1 \mid q_2$, per opportuni processi estesi $q_1, q_2 \in \mathcal{P}^{\mathcal{N}}$, la prova è analoga alla precedente;
- se $q = (\nu a)q'$, per un opportuno nome $a \in \mathcal{L}$ e per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, possiamo distinguere due casi:
  - quando $a, \overline{a} \in fn(q')$, abbiamo

$$\begin{aligned} eid(q, e, i, A) &= eid((\nu a)q', e, i, A) = [\widehat{\eta_e}, \eta_i] \\ &= eid((\nu a)q', e, i, A \cup B) = eid(q, e, i, A \cup B) \end{aligned}$$

  - quando $a \notin fn(q') \vee \overline{a} \notin fn(q')$, abbiamo

$$\begin{aligned} eid(q, e, i, A) &= eid((\nu a)q', e, i, A) = \epsilon = eid((\nu a)q', e, i, A \cup B) \\ &= eid(q, e, i, A \cup B) \end{aligned}$$

- se $q = D$, dove $D \overset{def}{=} q'$ per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, dalle ipotesi abbiamo $D \notin B$; è opportuno distinguere due casi:
  - quando $D \in A$, e quindi anche $D \in A \cup B$, banalmente abbiamo

$$\begin{aligned} eid(q, e, i, A) &= eid(D, e, i, A) = \epsilon = eid(D, e, i, A \cup B) \\ &= eid(q, e, i, A \cup B) \end{aligned}$$

  - quando $D \notin A$, e quindi (avendo $D \notin B$) anche $D \notin A \cup B$, dalla definizione di *eid* abbiamo

$$eid(D, e, i, A) = eid(q', e, i, A \cup \{D\})$$

    e, dal momento che è evidente che $c(D) = c(q') \cup \{D\}$, abbiamo

$$(c(q') \subseteq c(D) = c(q) \wedge c(q) \cap B = \emptyset) \implies c(q') \cap B = \emptyset$$

    ovvero possiamo applicare l'induzione strutturale a q' e, applicandola non ad A e B ma ad $A \cup \{D\}$ e B, otteniamo

$$eid(q', e, i, A \cup \{D\}) = eid(q', e, i, A \cup \{D\} \cup B)$$

    e quindi

$$\begin{aligned} eid(q, e, i, A) &= eid(D, e, i, A) = eid(q', e, i, A \cup \{D\}) \\ &= eid(q', e, i, A \cup \{D\} \cup B) = eid(D, e, i, A \cup B) \\ &= eid(q, e, i, A \cup B) \end{aligned}$$

□

Ora vediamo un risultato relativo alla sostituzione di un nome visibile con un altro nome (visibile o meno) e all'identificativo extra.

Proposizione 3.2.2. *Per ogni processo esteso $q \in \mathcal{P}^{\mathcal{N}}$, per ogni nome – visibile o ristretto – $\mu \in \mathcal{L} \cup \mathcal{N}$, per ogni nome $a \in \mathcal{L}$, per ogni coppia di valori $e, i \in \mathbb{N}$ e per ogni insieme di costanti $A \in \mathcal{P}(\mathcal{C})$, è vero che*

$$eid(q\{\mu/a\}, e, i, A') = eid(q, e, i, A) = eid(q, e, i, A)\{\mu/a\}$$

dove A' è l'insieme di costanti definito come segue

$$\begin{aligned} A' = \{C' \mid \quad & (C' \in A \land a, \overline{a} \notin fn(C')) \\ & \lor (C' = C_{\{\mu/a\}} \land C \in A \land (a \in fn(C) \lor \overline{a} \in fn(C))\} \end{aligned}$$

Dimostrazione. La seconda eguaglianza è banalmente vera, considerando e il fatto che $eid(q, e, i, A)$ può essere uguale solo all'identificativo nullo ϵ – sul quale una sostituzione è, ovviamente, ininfluente – o essere nella forma $[\widehat{\eta_e}, \eta_i]$ e quindi, tenendo conto della condizione 3.7 (pag. 65), trasparente a qualunque sostituzione di un nome $a \in \mathcal{L}$.

La prima uguaglianza la dimostriamo per induzione strutturale su q:

- se $q = \mathbf{0}$, abbiamo banalmente

$$\begin{aligned} eid(q\{\mu/a\}, e, i, A') &= eid(\mathbf{0}\{\mu/a\}, e, i, A') = eid(\mathbf{0}, e, i, A') \\ &= \epsilon = eid(\mathbf{0}, e, i, A) = eid(q, e, i, A) \end{aligned}$$

- se $q = a.q'$, per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, per induzione strutturale abbiamo

$$eid(q'\{\mu/a\}, e, i, A') = eid(q', e, i, A)$$

  e quindi

$$\begin{aligned} eid(q\{\mu/a\}, e, i, A') &= eid((a.q')\{\mu/a\}, e, i, A') \\ &= eid(\mu.(q'\{\mu/a\}), e, i, A') \\ &= eid(q'\{\mu/a\}, e, i, A') = eid(q', e, i, A) \\ &= eid(a.q', e, i, A) = eid(q, e, i, A) \end{aligned}$$

- se $q = \overline{a}.q'$, $q = \underline{a}.q'$, $q = \underline{\overline{a}}.q'$, $q = \mu'.q'$ o $q = \underline{\mu'}.q'$, per un'opportuna azione $\mu' \in \mathcal{A}ct$ con $\mu' \notin \{a, \overline{a}\}$ e per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, la prova è analoga a quella del caso precedente;

- se $q = p_1 + p_2$, per opportuni processi estesi sequenziali $p_1, p_2 \in \mathcal{P}^{\mathcal{N}}_{seq}$, per induzione strutturale abbiamo

$$eid(p_1\{\mu/a\}, e + 1, i, A') = eid(p_1, e + 1, i, A)$$

$$eid(p_2\{\mu/a\}, e+1, i+2^e, A') = eid(p_2, e+1, i+2^e, A)$$

e quindi, tenendo anche conto della casistica nella definizione di eid nel caso $p_1 + p_2$, quando è vero che

$$eid(p_1\{\mu/a\}, e+1, i, A') \neq \epsilon \neq eid(p_1, e+1, i, A)$$

abbiamo

$$\begin{aligned} eid(q\{\mu/a\}, e, i, A') &= eid((p_1 + p_2)\{\mu/a\}, e, i, A') \\ &= eid(p_1\{\mu/a\} + p_2\{\mu/a\}, e, i, A') \\ &= eid(p_1\{\mu/a\}, e+1, i, A') \\ &= eid(p_1, e+1, i, A) \\ &= eid(p_1 + p_2, e, i, A) = eid(q, e, i, A) \end{aligned}$$

Quando invece è vero che

$$eid(p_1\{\mu/a\}, e+1, i, A') = \epsilon = eid(p_1, e+1, i, A)$$

la dimostrazione è analoga ma passando da p_2;

- se $q = q_1 \mid q_2$, per opportuni processi estesi $q_1, q_2 \in \mathcal{P}^{\mathcal{N}}$, la prova è analoga a quella del caso precedente;
- se $q = (\nu a)q'$, per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, possiamo distinguere due casi:
  - quando $a, \overline{a} \in fn(q')$, abbiamo

$$\begin{aligned} eid(q\{\mu/a\}, e, i, A') &= eid(((\nu a)q')\{\mu/a\}, e, i, A') \\ &= eid((\nu a)q', e, i, A') = [\widehat{\eta_e}, \eta_i] \\ &= eid((\nu a)q', e, i, A) = eid(q, e, i, A) \end{aligned}$$

  - quando $\{a, \overline{a}\} \not\subseteq fn(q')$, la dimostrazione è analoga a quella del sottocaso precedente tranne che al posto di $[\widehat{\eta_e}, \eta_i]$ si ottiene l'identificativo nullo ϵ;

- se $q = (\nu b)q'$, per un opportuno nome $b \in \mathcal{L}$ con $b \neq a \wedge b \neq \mu$ e per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, osserviamo che

$$\{b, \overline{b}\} \cap fn(q) = \{b, \overline{b}\} \cap fn(q\{\mu/a\})$$

in quanto la sostituzione di μ al posto di a (con b diverso sia da μ che da a) non ha influenza sulla presenza di b o del suo complementare nell'insieme dei nomi liberi. Abbiamo quindi i due casi:

  – quando $b, \overline{b} \in fn(q')$, e quindi $b, \overline{b} \in fn(q'\{\mu/a\})$, abbiamo che

$$\begin{aligned} eid(q\{\mu/a\}, e, i, A') &= eid(((\nu b)q')\{\mu/a\}, e, i, A') \\ &= eid((\nu b)(q'\{\mu/a\}), e, i, A') = [\widehat{\eta_e}, \eta_i] \\ &= eid((\nu b)q', e, i, A) = eid(q, e, i, A) \end{aligned}$$

  – quando $\{b, \overline{b}\} \not\subseteq fn(q')$, la dimostrazione è analoga a quella del sottocaso precedente tranne che al posto di $[\widehat{\eta_e}, \eta_i]$ si ottiene l'identificativo nullo ϵ;

- se $q = (\nu\mu)q'$, per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, ricordiamo che le regole di alpha-conversione ci dicono che $((\nu\mu)q')\{\mu/a\} = ((\nu c)(q'\{c/\mu\}))\{\mu/a\}$, dove $c \notin n(q')$; è evidente che

$$(\{\mu, \overline{\mu}\} \cap fn(q'))\{c/\mu\} = \{c, \overline{c}\} \cap fn(q'\{c/\mu\})$$

e quindi che, per come è definita *eid*,

$$eid(((\nu c)(q'\{c/\mu\}), e, i, A) = eid(((\nu\mu)q', e, i, A)$$

e poiché, riconducendoci al caso precedente, abbiamo

$$eid(((\nu c)(q'\{c/\mu\})\{\mu/a\}, e, i, A') = eid(((\nu c)(q'\{c/\mu\}), e, i, A)$$

possiamo concludere

$$eid(q\{\mu/a\}, e, i, A') = eid(((\nu\mu)q')\{\mu/a\}, e, i, A')$$

$$\begin{aligned} &= eid(((\nu c)(q'\{c/\mu\})\{\mu/a\}, e, i, A') \\ &= eid(((\nu c)(q'\{c/\mu\}), e, i, A) \\ &= eid(((\nu\mu)q', e, i, A) = eid(q, e, i, A) \end{aligned}$$

- se $q = B$, dove $B \stackrel{def}{=} q'$ per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, è opportuno distinguere quattro casi:
  - quando $\{a, \overline{a}\} \cap fn(q') \neq \emptyset$, ovvero quando la sostituzione $\{\mu/a\}$ è significativa su q' (ovvero su B), e $B \in A$, osserviamo che $B\{\mu/a\} = B_{\{\mu/a\}}$ e che, per costruzione di A', $B_{\{\mu/a\}} \in A'$; abbiamo quindi

$$\begin{aligned} eid(q\{\mu/a\}, e, i, A') &= eid(B\{\mu/a\}, e, i, A') = eid(B_{\{\mu/a\}}, e, i, A') \\ &= \epsilon = eid(B, e, i, A) = eid(q, e, i, A) \end{aligned}$$

  - quando $\{a, \overline{a}\} \cap fn(q') \neq \emptyset$, ovvero quando la sostituzione $\{\mu/a\}$ è significativa su q' (ovvero su B), e $B \notin A$, osserviamo che $B\{\mu/a\} = B_{\{\mu/a\}}$ e che, per costruzione di A', $B_{\{\mu/a\}} \notin A'$; per induzione strutturale su q' abbiamo

$$eid(q'\{\mu/a\}, e, i, A' \cup \{B_{\{\mu/a\}}\}) = eid(q', e, i, A \cup \{B\})$$

  e quindi

$$\begin{aligned} eid(q\{\mu/a\}, e, i, A') &= eid(B\{\mu/a\}, e, i, A') = eid(B_{\{\mu/a\}}, e, i, A') \\ &= eid(q'\{\mu/a\}, e, i, A' \cup \{B_{\{\mu/a\}}\}) \\ &= eid(q', e, i, A \cup \{B\}) = eid(B, e, i, A) \\ &= eid(q, e, i, A) \end{aligned}$$

  - quando $a, \overline{a} \notin fn(q')$, ovvero quando la sostituzione $\{\mu/a\}$ non è significativa su q' (ovvero su B), e $B \in A$, osserviamo che $B\{\mu/a\} = B$ e che, per costruzione di A', $B \in A'$; abbiamo quindi

$$\begin{aligned} eid(q\{\mu/a\}, e, i, A') &= eid(B\{\mu/a\}, e, i, A') = eid(B, e, i, A') \\ &= \epsilon = eid(B, e, i, A) = eid(q, e, i, A) \end{aligned}$$

– quando $a, \overline{a} \notin fn(q')$, ovvero quando la sostituzione $\{\mu/a\}$ non è significativa su q' (ovvero su B), e $B \notin A$, osserviamo che $B\{\mu/a\} = B$ e che, per costruzione di A', $B \notin A'$; per induzione strutturale su q' abbiamo

$$eid(q'\{\mu/a\}, e, i, A' \cup \{B\}) = eid(q', e, i, A \cup \{B\})$$

e quindi

$$\begin{aligned} eid(q\{\mu/a\}, e, i, A') &= eid(B\{\mu/a\}, e, i, A') = eid(B, e, i, A') \\ &= eid(q', e, i, A' \cup \{B\}) \\ &= eid(q', e, i, A \cup \{B\}) = eid(B, e, i, A) \\ &= eid(q, e, i, A) \end{aligned}$$

□

Data la precedente proposizione, abbiamo banalmente il seguente

Corollario 3.2.3. *Per ogni processo esteso $q \in \mathcal{P}^{\mathcal{N}}$, per ogni nome – visibile o ristretto – $\mu \in \mathcal{L} \cup \mathcal{N}$, per ogni nome $a \in \mathcal{L}$ e per ogni coppia di valori $e, i \in \mathbb{N}$ è vero che*

$$eid(q\{\mu/a\}, e, i, \emptyset) = eid(q, e, i, \emptyset) = eid(q, e, i, \emptyset)\{\mu/a\}$$

Dimostrazione. Immediata conseguenza della proposizione 3.2.2 (pag. 73), osservando che quando $A = \emptyset$, anche $A' = \emptyset$. □

Ora vediamo che il fatto che l'identificativo extra sia nullo o meno dipende dal primo parametro (il processo esteso), e dal quarto (che però è solo un indice del livello di ricorsione, nella sostituzione delle costanti, raggiunto), e non dai due parametri interi di supporto. Questo risultato può essere intuitivamente ovvio ma – dal momento che verrà usato, in seguito, in una dimostrazione importante – ho preferito darne una dimostrazione formale.

Proposizione 3.2.4. *Per ogni processo esteso $q \in \mathcal{P}^{\mathcal{N}}$, o l'identificativo extra associato è sempre nullo (pari a ϵ), a prescindere dai valori assunti dai parametri di supporto, o non è mai nullo, ovvero*

$$\begin{aligned}\forall q \in \mathcal{P}^{\mathcal{N}}.\forall e, f, i, j \in \mathbb{N}.\forall A \in \mathcal{P}(\mathcal{C}).(\quad & (eid(q, e, i, A) = \epsilon = eid(q, f, j, A)) \\ & \vee\ (eid(q, e, i, A) \neq \epsilon \neq eid(q, f, j, A)))\end{aligned}$$

Dimostrazione. Si prova per induzione strutturale su q:

- se $q = \mathbf{0}$, abbiamo banalmente

$$eid(q, e, i, A) = eid(\mathbf{0}, e, i, A) = \epsilon = eid(\mathbf{0}, f, j, A) = eid(q, f, j, A)$$

- se $q = \mu.q'$, per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$ e per un'opportuna azione $\mu \in \mathcal{A}ct'$, per induzione strutturale su q' abbiamo

$$\begin{aligned}& eid(q', e, i, A) = \epsilon = eid(q', f, j, A) \\ \vee\ & eid(q', e, i, A) \neq \epsilon \neq eid(q', f, j, A)\end{aligned}$$

  Supponiamo che i due *eid* di q' coincidano con ϵ; abbiamo

$$\begin{aligned}eid(q, e, i, A) &= eid(\mu.q', e, i, A) = eid(q', e, i, A) = \epsilon \\ &= eid(q', f, j, A) = eid(\mu.q', f, j, A) = eid(q, f, j, A)\end{aligned}$$

  Se i due *eid* di q' sono entrambi diversi da ϵ, si dimostra in modo analogo che i due *eid* di q sono entrambi diversi da ϵ;

- se $q = \underline{\mu}.q'$, per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$ e un'opportuna azione $\mu \in \mathcal{A}ct'$, la prova è analoga a quella del caso precedente;

- se $q = p_1 + p_2$, per opportuni processi estesi sequenziali $p_1, p_2 \in \mathcal{P}^{\mathcal{N}}_{seq}$, per induzione strutturale su p_1 abbiamo

$$\begin{aligned}& eid(p_1, e+1, i, A) = \epsilon = eid(p_1, f+1, j, A) \\ \vee\ & eid(p_1, e+1, i, A) \neq \epsilon \neq eid(p_1, f+1, j, A)\end{aligned}$$

e, per induzione strutturale su p_2, abbiamo

$$\begin{aligned} & eid(p_2, e+1, i+2^e, A) = \epsilon = eid(p_2, f+1, j+2^f, A) \\ \vee\ & eid(p_2, e+1, i+2^e, A) \neq \epsilon \neq eid(p_2, f+1, j+2^f, A) \end{aligned}$$

Dobbiamo trattare tre casi:

- quando entrambi gli *eid* di p_1 sono non nulli, abbiamo

$$\begin{aligned} eid(q, e, i, A) &= eid(p_1 + p_2, e, i, A) = eid(p_1, e+1, i, A) \neq \epsilon \\ &\neq eid(p_1, f+1, j, A) = eid(p_1 + p_2, f, j, A) \\ &= eid(q, f, j, A) \end{aligned}$$

- quando gli *eid* di p_1 sono entrambi nulli e gli *eid* di p_2 sono entrambi non nulli, abbiamo

$$\begin{aligned} eid(q, e, i, A) &= eid(p_1 + p_2, e, i, A) = eid(p_2, e+1, i+2^e, A) \neq \epsilon \\ &\neq eid(p_2, f+1, j+2^f, A) = eid(p_1 + p_2, f, j, A) \\ &= eid(q, f, j, A) \end{aligned}$$

- quando sia entrambi gli *eid* di p_1 che entrambi gli *eid* di p_2 sono nulli, abbiamo

$$\begin{aligned} eid(q, e, i, A) &= eid(p_1 + p_2, e, i, A) = eid(p_2, e+1, i+2^e, A) = \epsilon \\ &= eid(p_2, f+1, j+2^f, A) = eid(p_1 + p_2, f, j, A) \\ &= eid(q, f, j, A) \end{aligned}$$

Abbiamo quindi, in tutti e tre i casi, che entrambi gli *eid* di q sono nulli o sono entrambi non nulli;

- se $q = q_1 \mid q_2$, per opportuni processi estesi $q_1, q_2 \in \mathcal{P}^{\mathcal{N}}$, la prova è analoga a quella del caso precedente;
- se $q = (\nu a)q'$, per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, abbiamo due casi:

– quando $a, \overline{a} \in fn(q')$, abbiamo

$$\begin{aligned} eid(q, e, i, A) = eid((\nu a)q', e, i, A) = [\widehat{\eta_e}, \eta_i] \neq \epsilon \\ \neq [\widehat{\eta_f}, \eta_j] = eid((\nu a)q', f, j, A) = eid(q, f, j, A) \end{aligned}$$

– quando $\{a, \overline{a}\} \not\subseteq fn(q')$, abbiamo

$$\begin{aligned} eid(q, e, i, A) = eid((\nu a)q', e, i, A) = \epsilon \\ = eid((\nu a)q', f, j, A) = eid(q, f, j, A) \end{aligned}$$

In entrambi i casi, la tesi è verificata;

- se $q = B$, dove $B \stackrel{def}{=} q'$ – con $B \in A$ – per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, abbiamo banalmente

$$eid(q, e, i, A) = eid(B, e, i, A) = \epsilon = eid(B, f, j, A) = eid(q, f, j, A)$$

- se $q = B$, dove $B \stackrel{def}{=} q'$ – con $B \notin A$ – per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, per induzione strutturale su q' abbiamo

$$\begin{aligned} eid(q', e, i, A \cup \{B\}) = \epsilon = eid(q', f, j, A \cup \{B\}) \\ \vee\ eid(q', e, i, A \cup \{B\}) \neq \epsilon \neq eid(q', f, j, A \cup \{B\}) \end{aligned}$$

e, supponendo che entrambi gli *eid* di q' siano non nulli, abbiamo

$$\begin{aligned} eid(q, e, i, A) = eid(B, e, i, A) = eid(q', e, i, A \cup \{B\}) \neq \epsilon \\ \neq eid(q', f, j, A \cup \{B\}) = eid(B, f, j, A) \\ = eid(q, f, j, A) \end{aligned}$$

Quando entrambi gli *eid* di q' sono nulli, si prova in maniera analoga che anche entrambi gli *eid* di q sono nulli.

$\square$

3.2.5 Nuova definizione di *dec*

Possiamo quindi ridefinire *dec* che però, analogamente alla già vista *eid*, avrà bisogno di un paio di nuovi argomenti interi (l'insieme delle constanti incontrate non serve a *dec*) che serviranno a fissare in maniera deterministica le nuove azioni ristrette non banali.

Definizione 3.2.9. *La funzione*

$$dec : (\mathcal{P}^{\mathcal{N}} \times \mathbb{N} \times \mathbb{N}) \to \mathcal{M}_{fin}(S_{MCCS})$$

è definita induttivamente come segue

$$\begin{aligned}
dec(\mathbf{0}, e, i) &= \emptyset \\
dec(\mu.q, e, i) &= 1 \cdot (\mu.q, eid(q, e, i, \emptyset)) \\
dec(\underline{\mu}.q, e, i) &= 1 \cdot (\underline{\mu}.q, eid(q, e, i, \emptyset)) \\
dec(p_1 + p_2, e, i) &= 1 \cdot (p_1 + p_2, eid(p_1 + p_2, e, i, \emptyset)) \\
dec(q_1 \mid q_2, e, i) &= dec(q_1, e+1, i) \oplus dec(q_2, e+1, i+2^e) \\
dec((\nu a)q, e, i) &= \begin{cases} dec(q\{\eta_i/a\}, e, i+2^e) & se\ a, \overline{a} \in fn(q) \\ dec(q\{\eta_d/a\}, e, i) & se\ a \in fn(q) \wedge \overline{a} \notin fn(q) \\ dec(q\{\eta_c/a\}, e, i) & se\ a \notin fn(q) \wedge \overline{a} \in fn(q) \\ dec(q, e, i) & se\ a, \overline{a} \notin fn(q) \end{cases} \\
dec(C, e, i) &= dec(q, e, i) \qquad se\ C \stackrel{def}{=} q
\end{aligned}$$

Ricordiamo che rappresenteremo (p, ϵ) come $\{p\}$ mentre rappresenteremo $(p, (\widehat{\eta_i}, \eta_j))$ come $\{p\}[\widehat{\eta_i}, \eta_j]$. Per fare un paio di esempi, $(a.B, \epsilon)$ come $\{a.B\}$ e $(a.B, (\widehat{\eta_{12}}, \eta_{47}))$ come $\{a.B\}[\widehat{\eta_{12}}, \eta_{47}]$.

L'idea di fondo è di utilizzare i nomi ristretti banali (η_d o η_c) quando le circostanze lo consentono, di ignorare del tutto le sostituzioni degeneri (quando si sostituisce un nome che non appare – direttamente o complementato – tra i nomi liberi) e di usare un nome ristretto non banale solo quando necessario.

Per quanto riguarda la determinazione dell'indice del nome ristretto non banale, si usano due valori interi: e ed i. Il primo indica l'esponente del passo, visto come potenza di due, per la disponibilità di un nuovo indice; il secondo indica il primo indice disponibile. Se, per esempio, entrambi i valori sono pari a zero, vuol dire che il primo indice disponibile è zero (nome η_0), il seguente è uno (ovvero $0 + 2^0$), quindi due (ovvero $1 + 2^0$) e così via. Nei casi induttivi questi indici vengono tendenzialmente trasmessi; le uniche due eccezioni sono le seguenti:

- quando un indice viene utilizzato (nel caso della restrizione, quando $a, \overline{a} \in fn(q)$); in tal caso si "consuma" il valore indicato da i, si passa $i+2^e$ (precedente indice disponibile più il passo) al posto del precedente i;

- quando siamo in presenza di due termini paralleli; in tal caso si "spezza" la sequenza dei valori disponibili, incrementando di uno l'esponente e e passando come valori iniziali delle due nuove sequenze i primi due valori inizialmente disponibili.

Per descrivere meglio quest'ultima casistica si consideri il seguente esempio:

$$dec(q_1 \mid q_2, 0, 0) = dec(q_1, 1, 0) \oplus dec(q_2, 1, 1)$$

Inizialmente $q_1 \mid q_2$ dispone come primo indice di zero e come passo uno (ossia 2^0) ovvero dispone, come indici, di tutti i numeri naturali; la sequenza viene poi "spezzata" in due sequenze distinte dove per q_1 sono disponibili tutti gli indici a partire da zero con passo due (ossia 2^1), ovvero tutti gli indici pari, mentre per q_2 sono disponibili tutti gli indici a partire da uno, sempre con passo due, ovvero tutti i numeri dispari.

Vediamo ora un risultato che tornerà utile in seguito.

Proposizione 3.2.5. *Per ogni processo esteso $q \in \mathcal{P}^{\mathcal{N}}$, per ogni nome – visibile o ristretto – $\mu \in \mathcal{L} \cup \mathcal{N}$, per ogni nome $a \in \mathcal{L}$ e per ogni coppia di numeri naturali $e, i \in \mathbb{N}$, è vero che*

$$dec(q\{\mu/a\}, e, i) = dec(q, e, i)\{\mu/a\}$$

Dimostrazione. Lo dimostriamo per induzione strutturale su q:

- se $q = \mathbf{0}$, abbiamo

$$\begin{aligned} dec(q\{\mu/a\}, e, i) &= dec(\mathbf{0}\{\mu/a\}, e, i) = dec(\mathbf{0}, e, i) = \emptyset \\ &= \emptyset\{\mu/a\} = dec(\mathbf{0}, e, i)\{\mu/a\} = dec(q, e, i)\{\mu/a\} \end{aligned}$$

- se $q = a.q'$, per un opportuno nome $a \in \mathcal{L}$ e un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, dal corollario 3.2.3 (pag. 78) abbiamo

$$eid(q'\{\mu/a\}, e, i, \emptyset) = eid(q', e, i, \emptyset) = eid(q', e, i, \emptyset)\{\mu/a\}$$

  e quindi

$$\begin{aligned} dec(q\{\mu/a\}, e, i) &= dec((a.q')\{\mu/a\}, e, i) = dec(\mu.(q'\{\mu/a\}), e, i) \\ &= 1 \cdot (\mu.(q'\{\mu/a\}), eid(q'\{\mu/a\}, e, i, \emptyset)) \\ &= 1 \cdot ((a.q')\{\mu/a\}, eid(q'\{\mu/a\}, e, i, \emptyset)) \\ &= 1 \cdot ((a.q')\{\mu/a\}, eid(q', e, i, \emptyset)\{\mu/a\}) \\ &= 1 \cdot ((a.q', eid(q', e, i, \emptyset))\{\mu/a\}) \\ &= (1 \cdot (a.q', eid(q', e, i, \emptyset)))\{\mu/a\} \\ &= dec(a.q', e, i)\{\mu/a\} = dec(q, e, i)\{\mu/a\} \end{aligned}$$

- se $q = \overline{a}.q'$, $q = \underline{a}.q'$, $q = \underline{\overline{a}}.q'$, $q = \mu.q'$ o $q = \underline{\mu}.q'$ – per un'opportuna azione estesa $\mu \in \mathcal{A}ct'$ tale che $\mu \notin \{a, \overline{a}\}$ e per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$ – la dimostrazione è del tutto analoga a quella del caso precedente;

- se $q = p_1 + p_2$, per opportuni processi estesi sequenziali $p_1, p_2 \in \mathcal{P}^{\mathcal{N}}_{seq}$, dal corollario 3.2.3 abbiamo

$$\begin{aligned} eid((p_1 + p_2)\{\mu/a\}, e, i, \emptyset) &= eid(p_1 + p_2, e, i, \emptyset) \\ &= eid(p_1 + p_2, e, i, \emptyset)\{\mu/a\} \end{aligned}$$

  e quindi

$$dec(q\{\mu/a\}, e, i) = dec((p_1 + p_2)\{\mu/a\}, e, i)$$

$$
\begin{aligned}
&= dec(p_1\{\mu/a\} + p_2\{\mu/a\}, e, i) \\
&= 1 \cdot (p_1\{\mu/a\} + p_2\{\mu/a\}, \\
&\qquad eid(p_1\{\mu/a\} + p_2\{\mu/a\}, e, i, \emptyset)) \\
&= 1 \cdot ((p_1 + p_2)\{\mu/a\}, eid((p_1 + p_2)\{\mu/a\}, e, i, \emptyset)) \\
&= 1 \cdot ((p_1 + p_2)\{\mu/a\}, eid(p_1 + p_2, e, i, \emptyset)\{\mu/a\}) \\
&= 1 \cdot ((p_1 + p_2, eid(p_1 + p_2, e, i, \emptyset))\{\mu/a\}) \\
&= (1 \cdot (p_1 + p_2, eid(p_1 + p_2, e, i, \emptyset)))\{\mu/a\} \\
&= dec(p_1 + p_2, e, i)\{\mu/a\} = dec(q, e, i)\{\mu/a\}
\end{aligned}
$$

- se $q = q_1 \mid q_2$, per opportuni processi estesi $q_1, q_2 \in \mathcal{P}^{\mathcal{N}}$, per induzione strutturale su q_1 e su q_2 abbiamo

$$
\begin{aligned}
dec(q_1\{\mu/a\}, e+1, i) &= dec(q_1, e+1, i)\{\mu/a\} \\
dec(q_2\{\mu/a\}, e+1, i+2^e) &= dec(q_2, e+1, i+2^e)\{\mu/a\}
\end{aligned}
$$

  e quindi

$$
\begin{aligned}
dec(q\{\mu/a\}, e, i) &= dec((q_1 \mid q_2)\{\mu/a\}, e, i) \\
&= dec(q_1\{\mu/a\} \mid q_2\{\mu/a\}, e, i) \\
&= dec(q_1\{\mu/a\}, e+1, i) \oplus dec(q_2\{\mu/a\}, e+1, i+2^e) \\
&= dec(q_1, e+1, i)\{\mu/a\} \oplus dec(q_2, e+1, i+2^e)\{\mu/a\} \\
&= (dec(q_1, e+1, i) \oplus dec(q_2, e+1, i+2^e))\{\mu/a\} \\
&= dec(q_1 \mid q_2, e, i)\{\mu/a\} = dec(q, e, i)\{\mu/a\}
\end{aligned}
$$

- se $q = (\nu a)q'$, per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, abbiamo quattro casi, a seconda dei valori di $\{a, \overline{a}\} \cap fn(q)$:

  – quando $a, \overline{a} \in fn(q')$ è necessario utilizzare un nuovo nome ristretto η_i; per induzione strutturale abbiamo

$$
dec(q'\{\eta_i/a\}, e, i+2^e) = dec(q', e, i+2^e)\{\eta_i/a\}
$$

e quindi, osservando che la doppia sostituzione $\{\eta_i/a\}\{\mu/a\}$ ha lo stesso effetto della singola $\{\eta_i/a\}$ (dal momento che la seconda sostituzione, agendo nuovamente sulla già sostituita a, è inefficace), abbiamo

$$\begin{aligned}
dec(q\{\mu/a\}, e, i) &= dec(((\nu a)q')\{\mu/a\}, e, i) = dec((\nu a)q', e, i) \\
&= dec(q'\{\eta_i/a\}, e, i + 2^e) \\
&= dec(q', e, i + 2^e)\{\eta_i/a\} \\
&= dec(q', e, i + 2^e)\{\eta_i/a\}\{\mu/a\} \\
&= dec(q'\{\eta_i/a\}, e, i + 2^e)\{\mu/a\} \\
&= dec((\nu a)q', e, i)\{\mu/a\} = dec(q, e, i)\{\mu/a\}
\end{aligned}$$

– quando $a \in fn(q') \wedge \overline{a} \notin fn(q')$ è necessario utilizzare il nome ristretto banale η_d; tenendo conto che, per induzione strutturale,

$$dec(q'\{\eta_d/a\}, e, i) = dec(q', e, i)\{\eta_d/a\}$$

e quindi, osservando – analogamente al caso precedente – che la doppia sostituzione $\{\eta_d/a\}\{\mu/a\}$ ha lo stesso effetto della singola $\{\eta_d/a\}$, abbiamo

$$\begin{aligned}
dec(q\{\mu/a\}, e, i) &= dec(((\nu a)q')\{\mu/a\}, e, i) = dec((\nu a)q', e, i) \\
&= dec(q'\{\eta_d/a\}, e, i) = dec(q', e, i)\{\eta_d/a\} \\
&= dec(q', e, i)\{\eta_d/a\}\{\mu/a\} \\
&= dec(q'\{\eta_d/a\}, e, i)\{\mu/a\} \\
&= dec((\nu a)q', e, i)\{\mu/a\} = dec(q, e, i)\{\mu/a\}
\end{aligned}$$

– quando $a \notin fn(q') \wedge \overline{a} \in fn(q')$, la dimostrazione è del tutto analoga al caso precedente, utilizzando il nome ristretto banale η_c al posto di η_d;

– quando $a, \overline{a} \notin fn(q')$ abbiamo che il nome a e il suo co-nome non sono usati (se non protetti da altre sostituzioni) in q' e quindi

$q' = q'\{\mu/a\}$; l'induzione strutturale ci dice anche che

$$dec(q'\{\mu/a\}, e, i) = dec(q', e, i)\{\mu/a\}$$

e quindi abbiamo

$$\begin{aligned}
dec(q\{\mu/a\}, e, i) &= dec(((\nu a)q')\{\mu/a\}, e, i) = dec((\nu a)q', e, i) \\
&= dec(q', e, i) = dec(q'\{\mu/a\}, e, i) \\
&= dec(q', e, i)\{\mu/a\} = dec((\nu a)q', e, i)\{\mu/a\} \\
&= dec(q, e, i)\{\mu/a\}
\end{aligned}$$

- se $q = (\nu b)q'$, per un opportuno nome $b \in \mathcal{L}$ tale che $b \neq a \wedge b \neq \mu$ e per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, abbiamo quattro casi, a seconda dei valori di $\{b, \overline{b}\} \cap fn(q')$:
  - quando $b, \overline{b} \in fn(q')$ è necessario utilizzare un nuovo nome ristretto η_i; per induzione strutturale abbiamo che

    $$dec(q'\{\eta_i/b\}\{\mu/a\}, e, i + 2^e) = dec(q'\{\eta_i/b\}, e, i + 2^e)\{\mu/a\}$$

    e quindi

    $$\begin{aligned}
    dec(q\{\mu/a\}, e, i) &= dec(((\nu b)q')\{\mu/a\}, e, i) \\
    &= dec((\nu b)(q'\{\mu/a\}), e, i) \\
    &= dec(q'\{\mu/a\}\{\eta_i/b\}, e, i + 2^e) \\
    &= dec(q'\{\eta_i/b\}\{\mu/a\}, e, i + 2^e) \\
    &= dec(q'\{\eta_i/b\}, e, i + 2^e)\{\mu/a\} \\
    &= dec((\nu b)q', e, i)\{\mu/a\} = dec(q, e, i)\{\mu/a\}
    \end{aligned}$$

  - quando $b \in fn(q') \wedge \overline{b} \notin fn(q')$ è necessario utilizzare il nome ristretto banale η_d; per induzione strutturale abbiamo che

    $$dec(q'\{\eta_d/b\}\{\mu/a\}, e, i) = dec(q'\{\eta_d/b\}, e, i)\{\mu/a\}$$

    e quindi

    $$dec(q\{\mu/a\}, e, i) = dec(((\nu b)q')\{\mu/a\}, e, i)$$

$$
\begin{aligned}
&= dec((\nu b)(q'\{\mu/a\}), e, i) \\
&= dec(q'\{\mu/a\}\{\eta_d/b\}, e, i) \\
&= dec(q'\{\eta_d/b\}\{\mu/a\}, e, i) \\
&= dec(q'\{\eta_d/b\}, e, i)\{\mu/a\} \\
&= dec((\nu b)q', e, i)\{\mu/a\} \\
&= dec(q, e, i)\{\mu/a\}
\end{aligned}
$$

- quando $b \notin fn(q') \wedge \overline{b} \in fn(q')$, la dimostrazione è del tutto analoga al caso precedente, utilizzando il nome ristretto banale η_c al posto di η_d;
- quando $b, \overline{b} \notin fn(q')$, per induzione strutturale abbiamo che

$$dec(q'\{\mu/a\}, e, i) = dec(q', e, i)\{\mu/a\}$$

e quindi

$$
\begin{aligned}
dec(q\{\mu/a\}, e, i) &= dec(((\nu b)q')\{\mu/a\}, e, i) \\
&= dec((\nu b)(q'\{\mu/a\}), e, i) \\
&= dec(q'\{\mu/a\}, e, i) = dec(q', e, i)\{\mu/a\} \\
&= dec((\nu b)q', e, i)\{\mu/a\} \\
&= dec(q, e, i)\{\mu/a\}
\end{aligned}
$$

- se $q = (\nu\mu)q'$, per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, ricordiamo che le regole di alpha-conversione ci dicono che $((\nu\mu)q')\{\mu/a\} = ((\nu c)(q'\{c/\mu\}))\{\mu/a\}$, con $c \notin n(q')$; è evidente che

$$(\{\mu, \overline{\mu}\} \cap fn(q'))\{c/\mu\} = \{c, \overline{c}\} \cap fn(q'\{c/\mu\})$$

Riconducendoci al caso precedente abbiamo

$$dec(((\nu c)(q'\{c/\mu\})\{\mu/a\}, e, i) = dec(((\nu c)(q'\{c/\mu\}), e, i)\{\mu/a\}$$

Vediamo la casistica relativa a $\{\mu, \overline{\mu}\} \subseteq fn(q')$:

– quando $\mu, \overline{\mu} \in fn(q')$, e quindi $c, \overline{c} \in fn(q'\{c/\mu\})$, abbiamo – considerando anche che $c \notin n(q')$, e quindi che $q'\{c/\mu\}\{\eta_i/c\} = q'\{\eta_i/\mu\}$ (dal momento che la sostituzione $\{\eta_i/\mu\}$ non tralascia nessuna c presente in q') – che

$$\begin{aligned} dec(q\{\mu/a\}, e, i) &= dec(((\nu\mu)q')\{\mu/a\}, e, i) \\ &= dec((\nu c)(q'\{c/\mu\})\{\mu/a\}, e, i) \\ &= dec((\nu c)(q'\{c/\mu\}), e, i)\{\mu/a\} \\ &= dec(q'\{c/\mu\}\{\eta_i/c\}, e, i + 2^e)\{\mu/a\} \\ &= dec(q'\{\eta_i/\mu\}, e, i + 2^e)\{\mu/a\} \\ &= dec((\nu\mu)q', e, i)\{\mu/a\} = dec(q, e, i)\{\mu/a\} \end{aligned}$$

– quando $\mu \in fn(q') \wedge \overline{\mu} \notin fn(q')$, e quindi $c \in fn(q'\{c/\mu\}) \wedge \overline{c} \notin fn(q'\{c/\mu\})$, abbiamo – analogamente al caso precedente – che $q'\{c/\mu\}\{\eta_d/c\} = q'\{\eta_d/\mu\}$ e quindi

$$\begin{aligned} dec(q\{\mu/a\}, e, i) &= dec(((\nu\mu)q')\{\mu/a\}, e, i) \\ &= dec((\nu c)(q'\{c/\mu\})\{\mu/a\}, e, i) \\ &= dec((\nu c)(q'\{c/\mu\}), e, i)\{\mu/a\} \\ &= dec(q'\{c/\mu\}\{\eta_d/c\}, e, i)\{\mu/a\} \\ &= dec(q'\{\eta_d/\mu\}, e, i)\{\mu/a\} \\ &= dec((\nu\mu)q', e, i)\{\mu/a\} = dec(q, e, i)\{\mu/a\} \end{aligned}$$

– quando $\mu \notin fn(q') \wedge \overline{\mu} \in fn(q')$, e quindi $c \notin fn(q'\{c/\mu\}) \wedge \overline{c} \in fn(q'\{c/\mu\})$, la dimostrazione è del tutto analoga al caso precedente, usando η_c al posto di η_d;

– quando $\mu, \overline{\mu} \notin fn(q')$, e quindi $c, \overline{c} \notin fn(q'\{c/\mu\})$, rileviamo che $q'\{c/\mu\} = q'$ (dal momento che μ e il suo co-nome non sono presenti liberi in q', la loro sostituzione è ininfluente) e quindi

$$\begin{aligned} dec(q\{\mu/a\}, e, i) &= dec(((\nu\mu)q')\{\mu/a\}, e, i) \\ &= dec((\nu c)(q'\{c/\mu\})\{\mu/a\}, e, i) \end{aligned}$$

$$
\begin{aligned}
&= dec((\nu c)(q'\{c/\mu\}), e, i)\{\mu/a\} \\
&= dec(q'\{c/\mu\}, e, i)\{\mu/a\} \\
&= dec(q', e, i)\{\mu/a\} \\
&= dec((\nu\mu)q', e, i)\{\mu/a\} \\
&= dec(q, e, i)\{\mu/a\}
\end{aligned}
$$

- se $q = B$, con $B \stackrel{def}{=} q'$ per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, per induzione strutturale abbiamo che

$$dec(q'\{\mu/a\}, e, i) = dec(q', e, i)\{\mu/a\}$$

e quindi

$$
\begin{aligned}
dec(q\{\mu/a\}, e, i) &= dec(B\{\mu/a\}, e, i) = dec(q'\{\mu/a\}, e, i) \\
&= dec(q', e, i)\{\mu/a\} = dec(B, e, i)\{\mu/a\} \\
&= dec(q, e, i)\{\mu/a\}
\end{aligned}
$$

□

3.2.6 Nuove regole di transizione

Stante la nuova definizione di *dec*, ho riscritto le regole di transizione di tabella 3.1 (pag. 62) tenendo conto dei nuovi parametri della funzione. Le nuove regole sono quelle di tabella 3.2 (pagina a fronte), con il supporto di regole di sincronizzazione di tabella 3.3 (pag. 92), che corrispondono a quelle della semantica sugli LTS – ovvero quelle di tabella 2.2 (pag. 39) – ma che tengono conto anche delle azioni ristrette presenti in $\mathcal{N} \cup \overline{\mathcal{N}}$, e delle regole di concatenazione di tabella 3.4 (pag. 93), che sono sostanzialmente le stesse di quelle della semantica sugli LTS, ovvero quelle di tabella 2.3 (pag. 39).

Nelle tabelle indichiamo con μ le azioni in $\mathcal{A}ct'$, comprese quindi le azioni ristrette δ. È necessario prevedere anche le transizioni delle azioni ristrette (ad esempio: nella regola (pref)) per permetterne la sincronizzazione con la regola (com). Con σ (σ_1, σ_2) intendiamo sequenze di una o più azioni in

$$\text{(pref)}\ dec(\mu.q, e, i) \xrightarrow{\mu} dec(q, e, i)$$

$$\text{(s-pref)}\ \frac{m_1 \xrightarrow{\sigma'} m'_1}{dec(\underline{\mu}.q, e, i) \xrightarrow{\sigma} m'_1 \oplus m_2} \quad dec(q, e, i) = m_1 \oplus m_2 \quad conc(\mu, \sigma', \sigma)$$

$$\text{(sum-1)}\ \frac{dec(p_1, e+1, i) \xrightarrow{\sigma} m}{dec(p_1 + p_2, e, i) \xrightarrow{\sigma} m}$$

$$\text{(sum-2)}\ \frac{dec(p_2, e+1, i+2^e) \xrightarrow{\sigma} m}{dec(p_1 + p_2, e, i) \xrightarrow{\sigma} m}$$

$$\text{(com)}\ \frac{m_1 \xrightarrow{\sigma_1} m'_1 \quad m_2 \xrightarrow{\sigma_2} m'_2}{m_1 \oplus m_2 \xrightarrow{\sigma} m'_1 \oplus m'_2} \quad sync(\sigma_1, \sigma_2, \sigma)$$

Tabella 3.2: Regole aggiornate per le transizioni in $\mathcal{M}_{fin}(S_{MCCS})$

$$sync(\alpha, \overline{\alpha}, \tau) \qquad sync(\delta, \overline{\delta}, \tau)$$

$$\frac{merge(\sigma_1, \sigma_2, \sigma)}{sync(\alpha\sigma_1, \overline{\alpha}\sigma_2, \sigma)} \qquad \frac{merge(\sigma_1, \sigma_2, \sigma)}{sync(\delta\sigma_1, \overline{\delta}\sigma_2, \sigma)}$$

$$\frac{sync(\sigma_1, \sigma_2, \sigma')}{sync(\mu\sigma_1, \sigma_2, \sigma)} \quad conc(\mu, \sigma', \sigma)$$

$$\frac{sync(\sigma_1, \sigma_2, \sigma')}{sync(\sigma_1, \mu\sigma_2, \sigma)} \quad conc(\mu, \sigma', \sigma)$$

$$merge(\alpha, \overline{\alpha}, \tau) \qquad merge(\delta, \overline{\delta}, \tau)$$

$$merge(\mu, \epsilon, \mu) \qquad merge(\epsilon, \mu, \mu)$$

$$\frac{merge(\sigma_1, \sigma_2, \sigma)}{merge(\alpha\sigma_1, \overline{\alpha}\sigma_2, \sigma)} \qquad \frac{merge(\sigma_1, \sigma_2, \sigma)}{merge(\delta\sigma_1, \overline{\delta}\sigma_2, \sigma)}$$

$$\frac{merge(\sigma_1, \sigma_2, \sigma')}{merge(\mu\sigma_1, \sigma_2, \sigma)} \quad conc(\mu, \sigma', \sigma)$$

$$\frac{merge(\sigma_1, \sigma_2, \sigma')}{merge(\sigma_1, \mu\sigma_2, \sigma)} \quad conc(\mu, \sigma', \sigma)$$

Tabella 3.3: Regole di sincronizzazione con azioni ristrette

$(\mathcal{V} \cup \mathcal{N} \cup \overline{\mathcal{N}})^+ \cup \{\tau\}$[7]. Con α (solo tabella 3.3) intendiamo solo le azioni visibili ($\mathcal{V}$) mentre con δ solo le azioni ristrette ($\mathcal{N} \cup \overline{\mathcal{N}}$).

$$conc(\mu, \tau, \mu) \qquad conc(\tau, \sigma, \sigma) \qquad \frac{\mu \neq \tau \neq \sigma}{conc(\mu, \sigma, \mu\sigma)}$$

Tabella 3.4: Regole di concatenazione per azioni in $\mathcal{M}_{fin}(S_{MCCS})$

A noi però, concretamente, interessano solo le transizioni basate su azioni visibili o sulla sincronizzazione τ; le transizioni basate su azioni ristrette ci interessano solo quando si sincronizzano in azioni τ. Come vedremo più avanti, le transizioni basate su azioni ristrette e non sincronizzate verranno filtrate ed escluse.

Esplicitamente osserviamo che

Osservazione 3.2.6. *Se abbiamo* $dec(\underline{\mu}.q, e, i) \xrightarrow{\sigma} m$*, dalla premessa di (s-pref) abbiamo che* $dec(q, e, i)[t\rangle m$*, per un'opportuna transizione* t *tale che* $l(t) = \sigma'$ *e che* $conc(\mu, \sigma', \sigma)$*.*

3.2.7 Rete P/T e sistemi P/T associati al Multi-CCS

Disponendo della nuova *dec* e delle nuove regole di transizione, possiamo procedere con alcune definizioni.

Definizione 3.2.10. *Sia*

$$\longrightarrow \;\subseteq\; (\mathcal{M}_{fin}(S_{MCCS}) \setminus \emptyset) \times ((\mathcal{V} \cup \mathcal{N} \cup \overline{\mathcal{N}})^+ \cup \{\tau\}) \times \mathcal{M}_{fin}(S_{MCCS})$$

il più piccolo insieme di transizioni generato dalle regole di tabella 3.2 (pag. 91) – nella quale m*,* m_1*,* m_2*,* m_1' *e* m_2' *rappresentano elementi di*

[7]si osservi infatti che le regole di sincronizzazione di tabella 3.3, così come le originarie di tabella 2.2, prevedono il "collasso" delle τ

$\mathcal{M}_{fin}(S_{MCCS})$ – col supporto delle regole di sincronizzazione sync di tabella 3.3 (pag. 92) e delle regole di concatenazione conc di tabella 3.4 (pagina precedente).

Definizione 3.2.11. *La* rete P/T *associata al Multi-CCS è la tripla*

$$N_{MCCS} = (S_{MCCS}, \mathcal{A}, T_{MCCS})$$

dove S_{MCCS} e $\mathcal{A}$, conformemente alle definizioni 3.2.2 (pag. 66) e 2.3.1 (pag. 37), sono pari a

$$\begin{aligned} S_{MCCS} &= \mathcal{P}_{seq}^{\mathcal{N}} \times ((\widehat{\mathcal{N}_{\mathbb{N}}} \times \mathcal{N}_{\mathbb{N}}) \cup \{\epsilon\}) \\ \mathcal{A} &= \mathcal{V}^{+} \cup \{\tau\} \end{aligned}$$

e dove T_{MCCS} è qui definito pari a

$$T_{MCCS} = \{(m, \sigma, m') \mid m \xrightarrow{\sigma} m' \wedge \sigma \in \mathcal{A}\}$$

In altre parole, la rete P/T è ottenuta filtrando le transizioni ottenute dalle regole di tabella 3.2 in maniera tale che nelle σ non appaiano azioni ristrette δ.

Dato un generico processo Multi-CCS $q \in \mathcal{P}$, possiamo individuare una sottorete di N_{MCCS} comprendente i posti raggiungibili a partire da $dec(q, 0, 0)$, le transizioni attivabili, sempre a partire da $dec(q, 0, 0)$, e le etichette corrispondenti a tali transizioni; formalmente

Definizione 3.2.12. *Sia $q \in \mathcal{P}$ un generico processo Multi-CCS; il* sistema P/T associato a *q è*

$$Net(q) = (S_q, A_q, T_q, m_0)$$

dove

$$\begin{aligned} m_0 &= dec(q, 0, 0) \\ S_q &= \{s \in S_{MCCS} \mid \exists m \in [m_0\rangle . m(s) > 0\} \\ T_q &= \{t \in T_{MCCS} \mid \exists m \in [m_0\rangle . m[t\rangle\} \\ A_q &= \{\sigma \in \mathcal{A} \mid \exists t \in T_q . \sigma = l(t)\} \end{aligned}$$

3.2.8 Pesi di transizioni e di multiset di transizioni

Sulla base delle definizioni delle transizioni, possiamo definire il “peso” di una transizione come segue

Definizione 3.2.13. *Sia* $t \in\longrightarrow$ *una transizione generata dalle regole di tabella 3.2 (pag. 91), definiamo induttivamente il* peso *della transizione – e lo indichiamo sinteticamente come* $\gamma(t)$ *– come segue:*

- $\gamma(t) = 1$*, se* t *è giustificata dalla regola (pref);*
- $\gamma(t) = \gamma(t') + 1$*, se* t *è giustificata dalla regola (s-pref), (sum-1) o (sum-2), dove* t' *è la transizione della premessa;*
- $\gamma(t) = \gamma(t_1) + \gamma(t_2) + 1$*, se* t *è giustificata dalla regola (com), dove* t_1 *e* t_2 *sono le transizioni della premessa.*

Nel caso la transizione t *possa essere giustificata da diverse regole o alberi di prova, il peso* $\gamma(t)$ *è il minore dei pesi ottenibili dall'applicazione delle regole indicate.*

Grazie al peso delle transizioni potremo, in alcune delle dimostrazioni per induzione, disporre di un valore numerico decrescente che giustificherà l'induzione stessa.

Definito il peso di una transizione, la definizione di peso di un multiset di transizioni deriva banalmente.

Definizione 3.2.14. *Dato un multiset* $u \in \mathcal{M}_{fin}(\longrightarrow)$ *di transizioni pari a* $u = k_1t_1 \oplus k_2t_2 \oplus \ldots \oplus k_nt_n$*, con* $k_1, k_2, \ldots, k_n \in \mathbb{N}$ *e* $t_1, t_2, \ldots, t_n \in\longrightarrow$*, definiamo il peso del multiset* u *– che indichiamo con* $\gamma(u)$ *– come combinazione lineare dei pesi delle componenti, ovvero*

$$\gamma(u) = \sum_{i=1}^{n} k_i \cdot \gamma(t_i) \tag{3.10}$$

3.3 Esempi

Vediamo ora un paio di esempi relativi alla costruzione di sistemi P/T.

3.3.1 $A = (\nu b)(a.A \mid b.A)$

Per primo vediamo l'esempio riportato dall'equazione 3.2 (pag. 61) che in [7] era indicata come esempio di termine Multi-CCS che, in presenza delle regole originarie, avrebbe generato un sistema P/T con un numero infinito di posti ma che veniva indicato come un potenziale sistema P/T finito in presenza di regole migliorate.

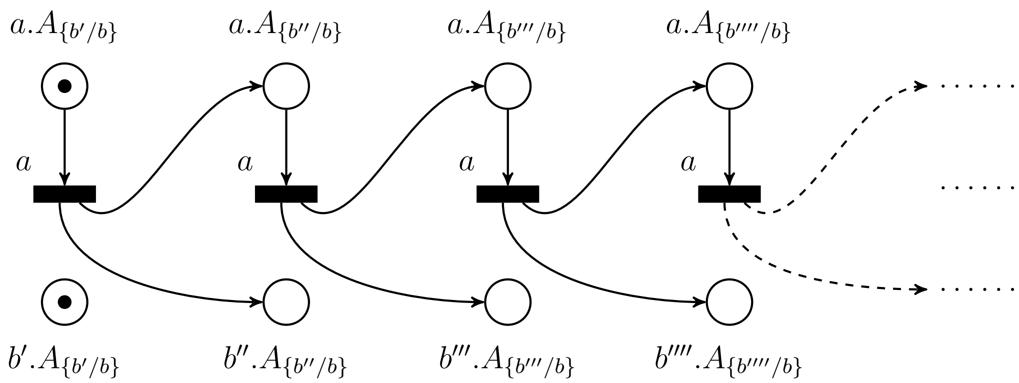

Figura 3.1: Sistema P/T, vecchia versione, per $A = (\nu b)(a.A \mid b.A)$

In figura 3.1 un frammento (comprendente i due posti della marcatura iniziale) dell'infinito sistema P/T risultante dalla semantica originaria; si tenga presente che i nomi b sono stati sostituiti da nomi ristretti b', b'', b''', ecc. – secondo lo stile usato in [6] e in [7] – e che le costanti A, secondo le regole di sostituzione di [7], sono state sostituite da costanti con sostituzione, che pure sarebbero inefficaci, per evitare l'unificazione di posti che devono rimanere separati.

Vediamo ora il sistema P/T generato dalla nuova semantica.

Per prima cosa osserviamo che, in base alle nuove regole di sostituzione, dal momento che $b, \overline{b} \notin fn(A)$, abbiamo $A\{\eta/b\} = A$.

Riguardo alla marcatura iniziale del sistema, premesso che è evidente che

$$\{b, \overline{b}\} \cap fn(a.A \mid b.A) = \{b\}$$

e quindi che, per ogni $e, i \in \mathbb{N}$,

$$eid(A, e, i, \emptyset) = eid(((\nu b)(a.A \mid b.A)), e, i, \{A\}) = \epsilon$$

abbiamo

$$\begin{aligned}
dec(A, 0, 0) &= dec((\nu b)(a.A \mid b.A), 0, 0) = dec((a.A \mid b.A)\{\eta_d/b\}, 0, 0)\\
&= dec((a.A)\{\eta/d\} \mid (b.A)\{\eta_d/b\}, 0, 0)\\
&= dec(a.(A\{\eta_d/b\}) \mid \eta_d.(A\{\eta_d/b\}), 0, 0)\\
&= dec(a.A \mid \eta_d.A, 0, 0) = dec(a.A, 1, 0) \oplus dec(\eta_d.A, 1, 1)\\
&= 1 \cdot (a.A, eid(A, 1, 0, \emptyset)) \oplus 1 \cdot (\eta_d.A, eid(A, 1, 1, \emptyset))\\
&= 1 \cdot (a.A, \epsilon) \oplus 1 \cdot (\eta_d.A, \epsilon) = \{a.A\} \oplus \{\eta_d.A\}
\end{aligned}$$

Abbiamo quindi ottenuto due posti – $\{a.A\}$, da $dec(a.A, 1, 0)$ e $\{\eta_d.A\}$, da $dec(\eta_d.A, 1, 1)$ – e dobbiamo determinare le relative transizioni ed eventuali nuovi posti.

Dai due posti è applicabile la regola (pref) che diventa

$$\begin{aligned}
dec(a.A, 1, 0) &\xrightarrow{a} dec(A, 1, 0)\\
dec(\eta_d.A, 1, 1) &\xrightarrow{\eta_d} dec(A, 1, 1)
\end{aligned}$$

Abbiamo due nuovi dec da calcolare; riguardo al primo

$$\begin{aligned}
dec(A, 1, 0) &= dec(((\nu b)(a.A \mid b.A)), 1, 0) = dec((a.A \mid b.A)\{\eta_d/b\}, 1, 0)\\
&= dec((a.A)\{\eta_d/b\} \mid (b.A)\{\eta_d/b\}, 1, 0)\\
&= dec(a.(A\{\eta_d/b\}) \mid \eta_d.(A\{\eta_d/b\}), 1, 0)\\
&= dec(a.A, 2, 0) \oplus dec(\eta_d.A, 2, 2)\\
&= 1 \cdot (a.A, eid(A, 2, 0, \emptyset)) \oplus 1 \cdot (\eta_d.A, eid(A, 2, 2, \emptyset))\\
&= 1 \cdot (a.A, \epsilon) \oplus 1 \cdot (\eta_d.A, \epsilon) = \{a.A\} \oplus \{\eta_d.A\}
\end{aligned}$$

e analogamente si prova che

$$dec(A, 1, 1) = \{a.A\} \oplus \{\eta_d.A\}$$

Abbiamo quindi che entrambe le transizioni portano ai due posti già noti. Non abbiamo quindi bisogno di iterare ulteriormente le regole. Osserviamo anche che alla transizione la cui etichetta è l'azione ristretta η_d non corrisponde alcuna transizione con azione ristretta complementare (non sarebbe comunque possibile, per come abbiamo costruito la nuova *dec*) quindi la seconda delle due transizioni viene filtrata.

In estrema sintesi, abbiamo che $Net(A) = (S_A, A_A, T_A, m_0)$ è pari a

$$\begin{aligned} S_A &= \{\{a.A\}, \{\eta_d.A\}\} \\ A_A &= \{a\} \\ T_A &= \{(\{a.A\}, a, \{a.A\} \oplus \{\eta_d.A\})\} \\ m_0 &= \{a.A\} \oplus \{\eta_d.A\} \end{aligned}$$

In figura 3.2 il sistema P/T risultante dalla nuova semantica.

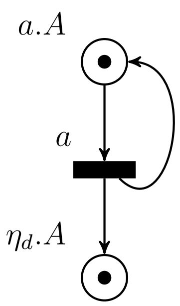

Figura 3.2: Sistema P/T, nuova versione, per $A = (\nu b)(a.A \mid b.A)$

Grazie al miglioramento relativo all'utilizzo delle azioni ristrette banali, nel caso di utilizzo delle azioni sostituite non significative (o solo diretto o solo complementare), abbiamo ottenuto un sistema P/T finito (due soli posti, un'unica transizione) al posto del sistema con infiniti posti (a causa delle infinite sostituzioni di b con infinite differenti azioni ristrette) e conseguenti infinite transizioni che avremmo ottenuto con la *dec* originaria.

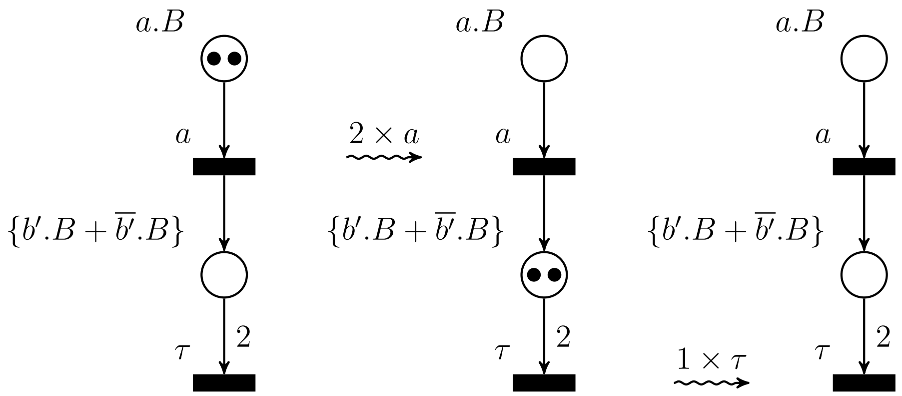

Figura 3.3: Sistema P/T, vecchia versione, per $q = a.B \mid a.B$

3.3.2 $q = a.B \mid a.B$, con $B = (\nu b)(b.\mathbf{0} + \overline{b}.\mathbf{0})$

Vediamo ora l'esempio riportato dall'equazione 3.6 (pag. 63) che la vecchia *dec* e le vecchie regole di transizione portavano – come da figura 3.3 – ad avere una sincronizzazione sulle azioni ristrette non presente nel termine Multi-CCS di partenza.

È evidente che

$$\{b, \overline{b}\} = fn(b.\mathbf{0} + \overline{b}.\mathbf{0})$$

e quindi le sostituzioni richiedono azioni ristrette non banali; osserviamo anche che, per ogni $e, i \in \mathbb{N}$,

$$eid(B, e, i, \emptyset) = eid((\nu b)(b.\mathbf{0} + \overline{b}.\mathbf{0}), e, i, \{B\}) = [\widehat{\eta_e}, \eta_i]$$

e quindi abbiamo

$$\begin{aligned} dec(q, 0, 0) &= dec(a.B \mid a.B, 0, 0) = dec(a.B, 1, 0) \oplus dec(a.B, 1, 1) \\ &= 1 \cdot (a.B, eid(B, 1, 0, \emptyset)) \oplus 1 \cdot (a.B, eid(B, 1, 1, \emptyset)) \\ &= 1 \cdot (a.B, [\widehat{\eta_1}, \eta_0]) \oplus 1 \cdot (a.B, [\widehat{\eta_1}, \eta_1]) \\ &= \{a.B\}[\widehat{\eta_1}, \eta_0] \oplus \{a.B\}[\widehat{\eta_1}, \eta_1] \end{aligned}$$

Abbiamo quindi ottenuto due posti distinti – $\{a.B\}[\widehat{\eta_1}, \eta_0]$, da $dec(a.B, 1, 0)$, e $\{a.B\}[\widehat{\eta_1}, \eta_1]$, da $dec(a.B, 1, 1)$ – e dobbiamo determinare le relative transizioni ed eventuali nuovi posti.

Dai due posti è applicabile la regola (pref) che, nello specifico, diventa

$$dec(a.B, 1, 0) \xrightarrow{a} dec(B, 1, 0)$$
$$dec(a.B, 1, 1) \xrightarrow{a} dec(B, 1, 1)$$

Abbiamo due nuovi *dec* da calcolare; per prima cosa osserviamo che

$$eid(\eta_0.\mathbf{0}, 2, 2, \emptyset) = eid(\mathbf{0}, 2, 2, \emptyset) = \epsilon$$
$$eid(\overline{\eta_0}.\mathbf{0}, 2, 4, \emptyset) = eid(\mathbf{0}, 2, 4, \emptyset) = \epsilon$$
$$eid(\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}, 1, 2, \emptyset) = eid(\overline{\eta_0}.\mathbf{0}, 2, 4, \emptyset) = \epsilon$$

Per il primo nuovo *dec* abbiamo

$$\begin{aligned} dec(B, 1, 0) &= dec((\nu b)(b.\mathbf{0} + \overline{b}.\mathbf{0}), 1, 0) = dec((b.\mathbf{0} + \overline{b}.\mathbf{0})\{\eta_0/b\}, 1, 2) \\ &= dec(\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}, 1, 2) \\ &= 1 \cdot (\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}, eid(\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}, 1, 2, \emptyset)) \\ &= 1 \cdot (\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}, \epsilon) = \{\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}\} \end{aligned}$$

Analogamente si prova che

$$dec(B, 1, 1) = dec(\eta_1.\mathbf{0} + \overline{\eta_1}.\mathbf{0}, 1, 3) = \{\eta_1.\mathbf{0} + \overline{\eta_1}.\mathbf{0}\}$$

Le due transizioni di cui sopra quindi diventano quindi

$$\{a.B\}[\widehat{\eta_1}, \eta_0] \xrightarrow{a} \{\eta_0.\mathbf{0} + \overline{\eta}_0.\mathbf{0}\}$$
$$\{a.B\}[\widehat{\eta_1}, \eta_1] \xrightarrow{a} \{\eta_1.\mathbf{0} + \overline{\eta}_1.\mathbf{0}\}$$

Le transizioni hanno determinato due nuovi posti: $\{\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}\}$, da $dec(\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}, 1, 2)$, e $\{\eta_1.\mathbf{0} + \overline{\eta_1}.\mathbf{0}\}$, da $dec(\eta_1.\mathbf{0} + \overline{\eta_1}.\mathbf{0}, 1, 3)$. Dobbiamo determinare ulteriori possibili transizioni; trattandosi di posti somma, le regole applicabili sono (sum-1) e (sum-2) che, limitandosi al primo dei due posti, diventano

$$\frac{dec(\eta_0.\mathbf{0}, 2, 2) \xrightarrow{\sigma} m}{dec(\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}, 1, 2) \xrightarrow{\sigma} m}$$

$$\frac{dec(\overline{\eta_0}.\mathbf{0}, 2, 4) \xrightarrow{\sigma} m}{dec(\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}, 1, 2) \xrightarrow{\sigma} m}$$

Dobbiamo quindi determinare i due posti $dec(\eta_0.\mathbf{0}, 2, 2)$ e $dec(\overline{\eta_0}.\mathbf{0}, 2, 4)$ e le possibili transizioni; abbiamo

$$dec(\eta_0.\mathbf{0}, 2, 2) = 1 \cdot (\eta_0.\mathbf{0}, eid(\mathbf{0}, 2, 2, \emptyset)) = 1 \cdot (\eta_0.\mathbf{0}, \epsilon) = \{\eta_0.\mathbf{0}\}$$
$$dec(\overline{\eta_0}.\mathbf{0}, 2, 4) = 1 \cdot (\overline{\eta_0}.\mathbf{0}, eid(\mathbf{0}, 2, 4, \emptyset)) = 1 \cdot (\overline{\eta_0}.\mathbf{0}, \epsilon) = \{\overline{\eta_0}.\mathbf{0}\}$$

Da $\{\eta_0.\mathbf{0}\}$, alias $dec(\eta_0.\mathbf{0}, 2, 2)$, e da $\{\overline{\eta_0}.\mathbf{0}\}$, alias $dec(\overline{\eta_0}.\mathbf{0}, 2, 4)$, sono possibili le seguenti transizioni (pref)

$$dec(\eta_0.\mathbf{0}, 2, 2) \xrightarrow{\eta_0} dec(\mathbf{0}, 2, 2)$$
$$dec(\overline{\eta_0}.\mathbf{0}, 2, 4) \xrightarrow{\overline{\eta_0}} dec(\mathbf{0}, 2, 4)$$

È però ovvio che, per ogni $e, i \in \mathbb{N}$,

$$dec(\mathbf{0}, e, i) = \emptyset$$

quindi le due transizioni precedenti diventano

$$\{\eta_0.\mathbf{0}\} \xrightarrow{\eta_0} \emptyset$$
$$\{\overline{\eta_0}.\mathbf{0}\} \xrightarrow{\overline{\eta_0}} \emptyset$$

e quindi, le transizioni (sum-1) e (sum-2) da cui eravamo partiti,

$$\frac{\{\eta_0.\mathbf{0}\} \xrightarrow{\eta_0} \emptyset}{\{\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}\} \xrightarrow{\eta_0} \emptyset} \qquad \frac{\{\overline{\eta_0}.\mathbf{0}\} \xrightarrow{\overline{\eta_0}} \emptyset}{\{\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}\} \xrightarrow{\overline{\eta_0}} \emptyset}$$

ovvero sono previste le transizioni

$$\{\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}\} \xrightarrow{\eta_0} \emptyset \qquad \{\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}\} \xrightarrow{\overline{\eta_0}} \emptyset$$

Analogamente si può provare che dal posto $\{\eta_1.\mathbf{0} + \overline{\eta_1}.\mathbf{0}\}$ sono previste le transizioni

$$\{\eta_1.\mathbf{0} + \overline{\eta_1}.\mathbf{0}\} \xrightarrow{\eta_1} \emptyset \qquad \{\eta_1.\mathbf{0} + \overline{\eta_1}.\mathbf{0}\} \xrightarrow{\overline{\eta_1}} \emptyset$$

Ricordiamo però che dobbiamo filtrare le transizioni basate su azioni ristrette, come η_0, η_1 e le relative complementari, che ci interessano solo se si sincronizzano in un'azione τ. Le ultime transizioni viste sono quindi rilevanti quando si sincronizzano, grazie alla regola (com), ovvero se nel posto $\{\eta_0.\mathbf{0}+\overline{\eta_0}.\mathbf{0}\}$ (o, analogamente, nel posto $\{\eta_1.\mathbf{0}+\overline{\eta_1}.\mathbf{0}\}$) sono presenti almeno due token in grado di causare una auto-sincronizzazione. Così non può essere: partendo con un solo token in $\{a.B\}[\widehat{\eta_1},\eta_0]$ e un solo token in $\{a.B\}[\widehat{\eta_1},\eta_1]$ potremo avere al massimo un token sia in $\{\eta_0.\mathbf{0}+\overline{\eta_0}.\mathbf{0}\}$ che in $\{\eta_1.\mathbf{0}+\overline{\eta_1}.\mathbf{0}\}$ e quindi le transizioni basate sulle azioni η_0, η_1 e complementari verranno eliminate.

In sintesi, abbiamo che $Net(q) = (S_q, A_q, T_q, m_0)$ è pari a

$$
\begin{aligned}
S_q &= \{\{a.B\}[\widehat{\eta_1},\eta_0], \{a.B\}[\widehat{\eta_1},\eta_1], \{\eta_0.\mathbf{0}+\overline{\eta_0}.\mathbf{0}\}, \{\eta_1.\mathbf{0}+\overline{\eta_1}.\mathbf{0}\}\}\\
A_q &= \{a\}\\
T_q &= \{(\{a.B\}[\widehat{\eta_1},\eta_0], a, \{\eta_0.\mathbf{0}+\overline{\eta_0}.\mathbf{0}\}), (\{a.B\}[\widehat{\eta_1},\eta_1], a, \{\eta_1.\mathbf{0}+\overline{\eta_1}.\mathbf{0}\})\}\\
m_0 &= \{a.B\}[\widehat{\eta_1},\eta_0] \oplus \{a.B\}[\widehat{\eta_1},\eta_1]
\end{aligned}
$$

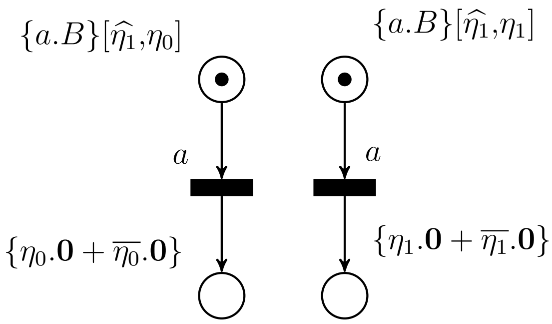

Figura 3.4: Sistema P/T, nuova versione, per $q = a.B \mid a.B$

Non è quindi più presente, come possiamo vedere in figura 3.4, l'azione τ che era invece (erroneamente) presente nel sistema $Net(q)$ ottenuto con le vecchie regole; infatti, distinguendo $\{a.B\}[\widehat{\eta_1},\eta_0]$ da $\{a.B\}[\widehat{\eta_1},\eta_1]$ – che con le vecchie regole erano unificati – siamo riusciti a impedire di avere due token su un posto del tipo $\{\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}\}$ e quindi a impedire un'auto-sincronizzazione che nell'originale termine Multi-CCS non era prevista.

3.3.3 $q = a.(\nu a)B \mid a.(\nu a)B$, con $B = b.(\nu b)C + (a.\mathbf{0} + \overline{a}.\mathbf{0})$ e con $C = b.\mathbf{0} + \overline{b}.\mathbf{0}$

Come abbiamo visto, la nuova versione della semantica prevede, per tenere separati posti che con la versione originaria si unificavano erroneamente, il calcolo di un "identificativo extra" che non aggiunge nulla (o, se vogliamo, aggiunge l'identificativo nullo ϵ) quando non è necessario tenere separati i posti e invece aggiunge un termine costituito dalla coppia $[\widehat{\eta_e}, \eta_i]$ (secondo la rappresentazione che ho adottato) quando è opportuno farlo. Abbiamo concretamente visto, nell'esempio precedente, perché questo identificativo extra è necessario e la sua efficacia nell'evitare, nella nuova versione della semantica P/T, una sincronizzazione sgradita e non prevista nella semantica LTS.

È però assolutamente legittimo chiedersi: perché usare due termini in tale identificativo? Non si poteva usare un identificativo costituito (quando necessario) dal solo valore $[\eta_i]$, ovvero da un nome ristretto associato al termine?

In effetti, questa era la mia prima ipotesi. Ero inizialmente convinto che fosse sufficiente un identificativo extra più sintetico, costituito dal solo $[\eta_i]$.

Il professor Gorrieri mi ha però segnalato un controesempio che mi ha dimostrato il contrario. Ne mostro di seguito una versione rielaborata.

Sia q il processo definito come segue:

$$\begin{aligned} q &= a.(\nu a)B \mid a.(\nu a)B \\ B &\stackrel{def}{=} b.(\nu b)C + (a.\mathbf{0} + \overline{a}.\mathbf{0}) \\ C &\stackrel{def}{=} b.\mathbf{0} + \overline{b}.\mathbf{0} \end{aligned}$$

Supponiamo che la funzione *eid* restituisca un identificativo extra "ridotto", ovvero costituito solo (quando opportuno) dal terzo parametro.

Osserviamo che $a, \overline{a} \in fn(B)$ e quindi, per e ed i generici, abbiamo che

$$eid(a.(\nu a)B, e, i, \emptyset) = eid((\nu a)B, e, i, \emptyset) = \eta_i$$

Calcoliamo ora $dec(a.(\nu a)B, e, i)$, ancora per e ed i generici

$$\begin{aligned} dec(a.(\nu a)B, e, i) &= 1 \cdot (a.(\nu a)B, eid(a.(\nu a)B, e, i, \emptyset)) \\ &= 1 \cdot (a.(\nu a)B, \eta_i) = \{a.(\nu a)B\}[\eta_i] \end{aligned}$$

Possiamo rilevare che l'ultimo valore calcolato non dipende dal secondo termine, ovvero che, per ogni $e, f \in \mathbb{N}$, abbiamo

$$dec(a.(\nu a)B, e, i) = \{a.(\nu a)B\}[\eta_i] = dec(a.(\nu a)B, f, i)$$

Possiamo ora calcolare la decomposizione di q che ci restituisce

$$\begin{aligned} dec(q, 0, 0) &= dec(a.(\nu a)B \mid a.(\nu a)B, 0, 0) \\ &= dec(a.(\nu a)B, 1, 0) \oplus dec(a.(\nu a)B, 1, 1) \\ &= \{a.(\nu a)B\}[\eta_0] \oplus \{a.(\nu a)B\}[\eta_1] \end{aligned}$$

Possiamo quindi rilevare che, almeno per ora, l'identificativo extra ridotto è adeguato in quanto riesce a tenere separati i due posti della marcatura iniziale.

Vediamo ora le transizioni.

Abbiamo due posti ottenuti da processi normalmente prefissi; la regola applicabile è la (pref), ovvero (ancora per e ed i generici),

$$dec(a.(\nu a)B, e, i) \xrightarrow{a} dec((\nu a)B, e, i)$$

Ricordando però che, per ogni $e, f \in \mathbb{N}$, $dec(a.(\nu a)B, e, i) = [\eta_i] = dec(a.(\nu a)B, f, i)$, abbiamo che

$$\begin{aligned} \{a.(\nu a)B\}[\eta_0] = dec(a.(\nu a)B, e, 0) \xrightarrow{a} dec((\nu a)B, e, 0) \\ \{a.(\nu a)B\}[\eta_1] = dec(a.(\nu a)B, e, 1) \xrightarrow{a} dec((\nu a)B, e, 1) \end{aligned}$$

per ogni $e \in \mathbb{N}$.

Calcoliamo ora $dec((\nu a)B, e, i)$, al variare di $e, i \in \mathbb{N}$; per prima cosa osserviamo che $b, \overline{b} \in fn(C)$; abbiamo perciò, per e ed i e η_k generici

$$\begin{aligned} eid(b.(\nu b)C + (\eta_k.\mathbf{0} + \overline{\eta_k}.\mathbf{0}), e, i, \emptyset) &= eid(b.(\nu b)C, e+1, i, \emptyset) \\ &= eid((\nu b)C, e+1, i, \emptyset) = \eta_i \end{aligned}$$

e quindi, ricordando che $a, \overline{a} \in fn(B)$ e osservando che $a, \overline{a} \notin fn(C)$, abbiamo

$$\begin{aligned} dec((\nu a)B, e, i) &= dec(B\{\eta_i/a\}, e, i+2^e) = dec(B_{\{\eta_i/a\}}, e, i+2^e) \\ &= dec((b.(\nu b)C + (a.\mathbf{0} + \overline{a}.\mathbf{0}))\{\eta_i/a\}, e, i+2^e) \\ &= dec((b.(\nu b)C)\{\eta_i/a\} + (a.\mathbf{0} + \overline{a}.\mathbf{0})\{\eta_i/a\}, e, i+2^e) \\ &= dec(b.(((\nu b)C)\{\eta_i/a\}) \\ &\qquad + ((a.\mathbf{0})\{\eta_i/a\} + (\overline{a}.\mathbf{0})\{\eta_i/a\}), e, i+2^e) \\ &= dec(b.(\nu b)(C\{\eta_i/a\}) \\ &\qquad + (\eta_i.(\mathbf{0}\{\eta_i/a\}) + \overline{\eta_i}.(\mathbf{0}\{\eta_i/a\})), e, i+2^e) \\ &= dec(b.(\nu b)C + \eta_i.\mathbf{0} + \overline{\eta_i}.\mathbf{0}, e, i+2^e) \\ &= 1 \cdot (b.(\nu b)C + \eta_i.\mathbf{0} + \overline{\eta_i}.\mathbf{0}, \\ &\qquad eid(b.(\nu b)C + \eta_i.\mathbf{0} + \overline{\eta_i}.\mathbf{0}, e, i+2^e)) \\ &= 1 \cdot (b.(\nu b)C + \eta_i.\mathbf{0} + \overline{\eta_i}.\mathbf{0}, \eta_{i+2^e}) \\ &= \{b.(\nu b)C + \eta_i.\mathbf{0} + \overline{\eta_i}.\mathbf{0}\}[\eta_{i+2^e}] \end{aligned}$$

Ricordando quanto visto in precedenza, abbiamo che, per ogni $e \in \mathbb{N}$, sono attive le seguenti transizioni

$$\{a.(\nu a)B\}[\eta_0] \xrightarrow{a} dec((\nu a)B, e, 0) = \{b.(\nu b)C + \eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}\}[\eta_{2^e}]$$
$$\{a.(\nu a)B\}[\eta_1] \xrightarrow{a} dec((\nu a)B, e, 1) = \{b.(\nu b)C + \eta_1.\mathbf{0} + \overline{\eta_1}.\mathbf{0}\}[\eta_{1+2^e}]$$

Possiamo già vedere una cosa estremamente sgradevole: abbiamo – al variare di $e \in \mathbb{N}$ – infinite transizioni da singoli posti.

Ma non è finita: fissando $e = 1$ (nel primo gruppo di transizioni) ed $e = 0$ (nel secondo) otteniamo le due seguenti transizioni

$$\{a.(\nu a)B\}[\eta_0] \xrightarrow{a} dec((\nu a)B, 1, 0) = \{b.(\nu b)C + \eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}\}[\eta_2]$$

$$\{a.(\nu a)B\}[\eta_1] \xrightarrow{a} dec((\nu a)B, 0, 1) = \{b.(\nu b)C + \eta_1.\mathbf{0} + \overline{\eta_1}.\mathbf{0}\}[\eta_2]$$

Vediamo ora quali transizioni otteniamo applicando la regola (sum-1) ai due termini di destinazione, ovvero a $dec((\nu a)B, 1, 0)$ e a $dec((\nu a)B, 0, 1)$; ricordando che

$$\begin{aligned} dec((\nu a)B, 1, 0) &= dec(b.(\nu b)C + (\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}), 1, 2) \\ dec((\nu a)B, 0, 1) &= dec(b.(\nu b)C + (\eta_1.\mathbf{0} + \overline{\eta_1}.\mathbf{0}), 0, 2) \end{aligned}$$

applicando la regola (sum-1) abbiamo

$$\frac{dec(b.(\nu b)C, 2, 2) \xrightarrow{\sigma} m}{dec(b.(\nu b)C + (\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}), 1, 2) \xrightarrow{\sigma} m}$$

$$\frac{dec(b.(\nu b)C, 1, 2) \xrightarrow{\sigma} m}{dec(b.(\nu b)C + (\eta_1.\mathbf{0} + \overline{\eta_1}.\mathbf{0}), 0, 2) \xrightarrow{\sigma} m}$$

Dobbiamo quindi individuare le possibili transizioni in partenza da $dec(b.(\nu b)B, 2, 2)$ e da $dec(b.(\nu b)B, 1, 2)$. Osserviamo che, essendo $b.(\nu b)B$ un termine normalmente prefisso, da queste marcature sono possibili solo transizioni giustificate (una per ogni marcatura) dalla regola (pref), ovvero

$$\begin{aligned} dec(b.(\nu b)C, 2, 2) &\xrightarrow{b} dec((\nu b)C, 2, 2) \\ dec(b.(\nu b)C, 1, 2) &\xrightarrow{b} dec((\nu b)C, 1, 2) \end{aligned}$$

Calcoliamo quindi $dec((\nu b)C, e, i)$, al variare di $e, i \in \mathbb{N}$; dal momento che è evidente che, al variare di $e, i \in \mathbb{N}$ e di $\eta_x \in \mathcal{N}$, abbiamo

$$\begin{aligned} eid(\eta_x.\mathbf{0} + \overline{\eta_x}.\mathbf{0}, e, i) &= eid(\overline{\eta_x}.\mathbf{0}, e+1, i+2^e) \\ &= eid(\mathbf{0}, e+1, i+2^e) = \epsilon \end{aligned}$$

e ricordando che $b, \overline{b} \in fn(C)$, abbiamo

$$\begin{aligned} dec((\nu b)C, e, i) &= dec(C\{\eta_i/b\}, e, i+2^e) = dec(C_{\{\eta_i/b\}}, e, i+2^e) \\ &= dec((b.\mathbf{0} + \overline{b}.\mathbf{0})\{\eta_i/b\}, e, i+2^e) \end{aligned}$$

$$
\begin{aligned}
&= dec((b.\mathbf{0})\{\eta_i/b\} + (\bar{b}.\mathbf{0})\{\eta_i/b\}, e, i + 2^e) \\
&= dec(\eta_i.(\mathbf{0}\{\eta_i/b\}) + \overline{\eta_i}.(\mathbf{0}\{\eta_i/b\}), e, i + 2^e) \\
&= dec(\eta_i.\mathbf{0} + \overline{\eta_i}.\mathbf{0}, e, i + 2^e) \\
&= 1 \cdot (\eta_i.\mathbf{0} + \overline{\eta_i}.\mathbf{0}, eid(\eta_i.\mathbf{0} + \overline{\eta_i}.\mathbf{0}, e, i + 2^e)) \\
&= 1 \cdot (\eta_i.\mathbf{0} + \overline{\eta_i}.\mathbf{0}, \epsilon) = \{\eta_i.\mathbf{0} + \overline{\eta_i}.\mathbf{0}\}
\end{aligned}
$$

Abbiamo quindi

$$
\begin{aligned}
dec(b.(\nu b)C, 2, 2) &\xrightarrow{b} dec((\nu b)C, 2, 2) = \{\eta_2.\mathbf{0} + \overline{\eta_2}.\mathbf{0}\} \\
dec(b.(\nu b)C, 1, 2) &\xrightarrow{b} dec((\nu b)C, 1, 2) = \{\eta_2.\mathbf{0} + \overline{\eta_2}.\mathbf{0}\}
\end{aligned}
$$

e, applicando la regola (sum-1),

$$
\frac{dec(b.(\nu b)C, 2, 2) \xrightarrow{b} \{\eta_2.\mathbf{0} + \overline{\eta_2}.\mathbf{0}\}}{dec(b.(\nu b)C + (\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}), 1, 2) \xrightarrow{b} \{\eta_2.\mathbf{0} + \overline{\eta_2}.\mathbf{0}\}}
$$

$$
\frac{dec(b.(\nu b)C, 1, 2) \xrightarrow{b} \{\eta_2.\mathbf{0} + \overline{\eta_2}.\mathbf{0}\}}{dec(b.(\nu b)C + (\eta_1.\mathbf{0} + \overline{\eta_1}.\mathbf{0}), 0, 2) \xrightarrow{b} \{\eta_2.\mathbf{0} + \overline{\eta_2}.\mathbf{0}\}}
$$

ovvero

$$
\begin{aligned}
dec((\nu a)B, 1, 0) &= dec(b.(\nu b)C + (\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}), 1, 2) \xrightarrow{b} \{\eta_2.\mathbf{0} + \overline{\eta_2}.\mathbf{0}\} \\
dec((\nu a)B, 0, 1) &= dec(b.(\nu b)C + (\eta_1.\mathbf{0} + \overline{\eta_1}.\mathbf{0}), 0, 2) \xrightarrow{b} \{\eta_2.\mathbf{0} + \overline{\eta_2}.\mathbf{0}\}
\end{aligned}
$$

Per quanto riguarda $dec((\nu b)C, e, 2) = \{\eta_2.\mathbf{0} + \overline{\eta_2}.\mathbf{0}\}$ (al variare di $e \in \mathbb{N}$), ricordando che

$$
dec((\nu b)C, e, 2) = dec(\eta_2.\mathbf{0} + \overline{\eta_2}.\mathbf{0}, e, 2 + 2^e)
$$

abbiamo che le transizioni supportate possono essere giustificate solamente dalle regole (sum-1) e (sum-2), ovvero

$$
\frac{dec(\eta_2.\mathbf{0}, e + 1, 2 + 2^e) \xrightarrow{\sigma} m_1}{dec(\eta_2.\mathbf{0} + \overline{\eta_2}.\mathbf{0}, e, 2 + 2^e) \xrightarrow{\sigma} m_1}
$$

$$\frac{dec(\overline{\eta_2}.\mathbf{0}, e+1, 2+2^{e+1}) \xrightarrow{\sigma} m_2}{dec(\eta_2.\mathbf{0}+\overline{\sigma}.\mathbf{0}, e, 2+2^e) \xrightarrow{\sigma} m_2}$$

Per quanto riguarda $dec(\eta_2.\mathbf{0}, e+1, 2+2^e)$ e $dec(\overline{\eta_2}.\mathbf{0}, e+1, 2+2^{e+1})$, è evidente che possono supportare solo transizioni supportate dalla regola (pref), ovvero

$$dec(\eta_2.\mathbf{0}, e+1, 2+2^e) \xrightarrow{\eta_2} dec(\mathbf{0}, e+1, 2+2^e) = \emptyset$$
$$dec(\overline{\eta_2}.\mathbf{0}, e+1, 2+2^{e+1}) \xrightarrow{\overline{\eta_2}} dec(\mathbf{0}, e+1, 2+2^{e+1}) = \emptyset$$

ovvero, tornando a $dec(\eta_2.\mathbf{0}+\overline{\eta_2}.\mathbf{0}, e, 2+2^e) = \{\eta_2.\mathbf{0}+\overline{\eta_2}.\mathbf{0}\}$, abbiamo

$$\frac{dec(\eta_2.\mathbf{0}, e+1, 2+2^e) \xrightarrow{\eta_2} \emptyset}{dec(\eta_2.\mathbf{0}+\overline{\eta_2}.\mathbf{0}, e, 2+2^e) \xrightarrow{\eta_2} \emptyset}$$

$$\frac{dec(\overline{\eta_2}.\mathbf{0}, e+1, 2+2^{e+1}) \xrightarrow{\overline{\eta_2}} \emptyset}{dec(\eta_2.\mathbf{0}+\overline{\eta_2}.\mathbf{0}, e, 2+2^e) \xrightarrow{\overline{\eta_2}} \emptyset}$$

e quindi

$$\{\eta_2.\mathbf{0}+\overline{\eta_2}.\mathbf{0}\} = dec(\eta_2.\mathbf{0}+\overline{\eta_2}.\mathbf{0}, e, 2+2^e) \xrightarrow{\eta_2} \emptyset$$
$$\{\eta_2.\mathbf{0}+\overline{\eta_2}.\mathbf{0}\} = dec(\eta_2.\mathbf{0}+\overline{\eta_2}.\mathbf{0}, e, 2+2^e) \xrightarrow{\overline{\eta_2}} \emptyset$$

Queste due transizioni non sono effettivamente disponibili direttamente – dal momento che sono basate sulle azioni ristrette η_2 e $\overline{\eta_2}$ – ma possono sincronizzarsi, tramite la regola (com), nella seguente

$$\frac{\{\eta_2.\mathbf{0}+\overline{\eta_2}.\mathbf{0}\} \xrightarrow{\eta_2} \emptyset \quad \{\eta_2.\mathbf{0}+\overline{\eta_2}.\mathbf{0}\} \xrightarrow{\overline{\eta_2}} \emptyset}{\{\eta_2.\mathbf{0}+\overline{\eta_2}.\mathbf{0}\} \oplus \{\eta_2.\mathbf{0}+\overline{\eta_2}.\mathbf{0}\} \xrightarrow{\tau} \emptyset \oplus \emptyset} \qquad sync(\eta_2, \overline{\eta_2}, \tau)$$

ovvero

$$2 \cdot \{\eta_2.\mathbf{0}+\overline{\eta_2}.\mathbf{0}\} \xrightarrow{\tau} \emptyset$$

Riassumendo, in figura 3.5 (pagina a fronte) abbiamo un frammento dell'infinito sistema P/T generato da $dec(q, 0, 0)$; la marcatura iniziale è pari alla

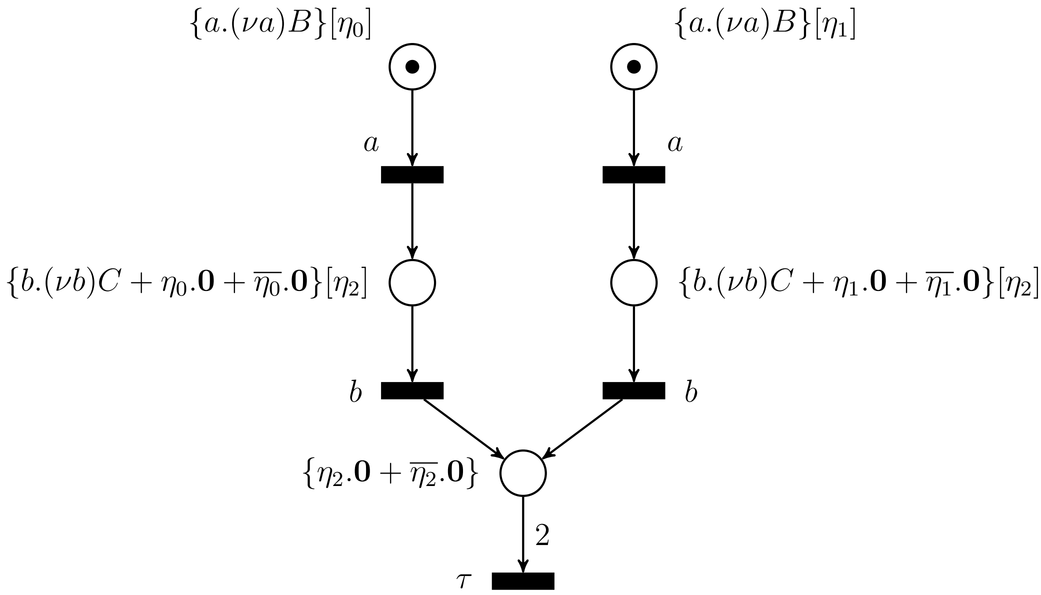

Figura 3.5: Frammento del sistema P/T, *eid* ridotto, per $q = a.(\nu a)B \mid a.(\nu a)B$

somma di due marcature (distinte) che supportano (ognuna) infinite transizioni (e già questo è sgradevole) e da due di queste (in figura, con etichetta a) è possibile arrivare (dopo un'altra transizione per parte, con etichetta b) a un medesimo posto – $\{\eta_2.\mathbf{0}+\overline{\eta_2}.\mathbf{0}\}$ – dalla quale i due token possono attivare una transizione con etichetta τ che non è prevista nell'LTS generato da q.

Ovvero: anche se dimostrarlo è stato molto più complicato rispetto all'esempio precedente, abbiamo che con un identificatore extra ridotto (col solo secondo elemento) abbiamo perso la bisimilitudine con la semantica LTS.

Vediamo ora cosa succede con l'identificativo extra completo, ovvero quello effettivamente previsto dalla definizione 3.2.8 (pag. 69).

Ricordando che $a, \overline{a} \in fn(B)$, per e ed i generici, abbiamo che

$$eid(a.(\nu a)B, e, i, \emptyset) = eid((\nu a)B, e, i, \emptyset) = [\widehat{\eta_e}, \eta_i]$$

Calcoliamo ora $dec(a.(\nu a)B, e, i)$, ancora per e ed i generici

$$dec(a.(\nu a)B, e, i) = 1 \cdot (a.(\nu a)B, eid(a.(\nu a)B, e, i, \emptyset))$$

$$= 1 \cdot (a.(\nu a)B, \eta_i) = \{a.(\nu a)B\}[\widehat{\eta_e}, \eta_i]$$

Contrariamente a quanto visto nell'ipotesi identificativo ridotto, ora abbiamo che anche il primo valore numerico di supporto è significativo; infatti, se $e \neq f$, abbiamo

$$\begin{aligned} dec(a.(\nu a)B, e, i) &= \{a.(\nu a)B\}[\widehat{\eta_e}, \eta_i] \\ &\neq \{a.(\nu a)B\}[\widehat{\eta_f}, \eta_i] = dec(a.(\nu a)B, f, i) \end{aligned}$$

Possiamo ora ricalcolare la decomposizione di q che ci restituisce

$$\begin{aligned} dec(q, 0, 0) &= dec(a.(\nu a)B \mid a.(\nu a)B, 0, 0) \\ &= dec(a.(\nu a)B, 1, 0) \oplus dec(a.(\nu a)B, 1, 1) \\ &= \{a.(\nu a)B\}[\widehat{\eta_1}, \eta_0] \oplus \{a.(\nu a)B\}[\widehat{\eta_1}, \eta_1] \end{aligned}$$

Per ora non cambia nulla: i due posti sono distinti ma lo erano anche prima.

Vediamo ora le transizioni.

Applicando ancora la regola (pref), ovvero (ancora per e ed i generici), abbiamo

$$dec(a.(\nu a)B, e, i) \xrightarrow{a} dec((\nu a)B, e, i)$$

ovvero

$$\begin{aligned} \{a.(\nu a)B\}[\widehat{\eta_1}, \eta_0] = dec(a.(\nu a)B, 1, 0) \xrightarrow{a} dec((\nu a)B, 1, 0) \\ \{a.(\nu a)B\}[\widehat{\eta_1}, \eta_1] = dec(a.(\nu a)B, 1, 1) \xrightarrow{a} dec((\nu a)B, 1, 1) \end{aligned}$$

Osserviamo però che ora i posti della marcatura iniziale non dipendono più dal solo secondo valore numerico, ma anche dal primo; supportano quindi una sola (per ognuno) e non infinite transizioni.

Calcoliamo ora $dec((\nu a)B, e, i)$, al variare di e ed i generici; per prima cosa osserviamo che $b, \overline{b} \in fn(C)$; abbiamo, per e ed i e η_k generici

$$eid(b.(\nu b)C + (\eta_k.\mathbf{0} + \overline{\eta_k}.\mathbf{0}), e, i, \emptyset) = eid(b.(\nu b)C, e + 1, i, \emptyset)$$

$$= eid((\nu b)C, e+1, i, \emptyset) = (\widehat{\eta_{e+1}}, \eta_i)$$

e quindi, ricordando che $a, \overline{a} \in fn(B)$ e osservando che $a, \overline{a} \notin fn(C)$, abbiamo (saltando i passaggi uguali all'ipotesi precedente)

$$\begin{aligned} dec((\nu a)B, e, i) &= 1 \cdot (b.(\nu b)C + \eta_i.\mathbf{0} + \overline{\eta_i}.\mathbf{0}, \\ &\qquad eid(b.(\nu b)C + \eta_i.\mathbf{0} + \overline{\eta_i}.\mathbf{0}, e, i+2^e)) \\ &= 1 \cdot (b.(\nu b)C + \eta_i.\mathbf{0} + \overline{\eta_i}.\mathbf{0}, (\widehat{\eta_{e+1}}, \eta_{i+2^e})) \\ &= \{b.(\nu b)C + \eta_i.\mathbf{0} + \overline{\eta_i}.\mathbf{0}\}[\widehat{\eta_{e+1}}, \eta_{i+2^e}] \end{aligned}$$

Quindi, in questo caso dai posti della marcatura iniziale sono attive solamente le seguenti transizioni

$$\{a.(\nu a)B\}[\widehat{\eta_1}, \eta_0] \xrightarrow{a} dec((\nu a)B, 1, 0) = \{b.(\nu b)C + \eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}\}[\widehat{\eta_2}, \eta_2]$$
$$\{a.(\nu a)B\}[\widehat{\eta_1}, \eta_1] \xrightarrow{a} dec((\nu a)B, 1, 1) = \{b.(\nu b)C + \eta_1.\mathbf{0} + \overline{\eta_1}.\mathbf{0}\}[\widehat{\eta_2}, \eta_3]$$

Prendiamo ora in considerazione i posti di destinazione di queste marcature; per prima cosa osserviamo che prevedono transizioni (applicando la regola (sum-2) e poi la regola (sum-1) o la regola (sum-2)) con etichette pari ad azioni ristrettte, ovvero η_0, $\overline{\eta_0}$, η_1, $\overline{\eta_1}$. Tali transizioni sono significative solo in caso di sincronizzazioni, ovvero solo se nel posto in questione sono presenti 2 token; partendo con un solo token nei due posti inziali, in questi posti possono essere presenti zero o un solo token (vedremo che non potranno arrivare altri token da altre transizioni) quindi queste transizioni non faranno parte della sottorete raggiungibile dalla marcatura iniziale.

Invece, applicando la regola (sum-1); ricordiamo che

$$dec((\nu a)B, 1, 0) = dec(b.(\nu b)C + (\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}), 1, 2)$$
$$dec((\nu a)B, 1, 1) = dec(b.(\nu b)C + (\eta_1.\mathbf{0} + \overline{\eta_1}.\mathbf{0}), 1, 3)$$

Applicando la regola (sum-1) abbiamo

$$\frac{dec(b.(\nu b)C, 2, 2) \xrightarrow{\sigma} m}{dec(b.(\nu b)C + (\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}), 1, 2) \xrightarrow{\sigma} m}$$

$$\frac{dec(b.(\nu b)C, 2, 3) \xrightarrow{\sigma} m}{dec(b.(\nu b)C + (\eta_1.\mathbf{0} + \overline{\eta_1}.\mathbf{0}), 1, 3) \xrightarrow{\sigma} m}$$

Dobbiamo quindi individuare le possibili transizioni in partenza da $dec(b.(\nu b)B, 2, 2)$ e da $dec(b.(\nu b)B, 2, 3)$. Applichiamo anche in questo caso la regola (pref), ovvero

$$dec(b.(\nu b)C, 2, 2) \xrightarrow{b} dec((\nu b)C, 2, 2)$$
$$dec(b.(\nu b)C, 2, 3) \xrightarrow{b} dec((\nu b)C, 2, 3)$$

Calcoliamo quindi, con le nuove regole, $dec((\nu b)C, e, i)$, al variare di $e, i \in \mathbb{N}$; al variare di $e, i \in \mathbb{N}$ e di $\eta_x \in \mathcal{N}$, abbiamo anche in questo caso

$$\begin{aligned} eid(\eta_x.\mathbf{0} + \overline{\eta_x}.\mathbf{0}, e, i) &= eid(\overline{\eta_x}.\mathbf{0}, e+1, i+2^e) \\ &= eid(\mathbf{0}, e+1, i+2^e) = \epsilon \end{aligned}$$

e ricordando che $b, \overline{b} \in fn(C)$, abbiamo ancora (salto i passaggi intermedi, uguali a quelli dell'ipotesi precedente)

$$dec((\nu b)C, e, i) = \{\eta_i.\mathbf{0} + \overline{\eta_i}.\mathbf{0}\}$$

Abbiamo quindi

$$dec(b.(\nu b)C, 2, 2) \xrightarrow{b} dec((\nu b)C, 2, 2) = \{\eta_2.\mathbf{0} + \overline{\eta_2}.\mathbf{0}\}$$
$$dec(b.(\nu b)C, 2, 3) \xrightarrow{b} dec((\nu b)C, 2, 3) = \{\eta_3.\mathbf{0} + \overline{\eta_3}.\mathbf{0}\}$$

e, applicando la regola (sum-1),

$$\frac{dec(b.(\nu b)C, 2, 2) \xrightarrow{b} \{\eta_2.\mathbf{0} + \overline{\eta_2}.\mathbf{0}\}}{dec(b.(\nu b)C + (\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}), 1, 2) \xrightarrow{b} \{\eta_2.\mathbf{0} + \overline{\eta_2}.\mathbf{0}\}}$$

$$\frac{dec(b.(\nu b)C, 2, 3) \xrightarrow{b} \{\eta_3.\mathbf{0} + \overline{\eta_3}.\mathbf{0}\}}{dec(b.(\nu b)C + (\eta_1.\mathbf{0} + \overline{\eta_1}.\mathbf{0}), 1, 3) \xrightarrow{b} \{\eta_3.\mathbf{0} + \overline{\eta_3}.\mathbf{0}\}}$$

ovvero

$$dec((\nu a)B, 1, 0) = dec(b.(\nu b)C + (\eta_0.\mathbf{0} + \overline{\eta_0}.\mathbf{0}), 1, 2) \xrightarrow{b} \{\eta_2.\mathbf{0} + \overline{\eta_2}.\mathbf{0}\}$$
$$dec((\nu a)B, 1, 1) = dec(b.(\nu b)C + (\eta_1.\mathbf{0} + \overline{\eta_1}.\mathbf{0}), 1, 3) \xrightarrow{b} \{\eta_3.\mathbf{0} + \overline{\eta_3}.\mathbf{0}\}$$

Per quanto riguarda $dec((\nu b)C, 2, 2) = \{\eta_2.\mathbf{0}+\overline{\eta_2}.\mathbf{0}\}$ (questa volta col primo parametro intero fissato), non sto a rivedere tutti i passaggi (applicazione delle regole (sum-1), (sum-2), (pref)) ma si ottiene, ovviamente o quasi, che

$$\{\eta_2.\mathbf{0} + \overline{\eta_2}.\mathbf{0}\} \xrightarrow{\eta_2} \emptyset$$
$$\{\eta_2.\mathbf{0} + \overline{\eta_2}.\mathbf{0}\} \xrightarrow{\overline{\eta_2}} \emptyset$$

Anche in questo caso queste due transizioni non sono effettivamente disponibili direttamente – dal momento che sono basate sulle azioni ristrette η_2 e $\overline{\eta_2}$ – ma, contrariamente all'ipotesi con identificativo extra ridottto, sarà disponibile in questo posto un massimo di un token, dal momento che non potrà arrivare il token partito dal secondo posto della marcatura iniziale. Quest'ultimo infatti arriverà nel posto $\{\eta_3.\mathbf{0} + \overline{\eta_3}.\mathbf{0}\}$ che – analogamente – si può provare che supporta le transizioni

$$\{\eta_3.\mathbf{0} + \overline{\eta_3}.\mathbf{0}\} \xrightarrow{\eta_3} \emptyset$$
$$\{\eta_3.\mathbf{0} + \overline{\eta_3}.\mathbf{0}\} \xrightarrow{\overline{\eta_3}} \emptyset$$

che – analogamente – non disponendo di due token il posto che le supporta non postranno sincronizzarsi.

Riassumendo, in figura 3.6 (pagina seguente) abbiamo l'intero sistema P/T (non più un frammento) generato da $dec(q, 0, 0)$; con l'identificativo extra completo, i due token della marcatura iniziale seguono due percorsi disgiunti e non possono compiere l'autosincronizzazione finale che non aveva corrispondenza nell'LTS generato da q. Con l'identificativo extra abbiamo quindi ripristinato la bisimilitudine con la semantica LTS.

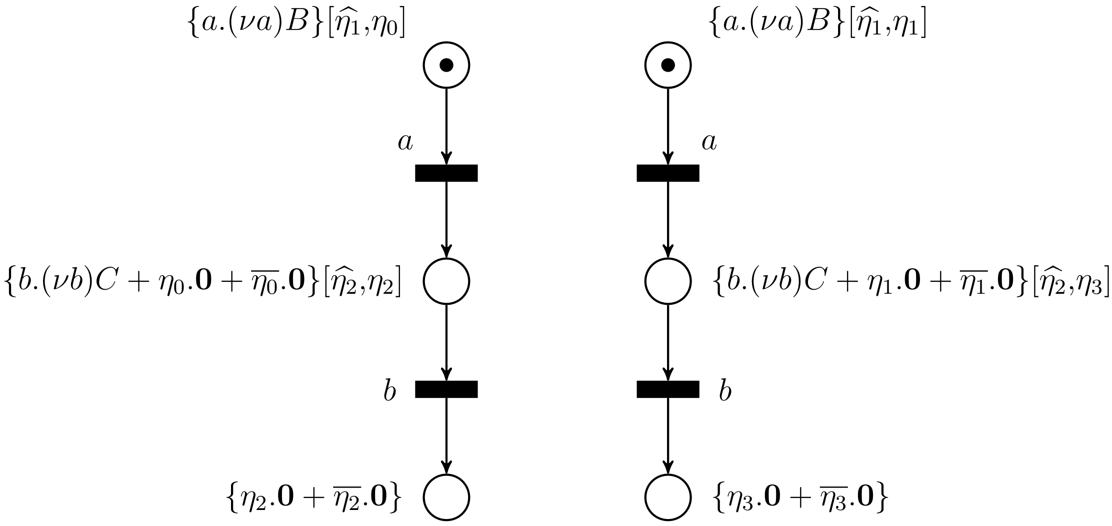

Figura 3.6: Sistema P/T, *eid* completo, per $q = a.(\nu a)B \mid a.(\nu a)B$

Capitolo 4

Correttezza dei sistemi P/T

In questo capitolo proverò la correttezza della nuova versione della semantica P/T, ovvero che – per ogni termine Multi-CCS $q \in \mathcal{P}$ – la marcatura ottenuta applicando *dec* a q è bisimile allo stato q ottenuto nella semantica LTS, ai sensi della definizione 1.3.4 (pag. 23).

4.1 Definizioni e risultati preliminari

Prima di procedere con la dimostrazione vera e propria, è necessario premettere alcune definizioni e alcuni risultati che saranno utili in seguito.

4.1.1 Nomi di posti e marcature

Per prima cosa osserviamo che tutti i posti in S_{MCCS} sono generati applicando la funzione di decomposizione *dec* a processi estesi sequenziali e che le marcature sono composizioni di posti; diamo una definizione dei "nomi" associati alle marcature in $\mathcal{M}_{fin}(S_{MCCS})$ partendo dai singoli posti.

Definizioni 4.1.1. *Sia $s \in \mathcal{S}_{MCCS}$ un posto ottenuto dall'applicazione della funzione dec a un processo esteso sequenziale $p \in \mathcal{P}_{seq}^{\mathcal{N}}$ e a due numeri interi $e, i \in \mathbb{N}$; i* nomi *associati al posto $s = dec(p, e, i)$, indicati con $n(s)$ o anche $n(dec(p, e, i))$, sono ottenuti dalla* funzione di estrazione dei nomi

$N(p, e, i, \emptyset)$, dove

$$N : \mathcal{P}^{\mathcal{N}} \times \mathbb{N} \times \mathbb{N} \times \mathcal{P}(\mathcal{M}_{fin}(S_{MCCS})) \rightarrow \mathcal{P}(\mathcal{A}ct')$$

è definita induttivamente come segue:

$$\begin{aligned}
N(\mathbf{0}, e, i, P) &= \emptyset \\
N(\mu.q, e, i, P) &= \{\mu\} \cup N(q, e, i, P) \\
N(\underline{\mu}.q, e, i, P) &= \{\mu\} \cup N(q, e, i, P) \\
N(p_1 + p_2, e, i, P) &= N(p_1, e+1, i, P) \cup N(p_2, e+1, i+2^e, P) \\
N(q_1 \mid q_2, e, i, P) &= N(q_1, e+1, i, P) \cup N(q_2, e+1, i+2^e, P) \\
N((\nu a)q, e, i, P) &= \begin{cases} N(q\{\eta_i/a\}, e, i+2^e, P) & \textit{se } a, \overline{a} \in fn(q) \\ N(q\{\eta_d/a\}, e, i, P) & \textit{se } a \in fn(q) \wedge \overline{a} \notin fn(q) \\ N(q\{\eta_c/a\}, e, i, P) & \textit{se } a \notin fn(q) \wedge \overline{a} \in fn(q) \\ N(q, e, i, P) & \textit{altrimenti} \end{cases} \\
N(A, e, i, P) &= \begin{cases} \emptyset & \textit{se } dec(A, e, i) \in P \\ N(q, e, i, P \cup \{dec(A, e, i)\}) & \textit{altrimenti, con } A \stackrel{def}{=} q \end{cases}
\end{aligned}$$

Sia $m \in \mathcal{M}_{fin}(S_{MCCS})$ una generica marcatura costituita dagli $n \geqslant 0$ posti $s_1, s_2, \ldots, s_n \in S_{MCCS}$; i nomi *associati alla marcatura m – che indicheremo con $n(m)$ sono l'unione dei nomi associati ai singoli posti che compongono m, ovvero*

$$n(m) = \bigcup_{i=1}^{n} n(s_i)$$

Osserviamo che la precedente definizione, contrariamente alle analoghe definizioni relative a nomi liberi e nomi legati del Multi-CCS, è basata su una funzione, N, che è definita induttivamente ma che non ha necessariamente termine; osserviamo infatti che, nel caso dell'esempio 3.3 (pag. 61), ovvero

$$C \stackrel{def}{=} (\nu b)(a.C \mid (b.c.\mathbf{0} + \overline{b}.\mathbf{0}))$$

avremo, applicando $N(C, 0, 0, \emptyset)$, infinite sostituzioni del nome b (con η_0, con η_1, con η_3, con η_7, ecc.) dovute al fatto che, dal momento che $b, \overline{b} \in fn(C)$, i vari $dec(C, e, i)$ che vengono aggiunti nel quarto argomento corrispondono a marcature sempre diverse e non causano il termine della funzione N. Del resto, questo rispecchia il fatto che $Net(C)$ è costituita da un'insieme infinito di posti, un'insieme infinito di transizioni e un insieme infinito di nomi; dal momento che $n(dec(C, 0, 0))$ ha l'obiettivo di determinare tutti i nomi "raggiungibili" (in un certo senso) da $dec(C, 0, 0)$ e che questi sono infiniti, è ragionevole che la funzione N non abbia termine, quando applicata a C.

Ci può venire il dubbio se $n(dec(q, e, i))$ può essere calcolato con funzione N, della definizione 4.1.1 (pag. 115), anche quando q è un processo non sequenziale. Per provarlo abbiamo bisogno di alcuni risultati preliminari.

Proposizione 4.1.1. *Per ogni processo esteso $q \in \mathcal{P}^{\mathcal{N}}$, per ogni nome ristretto $\eta \in \mathcal{N}$, per ogni nome $a \in \mathcal{L}$, per ogni coppia di numeri naturali $e, i \in \mathbb{N}$ e per ogni insieme di marcature $P \in \mathcal{P}(\mathcal{M}_{fin}(S_{MCCS}))$ è vero che*

$$N(q\{\eta/a\}, e, i, P') = N(q, e, i, P)\{\eta/a\}$$

dove

$$P' = \{dec(r\{\eta/a\}, f, j) \mid dec(r, f, j) \in P\}$$

Dimostrazione. Dimostriamo la tesi per induzione strutturale su q:

- se $q = \mathbf{0}$, abbiamo

$$\begin{aligned} N(q\{\eta/a\}, e, i, P') &= N(\mathbf{0}\{\eta/a\}, e, i, P') = N(\mathbf{0}, e, i, P') \\ &= \emptyset = \emptyset\{\eta/a\} = N(\mathbf{0}, e, i, P)\{\eta/a\} \\ &= N(q, e, i, P)\{\eta/a\} \end{aligned}$$

- se $q = a.q'$, per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, per induzione strutturale su q' abbiamo

$$N(q'\{\eta/a\}, e, i, P') = N(q', e, i, P)\{\eta/a\}$$

e quindi

$$\begin{aligned} N(q\{\eta/a\}, e, i, P') &= N((a.q')\{\eta/a\}, e, i, P') = N(\eta.(q'\{\eta/a\}), e, i, P') \\ &= \{\eta\} \cup N(q'\{\eta/a\}, e, i, P') \\ &= \{a\}\{\eta/a\} \cup N(q', e, i, P)\{\eta/a\} \\ &= (\{a\} \cup N(q', e, i, P))\{\eta/a\} \\ &= N(a.q', e, i, P)\{\eta/a\} = N(q, e, i, P)\{\eta/a\} \end{aligned}$$

- se $q = \overline{a}.q'$, $q = \underline{a}.q'$, $q = \underline{\overline{a}}.q'$, $q = \mu.q'$ o $q = \underline{\mu}.q'$, per un'opportuna azione estesa $\mu \in \mathcal{A}ct'$ tale che $\mu \notin \{a, \overline{a}\}$ e per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, la dimostrazione è del tutto analoga a quella del caso precedente;

- se $q = p_1 + p_2$, per opportuni processi estesi sequenziali $p_1, p_2 \in \mathcal{P}^{\mathcal{N}}_{seq}$, per induzione strutturale su p_1 e su p_2 abbiamo

$$N(p_1\{\eta/a\}, e+1, i, P') = N(p_1, e+1, i, P)\{\eta/a\}$$
$$N(p_2\{\eta/a\}, e+1, i+2^e, P') = N(p_2, e+1, i+2^e, P)\{\eta/a\}$$

e quindi

$$\begin{aligned} N(q\{\eta/a\}, e, i, P') &= N((p_1 + p_2)\{\eta/a\}, e, i, P') \\ &= N(p_1\{\eta/a\} + p_2\{\eta/a\}, e, i, P') \\ &= N(p_1\{\eta/a\}, e+1, i, P') \\ &\quad \cup N(p_2\{\eta/a\}, e+1, i+2^e, P') \\ &= N(p_1, e+1, i, P)\{\eta/a\} \\ &\quad \cup N(p_2, e+1, i+2^e, P)\{\eta/a\} \\ &= (N(p_1, e+1, i, P) \cup N(p_2, e+1, i+2^e, P))\{\eta/a\} \\ &= N(p_1 + p_2, e, i, P)\{\eta/a\} = N(q, e, i, P)\{\eta/a\} \end{aligned}$$

- se $q = q_1 \mid q_2$, per opportuni processi estesi $q_1, q_2 \in \mathcal{P}^{\mathcal{N}}$, la dimostrazione è del tutto analoga a quella del caso precedente;

- se $q = (\nu a)q'$, per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, abbiamo quattro casi, a seconda del valore di $\{a, \overline{a}\} \cap fn(q')$:
  - quando $a, \overline{a} \in fn(q')$ è necessario utilizzare un nuovo nome ristretto η_i; osserviamo, per prima cosa, che operando per induzione strutturale su q', abbiamo che esiste un P_0 tale che

    $$\begin{aligned} N(q, e, i, P) &= N((\nu a)q', e, i, P) = N(q'\{\eta_i/a\}, e, i + 2^e, P) \\ &= N(q', e, i + 2^e, P_0)\{\eta_i/a\} \end{aligned}$$

    e tale che

    $$P = \{dec(r\{\eta_i/a\}, f, j) \mid dec(r, f, j) \in P_0\}$$

    Quanto sopra implica che P' è pari a

    $$P' = \{dec(r\{\eta_i/a\}\{\eta/a\}, f, j) \mid dec(r, f, j) \in P_0\}$$

    ovvero, dal momento che la seconda sostituzione su a è inefficace, che $P' = P$; abbiamo quindi, sfruttando ancora l'inefficacia della seconda sostituzione su a e, due volte, l'induzione strutturale

    $$\begin{aligned} N(q\{\eta/a\}, e, i, P') &= N(q\{\eta/a\}, e, i, P) \\ &= N(((\nu a)q')\{\eta/a\}, e, i, P) \\ &= N((\nu a)q', e, i, P) \\ &= N(q'\{\eta_i/a\}, e, i + 2^e, P) \\ &= N(q', e, i + 2^e, P_0)\{\eta_i/a\} \\ &= N(q', e, i + 2^e, P_0)\{\eta_i/a\}\{\eta/a\} \\ &= N(q'\{\eta_i/a\}, e, i + 2^e, P)\{\eta/a\} \\ &= N((\nu a)q', e, i, P)\{\eta/a\} \\ &= N(q, e, i, P)\{\eta/a\} \end{aligned}$$

  - quando $a \in fn(q') \wedge \overline{a} \notin fn(q')$, la dimostrazione è simile a quella precedente; osserviamo che operando per induzione strutturale su

q', abbiamo che esiste un P_0 tale che

$$\begin{aligned} N(q, e, i, P) &= N((\nu a)q', e, i, P) = N(q'\{\eta_d/a\}, e, i, P) \\ &= N(q', e, i, P_0)\{\eta_d/a\} \end{aligned}$$

e tale che

$$P = \{dec(r\{\eta_d/a\}, f, j) \mid dec(r, f, j) \in P_0\}$$

Quanto sopra implica che P' è pari a

$$P' = \{dec(r\{\eta_d/a\}\{\eta/a\}, f, j) \mid dec(r, f, j) \in P_0\}$$

ovvero, dal momento che la seconda sostituzione su a è inefficace, che $P' = P$; abbiamo quindi, sfruttando ancora l'inefficacia della seconda sostituzione su a e, due volte, l'induzione strutturale

$$\begin{aligned} N(q\{\eta/a\}, e, i, P') &= N(q\{\eta/a\}, e, i, P) \\ &= N(((\nu a)q')\{\eta/a\}, e, i, P) \\ &= N((\nu a)q', e, i, P) \\ &= N(q'\{\eta_d/a\}, e, i, P) \\ &= N(q', e, i, P_0)\{\eta_d/a\} \\ &= N(q', e, i, P_0)\{\eta_d/a\}\{\eta/a\} \\ &= N(q'\{\eta_i/a\}, e, i, P)\{\eta/a\} \\ &= N((\nu a)q', e, i, P)\{\eta/a\} \\ &= N(q, e, i, P)\{\eta/a\} \end{aligned}$$

- quando $a \notin fn(q') \wedge \overline{a} \in fn(q')$, la dimostrazione è del tutto analoga al caso precedente, utilizzando il nome ristretto banale η_c al posto di η_d;
- quando $a, \overline{a} \notin fn(q')$, per induzione strutturale su q' abbiamo

$$N(q'\{\eta/a\}, e, i, P') = N(q', e, i, P)\{\eta/a\}$$

e quindi, ricordando che in questo caso abbiamo $q' = q'\{\eta/a\}$, abbiamo

$$\begin{aligned} N(q\{\eta/a\}, e, i, P') &= N(((\nu a)q')\{\eta/a\}, e, i, P') \\ &= N((\nu a)q', e, i, P') = N(q', e, i, P') \\ &= N(q'\{\eta/a\}, e, i, P') = N(q', e, i, P)\{\eta/a\} \\ &= N((\nu a)q', e, i, P)\{\eta/a\} \\ &= N(q, e, i, P)\{\eta/a\} \end{aligned}$$

- se $q = (\nu b)q'$, per un opportuno nome $b \in \mathcal{L}$ tale che $b \neq a$ e per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, abbiamo quattro casi, a seconda dei valori di $\{b, \overline{b}\} \cap fn(q')$:
  - quando $b, \overline{b} \in fn(q')$, per induzione strutturale abbiamo che

    $$N(q'\{\eta_i/b\}\{\eta/a\}, e, i + 2^e, P') = N(q'\{\eta_i/b\}, e, i + 2^e, P)\{\eta/a\}$$

    e quindi – tenendo anche conto del fatto che $a \neq b$, $b \neq \eta$ e $\eta_i \neq a$ ci permette di invertire l'ordine delle sostituzioni $\{\eta_i/b\}$ e $\{\eta/a\}$ – abbiamo

    $$\begin{aligned} N(q\{\eta/a\}, e, i, P') &= N(((\nu b)q')\{\eta/a\}, e, i, P') \\ &= N((\nu b)(q'\{\eta/a\}), e, i, P') \\ &= N(q'\{\eta/a\}\{\eta_i/b\}, e, i + 2^e, P') \\ &= N(q'\{\eta_i/b\}\{\eta/a\}, e, i + 2^e, P') \\ &= N(q'\{\eta_i/b\}, e, i + 2^e, P)\{\eta/a\} \\ &= N((\nu b)q', e, i, P)\{\eta/a\} = N(q, e, i, P)\{\eta/a\} \end{aligned}$$

  - quando $b \in fn(q') \wedge \overline{b} \notin fn(q')$, la dimostrazione è simile a quella del caso precedente; per induzione strutturale abbiamo che

    $$N(q'\{\eta_d/b\}\{\eta/a\}, e, i, P') = N(q'\{\eta_d/b\}, e, i, P)\{\eta/a\}$$

e quindi – tenendo anche conto del fatto che $a \neq b$, $b \neq \eta$ e $\eta_d \neq a$ ci permette di invertire l'ordine delle sostituzioni $\{\eta_d/b\}$ e $\{\eta/a\}$ – abbiamo

$$\begin{aligned} N(q\{\eta/a\}, e, i, P') &= N(((\nu b)q')\{\eta/a\}, e, i, P') \\ &= N((\nu b)(q'\{\eta/a\}), e, i, P') \\ &= N(q'\{\eta/a\}\{\eta_d/b\}, e, i, P') \\ &= N(q'\{\eta_d/b\}\{\eta/a\}, e, i, P') \\ &= N(q'\{\eta_d/b\}, e, i, P)\{\eta/a\} \\ &= N((\nu b)q', e, i, P)\{\eta/a\} = N(q, e, i, P)\{\eta/a\} \end{aligned}$$

- quando $b \notin fn(q') \wedge \overline{b} \in fn(q')$, la dimostrazione è del tutto analoga al caso precedente, utilizzando il nome ristretto banale η_c al posto di η_d;
- quando $b, \overline{b} \notin fn(q')$, per induzione strutturale abbiamo che

$$N(q'\{\eta/a\}, e, i, P') = N(q', e, i, P)\{\eta/a\}$$

e quindi

$$\begin{aligned} N(q\{\eta/a\}, e, i, P') &= N(((\nu b)q')\{\eta/a\}, e, i, P') \\ &= N((\nu b)(q'\{\eta/a\}), e, i, P') \\ &= N(q'\{\eta/a\}, e, i, P') = N(q', e, i, P)\{\eta/a\} \\ &= N((\nu b)q', e, i, P)\{\eta/a\} = N(q, e, i, P)\{\eta/a\} \end{aligned}$$

- se $q = B$, dove $B \stackrel{def}{=} q'$, per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$ tale che $a \in fn(q') \vee \overline{a} \in fn(q')$, possiamo distinguere due casi:
  - quando $dec(B, e, i) \notin P$ – e quindi, per costruzione di P', $dec(B_{\{\eta/a\}}, e, i) \notin P'$ – per induzione strutturale abbiamo

$$\begin{aligned} &N(q'\{\eta/a\}, e, i, P' \cup \{dec(B_{\{\eta/a\}}, e, i)\}) \\ &\quad = N(q', e, i, P \cup \{dec(B, e, i)\})\{\eta/a\} \end{aligned}$$

e quindi

$$\begin{aligned}
N(q\{\eta/a\}, e, i, P') &= N(B\{\eta/a\}, e, i, P') = N(B_{\{\eta/a\}}, e, i, P') \\
&= N(q'\{\eta/a\}, e, i, P' \cup \{dec(B_{\{\eta/a\}}, e, i)\}) \\
&= N(q', e, i, P \cup \{dec(B, e, i)\})\{\eta/a\} \\
&= N(B, e, i, P)\{\eta/a\} = N(q, e, i, P)\{\eta/a\}
\end{aligned}$$

– quando $dec(B, e, i) \in P$ – e quindi, per costruzione di P', $dec(B_{\{\eta/a\}}, e, i) \in P'$ – abbiamo banalmente

$$\begin{aligned}
N(q\{\eta/a\}, e, i, P') &= N(B\{\eta/a\}, e, i, P') = N(B_{\{\eta/a\}}, e, i, P') \\
&= \emptyset = \emptyset\{\eta/a\} = N(B, e, i, P)\{\eta/a\} \\
&= N(q, e, i, P)\{\eta/a\}
\end{aligned}$$

- se $q = B$, dove $B \stackrel{def}{=} q'$, per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$ tale che $a, \overline{a} \notin fn(q')$ (ovvero tale che $B\{\eta/a\} = B$), possiamo distinguere due casi:

  – quando $dec(B, e, i) \notin P$ – e quindi, per costruzione di P', $dec(B, e, i) \notin P'$ – per induzione strutturale abbiamo

$$\begin{aligned}
&N(q'\{\eta/a\}, e, i, P' \cup \{dec(B, e, i)\}) \\
&\quad = N(q', e, i, P \cup \{dec(B, e, i)\})\{\eta/a\}
\end{aligned}$$

  e quindi

$$\begin{aligned}
N(q\{\eta/a\}, e, i, P') &= N(B\{\eta/a\}, e, i, P') = N(B, e, i, P') \\
&= N(q'\{\eta/a\}, e, i, P' \cup \{dec(B, e, i)\}) \\
&= N(q', e, i, P \cup \{dec(B, e, i)\})\{\eta/a\} \\
&= N(B, e, i, P)\{\eta/a\} = N(q, e, i, P)\{\eta/a\}
\end{aligned}$$

  – quando $dec(B, e, i) \in P$ – e quindi, per costruzione di P', $dec(B, e, i) \in P'$ – abbiamo banalmente

$$N(q\{\eta/a\}, e, i, P') = N(B\{\eta/a\}, e, i, P') = N(B, e, i, P')$$

$$= \emptyset = \emptyset\{\eta/a\} = N(B, e, i, P)\{\eta/a\}$$
$$= N(q, e, i, P)\{\eta/a\}$$

□

Dalla precedente proposizione abbiamo banalmente il seguente

Corollario 4.1.2. *Per ogni processo esteso $q \in \mathcal{P}^{\mathcal{N}}$, per ogni nome ristretto $\eta \in \mathcal{N}$, per ogni nome $a \in \mathcal{L}$ e per ogni coppia di numeri naturali $e, i \in \mathbb{N}$ è vero che*

$$N(q\{\eta/a\}, e, i, \emptyset) = N(q, e, i, \emptyset)\{\eta/a\}$$

Dimostrazione. Ovvia conseguenza della proposizione 4.1.1 (pag. 117), con $P = \emptyset$ (e quindi anche $P' = \emptyset$). □

Applichiamo ora il corollario appena dimostrato ai processi estesi sequenziali.

Corollario 4.1.3. *Per ogni processo sequenziale esteso $p \in \mathcal{P}^{\mathcal{N}}_{seq}$, per ogni nome ristretto $\eta \in \mathcal{N}$, per ogni nome $a \in \mathcal{L}$ e per ogni coppia di numeri naturali $e, i \in \mathbb{N}$ è vero che*

$$n(dec(p\{\eta/a\}, e, i)) = n(dec(p, e, i))\{\eta/a\}$$

Dimostrazione. Si dimostra per per casistica su p, sfruttando il fatto che p è sequenziale e, quindi, che $n(p)$ è definito in funzione di N:

- se $p = 0$, abbiamo banalmente che

$$\begin{aligned} n(dec(p\{\eta/a\}, e, i)) &= n(dec(\mathbf{0}\{\eta/a\}, e, i)) = n(dec(\mathbf{0}, e, i)) = n(\emptyset) \\ &= \emptyset = \emptyset\{\eta/a\} = N(\mathbf{0}, e, i, \emptyset)\{\eta/a\} \\ &= N(p, e, i, \emptyset)\{\eta/a\} = n(dec(p, e, i))\{\eta/a\} \end{aligned}$$

- se $p = a.q$, per un opportuno processo esteso $q \in \mathcal{P}^{\mathcal{N}}$, applicando il corollario 4.1.2 a q abbiamo

$$N(q\{\eta/a\}, e, i, \emptyset) = N(q, e, i, \emptyset)\{\eta/a\}$$

e quindi

$$\begin{aligned} n(dec(p\{\eta/a\}, e, i)) &= n(dec((a.q)\{\eta/a\}, e, i)) \\ &= n(dec(\eta.(q\{\eta/a\}), e, i)) \\ &= N(\eta.(q\{\eta/a\}), e, i, \emptyset) \\ &= \{\eta\} \cup N(q\{\eta/a\}, e, i, \emptyset) \\ &= \{a\}\{\eta/a\} \cup N(q, e, i, \emptyset)\{\eta/a\} \\ &= (\{a\} \cup N(q, e, i, \emptyset))\{\eta/a\} \\ &= N(a.q, e, i, \emptyset)\{\eta/a\} = N(p, e, i, \emptyset)\{\eta/a\} \\ &= n(dec(p, e, i))\{\eta/a\} \end{aligned}$$

- se $p = \overline{a}.q$, $p = \underline{a}.q$, $p = \underline{\overline{a}}.q$, $p = \mu.q$ o $p = \underline{\mu}.q$, per un'opportuna azione estesa $\mu \in \mathcal{A}ct'$ tale che $\mu \notin \{a, \overline{a}\}$ e per un opportuno processo esteso $q \in \mathcal{P}^{\mathcal{N}}$, la dimostrazione è del tutto analoga a quella del caso precedente;

- se $p = p_1 + p_2$, per opportuni processi estesi sequenziali $p_1, p_2 \in \mathcal{P}^{\mathcal{N}}_{seq}$, applicando il corollario 4.1.2 sia a p_1 che a p_2 abbiamo

$$\begin{aligned} N(p_1\{\eta/a\}, e+1, i, \emptyset) &= N(p_1, e+1, i, \emptyset)\{\eta/a\} \\ N(p_2\{\eta/a\}, e+1, i+2^e, \emptyset) &= N(p_2, e+1, i+2^e, \emptyset)\{\eta/a\} \end{aligned}$$

e quindi

$$\begin{aligned} n(dec(p\{\eta/a\}, e, i)) &= n(dec((p_1 + p_2)\{\eta/a\}, e, i)) \\ &= n(dec(p_1\{\eta/a\} + p_2\{\eta/a\}, e, i)) \\ &= N(p_1\{\eta/a\} + p_2\{\eta/a\}, e, i, \emptyset) \\ &= N(p_1\{\eta/a\}, e+1, i, \emptyset) \\ &\quad \cup N(p_2\{\eta/a\}, e+1, i+2^e, \emptyset) \\ &= N(p_1, e+1, i, \emptyset)\{\eta/a\} \\ &\quad \cup N(p_2, e+1, i+2^e, \emptyset)\{\eta/a\} \end{aligned}$$

$$\begin{aligned}
&= (N(p_1, e+1, i, \emptyset) \cup N(p_2, e+1, i+2^e, \emptyset))\{\eta/a\} \\
&= N(p_1 + p_2, e, i, \emptyset)\{\eta/a\} = N(p, e, i, \emptyset)\{\eta/a\} \\
&= n(dec(p, e, i))\{\eta/a\}
\end{aligned}$$

□

Il corollario appena dimostrato può essere banalmente esteso a tutte le marcature.

Corollario 4.1.4. *Per ogni marcatura $m \in \mathcal{M}_{fin}(S_{MCCS})$, per ogni nome ristretto $\eta \in \mathcal{N}$ e per ogni nome $a \in \mathcal{L}$ è vero che*

$$n(m\{\eta/a\}) = n(m)\{\eta/a\}$$

Dimostrazione. Immediata conseguenza del corollario 4.1.3 (pag. 124) e del fatto che m è composizione di posti che sono a loro volta ottenuti da dec su processi estesi sequenziali; abbiamo infatti che, dato una generica marcatura $m \in \mathcal{M}_{fin}(S_{MCCS})$, esistono $n \geqslant 0$ processi sequenziali $p_1, p_2, \ldots, p_n \in \mathcal{P}_{seq}^{\mathcal{N}}$ e $2n$ numeri interi $e_1, e_2, \ldots, e_n, i_1, i_2, \ldots, i_n \in \mathbb{N}$ tali che

$$m = \bigoplus_{k=1}^{n} dec(p_k, e_k, i_k)$$

Applicando sia la proposizione 3.2.5 (pag. 83) che il corollario 4.1.3 (pag. 124) alle singole componenti di m otteniamo

$$\begin{aligned}
n(m\{\eta/a\}) &= n\left(\left(\bigoplus_{k=1}^{n} dec(p_k, e_k, i_k)\right)\{\eta/a\}\right) \\
&= n\left(\bigoplus_{k=1}^{n} dec(p_k, e_k, i_k)\{\eta/a\}\right) = n\left(\bigoplus_{k=1}^{n} dec(p_k\{\eta/a\}, e_k, i_k)\right) \\
&= \bigcup_{k=1}^{n} n(dec(p_k\{\eta/a\}, e_k, i_k)) = \bigcup_{k=1}^{n} n(dec(p_k, e_k, i_k))\{\eta/a\} \\
&= \left(\bigcup_{k=1}^{n} n(dec(p_k, e_k, i_k))\right)\{\eta/a\}
\end{aligned}$$

$$= n\left(\bigoplus_{k=1}^{n} dec(p_k, e_k, i_k)\right)\{\eta/a\} = n(m)\{\eta/a\}$$

□

Applicando il corollario precedente alle marcature ottenute da singoli processi otteniamo una generalizzazione del corollario 4.1.3 (pag. 124).

Corollario 4.1.5. *Per ogni processo esteso* $q \in \mathcal{P}^{\mathcal{N}}$*, per ogni nome ristretto* $\eta \in \mathcal{N}$*, per ogni nome* $a \in \mathcal{L}$ *e per ogni coppia di numeri naturali* $e, i \in \mathbb{N}$ *è vero che*

$$n(dec(q\{\eta/a\}, e, i)) = n(dec(q, e, i))\{\eta/a\}$$

Dimostrazione. Applicando prima la proposizione 3.2.5 (pag. 83) poi il corollario 4.1.4 (pagina a fronte) otteniamo

$$n(dec(q\{\eta/a\}, e, i)) = n(dec(q, e, i)\{\eta/a\}) = n(dec(q, e, i))\{\eta/a\}$$

□

Analizzando la definizione 4.1.1 (pag. 115), possiamo rilevare la seguente

Proposizione 4.1.6. *Sia* $B = q$*, per un opportuno processo esteso* $q \in \mathcal{P}^{\mathcal{N}}$*; siano* $e, i \in \mathbb{N}$ *due generici numeri interi; è vero che*

$$N(q, e, i, \emptyset) = N(q, e, i, \{dec(B, e, i)\})$$

Dimostrazione. La tesi della presente proposizione è conseguenza del fatto che $N(B, e, i, \emptyset) = N(q, e, i, \{dec(B, e, i)\})$ e del fatto che il quarto parametro di N ha esclusivamente la funzione di bloccare (quando opportuno) la ricorsione su B evitando (quando superfluo) una nuova sostituzione della stessa costante con q; l'unica differenza di comportamento tra $N(q, e, i, \emptyset)$ e $N(q, e, i, \{dec(B, e, i)\})$ è che il primo termine – incontrandola – sostituirà nuovamente con q la costante B, aggiungendo ancora una volta i nomi di q al risultato ma, essendo questo un insieme, l'effetto pratico sarà nullo.

□

Possiamo finalmente dimostrare che la funzione N di estrazione dei nomi calcola correttamente $n(dec(q, e, i))$ anche quando $q \in \mathcal{P}^{\mathcal{N}}$ è un generico processo esteso e non solo quando è sequenziale.

Proposizione 4.1.7. *Sia $q \in \mathcal{P}^{\mathcal{N}}$ un generico processo esteso; siano $e, i \in \mathbb{N}$ due generici numeri interi, allora è vero che*

$$n(dec(q, e, i)) = N(q, e, i, \emptyset)$$

Dimostrazione. Si dimostra per induzione strutturale su q:

- se $q = \mathbf{0}$, il caso è degenere e banalmente abbiamo

$$\begin{aligned} n(dec(q, e, i)) &= n(dec(\mathbf{0}, e, i)) = n(\emptyset) = \emptyset = N(\mathbf{0}, e, i, \emptyset) \\ &= N(q, e, i, \emptyset) \end{aligned}$$

- se q è un processo sequenziale – ovvero se $q = \mu.q'$ o $q = \underline{\mu}.q'$ o $q = p_1+p_2$, per un'opportuna azione estesa $\mu \in \mathcal{A}ct'$, per opportuni processi estesi sequenziali $p_1, p_2 \in \mathcal{P}^{\mathcal{N}}_{seq}$ e per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$ – $n(dec(q, e, i)) = N(q, e, i, \emptyset)$ per definizione di n;

- se $q = q_1 \mid q_2$, per opportuni processi estesi $q_1, q_2 \in \mathcal{P}^{\mathcal{N}}$, per induzione strutturale abbiamo

$$\begin{aligned} n(dec(q_1, e+1, i)) &= N(q_1, e+1, i, \emptyset) \\ n(dec(q_2, e+1, i+2^e)) &= N(q_2, e+1, i+2^e, \emptyset) \end{aligned}$$

e quindi

$$\begin{aligned} n(dec(q, e, i)) &= n(dec(q_1 \mid q_2, e, i)) \\ &= n(dec(q_1, e+1, i) \oplus dec(q_2, e+1, i+2^e)) \\ &= n(dec(q_1, e+1, i)) \cup n(dec(q_2, e+1, i+2^e)) \\ &= N(q_1, e+1, i, \emptyset) \cup N(q_2, e+1, i+2^e, \emptyset) \\ &= N(q_1 \mid q_2, e, i, \emptyset) = N(q, e, i, \emptyset) \end{aligned}$$

- se $q = (\nu a)q'$, per un'opportuna azione $a \in \mathcal{L}$ e per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, dobbiamo distinguere quattro casi, a seconda dei valori di $\{a, \overline{a}\} \cap fn(q')$:
  - quando $a, \overline{a} \in fn(q')$, per induzione strutturale abbiamo

    $$n(dec(q', e, i + 2^e)) = N(q', e, i + 2^e, \emptyset)$$

    e quindi, ricordando anche i corollari 4.1.5 (pag. 127) e 4.1.2 (pag. 124), abbiamo

    $$\begin{aligned} n(dec(q, e, i)) &= n(dec((\nu a)q', e, i)) = n(dec(q'\{\eta_i/a\}, e, i + 2^e)) \\ &= n(dec(q', e, i + 2^e))\{\eta_i/a\} \\ &= N(q', e, i + 2^e, \emptyset)\{\eta_i/a\} \\ &= N(q'\{\eta_i/a\}, e, i + 2^e, \emptyset) = N((\nu a)q', e, i, \emptyset) \\ &= N(q, e, i, \emptyset) \end{aligned}$$

  - quando $a \in fn(q') \wedge \overline{a} \notin fn(q')$, la dimostrazione è analoga a quella del caso precedente; per induzione strutturale abbiamo

    $$n(dec(q', e, i)) = N(q', e, i, \emptyset)$$

    e quindi, ricordando anche i corollari 4.1.5 e 4.1.2, abbiamo

    $$\begin{aligned} n(dec(q, e, i)) &= n(dec((\nu a)q', e, i)) = n(dec(q'\{\eta_d/a\}, e, i)) \\ &= n(dec(q', e, i))\{\eta_d/a\} \\ &= N(q', e, i, \emptyset)\{\eta_d/a\} \\ &= N(q'\{\eta_d/a\}, e, i, \emptyset) = N((\nu a)q', e, i, \emptyset) \\ &= N(q, e, i, \emptyset) \end{aligned}$$

  - quando $a \notin fn(q') \wedge \overline{a} \in fn(q')$, la dimostrazione è analoga a quella del sottocaso precedente;
  - quando $a, \overline{a} \notin fn(q')$, per induzione strutturale abbiamo

    $$n(dec(q', e, i)) = N(q', e, i, \emptyset)$$

e quindi

$$\begin{aligned} n(dec(q,e,i)) &= n(dec((\nu a)q',e,i)) = n(dec(q',e,i)) \\ &= N(q',e,i,\emptyset) = N((\nu a)q',e,i,\emptyset) = N(q,e,i,\emptyset) \end{aligned}$$

- se $q = B$, con $B \stackrel{def}{=} q'$ per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, per induzione strutturale abbiamo

  $$n(dec(q',e,i)) = N(q',e,i,\emptyset)$$

  e quindi, tenendo anche conto della proposizione 4.1.6 (pag. 127),

  $$\begin{aligned} n(dec(q,e,i)) &= n(dec(B,e,i)) = n(dec(q',e,i)) = N(q',e,i,\emptyset) \\ &= N(q',e,i,\{dec(B,e,i)\}) = N(B,e,i,\emptyset) = N(q,e,i,\emptyset) \end{aligned}$$

□

Tenendo conto della precedente proposizione, e della definizione 4.1.1 (pag. 115), possiamo banalmente rilevare quanto segue

Osservazione 4.1.8. *Siano $q, q_1, q_2 \in \mathcal{P}^{\mathcal{N}}$ generici processi estesi, siano $p_1, p_2 \in \mathcal{P}^{\mathcal{N}}_{seq}$ generici processi estesi sequenziali e sia $\mu \in \mathcal{A}ct'$ una generica azione estesa; abbiamo*

$$\begin{aligned} n(dec(q,e,i)) &\subseteq n(dec(\mu.q,e,i)) \\ n(dec(q,e,i)) &\subseteq n(dec(\underline{\mu}.q,e,i)) \\ n(dec(p_1,e+1,i)) &\subseteq n(dec(p_1+p_2,e,i)) \\ n(dec(p_2,e+1,i+2^e)) &\subseteq n(dec(p_1+p_2,e,i)) \\ n(dec(q_1,e+1,i)) &\subseteq n(dec(q_1 \mid q_2,e,i)) \\ n(dec(q_2,e+1,i+2^e)) &\subseteq n(dec(q_1 \mid q_2,e,i)) \end{aligned}$$

Vediamo ora alcuni utili risultati.

Proposizione 4.1.9. *Sia $q \in \mathcal{P}^{\mathcal{N}}$ un generico processo esteso; sia $\mu \in Act$ una generica azione non ristretta; abbiamo, per $e, i \in \mathbb{N}$ generici, che*

$$\mu \in fn(q) \iff \mu \in n(dec(q,e,i))$$

ovvero

$$fn(q) = n(dec(q, e, i)) \cap \mathcal{A}ct$$

Dimostrazione. Si dimostra per induzione strutturale su $q \in \mathcal{P}^{\mathcal{N}}$:

- se $q = \mathbf{0}$, è evidente che, con e ed i generici,

  $$\begin{aligned} &\mu \notin \emptyset = fn(\mathbf{0}) = fn(q) \\ &\mu \notin \emptyset = N(\mathbf{0}, e, i, \emptyset) = n(dec(\mathbf{0}, e, i)) = n(dec(q, e, i)) \end{aligned}$$

  La tesi, in questo caso, è quindi verificata;

- se $q = \mu.q'$, per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, è evidente che, con e ed i generici,

  $$\begin{aligned} \mu \in (\{\mu\} \cup fn(q')) &= fn(\mu.q') = fn(q) \\ \mu \in (\{\mu\} \cup N(q', e, i, \emptyset)) &= N(\mu.q', e, i, \emptyset) = n(dec(\mu.q', e, i)) \\ &= n(dec(q, e, i)) \end{aligned}$$

  Quindi, anche in questo caso, la tesi è verificata;

- se $q = \underline{\mu}.q'$, per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, la dimostrazione è del tutto analoga a quella del caso precedente;

- se $q = \mu'.q'$, per un'opportuna azione estesa $\mu' \in \mathcal{A}ct'$ tale che $\mu \neq \mu'$ e per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, abbiamo, per induzione strutturale con e ed i generici,

  $$\mu \in fn(q') \iff \mu \in n(dec(q', e, i))$$

  Quindi, ricordando la proposizione 4.1.7 (pag. 128) e che $\mu \neq \mu'$, abbiamo

  $$\begin{aligned} \mu \in fn(q) &\iff \mu \in fn(\mu'.q') \iff \mu \in (\{\mu'\} \cup fn(q')) \\ &\iff \mu \in fn(q') \iff \mu \in n(dec(q', e, i)) \\ &\iff \mu \in N(q', e, i, \emptyset) \iff \mu \in (\{\mu'\} \cup N(q', e, i, \emptyset)) \end{aligned}$$

$$\begin{aligned}&\iff \mu \in N(\mu'.q', e, i, \emptyset) \iff \mu \in n(dec(\mu'.q', e, i))\\&\iff \mu \in n(dec(q, e, i))\end{aligned}$$

- se $q = \underline{\mu}'.q'$, per un'opportuna azione estesa $\mu' \in \mathcal{A}ct'$ tale che $\mu \neq \mu'$ e per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, la dimostrazione è del tutto analoga a quella del caso precedente;
- se $q = p_1 + p_2$, per opportuni processi estesi sequenziali $p_1, p_2 \in \mathcal{P}^{\mathcal{N}}_{seq}$, abbiamo, per induzione strutturale con e ed i generici,

  $$\begin{aligned}\mu \in fn(p_1) &\iff \mu \in n(dec(p_1, e+1, i))\\ \mu \in fn(p_2) &\iff \mu \in n(dec(p_2, e+1, i+2^e))\end{aligned}$$

  e quindi

  $$\begin{aligned}\mu \in fn(q) &\iff \mu \in fn(p_1 + p_2) \iff \mu \in (fn(p_1) \cup fn(p_2))\\ &\iff \mu \in fn(p_1) \vee \mu \in fn(p_2)\\ &\iff \mu \in n(dec(p_1, e+1, i)) \vee \mu \in n(dec(p_2, e+1, i+2^e))\\ &\iff \mu \in (n(dec(p_1, e+1, i)) \cup n(dec(p_2, e+1, i+2^e)))\\ &\iff \mu \in (N(p_1, e+1, i, \emptyset) \cup N(p_2, e+1, i+2^e, \emptyset))\\ &\iff \mu \in N(p_1 + p_2, e, i, \emptyset) \iff \mu \in n(dec(p_1 + p_2, e, i))\\ &\iff \mu \in n(dec(q, e, i))\end{aligned}$$

- se $q = q_1 \mid q_2$, per opportuni processi estesi $q_1, q_2 \in \mathcal{P}^{\mathcal{N}}$, la dimostrazione, utilizzando anche la proposizione 4.1.7, è analoga a quella del caso precedente;
- se $q = (\nu\mu)q'$, per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, per quando riguarda l'appartenenza di μ a $fn(q)$ abbiamo evidentemente che

  $$\mu \notin (fn(q') \setminus \{\mu, \overline{\mu}\}) = fn((\nu\mu)q') = fn(q)$$

  Invece, per quanto riguarda l'appartenenza o meno di μ a $n(dec(q, e, i))$, dobbiamo distinguere i quattro casi relativi a $\{\mu, \overline{\mu}\} \cap fn(q')$:

– se $\mu, \overline{\mu} \in fn(q')$, tenendo anche conto della proposizione 4.1.7, è evidente che

$$\begin{aligned} \mu \notin N(q', e, i + 2^e, \emptyset)\{\eta_i/\mu\} &= N((\nu\mu)q', e, i, \emptyset) \\ &= N(q, e, i, \emptyset) = n(dec(q, e, i)) \end{aligned}$$

– se $\mu \in fn(q') \wedge \overline{\mu} \notin fn(q')$, tenendo anche conto della proposizione 4.1.7, è evidente, come nel sottocaso precedente, che

$$\begin{aligned} \mu \notin N(q', e, i, \emptyset)\{\eta_d/\mu\} &= N((\nu\mu)q', e, i, \emptyset) \\ &= N(q, e, i, \emptyset) = n(dec(q, e, i)) \end{aligned}$$

– se $\mu \notin fn(q') \wedge \overline{\mu} \in fn(q')$, la dimostrazione è analoga a quella del sottocaso precedente;

– se $\mu, \overline{\mu} \notin fn(q')$, per induzione strutturale su q' abbiamo che

$$fn(q') = n(dec(q', e, i)) \cap \mathcal{A}ct$$

e, tenendo conto che $\mu \in \mathcal{A}ct$, è evidente che

$$\mu \notin fn(q') \implies \mu \notin n(dec(q', e, i))$$

e quindi, tenendo anche conto della proposizione 4.1.7, abbiamo che

$$\begin{aligned} \mu \notin n(dec(q', e, i)) = N(q', e, i, \emptyset) &= N((\nu\mu)q', e, i, \emptyset) \\ &= N(q, e, i, \emptyset) = n(dec(q, e, i)) \end{aligned}$$

In tutti i quattro sottocasi abbiamo quindi anche che $\mu \notin n(dec(q, e, i))$; la tesi è perciò, anche in questo caso, verificata;

- se $q = (\nu\overline{\mu})q'$, per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, la dimostrazione è analoga a quella del caso precedente;
- se $q = (\nu b)q'$, per un opportuno nome $b \in \mathcal{L}$ tale che $\mu \notin \{b, \overline{b}\}$ e per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, dobbiamo trattare i quattro casi relativi a $\{b, \overline{b}\} \cap fn(q')$:

– quando $b, \overline{b} \in fn(q')$, per induzione strutturale su q' abbiamo

$$\mu \in fn(q') \iff \mu \in n(dec(q', e+1, i+2^e))$$

e, dal momento che le sostituzioni di b con un'azione ristretta non hanno effetto sulla presenza di μ libera in q' e ricordando anche la proposizione 4.1.7, abbiamo che

$$\begin{aligned}
\mu \in fn(q) &\iff \mu \in fn((\nu b)q') \iff \mu \in fn(q') \setminus \{b, \overline{b}\} \\
&\iff \mu \in fn(q') \iff \mu \in n(dec(q', e+1, i+2^e)) \\
&\iff \mu \in n(dec(q', e+1, i+2^e))\{\eta_i/b\} \\
&\iff \mu \in N(q', e+1, i+2^e, \emptyset)\{\eta_i/b\} \\
&\iff \mu \in N((\nu b)q', e, i, \emptyset) \\
&\iff \mu \in n(dec((\nu b)q', e, i)) \iff \mu \in n(dec(q, e, i))
\end{aligned}$$

– quando $b \in fn(q') \wedge \overline{b} \notin f(q')$, la dimostrazione è analoga a quella del sottocaso precedente; per induzione strutturale su q' abbiamo

$$\mu \in fn(q') \iff \mu \in n(dec(q', e, i))$$

e, dal momento che le sostituzioni di b con un'azione ristretta non hanno effetto sulla presenza di μ libera in q' e ricordando anche la proposizione 4.1.7, abbiamo che

$$\begin{aligned}
\mu \in fn(q) &\iff \mu \in fn((\nu b)q') \iff \mu \in fn(q') \setminus \{b, \overline{b}\} \\
&\iff \mu \in fn(q') \iff \mu \in n(dec(q', e, i)) \\
&\iff \mu \in n(dec(q', e, i))\{\eta_d/b\} \\
&\iff \mu \in N(q', e, i, \emptyset)\{\eta_d/b\} \\
&\iff \mu \in N((\nu b)q', e, i, \emptyset) \\
&\iff \mu \in n(dec((\nu b)q', e, i)) \iff \mu \in n(dec(q, e, i))
\end{aligned}$$

– quando $b \notin fn(q') \wedge \overline{b} \in f(q')$, la dimostrazione è analoga a quella del sottocaso precedente ma operando con la sostituzione $\{\eta_c/b\}$ al posto di $\{\eta_d/b\}$;

– quando $b, \bar{b} \notin fn(q')$, per induzione strutturale abbiamo

$$\mu \in fn(q') \iff \mu \in n(dec(q', e, i))$$

e quindi

$$\begin{aligned} \mu \in fn(q) \iff \mu \in fn((\nu b)q') &\iff \mu \in fn(q') \setminus \{b, \bar{b}\} \\ &\iff \mu \in fn(q') \iff \mu \in n(dec(q', e, i)) \\ &\iff \mu \in n(dec((\nu b)q', e, i)) \iff \mu \in n(dec(q, e, i)) \end{aligned}$$

- se $q = B$, con $B \stackrel{def}{=} q'$ per un opportuno processo esteso $q' \in \mathcal{P}^{\mathcal{N}}$, per induzione strutturale abbiamo

$$\mu \in fn(q') \iff \mu \in n(dec(q', e, i))$$

e quindi

$$\begin{aligned} \mu \in fn(q) &\iff \mu \in fn(B) \iff \mu \in fn(q') \\ &\iff \mu \in n(dec(q', e, i)) \iff \mu \in n(dec(B, e, i)) \\ &\iff \mu \in n(dec(q, e, i)) \end{aligned}$$

□

Abbiamo quindi verificato che la funzione *dec* mantiene, tra i nomi della marcatura generata – limitatamente ai nomi visibili, ai co-nomi visibili e all'azione invisibile τ – tutte e sole quelle che sono libere nel processo elaborato. Naturalmente questo risultato non può valere per le azioni ristrette poiché la funzione di decomposizione ha proprio il compito di generarle.

Da quanto appena dimostrato, deriva banalmente la seguente

Proposizione 4.1.10. *Sia $q \in \mathcal{P}^{\mathcal{N}}$ un generico processo esteso; sia $\mu \in Act$ una generica azione non ristretta; siano $e, f, i, j \in \mathbb{N}$ quattro generici numeri interi; è vero che*

$$\mu \in n(dec(q, e, i)) \iff \mu \in n(dec(q, f, j))$$

ovvero

$$n(dec(q, e, i)) \cap \mathcal{A}ct = n(dec(q, f, j)) \cap \mathcal{A}ct$$

Dimostrazione. Banale conseguenza della proposizione 4.1.9 (pag. 130); infatti, applicandola sia nei confronti di $dec(q, e, i)$ che di $dec(q, f, j)$ otteniamo

$$\mu \in n(dec(q, e, i)) \iff \mu \in fn(q) \iff \mu \in n(dec(q, f, j))$$

ovvero

$$n(dec(q, e, i)) \cap \mathcal{A}ct = fn(q) = n(dec(q, f, j)) \cap \mathcal{A}ct$$

□

Abbiamo quindi visto che la presenza di un'azione visibile, o dell'azione invisibile τ, tra i nomi di una marcatura $dec(q, e, i)$ non dipende dalla scelta degli interi $e, i \in \mathbb{N}$, come era ragionevole aspettarsi visto che i parametri di supporto della *dec* hanno esclusivamente il compito di determinare gli eventuali indici per le azioni ristrette e per i supporti ristretti.

I risultati visti fin'ora sono relativi a processi estesi; vediamo ora alcuni risultati relativi a processi Multi-CCS non estesi.

Proposizione 4.1.11. *Sia $q \in \mathcal{P}$, un generico processo Multi-CCS; siano $e, i \in \mathbb{N}$ due generici numeri naturali; sia $\eta_x \in \mathcal{N}_{\mathbb{N}}$ un generico nome ristretto non banale, abbiamo*

$$\eta_x \in n(dec(q, e, i)) \implies \exists k \in \mathbb{N}. x = i + k \cdot 2^e$$
$$\overline{\eta_x} \in n(dec(q, e, i)) \implies \exists k \in \mathbb{N}. x = i + k \cdot 2^e$$

Dimostrazione. Dimostriamo solo la prima delle due implicazioni, per induzione strutturale su q, poiché la dimostrazione della seconda è banalmente identica:

- se $q = \mathbf{0}$, abbiamo

  $$n(dec(q, e, i)) = N(q, e, i, \emptyset) = N(\mathbf{0}, e, i, \emptyset) = \emptyset$$

  e quindi, evidentemente, non può esistere un $\eta_x \in n(dec(q, e, i))$; dalla negazione della premessa deriva l'assunto;

- se $q = \mu.q'$, per un'opportuna azione $\mu \in \mathcal{A}ct$ e un opportuno processo $q' \in \mathcal{P}$, per induzione strutturale su q' abbiamo

$$\eta_x \in n(dec(q', e, i)) \implies \exists k \in \mathbb{N}.x = i + k \cdot 2^e$$

Abbiamo

$$\begin{aligned} n(dec(q, e, i)) &= N(q, e, i, \emptyset) = N(\mu.q', e, i, \emptyset) \\ &= \{\mu\} \cup N(q', e, i, \emptyset) = \{\mu\} \cup n(dec(q', e, i)) \end{aligned}$$

Dal momento che $\mu \in \mathcal{A}ct$, per definizione abbiamo che, per ogni $\eta_x \in \mathcal{N}_{\mathbb{N}}$, $\mu \neq \eta_x$; quindi, da quanto appena visto abbiamo che

$$\begin{aligned} \eta_x \in n(dec(q, e, i)) &\iff \eta_x \in n(dec(q', e, i)) \\ &\implies \exists k \in \mathbb{N}.x = i + k \cdot 2^e \end{aligned}$$

- se $q = \underline{\mu}.q'$, per un'opportuna azione $\mu \in \mathcal{A}ct$ e un opportuno processo $q' \in \mathcal{P}$, la dimostrazione è uguale a quella del caso precedente;

- se $q = p_1 + p_2$, per opportuni processi sequenziali $p_1, p_2 \in \mathcal{P}_{seq}$, per induzione strutturale su p_1 e su p_2 abbiamo

$$\begin{aligned} \eta_x \in n(dec(p_1, e+1, i)) &\implies \exists k \in \mathbb{N}.x = i + k \cdot 2^{e+1} \\ \eta_x \in n(dec(p_2, e+1, i+2^e)) &\implies \exists k \in \mathbb{N}.x = i + 2^e + k \cdot 2^{e+1} \end{aligned}$$

Abbiamo anche

$$\begin{aligned} n(dec(q, e, i)) &= N(q, e, i, \emptyset) = N(p_1 + p_2, e, i, \emptyset) \\ &= N(p_1, e+1, i, \emptyset) \cup N(p_1, e+1, i+2^e) \\ &= n(dec(p_1, e+1, i)) \cup n(dec(p_2, e+1, i+2^e)) \end{aligned}$$

Se $\eta_x \in n(dec(p_1 + p_2, e, i))$ abbiamo due casi:

– quando $\eta_x \in n(dec(p_1, e+1, i))$, ricordando l'ipotesi induttiva su p_1, abbiamo che esiste un $k_1 \in \mathbb{N}$ tale che

$$x = i + k_1 \cdot 2^{e+1} = i + 2 \cdot k_1 \cdot 2^e$$

La tesi è quindi verificata per $k = 2 \cdot k_1$;

– quando $\eta_x \in n(dec(p_2, e+1, i+2^e))$, ricordando l'ipotesi induttiva su p_2, abbiamo che esiste un $k_2 \in \mathbb{N}$ tale che

$$x = i + 2^e + k_2 \cdot 2^{e+1} = i + (2 \cdot k_2 + 1) \cdot 2^e$$

La tesi è quindi verificata per $k = 2 \cdot k_2 + 1$;

- se $q = q_1 \mid q_2$, per opportuni processi $q_1, q_2 \in \mathcal{P}$, la dimostrazione è identica a quella del caso precedente;

- se $q = (\nu a)q'$, per un opportuno nome $a \in \mathcal{L}$ e un opportuno processo $q' \in \mathcal{P}$, dobbiamo distinguere i seguenti casi:

  – quando $a, \overline{a} \in fn(q')$, per ipotesi induttiva su q' abbiamo

$$\eta_x \in n(dec(q', e, i+2^e)) \implies \exists k \in \mathbb{N}. x = i + 2^e + k \cdot 2^e$$

Abbiamo, ricordando anche la proposizione 3.2.5 (pag. 83) e il corollario 4.1.4 (pag. 126), che

$$\begin{aligned} n(dec(q, e, i)) &= n(dec((\nu a)q', e, i)) = n(dec(q'\{\eta_i/a\}, e, i+2^e)) \\ &= n(dec(q', e, i+2^e)\{\eta_i/a\}) \\ &= n(dec(q', e, i+2^e))\{\eta_i/a\} \end{aligned}$$

e, dal momento che la proposizione 4.1.9 (pag. 130) ci assicura che $a, \overline{a} \in fn(q') \implies a, \overline{a} \in n(dec(q', e, i+2^e))$, abbiamo

$$\begin{aligned} n(dec(q, e, i)) &= n(dec(q', e, i+2^e))\{\eta_i/a\} \\ &= ((n(dec(q', e, i+2^e)) \setminus \{a, \overline{a}\}) \cup \{a, \overline{a}\})\{\eta_i/a\} \end{aligned}$$

$$\begin{aligned}
&= (n(dec(q', e, i + 2^e)) \setminus \{a, \overline{a}\})\{\eta_i/a\} \\
&\qquad \cup \{a, \overline{a}\}\{\eta_i/a\} \\
&= (n(dec(q', e, i + 2^e)) \setminus \{a, \overline{a}\}) \cup \{\eta_i, \overline{\eta_i}\}
\end{aligned}$$

Quindi, quando $\eta_x \in n(dec(q, e, i))$ abbiamo 2 casi: (a) $\eta_x = \eta_i$, ovvero $i = i + k \cdot 2^e$ è verificata per $k = 0$, oppure (b) $\eta_x \in n(dec(q', e, i + 2^e))$ e quindi, per ipotesi induttiva su q', esiste un $k' \in \mathbb{N}$ tale che

$$x = i + 2^e + k' \cdot 2^e = i + (k' + 1) \cdot 2^e$$

e quindi la tesi è verificata per $k = k' + 1$;

- quando $a \in fn(q') \wedge \overline{a} \notin fn(q')$, la dimostrazione è una versione semplificata di quella del sottocaso precedente; per ipotesi induttiva su q' abbiamo

$$\eta_x \in n(dec(q', e, i)) \implies \exists k \in \mathbb{N}. x = i + k \cdot 2^e$$

Abbiamo, ricordando anche la proposizione 3.2.5 e il corollario 4.1.4, che

$$\begin{aligned}
n(dec(q, e, i)) &= n(dec((\nu a)q', e, i)) = n(dec(q'\{\eta_d/a\}, e, i)) \\
&= n(dec(q', e, i)\{\eta_d/a\}) = n(dec(q', e, i))\{\eta_d/a\}
\end{aligned}$$

e, dal momento che la sostituzione $\{\eta_d/a\}$ non ha influenza sui nomi ristretti non banali, abbiamo

$$\eta_x \in n(dec(q, e, i)) \iff \eta_x \in n(dec(q', e, i))$$

La tesi, in questo sottocaso, è quindi conseguenza dell'ipotesi induttiva;

- quando $a \notin fn(q') \wedge \overline{a} \in fn(q')$, la dimostrazione è del tutto analoga a quella del sottocaso precedente;

– quando $a, \overline{a} \notin fn(q')$ per ipotesi induttiva su q' abbiamo

$$\eta_x \in n(dec(q', e, i)) \implies \exists k \in \mathbb{N}.x = i + k \cdot 2^e$$

Abbiamo banalmente che

$$n(dec(q, e, i)) = n(dec((\nu a)q', e, i)) = n(dec(q', e, i))$$

e quindi, anche in questo sottocaso, la tesi è banalmente conseguenza dell'ipotesi induttiva;

- se $q = B$, dove $B \stackrel{def}{=} q'$ per un opportuno processo $q' \in \mathcal{P}$, per induzione strutturale su q' abbiamo che

$$\eta_x \in n(dec(q', e, i)) \implies \exists k \in \mathbb{N}.x = i + k \cdot 2^e$$

e poiché

$$n(dec(q, e, i)) = n(dec(B, e, i)) = n(dec(q', e, i))$$

la tesi, in questo caso, è conseguenza dell'ipotesi induttiva.

$\square$

Abbiamo quindi visto che la presenza di un'azione ristretta tra i nomi di una marcatura ottenuta dalla decomposizione di un processo q (non esteso) impone che il suo indice soddisfi un'equazione numerica dipendente dai parametri di supporto e ed i utilizzati per decomporre q. Che una tale dipendenza di fosse era sostanzialmente ovvio (dal momento che i parametri di supporto hanno come unica influenza quella di determinare gli indici delle azioni ristrette) ma con la proposizione appena vista abbiamo anche una precisa equazione.

Si noti che l'implicazione opposta di quella appena dimostrata non vale: nulla garantisce che se $x = i + k \cdot 2^e$, per un opportuno $k \in \mathbb{N}$, allora $\eta_x \in n(dec(q, e, i))$ o $\overline{\eta_x} \in n(dec(q, e, i))$; un controesempio trivialmente ovvio è quello dato dal processo nullo $q = \mathbf{0}$, per il quale $n(dec(q, e, i)) = \emptyset$.

Un altro risultato sostanzialmente ovvio (osservando come è definita la funzione *dec*) ma che ritengo sia comunque opportuno dimostrare (in maniera molto simile alla proposizione precedente) è il seguente.

Proposizione 4.1.12. *Sia $q \in \mathcal{P}$, un generico processo Multi-CCS; siano $e, i \in \mathbb{N}$ due generici numeri naturali; allora*

$$\eta_c, \overline{\eta_d} \notin n(dec(q, e, i))$$

Dimostrazione. Lo dimostriamo per induzione strutturale su q:

- se $q = \mathbf{0}$, abbiamo

  $$n(dec(q, e, i)) = N(q, e, i, \emptyset) = N(\mathbf{0}, e, i, \emptyset) = \emptyset$$

  ed è quindi ovvio che $\eta_c, \overline{\eta_d} \notin n(dec(q, e, i))$;

- se $q = \mu.q'$, per un'opportuna azione $\mu \in \mathcal{A}ct$ e per un opportuno processo $q' \in \mathcal{P}$, per induzione strutturale su q' abbiamo

  $$\eta_c, \overline{\eta_d} \notin n(dec(q, e, i))$$

  Abbiamo

  $$\begin{aligned} n(dec(q, e, i)) &= N(q, e, i, \emptyset) = N(\mu.q', e, i, \emptyset) \\ &= \{\mu\} \cup N(q', e, i, \emptyset) = \{\mu\} \cup n(dec(q', e, i)) \end{aligned}$$

  e dal momento che $\mu \in \mathcal{A}ct \implies \eta_c \neq \mu \neq \overline{\eta_d}$, la tesi deriva banalmente dall'ipotesi induttiva;

- se $q = \underline{\mu}.q'$, per un'opportuna azione $\mu \in \mathcal{A}ct$ e per un opportuno processo $q' \in \mathcal{P}$, la dimostrazione è uguale a quella del caso precedente;

- se $q = p_1 + p_2$, per opportuni processi sequenziali $p_1, p_2 \in \mathcal{P}_{seq}$, per induzione strutturale su p_1 e su p_2 abbiamo

  $$\eta_c, \overline{\eta_d} \notin n(dec(p_1, e + 1, i))$$

$$\eta_c, \overline{\eta_d} \notin n(dec(p_2, e+1, i+2^e))$$

Abbiamo

$$\begin{aligned} n(dec(q,e,i)) &= N(q,e,i,\emptyset) = N(p_1+p_2,e,i,\emptyset) \\ &= N(p_1,e+1,i,\emptyset) \cup N(p_1,e+1,i+2^e) \\ &= n(dec(p_1,e+1,i)) \cup n(dec(p_2,e+1,i+2^e)) \end{aligned}$$

È quindi evidente che

$$\begin{aligned} &(\eta_c \notin n(dec(p_1,e+1,i))) \wedge (\eta_c \notin n(dec(p_2,e+1,i+2^e))) \\ &\quad \Longrightarrow \eta_c \notin n(dec(p_1,e+1,i)) \cup n(dec(p_2,e+1,i+2^e)) \\ &\qquad = n(dec(q,e,i)) \end{aligned}$$

Analogamente si prova che $\overline{\eta_d} \notin n(dec(q,e,i))$;

- se $q = q_1 \mid q_2$, per opportuni processi $q_1, q_2 \in \mathcal{P}$, la dimostrazione è identica a quella del caso precedente;
- se $q = (\nu a)q'$, per un opportuno nome $a \in \mathcal{L}$ e per un opportuno processo $q' \in \mathcal{P}$, possiamo distinguere i seguenti casi:
  - quando $a, \overline{a} \in fn(q')$, per ipotesi induttiva su q' abbiamo

    $$\eta_c, \overline{\eta_d} \notin n(dec(q', e, i+2^e))$$

    Abbiamo, ricordando anche la proposizione 3.2.5 (pag. 83) e il corollario 4.1.4 (pag. 126), che

    $$\begin{aligned} n(dec(q,e,i)) &= n(dec((\nu a)q',e,i)) = n(dec(q'\{\eta_i/a\},e,i+2^e)) \\ &= n(dec(q',e,i+2^e)\{\eta_i/a\}) \\ &= n(dec(q',e,i+2^e))\{\eta_i/a\} \end{aligned}$$

    Dal momento che la sostituzione $\{\eta_i/a\}$ non ha nessun effetto sulla presenza di η_c o di $\overline{\eta_d}$ in $n(dec(q',e,i+2^e))$, la tesi è conseguenza dell'ipotesi induttiva;

– quando $a \in fn(q') \wedge \overline{a} \notin fn(q')$, per ipotesi induttiva su q' abbiamo

$$\eta_c, \overline{\eta_d} \notin n(dec(q', e, i))$$

Abbiamo, ricordando anche la proposizione 3.2.5 e il corollario 4.1.4, che

$$\begin{aligned} n(dec(q, e, i)) &= n(dec((\nu a)q', e, i)) = n(dec(q'\{\eta_d/a\}, e, i)) \\ &= n(dec(q', e, i)\{\eta_d/a\}) = n(dec(q', e, i))\{\eta_d/a\} \end{aligned}$$

e, dal momento che la sostituzione $\{\eta_d/a\}$ non ha influenza su η_c, il fatto che $\eta_c \notin n(dec(q, e, i))$ è immediata conseguenza dell'ipotesi induttiva; per quanto riguarda $\overline{\eta_d}$, osserviamo che dalla proposizione 4.1.9 (pag. 130) abbiamo che

$$\overline{a} \notin fn(q') \implies \overline{a} \notin n(dec(q', e, i))$$

e quindi la sostituzione $\{\eta_d/a\}$ potrà introdurre in $n(dec(q, e, i))$ delle azioni η_d ma non potrà introdurre delle azioni $\overline{\eta_d}$; dall'ipotesi induttiva possiamo quindi concludere anche che $\overline{\eta_d} \notin dec(q, e, i)$;

– quando $a \notin fn(q') \wedge \overline{a} \in fn(q')$, per ipotesi induttiva su q' abbiamo

$$\eta_c, \overline{\eta_d} \notin n(dec(q', e, i))$$

Abbiamo, ricordando anche la proposizione 3.2.5 e il corollario 4.1.4, che

$$\begin{aligned} n(dec(q, e, i)) &= n(dec((\nu a)q', e, i)) = n(dec(q'\{\eta_c/a\}, e, i)) \\ &= n(dec(q', e, i)\{\eta_c/a\}) = n(dec(q', e, i))\{\eta_c/a\} \end{aligned}$$

e, dal momento che la sostituzione $\{\eta_c/a\}$ non ha influenza su $\overline{\eta_d}$, il fatto che $\overline{\eta_d} \notin n(dec(q, e, i))$ è immediata conseguenza dell'ipotesi induttiva; per quanto riguarda η_c, osserviamo che dalla proposizione 4.1.9 abbiamo che

$$a \notin fn(q') \implies a \notin n(dec(q', e, i))$$

e quindi la sostituzione $\{\eta_c/a\}$ potrà introdurre in $n(dec(q,e,i))$ delle azioni $\overline{\eta_c}$ ma non potrà introdurre delle azioni η_c; dall'ipotesi induttiva possiamo quindi concludere anche che $\eta_c \notin dec(q,e,i)$;

- quando $a, \overline{a} \notin fn(q')$ per ipotesi induttiva su q' abbiamo

$$\eta_c, \overline{\eta_d} \notin n(dec(q',e,i))$$

Abbiamo banalmente che

$$n(dec(q,e,i)) = n(dec((\nu a)q',e,i)) = n(dec(q',e,i))$$

e quindi, sempre banalmente, la tesi, in questo sottocaso, è immediata conseguenza dell'ipotesi induttiva;

- se $q = B$, dove $B \stackrel{def}{=} q'$ per un opportuno processo $q' \in \mathcal{P}$, per induzione strutturale su q' abbiamo che

$$\eta_c, \overline{\eta_d} \notin n(dec(q',e,i))$$

e poiché

$$n(dec(q,e,i)) = n(dec(B,e,i)) = n(dec(q',e,i))$$

la tesi, in questo caso, è immediata conseguenza dell'ipotesi induttiva.

$\square$

Vediamo ora come si comportano i nomi delle marcature in corrispondenza delle transizioni.

Proposizione 4.1.13. *Siano $m, m' \in \mathcal{M}_{fin}(S_{MCCS})$ due marcature; se $m[t\rangle m'$, allora $n(m') \subseteq n(m)$.*

Dimostrazione. Si dimostra per casistica delle possibili transizioni e induzione sulle relative premesse:

- se la transizione t è conseguenza della regola (pref)[1], abbiamo che $m = m_1 \oplus dec(\mu.q, e, i)$, per un'opportuna azione estesa $\mu \in \mathcal{A}ct'$, per un opportuno processo esteso $q \in \mathcal{P}^{\mathcal{N}}$ e per opportuni valori interi $e, i \in \mathbb{N}$, con $dec(\mu.q, e, i) \xrightarrow{\mu} dec(q, e, i)$, con $l(t) = \mu$ e con $m' = m_1 \oplus dec(q, e, i)$; ricordando anche l'osservazione 4.1.8 (pag. 130) abbiamo

$$\begin{aligned} n(m') &= n(m_1 \oplus dec(q, e, i)) = n(m_1) \cup n(dec(q, e, i)) \\ &\subseteq n(m_1) \cup n(dec(\mu.q, e, i)) = n(m_1 \oplus dec(\mu.q, e, i)) = n(m) \end{aligned}$$

- se la transizione t è conseguenza della regola (s-pref), abbiamo che $m = m_1 \oplus dec(\underline{\mu}.q, e, i)$, per un'opportuna azione estesa $\mu \in \mathcal{A}ct'$, per un opportuno processo esteso $q \in \mathcal{P}^{\mathcal{N}}$ e per opportuni valori interi $e, i \in \mathbb{N}$, con $dec(\underline{\mu}.q, e, i) \xrightarrow{\sigma} m_2' \oplus m_3$, con $l(t) = \sigma$ e con $m' = m_1 + m_2' + m_3$; dalla premessa abbiamo che $m_2 \xrightarrow{\sigma'} m_2'$, che $dec(q, e, i) = m_2 \oplus m_3$ e che $conc(\mu, \sigma', \sigma)$; applicando l'induzione sulla premessa abbiamo che $n(m_2') \subseteq n(m_2)$; ricordando l'osservazione 4.1.8 abbiamo che $n(m_2) \cup n(m_3) = n(dec(q, e, i)) \subseteq n(dec(\underline{\mu}.q, e, i))$; tenendo conto di tutto questo abbiamo

$$\begin{aligned} n(m') &= n(m_1 \oplus m_2' \oplus m_3) = n(m_1) \cup n(m_2') \cup n(m_3) \\ &\subseteq n(m_1) \cup n(m_2) \cup n(m_3) = n(m_1) \cup n(m_2 \oplus m_3) \\ &= n(m_1) \cup n(dec(q, e, i)) \subseteq n(m_1) \cup n(dec(\underline{\mu}.q, e, i)) \\ &= n(m_1 \oplus dec(\underline{\mu}.q, e, i)) = n(m) \end{aligned}$$

- se la transizione t è conseguenza della regola (sum-1), abbiamo che $m = m_1 \oplus dec(p_1 + p_2, e, i)$, per opportuni processi sequenziali estesi $p_1, p_2 \in \mathcal{P}^{\mathcal{N}}_{seq}$ e per opportuni valori interi $e, i \in \mathbb{N}$, con $dec(p_1 + p_2, e, i) \xrightarrow{\sigma} m_2'$, con $l(t) = \sigma$ e con $m' = m_1 \oplus m_2'$; dall'induzione sulla premessa, ovvero $dec(p_1, e + 1, i) \xrightarrow{\sigma} m_2'$, abbiamo $n(m_2') \subseteq n(dec(p_1, e + 1, i))$; ricordando anche che l'osservazione 4.1.8 ci dice che $n(dec(p_1, e{+}1, i)) \subseteq n(dec(p_1 + p_2, e, i))$, abbiamo

$$n(m') = n(m_1 \oplus m_2') = n(m_1) \cup n(m_2') \subseteq n(m_1) \cup n(dec(p_1, e + 1, i))$$

[1]ovvero: se (pref) è l'ultima regola usata per giustificare la transizione

$$\begin{aligned}&\subseteq n(m_1) \cup n(dec(p_1 + p_2, e, i)) = n(m_1 \oplus dec(p_1 + p_2, e, i))\\&= n(m)\end{aligned}$$

- se la transizione t è conseguenza della regola (sum-2), la dimostrazione è analoga a quella del caso (sum-1) ma per induzione su $dec(p_2, e + 1, i + 2^e)$ anziché su $dec(p_1, e + 1, i)$;

- se la transizione t è conseguenza della regola (com), ovvero ottenuta dalla sincronizzazione di due transizioni t_1 e t_2, abbiamo che $m = m_1 \oplus m_2$, che $m' = m'_1 \oplus m'_2$, che $m_1[t_1\rangle m'_1$, che $m_2[t_2\rangle m'_2$ e che la sincronizzazione è $sync(l(t_1), l(t_2), l(t))$; per induzione sulla premesse abbiamo che $n(m'_1) \subseteq n(m_1)$ e che $n(m'_2) \subseteq n(m_2)$; abbiamo quindi

$$\begin{aligned}n(m') &= n(m'_1 \oplus m'_2) = n(m'_1) \cup n(m'_2)\\&\subseteq n(m_1) \cup n(m_2) = n(m_1 \oplus m_2) = n(m)\end{aligned}$$

□

Abbiamo quindi ottenuto un risultato che sarà molto utile in seguito: le transizioni non aumentano – anzi, a volte riducono – i nomi nelle marcature.

4.1.2 Azioni ristrette non banali di una marcatura

Definizione 4.1.2. *Dato un processo $q \in \mathcal{P}$, data una marcatura $m \in \mathcal{M}_{fin}(S_q)$ del sistema P/T Net(q), l'*insieme delle azioni ristrette attive *di m – che indichiamo con $r_a(m)$ – è l'insieme delle azioni ristrette non banali in m che possono dare origine a una sincronizzazione e quindi a una transizione, ovvero (per come abbiamo definito $\mathcal{N}_{\mathbb{N}}$ e dec)*

$$r_a(m) = n(m) \cap (\mathcal{N}_{\mathbb{N}} \cup \overline{\mathcal{N}_{\mathbb{N}}}) = \{\eta \mid \eta \in (\mathcal{N}_{\mathbb{N}} \cup \overline{\mathcal{N}_{\mathbb{N}}}) \wedge \eta \in n(m)\}$$

La precedente definizione è una versione modificata della definizione dei "*restricted names*" presente in [7] (appendice B) dalla quale abbiamo escluso le azioni ristrette banali. Osserviamo infatti che la precedente definizione è associata alle marcature presenti nei sistemi P/T, non alla più generica rete P/T; in quel contesto – per come è definita ora la funzione dec – i nomi ristretti banali η_d ed η_c possono essere considerati innocui (questo non è generalmente vero nella rete P/T) in quanto saranno presenti solo in maniera diretta, il primo, o complementato, il secondo, e quindi non potranno dare origine a sincronizzazione (necessaria per le azioni ristrette per essere rilevanti).

Dalla precedente proposizione 4.1.11 (pag. 136), deriva molto semplicemente il seguente

Corollario 4.1.14. *Sia $q \in \mathcal{P}$ un generico processo Multi-CCS; siano $e, i \in \mathbb{N}$ due generici numeri naturali; sia $\eta_x \in \mathcal{N}_\mathbb{N}$ un generico nome ristretto; abbiamo*

$$\eta_x \in r_a(dec(q, e, i)) \implies \exists k \in \mathbb{N}.x = i + k \cdot 2^e$$
$$\overline{\eta_x} \in r_a(dec(q, e, i)) \implies \exists k \in \mathbb{N}.x = i + k \cdot 2^e$$

Dimostrazione. Immediata conseguenza della definizione dei nomi ristretti attivi r_a di una marcatura e della proposizione 4.1.11 (pag. 136). □

Dal corollario appena dimostrato, è possibile ottenere un ulteriore corollario, molto utile in seguito.

Corollario 4.1.15. *Sia $q = q_1 \mid q_2$, con $q, q_1, q_2 \in \mathcal{P}$, un processo Multi-CCS ottenuto dalla composizione parallela di due generici processi; siano $e, i \in \mathbb{N}$ due generici numeri naturali; abbiamo*

$$r_a(dec(q_1, e+1, i)) \cap \overline{r_a(dec(q_2, e+1, i+2^e))} = \emptyset$$

ovvero abbiamo che la decomposizione di $dec(q_1 \mid q_2, e, i)$ genera due marcature che non possono sincronizzarsi tramite azioni ristrette.

Dimostrazione. Il presente corollario è una banale conseguenza del corollario 4.1.14: se, per assurdo, esistesse un nome ristretto $\eta_x \in \mathcal{N}_\mathbb{N}$ tale che $\eta_x \in r_a(dec(q_1, e+1, i)) \cap \overline{r_a(dec(q_2, e+1, i+2^e))}$, avremmo

$$\eta_x \in r_a(dec(q_1, e+1, i)) \implies \exists k_1 \in \mathbb{N}.x = i + k_1 \cdot 2^{e+1}$$

$$\overline{\eta_x} \in r_a(dec(q_2, e+1, i+2^e)) \implies \exists k_2 \in \mathbb{N}.x = i + 2^e + k_2 \cdot 2^{e+1}$$

Ovvero

$$\begin{aligned} i + k_1 \cdot 2^{e+1} = x = i + 2^e + k_2 \cdot 2^{e+1} &\implies k_1 \cdot 2^{e+1} = 2^e + k_2 \cdot 2^{e+1} \\ &\implies 2 \cdot k_1 \cdot 2^e = 2^e + 2 \cdot k_2 \cdot 2^e \\ &\implies 2 \cdot k_1 \cdot 2^e = (2 \cdot k_2 + 1) \cdot 2^e \\ &\implies 2 \cdot k_1 = 2 \cdot k_2 + 1 \end{aligned}$$

In altre parole, un numero pari $(2 \cdot k_1)$ dovrebbe essere uguale a un numero dispari $(2 \cdot k_2 + 1)$. Dal momento che questo è impossibile, la tesi è dimostrata.

In maniera del tutto analoga si prova che neppure un co-nome ristretto $\overline{\eta_x} \in \overline{\mathcal{N}_\mathbb{N}}$ può far parte sia di $r_a(dec(q_1, e+1, i))$ che di $\overline{r_a(dec(q_2, e+1, i+2^e))}$.

□

Dalla precedente proposizione 4.1.13, deriva molto semplicemente il seguente corollario

Corollario 4.1.16. *Siano $m, m' \in \mathcal{M}_{fin}(S_q)$ due marcature in un sistema P/T $Net(q)$, per un opportuno $q \in \mathcal{P}$; se esiste una transizione $t \in T_q$ tale che $m[t\rangle m'$, allora $r_a(m') \subseteq r_a(m)$.*

Dimostrazione. Dal momento che, per come è definito l'insieme dei nomi ristretti attivi, $r_a(m) = n(m) \cap (\mathcal{N}_\mathbb{N} \cup \overline{\mathcal{N}_\mathbb{N}})$, applicando la proposizione 4.1.13 (pag. 144) abbiamo

$$r_a(m') = n(m') \cap (\mathcal{N}_\mathbb{N} \cup \overline{\mathcal{N}_\mathbb{N}}) \subseteq n(m) \cap (\mathcal{N}_\mathbb{N} \cup \overline{\mathcal{N}_\mathbb{N}}) = r_a(m)$$

□

Ovvero, le transizioni non aumentano – e a volte riducono – anche l'insieme dei nomi ristretti non banali delle marcature.

4.1.3 Sostituzioni compatibili

Introduciamo ora il concetto di “sostituzione compatibile” che sarà centrale in seguito.

Definizione 4.1.3. *Sia* $m \in \mathcal{M}_{fin}(S_{MCCS})$ *una marcatura; diciamo che* $\{\eta/a\}$ *è una* sostituzione compatibile *per* m *quando* $a \in \mathcal{L}$, $\eta \in \mathcal{N}$ *e vale almeno una delle seguenti condizioni:*

- $\eta \in \mathcal{N}_{\mathbb{N}} \wedge \eta, \overline{\eta} \notin r_a(m)$
- $\eta = \eta_d \wedge \overline{a} \notin n(m)$
- $\eta = \eta_c \wedge a \notin n(m)$

In altre parole, una sostituzione è compatibile con una marcatura quando un nuovo nome ristretto non banale sostituisce un nome visibile oppure quando un nome ristretto banale sostituisce il tipo di nome visibile per la quale è stato predisposto o un nome visibile non presente libero (ovvero: senza la possibilità di fare “danni”).

In un certo senso, il concetto di “sostituzione compatibile” può anche essere visto come potenziale raccordo con altri, possibili, algoritmi di determinazione di nomi ristretti sostitutivi, a patto di usare il medesimo insieme $\mathcal{N}$ di nomi ristretti. In altre parole: fissato $\mathcal{N}$, il meccanismo che abbiamo visto, utilizzato per determinare i nomi ristretti sostitutivi (basato su una coppia di valori interi di supporto che determinano il primo indice disponibile e passo per il seguente indice, sull'individuazione della presenza delle azioni nel processo ristretto, ecc.), è solo uno dei possibili meccanismi; genera sostituzioni compatibili e possiamo ipotizzare altri meccanismi che (intuitivamente) saranno ragionevoli a patto di generare, a loro volta, sostituzioni compatibili.

Vediamo subito alcuni risultati relativi a questa definizione, che però sono strettamente legati allo specifico meccanismo che abbiamo adottato per stabilire i nomi ristretti da utilizzare.

Proposizione 4.1.17. *Per ogni processo $q \in \mathcal{P}$, per ogni nome visibile $a \in \mathcal{L}$ e per ogni terna di numeri naturali $e, i, j \in \mathbb{N}$, se vale la condizione*

$$\forall k \in \mathbb{N}. j \neq i + k \cdot 2^e$$

allora la sostituzione $\{\eta_j/a\}$ è compatibile con $dec(q, e, i)$.

Dimostrazione. Perché la sostituzione $\{\eta_j/a\}$ sia compatibile con $dec(q, e, i)$, deve valere la condizione $\eta_j, \overline{\eta_j} \notin r_a(dec(q, e, i))$.

Dal corollario 4.1.14 (pag. 147), rovesciandone le implicazioni, abbiamo

$$(\forall k \in \mathbb{N}. j \neq i + k \cdot 2^e) \implies \eta_j, \overline{\eta_j} \notin r_a(dec(q, e, i))$$

e quindi la tesi è verificata. □

Da tale proposizione discende banalmente il seguente corollario

Corollario 4.1.18. *Per ogni processo $q \in \mathcal{P}$, per ogni nome visibile $a \in \mathcal{L}$ e per ogni terna di numeri naturali $e, i, j \in \mathbb{N}$, se vale la condizione $j < i$, allora la sostituzione $\{\eta_j/a\}$ è compatibile con $dec(q, e, i)$.*

Dimostrazione. Osserviamo che

$$(j < i) \implies (\forall k \in \mathbb{N}. j \neq i + k \cdot 2^e)$$

La tesi è quindi conseguenza della proposizione 4.1.17. □

Disponiamo quindi, grazie al precedente corollario, di un criterio semplice per individuare, in alcune circostanze, delle sostituzioni compatibili.

Vediamo ora una proposizione che, in presenza di restrizione, ci permette di affermare l'esistenza di una sostituzione compatibile.

Proposizione 4.1.19. *Per ogni processo $q \in \mathcal{P}$, per ogni nome visibile $a \in \mathcal{L}$ e per ogni coppia di numeri naturali $e, i \in \mathbb{N}$, esistono un nome ristretto (banale o meno) $\eta \in \mathcal{N}$ e un numero naturale $j \in \mathbb{N}$ tali che $\{\eta/a\}$ è una sostituzione compatibile con $dec(q, e, j)$ e*

$$dec((\nu a)q, e, i) = dec(q, e, j)\{\eta/a\}$$

Dimostrazione. La funzione dec, in corrispondenza di una restrizione $(\nu a)q$, differenzia il proprio comportamento a seconda dei quattro possibili valori di $\{a, \overline{a}\} \cap fn(q)$; verifichiamo quindi i singoli casi:

- se $a, \overline{a} \in fn(q)$, ricordando anche la proposizione 3.2.5 (pag. 83), abbiamo

  $$dec((\nu a)q, e, i) = dec(q\{\eta_i/a\}, e, i + 2^e) = dec(q, e, i + 2^e)\{\eta_i/a\}$$

  Dal momento che $i < i + 2^e$, dal corollario 4.1.18 (pagina a fronte) abbiamo che $\{\eta_i/a\}$ è una sostituzione compatibile con $dec(q, e, i+2^e)$; abbiamo quindi che $\eta = \eta_i$ e $j = i + 2^e$ sono, rispettivamente, l'azione ristretta e il numero naturale cercati;

- se $a \in fn(q) \wedge \overline{a} \notin fn(q)$, ricordando anche la proposizione 3.2.5, abbiamo

  $$dec((\nu a)q, e, i) = dec(q\{\eta_d/a\}, e, i) = dec(q, e, i)\{\eta_d/a\}$$

  Tenendo conto della proposizione 4.1.9 (pag. 130) abbiamo che $\overline{a} \notin n(dec(q, e, i))$ e quindi la sostituzione $\{\eta_d/a\}$ è compatibile con la marcatura $dec(q, e, i)$; abbiamo quindi che $\eta = \eta_d$ e $j = i$ sono, rispettivamente, l'azione ristretta e il numero naturale cercati;

- se $a \notin fn(q) \wedge \overline{a} \in fn(q)$ si dimostra, in maniera analoga al caso precedente, che $dec((\nu a)q, e, i) = dec(q, e, i)\{\eta_c/a\}$, che $a \notin n(dec(q, e, i))$, che $\{\eta_c/a\}$ è una sostituzione compatibile con $dec(q, e, i)$ e quindi che $\eta = \eta_c$ e $j = i$ sono l'azione ristretta e il numero naturale cercati;

- se $a, \overline{a} \notin fn(q)$ abbiamo $dec((\nu a)q, e, i) = dec(q, e, i)$; rileviamo che né l'azione a né la sua azione complementare sono libere in q e, quindi, la sostituzione di a con qualunque altra azione, ristretta o meno, è ininfluente; grazie alla proposizione 4.1.9 sappiamo che $a, \overline{a} \notin n(dec(q, e, i))$ e questo rende compatibile con $dec(q, e, i)$ entrambe le sostituzioni

$\{\eta_d/a\}$ ed $\{\eta_c/a\}$; possiamo quindi affermare (scegliendo η_d, ma avremmo potuto scegliere anche η_c) che $\{\eta_d/a\}$ è una sostituzione compatibile con $dec(q, e, i)$ e che

$$dec((\nu a)q, e, i) = dec(q, e, i) = dec(q\{\eta_d/a\}, e, i) = dec(q, e, i)\{\eta_d/a\}$$

Abbiamo quindi che $\eta = \eta_d$ e $j = i$ sono, rispettivamente, l'azione ristretta e il numero naturale cercati

□

4.1.4 Restrizione e transizioni

Vediamo ora alcuni risultati, relativi agli effetti delle sostituzioni alle transizioni, che torneranno utili in alcune delle dimostrazioni che seguiranno.

Proposizione 4.1.20. *Se $t \in \longrightarrow$ è una generica transizione e $\{\eta/a\}$ è una generica sostituzione di un nome visibile $a \in \mathcal{L}$ con un nome ristretto $\eta \in \mathcal{N}$, allora esiste una transizione $t_1 \in \longrightarrow$ tale che $t_1 = ({}^\bullet t\{\eta/a\}, l(t)\{\eta/a\}, t^\bullet\{\eta/a\})$ e che $\gamma(t_1) = \gamma(t)$.*

Dimostrazione. Nel corso della dimostrazione utilizzeremo frequentemente il risultato della proposizione 3.2.5 (pag. 83), ovvero che, in generale,

$$dec(q\{\eta/a\}, e, i) = dec(q, e, i)\{\eta/a\}$$

Dimostriamo la tesi per casistica su t e per induzione sulle relative premesse:

- se t è giustificata dalla regola (pref), allora esistono un'opportuna azione estesa $\mu \in \mathcal{A}ct'$, un opportuno processo esteso $q \in \mathcal{P}^{\mathcal{N}}$ e due opportuni numeri naturali $e, i \in \mathbb{N}$ tali che ${}^\bullet t = dec(\mu.q, e, i)$; in base alla regola (pref) abbiamo $t = (dec(\mu.q, e, i), \mu, dec(q, e, i))$; riguardo a $\{\eta/a\}$ possiamo distinguere tre casi:

– quando $\mu = a$, abbiamo

$$\begin{aligned} {}^{\bullet}t\{\eta/a\} &= dec(a.q, e, i)\{\eta/a\} = dec((a.q)\{\eta/a\}, e, i) \\ &= dec(\eta.(q\{\eta/a\}), e, i) \end{aligned}$$

Applicando la regola (pref) anche a $dec(\eta.(q\{\eta/a\}), e, i)$, possiamo concludere che esiste una transizione $t_1 = (dec(\eta.(q\{\eta/a\}), \eta, dec(q\{\eta/a\}, e, i))$; la transizione $t_1 \in \longrightarrow$ è la transizione cercata; infatti, abbiamo (per costruzione) che ${}^{\bullet}t_1 = {}^{\bullet}t\{\eta/a\}$ e, per quanto riguarda etichetta e postset,

$$\begin{aligned} l(t_1) &= \eta = a\{\eta/a\} = l(t)\{\eta/a\} \\ t_1^{\bullet} &= dec(q\{\eta/a\}, e, i) = dec(q, e, i)\{\eta/a\} = t^{\bullet}\{\eta/a\} \end{aligned}$$

– quando $\mu = \overline{a}$, la dimostrazione è del tutto analoga a quella del sottocaso precedente;

– quando $a \neq \mu \neq \overline{a}$, abbiamo

$$\begin{aligned} {}^{\bullet}t\{\eta/a\} &= dec(\mu.q, e, i)\{\eta/a\} = dec((\mu.q)\{\eta/a\}, e, i) \\ &= dec(\mu.(q\{\eta/a\}), e, i) \end{aligned}$$

Applicando la regola (pref) anche a $dec(\mu.(q\{\eta/a\}), e, i)$, possiamo concludere che esiste una transizione $t_1 = (dec(\mu.(q\{\eta/a\}), \mu, dec(q\{\eta/a\}, e, i))$; la transizione $t_1 \in \longrightarrow$ è la transizione cercata; infatti, abbiamo (per costruzione) che ${}^{\bullet}t_1 = {}^{\bullet}t\{\eta/a\}$ e, per quanto riguarda etichetta e postset,

$$\begin{aligned} l(t_1) &= \mu = \mu\{\eta/a\} = l(t)\{\eta/a\} \\ t_1^{\bullet} &= dec(q\{\eta/a\}, e, i) = dec(q, e, i)\{\eta/a\} = t^{\bullet}\{\eta/a\} \end{aligned}$$

In tutti i tre sottocasi abbiamo anche che sia t che t_1 sono giustificate dalla regola (pref) e quindi, dalla definizione 3.2.13 (pag. 95), $\gamma(t) = 1 = \gamma(t_1)$;

- se t è giustificata dalla regola (s-pref), allora esistono un'opportuna azione estesa $\mu \in \mathcal{A}ct'$, un opportuno processo esteso $q \in \mathcal{P}^{\mathcal{N}}$ e due opportuni numeri naturali $e, i \in \mathbb{N}$ tali che ${}^\bullet t = dec(\underline{\mu}.q, e, i)$; la regola (s-pref) ci garantisce l'esistenza di una transizione $t' \in \longrightarrow$ tale che ${}^\bullet t' \subseteq dec(q, e, i)$, che $conc(\mu, l(t'), l(t))$ e che $t'^\bullet = t^\bullet$; applicando l'ipotesi induttiva alla transizione t', otteniamo che esiste una transizione $t'_1 = ({}^\bullet t'\{\eta/a\}, l(t')\{\eta/a\}, t'^\bullet\{\eta/a\})$ con $\gamma(t'_1) = \gamma(t')$; per quanto riguarda il preset di t'_1 osserviamo che da ${}^\bullet t' \subseteq dec(q, e, i)$ e da ${}^\bullet t'_1 = {}^\bullet t'\{\eta/a\}$ otteniamo ${}^\bullet t'_1 \subseteq dec(q, e, i)\{\eta/a\} = dec(q\{\eta/a\}, e, i)$ e quindi che possiamo usare la transizione t'_1 come premessa per la regola (s-pref); anche in questo caso, distinguiamo tre sottocasi relativi a μ e ad a:

  - quando $\mu = a$, abbiamo

    $$\begin{aligned} {}^\bullet t\{\eta/a\} &= dec(a.q, e, i)\{\eta/a\} = dec((a.q)\{\eta/a\}, e, i) \\ &= dec(\eta.(q\{\eta/a\}), e, i) \end{aligned}$$

    Applicando la regola (s-pref), usando la transizione t'_1 come premessa, a $dec(\eta.(q\{\eta/a\}), e, i)$, possiamo concludere che esiste una transizione $t_1 = (dec(\eta.(q\{\eta/a\}), \sigma, dec(q\{\eta/a\}, e, i))$, per un'opportuna etichetta σ tale che $conc(\eta, l(t'_1), \sigma)$; la transizione $t_1 \in \longrightarrow$ è la transizione cercata; infatti, abbiamo (per costruzione) che ${}^\bullet t_1 = {}^\bullet t\{\eta/a\}$; per quanto riguarda l'etichetta osserviamo che $l(t_1) = \sigma$ è ottenuta dalla concatenazione di $\eta = a\{\eta/a\}$ e di $l(t'_1)$ che è pari (per ipotesi induttiva) a $l(t')\{\eta/a\}$; dal momento che $l(t)$ è ottenuta come concatenazione di a e di $l(t')$, è evidente che $l(t_1) = l(t)\{\eta/a\}$; per quanto riguarda il postset, abbiamo (ricordando la regola (s-pref)) che

    $$t_1^\bullet = t_1'^\bullet = t'^\bullet\{\eta/a\} = t^\bullet\{\eta/a\}$$

  - quando $\mu = \overline{a}$, la dimostrazione è del tutto analoga a quella del sottocaso precedente;

– quando $a \neq \mu \neq \overline{a}$, abbiamo

$$
\begin{aligned}
{}^{\bullet}t\{\eta/a\} &= dec(\mu.q, e, i)\{\eta/a\} = dec((\mu.q)\{\eta/a\}, e, i) \\
&= dec(\mu.(q\{\eta/a\}), e, i)
\end{aligned}
$$

Applicando la regola (s-pref), usando la transizione t'_1 come premessa, a $dec(\mu.(q\{\eta/a\}), e, i)$, possiamo concludere che esiste una transizione $t_1 = (dec(\eta.(q\{\eta/a\}), \sigma, dec(q\{\eta/a\}, e, i))$, per un'opportuna etichetta σ tale che $conc(\mu, l(t'_1), \sigma)$; la transizione $t_1 \in\longrightarrow$ è la transizione cercata; infatti, abbiamo (per costruzione) che ${}^{\bullet}t_1 = {}^{\bullet}t\{\eta/a\}$; per quanto riguarda l'etichetta osserviamo che $l(t_1) = \sigma$ è ottenuta dalla concatenazione di $\mu = \mu\{\eta/a\}$ e di $l(t'_1)$ che è pari (per ipotesi induttiva) a $l(t')\{\eta/a\}$; dal momento che $l(t)$ è ottenuta come concatenazione di μ e di $l(t')$, è evidente che $l(t_1) = l(t)\{\eta/a\}$; per quanto riguarda il postset, abbiamo (ricordando la regola (s-pref)) che

$$t_1^{\bullet} = t_1'^{\bullet} = t'^{\bullet}\{\eta/a\} = t^{\bullet}\{\eta/a\}$$

In tutti i tre sottocasi abbiamo che sia t che t_1 sono ottenute dalla regola (s-pref) usando come premessa transizioni t' e t'_1 tali che $\gamma(t') = \gamma(t'_1)$; ricordando la definizione 3.2.13 abbiamo quindi

$$\gamma(t) = \gamma(t') + 1 = \gamma(t'_1) + 1 = \gamma(t_1)$$

- se t è giustificata dalla regola (sum-1), allora esistono due opportuni processi sequenziali estesi $p_1, p_2 \in \mathcal{P}^{\mathcal{N}}_{seq}$ e due opportuni numeri naturali $e, i \in \mathbb{N}$ tali che ${}^{\bullet}t = dec(p_1 + p_2, e, i)$; dalla premessa della regola (sum-1) sappiamo che esiste una transizione $t' \in\longrightarrow$ tale che $t' = (dec(p_1, e+1, i), l(t), t^{\bullet})$; applicando l'ipotesi induttiva alla transizione t', otteniamo l'esistenza di una transizione $t'_1 \in\longrightarrow$ pari a $t'_1 = ({}^{\bullet}t'\{\eta/a\}, l(t')\{\eta/a\}, t'^{\bullet}\{\eta/a\})$ con $\gamma(t'_1) = \gamma(t')$; dal momento che ${}^{\bullet}t'_1 = {}^{\bullet}t'\{\eta/a\} = dec(p_1, e+1, i)\{\eta/a\} = dec(p_1\{\eta/a\}, e+1, i)$, possiamo usare t'_1 come premessa della regola (sum-1) e ottenere l'esistenza di una transizione $t_1 = (dec(p_1\{\eta/a\} + p_2\{\eta/a\}, e, i), l(t'_1), t_1'^{\bullet})$ la transizione $t_1 \in \longrightarrow$ è la transizione cercata, infatti

$$\begin{aligned} {}^{\bullet}t_1 &= dec(p_1\{\eta/a\} + p_2\{\eta/a\}, e, i) = dec(p_1 + p_2, e, i)\{\eta/a\} \\ &= {}^{\bullet}t\{\eta/a\} \\ l(t_1) &= l(t'_1) = l(t')\{\eta/a\} = l(t)\{\eta/a\} \\ t_1^{\bullet} &= t_1'^{\bullet} = t'^{\bullet}\{\eta/a\} = t^{\bullet}\{\eta/a\} \end{aligned}$$

Dal momento che sia t che t_1 sono ottenute dalla regola (sum-1) usando come premessa transizioni t' e t'_1 tali che $\gamma(t') = \gamma(t'_1)$; ricordando la definizione 3.2.13 abbiamo anche

$$\gamma(t) = \gamma(t') + 1 = \gamma(t'_1) + 1 = \gamma(t_1)$$

- se t è giustificata dalla regola (sum-2), la dimostrazione è del tutto analoga a quella del caso precedente ma operando per induzione sulla seconda componente additiva anziché sulla prima;

- se t è giustificata dalla regola (com), allora esistono due transizioni $t_a, t_b \in \longrightarrow$ che si sincronizzano in t; applicando l'induzione strutturale sia a t_a che a t_b otteniamo che esistono due transizioni $t_{a1}, t_{b1} \in \longrightarrow$ pari a $t_{a1} = ({}^{\bullet}t_a\{\eta/a\}, l(t_a)\{\eta/a\}, t_a^{\bullet}\{\eta/a\})$ e $t_{b1} = ({}^{\bullet}t_b\{\eta/a\}, l(t_b)\{\eta/a\}, t_b^{\bullet}\{\eta/a\})$, con $\gamma(t_{a1}) = \gamma(t_a)$ e $\gamma(t_{b1}) = \gamma(t_b)$; dal momento che $sync(l(t_a), l(t_b), l(t))$, che $l(t_{a1}) = l(t_a)\{\eta/a\}$ e che $l(t_{b1}) = l(t_b)\{\eta/a\}$, possiamo sincronizzare le etichette di t_{a1} e di t_{b1} nello stesso modo (o quasi: alle sincronizzazioni tra a e $\overline{a}$ si sostituiscono sincronizzazioni tra η e $\overline{\eta}$) in cui si sincronizzano le etichette di t_a e di t_b ottenendo un'etichetta $l(t)\{\eta/a\}$; esiste quindi una transizione $t_1 \in \longrightarrow$, con etichetta $l(t)\{\eta/a\}$, ottenuta sincronizzando t_{a1} e t_{b1}; t_1 è la transizione cercata; abbiamo già verificato (o meglio, costruito t_1 in maniera tale che) $l(t_1) = l(t)\{\eta/a\}$; relativamente a preset e postset abbiamo

$$ {}^{\bullet}t_1 = {}^{\bullet}t_{a1} \oplus {}^{\bullet}t_{b1} = {}^{\bullet}t_a\{\eta/a\} \oplus {}^{\bullet}t_b\{\eta/a\} = ({}^{\bullet}t_a \oplus {}^{\bullet}t_b)\{\eta/a\} = {}^{\bullet}t\{\eta/a\}$$

$$t_1^\bullet = t_{a1}^\bullet \oplus t_{b1}^\bullet = t_a^\bullet\{\eta/a\} \oplus t_b^\bullet\{\eta/a\} = (t_a^\bullet \oplus t_b^\bullet)\{\eta/a\} = t^\bullet\{\eta/a\}$$

Dal momento che sia t che t_1 sono ottenute dalla regola (com) sincronizzando transizioni t_a e t_b (la prima), t_{a1} e t_{b1} (la seconda) tali che $\gamma(t_a) = \gamma(t_{a1})$ e $\gamma(t_b) = \gamma(t_{b1})$; ricordando la definizione 3.2.13 abbiamo anche

$$\gamma(t) = \gamma(t_a) + \gamma(t_b) + 1 = \gamma(t_{a1}) + \gamma(t_{b1}) + 1 = \gamma(t_1)$$

□

La proposizione appena vista può essere estesa alle marcature che supportano contemporaneamente multiset di transizioni.

Però, prima di procedere, ricordando che sulle reti P/T abbiamo definito, in 1.2.5 (pag. 14), i multiset di transazioni contemporaneamente supportate da una marcatura mentre per i processi Multi-CCS abbiamo definito, in 2.3.3 (pag. 46), i multiset di etichette di transizioni contemporaneamente supportate, definiamo una sorta di ovvio "raccordo" tra le due definizioni che sarà anche utile per formulare in maniera più sintetica il seguente corollario e non solo.

Definizione 4.1.4. *Dato un multiset $u \in \mathcal{M}_{fin}(T_{MCCS})$ di transizioni della rete P/T associato al Multi-CCS, pari a*

$$u = \bigoplus_{i=1}^{n} k_i \cdot t_i$$

dove $k_1, k_2, \ldots, k_n \in \mathbb{N}$ e $t_1, t_2, \ldots, t_n \in T_{MCCS}$, il multiset di $u' \in \mathcal{M}_{fin}(\mathcal{A})$ di etichette delle transizioni associato a u – che indichiamo sinteticamente con $u' = mtl(u)$ – è pari a

$$u' = mtl(u) = \bigoplus_{i=1}^{n} k_i \cdot l(t_i)$$

Possiamo ora dimostrare il seguente

Corollario 4.1.21. *Se $u \in \mathcal{M}_{fin}(\longrightarrow)$ è un generico multiset di transizioni – pari a $u = k_1t_1 \oplus k_2t_2 \oplus \ldots \oplus k_nt_n$, per opportuni numeri interi $k_1, k_2, \ldots, k_n \in \mathbb{N}$ e opportune transizioni $t_1, t_2, \ldots, t_n \in\longrightarrow$ – e $m, m' \in \mathcal{M}_{fin}(S_{MCCS})$ sono due marcature tali che $m[[u\rangle\rangle m'$, se $\{\eta/a\}$ una generica sostituzione di un nome visibile $a \in \mathcal{L}$ con un nome ristretto $\eta \in \mathcal{N}$, allora esiste un multiset di transizioni $u' \in \mathcal{M}_{fin}(\longrightarrow)$, pari a $u' = k_1t'_1 \oplus k_2t'_2 \oplus \ldots \oplus k_nt'_n$ – sui medesimi numeri interi $k_1, k_2, \ldots, k_n \in \mathbb{N}$ e con transizioni $t'_1, t'_2, \ldots, t'_n \in\longrightarrow$ tali, per $i \in [1 \ldots n]$, che $t'_i = ({}^\bullet t_i\{\eta/a\}, l(t_i)\{\eta/a\}, t_i^\bullet\{\eta/a\})$ e che $\gamma(t'_i) = \gamma(t_i)$ – tale che $m\{\eta/a\}[[u'\rangle\rangle m'\{\eta/a\}$.*

Dimostrazione. Se $m[[u\rangle\rangle m'$, con $u = k_1t_1 \oplus k_2t_2 \oplus \ldots \oplus k_nt_n$, ricordando anche la definizione 1.2.5 (pag. 14) abbiamo che la marcatura m può essere scissa in $n+1$ sub-marcature, le prime n delle quali sono pari al preset di $k_i \cdot t_i$[2], al variare di $i \in [1 \ldots n]$, e l'ultima viene lasciata inalterata dall'applicazione delle transizioni; più formalmente abbiamo che

$$m = \left(\bigoplus_{i=1}^{n} k_i \cdot {}^\bullet t_i \right) \oplus m_1$$

per un'opportuna marcatura m_1, potenzialmente anche nulla. Sempre dalla definizione 1.2.5 abbiamo che

$$\begin{aligned}
m' &= \left(m \setminus \left(\bigoplus_{i=1}^{n} k_i \cdot {}^\bullet t_i \right) \right) \oplus \left(\bigoplus_{i=1}^{n} k_i \cdot t_i^\bullet \right) \\
&= \left(\left(\left(\bigoplus_{i=1}^{n} k_i \cdot {}^\bullet t_i \right) \oplus m_1 \right) \setminus \left(\bigoplus_{i=1}^{n} k_i \cdot {}^\bullet t_i \right) \right) \oplus \left(\bigoplus_{i=1}^{n} k_i \cdot t_i^\bullet \right) \\
&= \left(\bigoplus_{i=1}^{n} k_i \cdot t_i^\bullet \right) \oplus m_1
\end{aligned}$$

Applicando alle transizioni $t_1, t_2, \ldots, t_n$ la proposizione 4.1.20 (pag. 152)[3] otteniamo che esistono n transizioni $t'_i = ({}^\bullet t_i\{\eta/a\}, l(t_i)\{\eta/a\}, t_i^\bullet\{\eta/a\})$, al variare di $i \in [1 \ldots n]$, con $\gamma(t'_i) = \gamma(t_i)$; $u' = k_1t'_1 \oplus k_2t_2 \oplus \ldots k_nt_n$ è il multiset di transizioni cercato; rimane solo da provare che $m\{\eta/a\}[[u'\rangle\rangle m'\{\eta/a\}$;

[2] o, più esattamente, sono pari a k_i volte il preset di t_i

[3] per l'esattezza, applicando tale proposizione k_1 volte a t_1, k_2 volte a t_2, ecc.

applicando u' a $m\{\eta/a\}$ otteniamo

$$
\begin{aligned}
&\left(m\{\eta/a\} \setminus \left(\bigoplus_{i=1}^{n} k_i{}^{\bullet}t'_i\right)\right) \oplus \left(\bigoplus_{i=1}^{n} k_i t'^{\bullet}_i\right)\\
&= \left(\left(\left(\bigoplus_{i=1}^{n} k_i{}^{\bullet}t_i\right) \oplus m_1\right)\{\eta/a\} \setminus \left(\bigoplus_{i=1}^{n} k_i{}^{\bullet}t'_i\right)\right) \oplus \left(\bigoplus_{i=1}^{n} k_i t'^{\bullet}_i\right)\\
&= \left(\left(\left(\bigoplus_{i=1}^{n} k_i{}^{\bullet}t_i\right)\{\eta/a\} \oplus m_1\{\eta/a\}\right) \setminus \left(\bigoplus_{i=1}^{n} k_i{}^{\bullet}t'_i\right)\right) \oplus \left(\bigoplus_{i=1}^{n} k_i t'^{\bullet}_i\right)\\
&= \left(\left(\left(\bigoplus_{i=1}^{n} (k_i{}^{\bullet}t_i)\{\eta/a\}\right) \oplus m_1\{\eta/a\}\right) \setminus \left(\bigoplus_{i=1}^{n} k_i{}^{\bullet}t'_i\right)\right) \oplus \left(\bigoplus_{i=1}^{n} k_i t'^{\bullet}_i\right)\\
&= \left(\left(\left(\bigoplus_{i=1}^{n} k_i({}^{\bullet}t_i\{\eta/a\})\right) \oplus m_1\{\eta/a\}\right) \setminus \left(\bigoplus_{i=1}^{n} k_i{}^{\bullet}t'_i\right)\right) \oplus \left(\bigoplus_{i=1}^{n} k_i t'^{\bullet}_i\right)\\
&= \left(\left(\left(\bigoplus_{i=1}^{n} k_i{}^{\bullet}t'_i\right) \oplus m_1\{\eta/a\}\right) \setminus \left(\bigoplus_{i=1}^{n} k_i{}^{\bullet}t'_i\right)\right) \oplus \left(\bigoplus_{i=1}^{n} k_i t'^{\bullet}_i\right)\\
&= m_1\{\eta/a\} \oplus \left(\bigoplus_{i=1}^{n} k_i t'^{\bullet}_i\right) = m_1\{\eta/a\} \oplus \left(\bigoplus_{i=1}^{n} k_i(t^{\bullet}_i\{\eta/a\})\right)\\
&= m_1\{\eta/a\} \oplus \left(\bigoplus_{i=1}^{n} (k_i t^{\bullet}_i)\{\eta/a\}\right) = m_1\{\eta/a\} \oplus \left(\bigoplus_{i=1}^{n} k_i t^{\bullet}_i\right)\{\eta/a\}\\
&= \left(m_1 \oplus \left(\bigoplus_{i=1}^{n} k_i t^{\bullet}_i\right)\right)\{\eta/a\} = m'\{\eta/a\}
\end{aligned}
$$

ovvero abbiamo verificato anche che $m\{\eta/a\}[[u'\rangle\rangle m'\{\eta/a\}$. □

Dato il corollario precedente, è immediato il seguente

Corollario 4.1.22. *Se $t \in\longrightarrow$ è una generica transizione e $m, m' \in \mathcal{M}_{fin}(S_{MCCS})$ due marcature tali che $m[t\rangle m'$, se $\{\eta/a\}$ una generica sostituzione di un nome visibile $a \in \mathcal{L}$ con un nome ristretto $\eta \in \mathcal{N}$, allora esiste una transizione $t_1 \in\longrightarrow$, pari a $t_1 = ({}^{\bullet}t\{\eta/a\}, l(t)\{\eta/a\}, t^{\bullet}\{\eta/a\})$, tale che $m\{\eta/a\}[t_1\rangle m'\{\eta/a\}$ e che $\gamma(t_1) = \gamma(t)$.*

Dimostrazione. Immediata conseguenza del corollario 4.1.21 (pagina a fronte), applicata al multiset di transizioni $u = 1 \cdot t$: otteniamo infatti un multiset $u_1 = 1 \cdot t_1$, con $t_1 = ({}^\bullet t\{\eta/a\}, l(t)\{\eta/a\}, t^\bullet\{\eta/a\})$ e $\gamma(t_1) = \gamma(t)$; t_1 è, ovviamente, la transizione cercata. $\square$

I risultati appena visti riguardano transizioni in $\longrightarrow$, ovvero con etichette che possono contenere una o più transizioni ristrette; il più delle volte saremo invece interessati a transizioni in T_{MCCS}, ovvero con etichette prive di azioni ristrette; i due risultati precedenti continuano ovviamente a valere, a patto che sia rispettata un'ulteriore – ovvia – condizione.

Corollario 4.1.23. *Sia $t \in T_{MCCS}$ è una generica transizione; sia $\{\eta/a\}$ una generica sostituzione di un nome visibile $a \in \mathcal{L}$ con un nome ristretto $\eta \in \mathcal{N}$; se $a, \overline{a} \notin n(l(t))$ esiste una transizione $t_1 \in T_{MCCS}$ pari a $t_1 = ({}^\bullet t\{\eta/a\}, l(t), t^\bullet\{\eta/a\})$ e con $\gamma(t_1) = \gamma(t)$.*

Dimostrazione. È evidente che

$$(t \in T_{MCCS} \wedge T_{MCCS} \subseteq \longrightarrow) \implies t \in \longrightarrow$$

e quindi che possiamo applicare la proposizione 4.1.20 (pag. 152) a t e ottenere che esiste una transizione $t_1 \in \longrightarrow$ pari a $t_1 = ({}^\bullet t\{\eta/a\}, l(t)\{\eta/a\}, t^\bullet\{\eta/a\})$ e con $\gamma(t_1) = \gamma(t)$.

Osserviamo che, siccome $a, \overline{a} \notin n(l(t))$, la sostituzione $\{\eta/a\}$, applicata all'etichetta di t, non sostituisce nulla, ovvero $l(t_1) = l(t)\{\eta/a\} = l(t)$; quest'ultima uguaglianza di etichette prova anche che $t_1 \in T_{MCCS}$ e quindi che t_1 è la transizione cercata. $\square$

Sul supporto contemporaneo di transizioni abbiamo

Corollario 4.1.24. *Se $u \in \mathcal{M}_{fin}(T_{MCCS})$ è un generico multiset di transizioni T_{MCCS} – pari a $u = k_1 t_1 \oplus k_2 t_2 \oplus \ldots \oplus k_n t_n$, per opportuni numeri interi $k_1, k_2, \ldots, k_2 \in \mathbb{N}$ e opportune transizioni $t_1, t_2, \ldots, t_n \in T_{MCCS}$ dove $a, \overline{a} \notin n(l(t_i))$, per ogni $i \in [1 \ldots n]$ – e $m, m' \in \mathcal{M}_{fin}(S_{MCCS})$ sono due marcature tali che $m[[u\rangle\rangle m'$, se $\{\eta/a\}$ una generica sostituzione di un nome visibile $a \in \mathcal{L}$ con un nome ristretto $\eta \in \mathcal{N}$, allora esiste un multiset di transizioni $u' \in \mathcal{M}_{fin}(T_{MCCS})$, pari a $u' = k_1 t'_1 \oplus k_2 t'_2 \oplus \ldots \oplus k_n t'_n$ – sui medesimi* *numeri interi* $k_1, k_2, \ldots, k_2 \in \mathbb{N}$ *e con transizioni* $t'_1, t'_2, \ldots, t'_n \in T_{MCCS}$ *tali, per* $i = [1 \ldots n]$*, che* $t'_i = ({}^\bullet t_i\{\eta/a\}, l(t_i), t_i^\bullet\{\eta/a\})$ *e che* $\gamma(t'_i) = \gamma(t_i)$ *– tale che* $m\{\eta/a\}[[u'\rangle\rangle m'\{\eta/a\}$*.*

Dimostrazione. Immediata conseguenza del corollario 4.1.21 (pag. 158), considerando che $u \in \mathcal{M}_{fin}(T_{MCCS}) \subseteq \mathcal{M}_{fin}(\longrightarrow)$; l'ipotesi aggiuntiva $a, \overline{a} \in n(l(t_i))$ ci garantisce infatti che la sostituzione $\{\eta/a\}$ è ininfluente sulle etichette delle transizioni, ovvero che $l(t'_1) = l(t_i)\{\eta/a\} = l(t_i)$, permettendoci anche di concludere che, per ogni $i \in [1 \cdot n]$, $t'_i \in T_{MCCS}$. □

Dal precedente corollario abbiamo

Corollario 4.1.25. *Sia* $t \in T_{MCCS}$ *è una generica transizione; siano* $m, m' \in \mathcal{M}_{fin}(S_{MCCS})$ *due marcature tali che* $m[t\rangle m'$*; sia* $\{\eta/a\}$ *una generica sostituzione di un nome visibile* $a \in \mathcal{L}$ *con un nome ristretto* $\eta \in \mathcal{N}$*; se* $a, \overline{a} \notin n(l(t))$*, esiste una transizione* $t_1 \in T_{MCCS}$*, pari a* $t_1 = ({}^\bullet t\{\eta/a\}, l(t), t^\bullet\{\eta/a\})$ *e con* $\gamma(t_1) = \gamma(t)$*, e vale* $m\{\eta/a\}[t_1\rangle m'\{\eta/a\}$*.*

Dimostrazione. Immediata conseguenza del corollario 4.1.24 (pagina a fronte), applicato al multiset di transizioni $u = 1 \cdot t$: otteniamo infatti un multiset $u_1 = 1 \cdot t_1$, con $t_1 = ({}^\bullet t\{\eta/a\}, l(t), t^\bullet\{\eta/a\})$ e $\gamma(t_1) = \gamma(t)$; $t_1 \in T_{MCCS}$ è, ovviamente, la transizione cercata. □

I risultati visti fin'ora sono relativi all'"aggiunta" di una sostituzione a una transizione; vediamo ora analoghi risultati relativi alla "sottrazione" di una sostituzione. Per prima, la proposizione corrispondente alla 4.1.20 (pag. 152).

Proposizione 4.1.26. *Sia* $t \in \longrightarrow$ *è una generica transizione; se esistono una marcatura* $m_1 \in \mathcal{M}_{fin}(S_{MCCS})$ *e una sostituzione* $\{\eta/a\}$ *– di un nome visibile* $a \in \mathcal{L}$ *con un nome ristretto* $\eta \in \mathcal{N}$ *– tali che* ${}^\bullet t = m_1\{\eta/a\}$*, allora esiste una transizione* $t_1 \in \longrightarrow$ *tale che* $t = ({}^\bullet t_1\{\eta/a\}, l(t_1)\{\eta/a\}, t_1^\bullet\{\eta/a\})$ *e che* $\gamma(t_1) = \gamma(t)$*.*

Dimostrazione. Nel corso della dimostrazione utilizzeremo frequentemente il risultato della proposizione 3.2.5 (pag. 83), ovvero che, in generale,

$$dec(q\{\eta/a\}, e, i) = dec(q, e, i)\{\eta/a\}$$

Dimostriamo la tesi per casistica su t e per induzione sulle relative premesse:

- se t è giustificata dalla regola (pref), allora esistono un'opportuna azione estesa $\mu \in \mathcal{A}ct'$, un opportuno processo esteso $q \in \mathcal{P}^{\mathcal{N}}$ e due opportuni numeri naturali $e, i \in \mathbb{N}$ tali che $m_1 = dec(\mu.q, e, i)$; possiamo distinguere tre casi:

  – quando $\mu = a$, abbiamo

  $$\begin{aligned} m_1\{\eta/a\} &= dec(a.q, e, i)\{\eta/a\} = dec((a.q)\{\eta/a\}, e, i) \\ &= dec(\eta.(q\{\eta/a\}), e, i) \end{aligned}$$

  Applicando la regola (pref) a $dec(\eta.(q\{\eta/a\}), e, i)$, possiamo concludere che la transizione t è pari a $t = (m_1\{\eta/a\}, \eta, dec(q\{\eta/a\}, e, i))$; d'altro canto possiamo applicare la regola (pref) anche a m_1, ovvero a $dec(a.q, e, i)$, ottenendo la transizione $t_1 = (m_1, a, dec(q, e, i))$; la transizione $t_1 \in T_{MCCS} \subseteq\longrightarrow$ è la transizione cercata; abbiamo infatti

  $$\begin{aligned} {}^\bullet t &= m_1\{\eta/a\} = {}^\bullet t_1\{\eta/a\} \\ l(t) &= \eta = a\{\eta/a\} = l(t_1)\{\eta/a\} \\ t^\bullet &= dec(q\{\eta/a\}, e, i) = dec(q, e, i)\{\eta/a\} = t_1^\bullet\{\eta/a\} \end{aligned}$$

  – quando $\mu = \overline{a}$, la dimostrazione è del tutto analoga a quella del sottocaso precedente;

  – quando $a \neq \mu \neq \overline{a}$, abbiamo

  $$m_1\{\eta/a\} = dec(\mu.q, e, i)\{\eta/a\} = dec((\mu.q)\{\eta/a\}, e, i)$$

$$= dec(\mu.(q\{\eta/a\}), e, i)$$

Applicando la regola (pref) a $dec(\mu.(q\{\eta/a\}), e, i)$, possiamo concludere che la transizione t è pari a $t = (m_1\{\eta/a\}, \mu, dec(q\{\eta/a\}, e, i))$; d'altro canto possiamo applicare la regola (pref) anche a m_1, ovvero a $dec(\mu.q, e, i)$, ottenendo la transizione $t_1 = (m_1, \mu, dec(q, e, i))$; la transizione $t_1 \in\longrightarrow$ è la transizione cercata; abbiamo infatti

$$\begin{aligned} {}^\bullet t &= m_1\{\eta/a\} = {}^\bullet t_1\{\eta/a\} \\ l(t) &= \mu = \mu\{\eta/a\} = l(t_1)\{\eta/a\} \\ t^\bullet &= dec(q\{\eta/a\}, e, i) = dec(q, e, i)\{\eta/a\} = t_1^\bullet\{\eta/a\} \end{aligned}$$

In tutti i tre sottocasi abbiamo anche che sia t che t_1 sono giustificate dalla regola (pref) e quindi, dalla definizione 3.2.13 (pag. 95), $\gamma(t) = 1 = \gamma(t_1)$;

- se t è giustificata dalla regola (s-pref), allora esistono un'opportuna azione estesa $\mu \in \mathcal{A}ct'$, un opportuno processo esteso $q \in \mathcal{P}^{\mathcal{N}}$ e due opportuni numeri naturali $e, i \in \mathbb{N}$ tali che $m_1 = dec(\underline{\mu}.q, e, i)$; anche in questo caso possiamo distinguere tre casi:
  - quando $\mu = a$, abbiamo

$$\begin{aligned} m_1\{\eta/a\} &= dec(\underline{a}.q, e, i)\{\eta/a\} = dec((\underline{a}.q)\{\eta/a\}, e, i) \\ &= dec(\underline{\eta}.(q\{\eta/a\}), e, i) \end{aligned}$$

Dal momento che t è giustificata dalla regola (s-pref), necessariamente esiste una transizione $t' \in\longrightarrow$ tale che ${}^\bullet t' \subseteq dec(q\{\eta/a\}, e, i) = dec(q, e, i)\{\eta/a\}$ e che $conc(\eta, l(t'), l(t))$; da ${}^\bullet t' \subseteq dec(q, e, i)\{\eta/a\}$ abbiamo che esiste una marcatura m'_1 tale che ${}^\bullet t' = m'_1\{\eta/a\}$ – e quindi $m'_1 \subseteq dec(q, e, i)$ – ovvero che possiamo applicare a t' l'induzione strutturale e concludere che esiste una transizione $t'_1 \in\longrightarrow$ tale che $t' = ({}^\bullet t'_1\{\eta/a\}, l(t'_1)\{\eta/a\}, t'^\bullet_1\{\eta/a\})$, ovvero ${}^\bullet t'_1 = m'_1$, e che $\gamma(t'_1) = \gamma(t')$; dal momento che ${}^\bullet t'_1 = m'_1 \subseteq dec(q, e, i)$, possiamo utilizzare la transizione t'_1 come premessa per la regola (s-pref) ottenendo la transizione $t_1 \in\longrightarrow$ tale che ${}^\bullet t_1 = dec(\underline{a}.q, e, i)$, che $conc(a, l(t'_1), l(t_1))$ e che $t_1^\bullet = t_1'^\bullet$; la transizione t_1 è la transizione cercata; osserviamo infatti che da $conc(\eta, l(t'), l(t))$, da $conc(a, l(t'_1), l(t_1))$, da $\eta = a\{\eta/a\}$ e da $l(t') = l(t'_1)\{\eta/a\}$ – ossia da $conc(a\{\eta/a\}, l(t'_1)\{\eta/a\}, l(t))$ – deriva $l(t) = l(t_1)\{\eta/a\}$; abbiamo anche

$$\begin{aligned} {}^\bullet t &= m_1\{\eta/a\} = dec(\underline{a}.q, e, i)\{\eta/a\} = {}^\bullet t_1\{\eta/a\} \\ t^\bullet &= t'^\bullet = t_1'^\bullet\{\eta/a\} = t_1^\bullet\{\eta/a\} \end{aligned}$$

- quando $\mu = \overline{a}$, la dimostrazione è del tutto analoga a quella del sottocaso precedente;
- quando $a \neq \mu \neq \overline{a}$, abbiamo

$$\begin{aligned} m_1\{\eta/a\} &= dec(\underline{\mu}.q, e, i)\{\eta/a\} = dec((\underline{\mu}.q)\{\eta/a\}, e, i) \\ &= dec(\underline{\mu}.(q\{\eta/a\}), e, i) \end{aligned}$$

Dal momento che t è giustificata dalla regola (s-pref), necessariamente esiste una transizione $t' \in\longrightarrow$ tale che ${}^\bullet t' \subseteq dec(q\{\eta/a\}, e, i) = dec(q, e, i)\{\eta/a\}$ e che $conc(\mu, l(t'), l(t))$; da ${}^\bullet t' \subseteq dec(q, e, i)\{\eta/a\}$ abbiamo che esiste una marcatura m'_1 tale che ${}^\bullet t' = m'_1\{\eta/a\}$ – e quindi $m'_1 \subseteq dec(q, e, i)$ – ovvero che possiamo applicare a t' l'induzione strutturale e concludere che esiste una transizione $t'_1 \in\longrightarrow$ tale che $t' = ({}^\bullet t'_1\{\eta/a\}, l(t'_1)\{\eta/a\}, t_1'^\bullet\{\eta/a\})$, ovvero ${}^\bullet t'_1 = m'_1$, e che $\gamma(t'_1) = \gamma(t')$; dal momento che ${}^\bullet t'_1 = m'_1 \subseteq dec(q, e, i)$, possiamo utilizzare la transizione t'_1 come premessa per la regola (s-pref) ottenendo la transizione $t_1 \in\longrightarrow$ tale che ${}^\bullet t_1 = dec(\underline{\mu}.q, e, i)$, che $conc(\mu, l(t'_1), l(t_1))$ e che $t_1^\bullet = t_1'^\bullet$; la transizione t_1 è la transizione cercata; osserviamo infatti che da $conc(\mu, l(t'), l(t))$, da $conc(\mu, l(t'_1), l(t_1))$, da $\mu = \mu\{\eta/a\}$ e da

$l(t') = l(t'_1)\{\eta/a\}$ – ossia da $conc(\mu\{\eta/a\}, l(t'_1)\{\eta/a\}, l(t))$ – deriva $l(t) = l(t_1)\{\eta/a\}$; abbiamo anche

$$\begin{aligned} {}^\bullet t &= m_1\{\eta/a\} = dec(\underline{\mu}.q, e, i)\{\eta/a\} = {}^\bullet t_1\{\eta/a\} \\ t^\bullet &= t'^\bullet = t_1'^\bullet\{\eta/a\} = t_1^\bullet\{\eta/a\} \end{aligned}$$

In tutti i tre sottocasi abbiamo che sia t che t_1 sono ottenute dalla regola (s-pref) usando come premessa transizioni t' e t'_1 tali che $\gamma(t') = \gamma(t'_1)$; ricordando la definizione 3.2.13 abbiamo quindi

$$\gamma(t) = \gamma(t') + 1 = \gamma(t'_1) + 1 = \gamma(t_1)$$

- se t è giustificata dalla regola (sum-1), allora esistono due opportuni processi sequenziali estesi $p_1, p_2 \in \mathcal{P}^{\mathcal{N}}_{seq}$ e due opportuni numeri naturali $e, i \in \mathbb{N}$ tali che $m_1 = dec(p_1 + p_2, e, i)$; abbiamo

$$\begin{aligned} m_1\{\eta/a\} &= dec(p_1 + p_2, e, i)\{\eta/a\} = dec((p_1 + p_2)\{\eta/a\}, e, i) \\ &= dec(p_1\{\eta/a\} + p_2\{\eta/a\}, e, i) \end{aligned}$$

  Dalla premessa della regola (sum-1) sappiamo che esiste una transizione $t' \in \longrightarrow$ tale che ${}^\bullet t' = dec(p_1\{\eta/a\}, e + 1, i) = dec(p_1, e, i)\{\eta/a\}$, che $l(t) = l(t')$ e che $t^\bullet = t'^\bullet$; dal momento che il preset di t' è pari a $dec(p_1, e + 1, i)\{\eta/a\}$, possiamo applicare l'induzione a t' e concludere che esiste una transizione $t'_1 \in \longrightarrow$ tale che $t' = ({}^\bullet t'_1\{\eta/a\}, l(t'_1)\{\eta/a\}, t_1'^\bullet\{\eta/a\})$, ovvero ${}^\bullet t'_1 = dec(p_1, e + 1, i)$, e che $\gamma(t'_1) = \gamma(t')$; utilizzando t'_1 come premessa della regola (sum-1) otteniamo l'esistenza di una transizione $t_1 \in \longrightarrow$ tale che ${}^\bullet t_1 = dec(p_1 + p_2, e + 1, i)$, che $l(t_1) = l(t'_1)$ e che $t_1^\bullet = t_1'^\bullet$; la transizione t_1 è la transizione cercata, infatti

$$\begin{aligned} {}^\bullet t &= m_1\{\eta/a\} = dec(p_1 + p_2, e, i)\{\eta/a\} = {}^\bullet t_1\{\eta/a\} \\ l(t) &= l(t') = l(t'_1)\{\eta/a\} = l(t_1)\{\eta/a\} \\ t^\bullet &= t'^\bullet = t_1'^\bullet\{\eta/a\} = t_1^\bullet\{\eta/a\} \end{aligned}$$

Inoltre abbiamo che sia t che t_1 sono ottenute dalla regola (sum-1), usando come premessa transizioni t' e t'_1 tali che $\gamma(t') = \gamma(t'_1)$; ricordando la definizione 3.2.13 abbiamo quindi

$$\gamma(t) = \gamma(t') + 1 = \gamma(t'_1) + 1 = \gamma(t_1)$$

- se t è giustificata dalla regola (sum-2), la dimostrazione è del tutto analoga a quella del caso precedente ma operando per induzione strutturale sulla seconda componente additiva anziché sulla prima;

- se t è giustificata dalla regola (com), allora esistono due transizioni $t_a, t_b \in \longrightarrow$ che si sincronizzano in t, ovvero $sync(l(t_a), l(t_b), l(t))$; dal momento che $m_1\{\eta/a\} = {}^\bullet t = {}^\bullet t_a \oplus {}^\bullet t_b$, esistono due marcature $m_{a1}, m_{b1} \in \mathcal{M}_{fin}(S_{MCCS})$ tali che $m_1 = m_{a1} \oplus m_{b1}$, che ${}^\bullet t_a = m_{a1}\{\eta/a\}$ e che ${}^\bullet t_b = m_{b1}\{\eta/a\}$; applicando l'induzione strutturale sia a t_a che a t_b otteniamo che esistono due transizioni $t_{a1}, t_{b1} \in \longrightarrow$ tali che $t_a = ({}^\bullet t_{a1}\{\eta/a\}, l(t_{a1})\{\eta/a\}, t_{a1}^\bullet\{\eta/a\})$, che $t_b = ({}^\bullet t_{b1}\{\eta/a\}, l(t_{b1})\{\eta/a\}, t_{b1}^\bullet\{\eta/a\})$, che $\gamma(t_{a1}) = \gamma(t_a)$ e che $\gamma(t_{b1}) = \gamma(t_b)$; la transizione $t_1 \in \longrightarrow$, ottenuta sincronizzando t_{a1} e t_{b1} nello stesso modo (salvo sostituzione $\{\eta/a\}$) di t_a e di t_b, è la transizione cercata; per quanto riguarda l'etichetta abbiamo $sync(l(t_a), l(t_b), l(t)) = sync(l(t_{a1})\{\eta/a\}, l(t_{b1})\{\eta/a\}, l(t))$ e, ricordando che $sync(l(t_{a1}), l(t_{b1}), l(t_1))$, otteniamo $l(t) = l(t_1)\{\eta/a\}$; abbiamo anche

$$\begin{aligned} {}^\bullet t &= {}^\bullet t_a \oplus {}^\bullet t_b = m_{a1}\{\eta/a\} \oplus m_{b1}\{\eta/a\} = {}^\bullet t_{a1}\{\eta/a\} \oplus {}^\bullet t_{b1}\{\eta/a\} \\ &= ({}^\bullet t_{a1} \oplus {}^\bullet t_{b1})\{\eta/a\} = {}^\bullet t_1\{\eta/a\} \\ t^\bullet &= t_a^\bullet \oplus t_b^\bullet = t_{a1}^\bullet\{\eta/a\} \oplus t_{b1}^\bullet\{\eta/a\} = (t_{a1}^\bullet \oplus t_{b1}^\bullet)\{\eta/a\} = t_1^\bullet\{\eta/a\} \end{aligned}$$

Dal momento che sia t che t_1 sono ottenute dalla regola (com) sincronizzando transizioni t_a e t_b (la prima), t_{a1} e t_{b1} (la seconda) tali che $\gamma(t_a) = \gamma(t_{a1})$ e $\gamma(t_b) = \gamma(t_{b1})$, e ricordando anche la definizione 3.2.13 abbiamo

$$\gamma(t) = \gamma(t_a) + \gamma(t_b) + 1 = \gamma(t_{a1}) + \gamma(t_{b1}) + 1 = \gamma(t_1)$$

□

Come nel caso dell'analogo corollario 4.1.21 (pag. 158), la proposizione appena vista può essere estesa alle marcature che supportano multiset di transizioni, ovvero

Corollario 4.1.27. *Se* $u \in \mathcal{M}_{fin}(\longrightarrow)$ *è un multiset di transizioni – pari a* $u = k_1t_1 \oplus k_2t_2 \oplus \ldots \oplus k_nt_n$*, per opportuni numeri interi* $k_1, k_2, \ldots, k_n \in \mathbb{N}$ *e opportune transizioni* $t_1, t_2, \ldots, t_n \in\longrightarrow$ *– e* $m_1, m' \in \mathcal{M}_{fin}(S_{MCCS})$ *sono due marcature e* $\{\eta/a\}$ *è la sostituzione di un nome visibile* $a \in \mathcal{L}$ *con un nome ristretto* $\eta \in \mathcal{N}$ *tali che* $m_1\{\eta/a\}[[u\rangle\rangle m'$*, allora esistono un multiset di transizioni* $u' \in \mathcal{M}_{fin}(\longrightarrow)$*, pari a* $u' = k_1t'_1 \oplus k_2t'_2 \oplus \ldots \oplus k_nt'_n$ *– sui medesimi numeri interi* $k_1, k_2, \ldots, k_n \in \mathbb{N}$ *e con transizioni* $t'_1, t'_2, \ldots, t'_n \in\longrightarrow$ *tali, per* $i \in [1 \ldots n]$*, che* $t_i = ({}^\bullet t'_i\{\eta/a\}, l(t'_i)\{\eta/a\}, t'^\bullet_i\{\eta/a\})$ *e che* $\gamma(t'_i) = \gamma(t_i)$ *– e una marcatura* $m'_1 \in \mathcal{M}_{fin}(S_{MCCS})$ *tali che* $m_1[[u'\rangle\rangle m'_1$ *e che* $m' = m'_1\{\eta/a\}$*.*

Dimostrazione. Se $m_1\{\eta/a\}[[u\rangle\rangle m'$, con $u = k_1t_1 \oplus k_2t_2 \oplus \ldots \oplus k_nt_n$, ricordando anche la definizione 1.2.5 (pag. 14) abbiamo che la marcatura m_1 può essere scissa in $n+1$ sub-marcature, le prime n delle quali sono pari, dopo l'applicazione della sostituzione, al preset di $k_i \cdot t_i$[4], al variare di $i \in [1 \ldots n]$, e l'ultima viene lasciata inalterata dall'applicazione delle transizioni; più formalmente abbiamo che

$$m_1 = \left(\bigoplus_{i=1}^{n} k_i \cdot m_{ai} \right) \oplus m_b$$

per opportune marcature $m_{a1}, m_{a2}, \ldots, m_{an}$ e m_b, l'ultima delle quali potenzialmente anche nulla – con ${}^\bullet t_i = m_{ai}\{\eta/a\}$, al variare di $i \in [1 \ldots n]$. Tenendo conto che ${}^\bullet t_i = m_{ai}\{\eta/a\}$, per ogni $i \in [1 \ldots n]$ possiamo applicare a t_i la proposizione 4.1.26 (pag. 161) ottenendo che esiste una transizione $t'_i \in\longrightarrow$ tale che $t_i = ({}^\bullet t'_i\{\eta/a\}, l(t'_i)\{\eta/a\}, t'^\bullet_i\{\eta/a\})$ e che $\gamma(t'_i) = \gamma(t_i)$; in particolare abbiamo, sempre per ogni $i \in [1 \ldots n]$, che ${}^\bullet t'_i = m_{ai}$; osserviamo che, applicando le transizioni $t'_1, t'_2, \ldots, t'_n$ a m_1, otteniamo una marcatura

[4] o, più esattamente, sono pari a k_i volte il preset di t_i

m'_1 pari a

$$\begin{aligned}
m'_1 &= \left(m_1 \setminus \left(\bigoplus_{i=1}^{n} k_i \cdot m_{ai}\right)\right) \oplus \left(\bigoplus_{i=1}^{n} k_i \cdot t_i'^{\bullet}\right) \\
&= \left(m_1 \setminus \left(\bigoplus_{i=1}^{n} k_i \cdot {}^{\bullet}t'_i\right)\right) \oplus \left(\bigoplus_{i=1}^{n} k_i \cdot t_i'^{\bullet}\right) \\
&= \left(\left(\left(\bigoplus_{i=1}^{n} k_i \cdot {}^{\bullet}t'_i\right) \oplus m_b\right) \setminus \left(\bigoplus_{i=1}^{n} k_i \cdot {}^{\bullet}t'_i\right)\right) \oplus \left(\bigoplus_{i=1}^{n} k_i \cdot t_i'^{\bullet}\right) \\
&= \left(\bigoplus_{i=1}^{n} k_i \cdot t_i'^{\bullet}\right) \oplus m_b
\end{aligned}$$

Il multiset $u' = k_1 t'_1 \oplus k_2 t'_2 \oplus \ldots \oplus k_n t'_n$ e la marcatura m'_1 sono il multiset di transizioni e la marcatura cercati; rimane solo da provare che $m' = m'_1\{\eta/a\}$; abbiamo infatti che

$$\begin{aligned}
m' &= \left(m_1\{\eta/a\} \setminus \left(\bigoplus_{i=1}^{n} k_i {}^{\bullet}t_i\right)\right) \oplus \left(\bigoplus_{i=1}^{n} k_i t_i^{\bullet}\right) \\
&= \left(\left(\left(\bigoplus_{i=1}^{n} k_i m_{ai}\right) \oplus m_b\right)\{\eta/a\} \setminus \left(\bigoplus_{i=1}^{n} k_i {}^{\bullet}t_i\right)\right) \oplus \left(\bigoplus_{i=1}^{n} k_i t_i^{\bullet}\right) \\
&= \left(\left(\left(\bigoplus_{i=1}^{n} k_i m_{ai}\right)\{\eta/a\} \oplus m_b\{\eta/a\}\right) \setminus \left(\bigoplus_{i=1}^{n} k_i {}^{\bullet}t_i\right)\right) \oplus \left(\bigoplus_{i=1}^{n} k_i t_i^{\bullet}\right) \\
&= \left(\left(\left(\bigoplus_{i=1}^{n} (k_i m_{ai})\{\eta/a\}\right) \oplus m_b\{\eta/a\}\right) \setminus \left(\bigoplus_{i=1}^{n} k_i {}^{\bullet}t_i\right)\right) \oplus \left(\bigoplus_{i=1}^{n} k_i t_i^{\bullet}\right) \\
&= \left(\left(\left(\bigoplus_{i=1}^{n} k_i (m_{ai}\{\eta/a\})\right) \oplus m_b\{\eta/a\}\right) \setminus \left(\bigoplus_{i=1}^{n} k_i {}^{\bullet}t_i\right)\right) \oplus \left(\bigoplus_{i=1}^{n} k_i t_i^{\bullet}\right) \\
&= \left(\left(\left(\bigoplus_{i=1}^{n} k_i {}^{\bullet}t_i\right) \oplus m_b\{\eta/a\}\right) \setminus \left(\bigoplus_{i=1}^{n} k_i {}^{\bullet}t_i\right)\right) \oplus \left(\bigoplus_{i=1}^{n} k_i t_i^{\bullet}\right) \\
&= m_b\{\eta/a\} \oplus \left(\bigoplus_{i=1}^{n} k_i t_i^{\bullet}\right) = m_b\{\eta/a\} \oplus \left(\bigoplus_{i=1}^{n} k_i (t_i'^{\bullet}\{\eta/a\})\right) \\
&= m_b\{\eta/a\} \oplus \left(\bigoplus_{i=1}^{n} (k_i t_i'^{\bullet})\{\eta/a\}\right) = m_b\{\eta/a\} \oplus \left(\bigoplus_{i=1}^{n} k_i t_i'^{\bullet}\right)\{\eta/a\} \\
&= \left(m_b \oplus \left(\bigoplus_{i=1}^{n} k_i t_i'^{\bullet}\right)\right)\{\eta/a\} = m'_1\{\eta/a\}
\end{aligned}$$

□

Anche in questo caso, dato il corollario precedente, è immediato il seguente

Corollario 4.1.28. *Se $t \in \longrightarrow$ è una transizione, $m_1, m' \in \mathcal{M}_{fin}(S_{MCCS})$ due marcature e $\{\eta/a\}$ la sostituzione di un nome visibile $a \in \mathcal{L}$ con un nome ristretto $\eta \in \mathcal{N}$ tali che $m_1\{\eta/a\}[t\rangle m'$, allora esistono una transizione $t_1 \in \longrightarrow$ e una marcatura $m'_1 \in \mathcal{M}_{fin}(S_{MCCS})$ tali che $t = ({}^\bullet t_1\{\eta/a\}, l(t_1)\{\eta/a\}, t_1^\bullet\{\eta/a\})$, che $m_1[t_1\rangle m'_1$, che $m' = m'_1\{\eta/a\}$ e che $\gamma(t_1) = \gamma(t)$.*

Dimostrazione. Immediata conseguenza del corollario 4.1.27 (pag. 167), applicato al multiset di transizioni $u = 1 \cdot t$: otteniamo infatti un multiset $u_1 = 1 \cdot t_1$ e una marcatura m'_1, con $t = ({}^\bullet t_1\{\eta/a\}, l(t_1)\{\eta/a\}, t_1^\bullet\{\eta/a\})$, con $\gamma(t_1) = \gamma(t)$, con $m_1[[u_1\rangle\rangle m'_1$ e con $m' = m'_1\{\eta/a\}$; ricordando le osservazioni 1.2.3 (pag. 15) abbiamo anche $m_1[t_1\rangle m'_1$ e, quindi, t_1 e m'_1 sono la transizione e la marcatura cercate. □

Anche in questo caso possiamo ridurci alle transizioni che veramente ci interessano – quelle in T_{MCCS}, ovvero prive di azioni ristrette nell'etichetta – ottenendo, per di più, l'identità dell'etichetta. In questo caso, a causa del cambiamento della prospettiva, quella che prima era una condizione aggiuntiva da imporre all'ipotesi – l'assenza delle azioni sostituite nell'etichetta della transizione – in questo caso diventa un risultato secondario.

Corollario 4.1.29. *Sia $t \in T_{MCCS}$ è una generica transizione; se esistono una marcatura $m_1 \in \mathcal{M}_{fin}(S_{MCCS})$ e una sostituzione $\{\eta/a\}$ – di un nome visibile $a \in \mathcal{L}$ con un nome ristretto $\eta \in \mathcal{N}$ – tali che ${}^\bullet t = m_1\{\eta/a\}$, allora esiste una transizione $t_1 \in T_{MCCS}$ – con la medesima etichetta di t – tale che*

(1) $t = ({}^\bullet t_1\{\eta/a\}, l(t_1), t_1^\bullet\{\eta/a\})$;

(2) $\gamma(t_1) = \gamma(t)$;

(3) $a, \overline{a} \notin n(l(t_1)) = n(l(t))$.

Dimostrazione. È evidente che

$$(t \in T_{MCCS} \wedge T_{MCCS} \subseteq \longrightarrow) \implies t \in \longrightarrow$$

e quindi che possiamo applicare la proposizione 4.1.26 (pag. 161) a t e ottenere che esiste una transizione $t_1 \in \longrightarrow$ tale che $t = ({}^\bullet t_1\{\eta/a\}, l(t_1)\{\eta/a\}, t_1^\bullet\{\eta/a\})$ e che $\gamma(t_1) = \gamma(t)$.

Osserviamo che la sostituzione $\{\eta/a\}$, applicata all'etichetta di t_1, non elimina alcuna azione ristretta ma, anzi, ne aggiungerebbe se fossero presenti in $l(t_1)$ le azioni a e $\overline{a}$; dal momento che sappiamo, dall'ipotesi $t \in T_{MCCS}$, che $l(t)$ non può contenere azioni ristrette, da $l(t) = l(t_1)\{\eta/a\}$ abbiamo sia che $l(t_1)$ non può contenere né azioni ristrette (che altrimenti sarebbero presenti anche in $l(t)$) né le azioni a e/o $\overline{a}$ (che diventerebbero, in $l(t)$, le azioni ristrette η e/o $\overline{\eta}$).

Dal fatto che $l(t_1)$ non può contenere azioni ristrette abbiamo che $t_1 \in T_{MCCS}$; dal fatto che $a, \overline{a} \notin n(l(t_1))$ otteniamo che la sostituzione $\{\eta/a\}$ è ininfluente in $l(t_1)$ ovvero che $l(t) = l(t_1)\{\eta/a\} = l(t_1)$; abbiamo quindi ottenuto che la transizione t è la transizione cercata e, combinando $a, \overline{a} \notin n(l(t_1))$ e $l(t) = l(t_1)$, possiamo anche concludere che $a, \overline{a} \notin n(l(t_1)) = n(l(t))$. □

Sul supporto contemporaneo di transizioni abbiamo

Corollario 4.1.30. *Sia* $u \in \mathcal{M}_{fin}(T_{MCCS})$ *è un generico multiset di transizioni* T_{MCCS} *pari a* $u = k_1t_1 \oplus k_2t_2 \oplus \ldots \oplus k_nt_n$*, per opportuni numeri interi* $k_1, k_2, \ldots, k_n \in \mathbb{N}$ *e opportune transizioni* $t_1, t_2, \ldots, t_n \in T_{MCCS}$*; siano* $m_1, m' \in \mathcal{M}_{fin}(S_{MCCS})$ *due marcature e* $\{\eta/a\}$ *una generica sostituzione di un nome visibile* $a \in \mathcal{L}$ *con un nome ristretto* $\eta \in \mathcal{N}$ *tali che* $m_1\{\eta/a\}[[u\rangle\rangle m'$*; allora esistono un multiset di transizioni* $u' \in \mathcal{M}_{fin}(T_{MCCS})$*, pari a* $u' = k_1t'_1 \oplus k_2t'_2 \oplus \ldots \oplus k_nt'_n$ *– sui medesimi numeri interi* $k_1, k_2, \ldots, k_n \in \mathbb{N}$ *e con transizioni* $t'_1, t'_2, \ldots, t'_n \in T_{MCCS}$ *con le medesime etichette delle corrispondenti* $t_1, t_2, \ldots, t_n$ *– e una marcatura* $m'_1 \in \mathcal{M}_{fin}(S_{MCCS})$ *tali che*

(1) $\forall i \in [1 \dots n].t_i = ({}^\bullet t'_i\{\eta/a\}, l(t'_i), t'^\bullet_i\{\eta/a\})$;

(2) $\forall i \in [1 \dots n].a, \overline{a} \notin n(l(t'_i)) = n(l(t_i))$;

(3) $\forall i \in [1 \dots n].\gamma(t'_i) = \gamma(t_i)$;

(4) $m_1[[u'\rangle\rangle m'_1$;

(5) $m' = m'_1\{\eta/a\}$;

Dimostrazione. Immediata conseguenza del corollario 4.1.27 (pag. 167) considerando che $u \in \mathcal{M}_{fin}(T_{MCCS}) \subseteq \mathcal{M}_{fin}(\longrightarrow)$; dal momento che abbiamo $l(t_i) = l(t'_i)\{\eta/a\}$, otteniamo anche che $a, \overline{a} \notin n(l(t'_i))$ perché, se così non fosse, avremmo nell'etichetta di $l(t_i)$ dei nomi ristretti e questo sarebbe in contrasto con $t_i \in T_{MCCS}$; da $a, \overline{a} \notin n(l(t'_i))$ otteniamo anche che la sostituzione $\{\eta/a\}$ sull'etichetta di t_i è ininfluente e quindi, per concludere, che $l(t_i) = l(t'_i)\{\eta/a\} = l(t'_i)$ e che $t'_i \in T_{MCCS}$. □

Dal precedente corollario abbiamo

Corollario 4.1.31. *Sia $t \in T_{MCCS}$ è una generica transizione; siano $m_1, m' \in \mathcal{M}_{fin}(S_{MCCS})$ due marcature e $\{\eta/a\}$ una generica sostituzione di un nome visibile $a \in \mathcal{L}$ con un nome ristretto $\eta \in \mathcal{N}$ tali che $m_1\{\eta/a\}[t\rangle m'$; allora esistono una transizione $t_1 \in T_{MCCS}$ – con la medesima etichetta di t – e una marcatura $m'_1 \in \mathcal{M}_{fin}(S_{MCCS})$ tali che*

(1) $t = ({}^\bullet t_1\{\eta/a\}, l(t_1), t_1^\bullet\{\eta/a\})$;

(2) $a, \overline{a} \notin n(l(t_1)) = n(l(t))$.

(3) $\gamma(t_1) = \gamma(t)$;

(4) $m_1[t_1\rangle m'_1$;

(5) $m' = m'_1\{\eta/a\}$;

Dimostrazione. Immediata conseguenza del corollario 4.1.30 (pag. 170) applicato al multiset di transizioni $u = 1 \cdot t$; otteniamo infatti un multiset $u_1 = 1 \cdot t_1$ – con $t = ({}^\bullet t_1\{\eta/a\}, l(t_1), t_1^\bullet\{\eta/a\})$, con $a, \overline{a} \notin n(l(t_1)) = n(l(t))$ e con $\gamma(t_1) = \gamma(t)$ – e una marcatura m'_1 per i quali vale $m_1[[u_1\rangle\rangle m'_1$ (ossia, ricordando le osservazioni 1.2.3 (pag. 15), $m_1[t_1\rangle m'_1$) e $m' = m'_1\{\eta/a\}$; $t_1 \in T_{MCCS}$ e m'_1 sono la transizione e la marcatura cercati. □

4.2 Dec-equivalenza e relativi risultati

Dimostrare che, per un generico processo Multi-CCS q, $dec(q, 0, 0)$ è bisimile allo stesso q visto come stato della semantica LTS (e lo indicheremo sinteticamente con $dec(q, 0, 0) \sim q$, considerando quindi quella LTS come semantica di default per il Multi-CCS) basandosi direttamente sulla bisimulazione, come si prova a fare in [7], non è per nulla agevole.

Un problema che si deve affrontare è che la congruenza definita in 2.2.7 (pag. 35) non si mantiene con la composizione parallela; infatti – come segnalatomi da Gorrieri – non è generalmente vero che $q_1 \sim q_2 \implies q_1 \mid q_3 \sim q_2 \mid q_3$: si prenda, ad esempio,

$$q_1 = a.a.\mathbf{0}$$
$$q_2 = a.\mathbf{0} \mid a.\mathbf{0}$$
$$q_3 = \underline{\overline{a}}.\underline{\overline{a}}.c.\mathbf{0}$$

e si osservi che è evidente che $q_1 \sim q_2$ ma che $q_2 \mid q_3$ può effettuare una transizione con etichetta c (grazie al fatto che q_3 può effettuare una doppia sincronizzazione con le due componenti di q_2) che non può essere replicata da $q_1 \mid q_3$ (dal momento che in q_1 è immediatamente disponibile una sola a per sincronizzarsi con q_3). Il problema indicato, ovviamente, vale anche per le marcature in $\mathcal{M}_{fin}(S_{MCCS})$: non è generalmente vero che $m_1 \sim m_2 \implies m_1 \mid m_3 \sim m_2 \mid m_3$; per un esempio è sufficiente applicare dec ai q_1, q_2 e q_3 di cui sopra.

Per aggirare questo problema, definisco di seguito una nuova relazione di equivalenza in $\mathcal{M}_{fin}(S_{MCCS}) \times \mathcal{M}_{fin}(S_{MCCS})$ che è più "forte" della bisimilitudine (ovvero: che implica la bisimilitudine) e che conserva la congruenza con la composizione parallela; sostituisco quindi alcuni risultati di [7] (scritto quando ancora non era noto il problema descritto sopra) relativi alla bisimilitudine tra marcature con risultati analoghi relativi alla nuova relazione di equivalenza. La bisimilitudine tra $dec(q, 0, 0)$ e q sarà quindi conseguenza di un risultato più "forte".

4.2.1 Definizione

Definiamo la dec-equivalenza come segue.

Definizione 4.2.1. *Siano $m_a, m_b \in \mathcal{M}_{fin}(S_{MCCS})$ due marcature; diciamo che m_a e m_b sono* sequenzialmente dec-equivalenti *quando esistono due processi sequenziali $p_a, p_b \in \mathcal{P}_{seq}$, per i quali vale $p_a \equiv p_b$, e quattro numeri naturali $e, f, i, j \in \mathbb{N}$ tali che*

$$m_a = dec(p_a, e, i) \qquad\qquad m_b = dec(p_b, f, j)$$

Siano $m_a, m_b \in \mathcal{M}_{fin}(S_{MCCS})$ due marcature; diciamo che m_a e m_b sono dec-equivalenti *– e lo indichiamo sinteticamente con $m_a \simeq_e m_b$ – quando*

- *m_a ed m_b sono sequenzialmente dec-equivalenti;*
- *oppure quando esistono due marcature $m'_a, m'_b \in \mathcal{M}_{fin}(S_{MCCS})$, tali che $m'_a \simeq_e m'_b$, e due sostituzioni $\{\eta_a/c\}$ ed $\{\eta_b/c\}$ – dello stesso nome $c \in \mathcal{L}$, con due nomi ristretti (coincidenti o anche non coincidenti) $\eta_a, \eta_b \in \mathcal{N}$, compatibili con m'_a la prima e con m'_b la seconda – tali che*

$$m_a = m'_a\{\eta_a/c\} \qquad\qquad m_b = m'_b\{\eta_b/c\}$$

- *oppure quando esistono quattro marcature a due a due dec-equivalenti $m_{a1}, m_{a2}, m_{b1}, m_{b2} \in \mathcal{M}_{fin}(S_{MCCS})$, ovvero tali che $m_{a1} \simeq_e m_{b1}$ e*

$m_{a2} \simeq_e m_{b2}$, *per le quali valgono le condizioni di non collisione dei nomi ristretti*

$$r_a(m_{a1}) \cap \overline{r_a(m_{a2})} = \emptyset \qquad r_a(m_{b1}) \cap \overline{r_a(m_{b2})} = \emptyset$$

e tali che

$$m_a = m_{a1} \oplus m_{a2} \qquad m_b = m_{b1} \oplus m_{b2}$$

In altre parole, due marcature sono dec-corrispondenti quando sono ottenibili – variando i parametri di supporto – da due processi sequenziali tra loro congruenti, o "sommando" due coppie di processi dec-equivalenti (a patto che non collidano i rispettivi nomi ristretti, ovvero a patto che la somma non possa "inventare" sincronizzazioni diseguali nelle due coppie) o aggiungendo restrizioni compatibili, sullo stesso nome, a una coppia di marcature dec-corrispondenti.

Queste definizioni sono alquanto elaborate e, probabilmente, prive di interesse autonomo. Però, come accennavo poc'anzi e come vedremo nel dettaglio in seguito, sono utili ai fini delle dimostrazioni dei risultati che consentiranno di concludere che, in generale, $dec(q, 0, 0) \sim q$.

Vista la definizione, è intuibile che la dec-equivalenza è una relazione di equivalenza; la seguente proposizione lo afferma e lo prova esplicitamente.

Proposizione 4.2.1. *La dec-equivalenza è una relazione di equivalenza.*

Dimostrazione. Una relazione di equivalenza è tale, per definizione, se è simmetrica, riflessiva e transitiva.

La simmetria a la riflessività sono ovvie, vista la definizione.

L'unica proprietà non immediatamente evidente è la transitività; supponiamo quindi che $m_a \simeq_e m_b$ e che $m_b \simeq_e m_c$, per generiche transizioni $m_a, m_b, m_c \in \mathcal{M}_{fin}(S_{MCCS})$, e dimostriamo, per induzione strutturale sulla definizione di dec-equivalenza, che è anche vero che $m_a \simeq_e m_c$:

- quando m_a e m_b sono sequenzialmente dec-equivalenti, allora $m_a = dec(p_a, e, i)$ e $m_b = dec(p_b, f, j)$, per opportuni processi sequenziali $p_a, p_b \in \mathcal{P}_{seq}$, tali che $p_a \equiv p_b$, e per opportuni numeri interi

$e, f, i, j \in \mathbb{N}$; poiché abbiamo anche $m_b \simeq_e m_c$, dal momento che m_b è nella forma indicata, abbiamo necessariamente che $m_c = dec(p_c, g, k)$, per un opportuno processo sequenziale $p_c \in \mathcal{P}_{seq}$, tale che $p_b \equiv p_c$, e per opportuni numeri interi $g, k \in \mathbb{N}$; dalla transitività della congruenza, abbiamo anche $p_a \equiv p_c$ e quindi la dec-equivalenza tra $m_a = dec(p_a, e, i)$ e $m_c = dec(p_c, g, k)$;

- quando m_a e m_b sono dec-equivalenti in quanto pari a $m_a = m'_a\{\eta_a/d\}$ e a $m_b = m'_b\{\eta_b/d\}$ – per opportune sostituzioni compatibili $\{\eta_a/d\}$ (con m'_a) ed $\{\eta_b/d\}$ (con m'_b), entrambe sul nome d, e con $m'_a \simeq_e m'_b$ – dal momento che $m_b \simeq_e m_c$, anche m_c è nella forma $m_c = m'_c\{\eta_c/d\}$, dove $\{\eta_c/d\}$ è una sostituzione compatibile con m'_c, con $m'_b \simeq_e m'_c$; applicando l'induzione strutturale a $m'_a \simeq_e m'_b$ e a $m'_b \simeq_e m'_c$ otteniamo che $m'_a \simeq_e m'_c$ e quindi, essendo m_a ed m_c ottenute da queste aggiungendo sostituzioni compatibili sullo stesso nome, anche $m_a \simeq_e m_c$;

- quando m_a e m_b sono dec-equivalenti in quanto pari a $m_a = m_{a1} \oplus m_{a2}$ e a $m_b = m_{b1} \oplus m_{b2}$ – con $m_{a1} \simeq_e m_{b1}$, con $m_{a2} \simeq_e m_{b2}$ e con in più la validità delle condizioni di non collisione dei nomi ristretti – dal momento che $m_b \simeq_e m_c$, anche m_c è nella forma $m_c = m_{c1} \oplus m_{c2}$, con con $m_{b1} \simeq_e m_{c1}$, con $m_{b2} \simeq_e m_{c2}$ e con la non collisione dei nomi ristretti anche tra m_{c1} e m_{c2}; applicando l'induzione strutturale a $m_{a1} \simeq_e m_{b1}$ e a $m_{b1} \simeq_e m_{c1}$ otteniamo che $m_{a1} \simeq_e m_{c1}$; analogamente, applicando l'induzione strutturale a $m_{a2} \simeq_e m_{b2}$ e a $m_{b2} \simeq_e m_{c2}$ otteniamo che $m_{a2} \simeq_e m_{c2}$; tenendo conto che le condizioni di non collisione dei nomi ristretti continuano a valere sulle sub-componenti di m_a e di m_b, dalla definizione di dec-equivalenza otteniamo che $m_a \simeq_e m_c$.

$\square$

4.2.2 Dec-equivalenza di un singolo processo

La definizione di dec-equivalenza è basata sulla dec-equivalenza sequenziale che, al netto della congruenza, equipara le marcature ottenute dal medesimo processo sequenziale al variare dei parametri di supporto. Ci si può domandare se la dec-equivalenza vale, al variare dei parametri di supporto, per tutti i processi. La risposta è positiva e abbiamo il seguente risultato.

Proposizione 4.2.2. *Sia $q \in \mathcal{P}$ un generico processo Multi-CCS e siano $e, f, i, j \in \mathbb{N}$ generici numeri interi; vale $dec(q, e, i) \simeq_e dec(q, f, j)$.*

Dimostrazione. Si dimostra per induzione strutturale su q:

- se q è sequenziale, abbiamo ovviamente $q \equiv q$ e quindi, dalle definizioni, $dec(q, e, i)$ e $dec(q, f, j)$ sono dec-equivalenti in quanto sequenzialmente dec-equivalenti;

- se $q = q_1 \mid q_2$, per opportuni processi $q_1, q_2 \in \mathcal{P}$, per induzione strutturale abbiamo (in particolare) che

$$dec(q_1, e+1, i) \simeq_e dec(q_1, f+1, j)$$
$$dec(q_2, e+1, i+2^e) \simeq_e dec(q_2, f+1, j+2^f)$$

  Dal corollario 4.1.15 (pag. 147), applicato sia a $dec(q_1 \mid q_2, e, i)$ che a $dec(q_1 \mid q_2, f, j)$, otteniamo che

$$r_a(dec(q_1, e+1, i)) \cap \overline{r_a(dec(q_2, e+1, i+2^e))} = \emptyset$$
$$r_a(dec(q_1, f+1, j)) \cap \overline{r_a(dec(q_2, f+1, j+2^f))} = \emptyset$$

  e, dal momento che le condizioni di non collisione dei nomi ristretti sono soddisfatte, dalla definizione di dec-equivalenza abbiamo

$$\begin{aligned} dec(q, e, i) = dec(q_1 \mid q_2, e, i) &= dec(q_1, e+1, i) \oplus dec(q_2, e+1, i+2^e) \\ &\simeq_e dec(q_1, f+1, j) \oplus dec(q_2, f+1, j+2^f) \\ &= dec(q_1 \mid q_2, f, j) = dec(q, f, j) \end{aligned}$$

- se $q = (\nu a)q'$, per un opportuno nome $a \in \mathcal{L}$ e un opportuno processo $q' \in \mathcal{P}$, possiamo distinguere i seguenti casi:

– quando $a, \overline{a} \in fn(q')$, per induzione strutturale abbiamo (in particolare) che

$$dec(q', e, i + 2^e) \simeq_e dec(q, f, j + 2^f)$$

Tenendo conto del corollario 4.1.18 (pag. 150), che ci garantisce la compatibilità delle sostituzioni utilizzate, e della proposizione 3.2.5 (pag. 83), dalla definizione di dec-equivalenza abbiamo

$$\begin{aligned} dec(q, e, i) &= dec((\nu a)q', e, i) = dec(q'\{\eta_i/a\}, e, i + 2^e) \\ &= dec(q', e, i + 2^e)\{\eta_i/a\} \\ &\simeq_e dec(q', f, j + 2^f)\{\eta_j/a\} \\ &= dec(q'\{\eta_j/a\}, f, j + 2^f) \\ &= dec((\nu a)q', f, j) = dec(q, f, j) \end{aligned}$$

– quando $a \in fn(q') \wedge \overline{a} \notin fn(q')$, per induzione strutturale abbiamo (in particolare) che

$$dec(q', e, i) \simeq_e dec(q, f, j)$$

Tenendo conto della proposizione 3.2.5 e del fatto che – dal momento che $\overline{a} \notin fn(q')$ e quindi, dalla proposizione 4.1.9 (pag. 130), anche $\overline{a} \notin n(dec(q', e, i))$ e $\overline{a} \notin n(dec(q', f, j))$ – $\{\eta_d/a\}$ è una sostituzione compatibile sia con $dec(q', e, i)$ che con $dec(q', f, j)$, dalla definizione di dec-equivalenza abbiamo

$$\begin{aligned} dec(q, e, i) &= dec((\nu a)q', e, i) = dec(q'\{\eta_d/a\}, e, i) \\ &= dec(q', e, i)\{\eta_d/a\} \simeq_e dec(q', f, j)\{\eta_d/a\} \\ &= dec(q'\{\eta_d/a\}, f, j) = dec((\nu a)q', f, j) = dec(q, f, j) \end{aligned}$$

– quando $a \notin fn(q') \wedge \overline{a} \in fn(q')$, la dimostrazione è analoga a quella del sottocaso precedente ma usando la sostituzione $\{\eta_c/a\}$ al posto di $\{\eta_d/a\}$;

– quando $a, \overline{a} \notin fn(q')$, per induzione strutturale abbiamo (in particolare) che

$$dec(q', e, i) \simeq_e dec(q, f, j)$$

e quindi, banalmente,

$$\begin{aligned} dec(q, e, i) &= dec((\nu a)q', e, i) = dec(q', e, i) \simeq_e dec(q', f, j) \\ &= dec((\nu a)q', f, j) = dec(q, f, j) \end{aligned}$$

- se $q = B$, con $B \overset{def}{=} q'$ per un opportuno processo $q' \in \mathcal{P}$, per induzione strutturale abbiamo (in particolare) che

$$dec(q', e, i) \simeq_e dec(q', f, j)$$

e quindi, banalmente,

$$\begin{aligned} dec(q, e, i) &= dec(B, e, i) = dec(q', e, i) \simeq_e dec(q', f, j) = dec(B, f, j) \\ &= dec(q, f, j) \end{aligned}$$

□

4.2.3 Restrizione e dec-equivalenza

Vediamo ora un corollario che, in presenza di restrizione, ci garantisce l'esistenza di una sostituzione compatibile.

Corollario 4.2.3. *Dati un generico processo $q \in \mathcal{P}$, una generica azione visibile $a \in \mathcal{L}$ e una generica coppia di numeri naturali $e, i \in \mathbb{N}$, esiste un'azione ristretta (banale o meno) $\eta \in \mathcal{N}$ tale che $\{\eta/a\}$ è una sostituzione compatibile per $dec(q, e, i + 2^e)$ e tale che*

$$dec((\nu a)q, e, i) \simeq_e dec(q, e, i + 2^e)\{\eta/a\}$$

Dimostrazione. Vediamo la casistica su $\{a, \overline{a}\} \cap fn(q)$:

- se $a, \overline{a} \in fn(q)$, ricordando anche la proposizione 3.2.5 (pag. 83), abbiamo che

$$dec((\nu a)q, e, i) = dec(q\{\eta_i/a\}, e, i + 2^e) = dec(q, e, i + 2^e)\{\eta_i/a\}$$

da cui, banalmente,

$$dec((\nu a)q, e, i) \simeq_e dec(q, e, i + 2^e)\{\eta_i/a\}$$

Dal momento che $q \in \mathcal{P}$ e che $i < i+2^e$, dal corollario 4.1.18 (pag. 150) abbiamo che la sostituzione $\{\eta_i/a\}$ è compatibile con $dec(q, e, i + 2^e)$; $\eta = \eta_i$ è quindi l'azione ristretta cercata;

- se $a \in fn(q) \wedge \overline{a} \notin fn(q)$, ricordando anche la proposizione 3.2.5, abbiamo

$$dec((\nu a)q, e, i) = dec(q\{\eta_d/a\}, e, i) = dec(q, e, i)\{\eta_d/a\}$$

Da $\overline{a} \notin fn(q)$ e dalla proposizione 4.1.9 (pag. 130) abbiamo sia che $\overline{a} \notin n(dec(q, e, i))$ sia che $\overline{a} \notin n(dec(q, e, i + 2^e))$ e quindi, dalla definizione, che la sostituzione $\{\eta_d/a\}$ è compatibile sia con $dec(q, e, i)$ che con $dec(q, e, i + 2^e)$; dalla proposizione 4.2.2 (pag. 176) abbiamo che

$$dec(q, e, i) \simeq_e dec(q, e, i + 2^e)$$

e quindi, essendo $\{\eta_d/a\}$ una sostituzione compatibile con entrambi i termini, dalla definizione di dec-corrispondenza abbiamo

$$dec((\nu a)q, e, i) = dec(q, e, i)\{\eta_d/a\} \simeq_e dec(q, e, i + 2^e)\{\eta_d/a\}$$

L'azione $\eta = \eta_d$ è quindi l'azione ristretta cercata.

- se $a \notin fn(q) \wedge \overline{a} \in fn(q)$, la dimostrazione è del tutto analoga a quella del caso precedente; l'azione ristretta cercata è $\eta = \eta_c$;

- se $a, \overline{a} \notin fn(q)$, il caso è degenere e abbiamo

$$dec((\nu a)q, e, i) = dec(q, e, i)$$

D'altro canto, dal momento che né a né il suo co-nome sono liberi in q, qualunque sostituzione di a è ininfluente e quindi, in particolare $q = q\{\eta_d/a\}$; ricordando anche la proposizione 3.2.5, abbiamo

$$dec((\nu a)q, e, i) = dec(q, e, i) = dec(q\{\eta_d/a\}, e, i) = dec(q, e, i)\{\eta_d/a\}$$

e tenendo conto che, in particolare, $\overline{a} \notin fn(q)$ e che dalla proposizione 4.1.9 abbiamo sia che $\overline{a} \notin n(dec(q, e, i))$ sia che $\overline{a} \notin n(dec(q, e, i + 2^e))$ e quindi, dalla definizione, che la sostituzione $\{\eta_d/a\}$ è compatibile sia con $dec(q, e, i)$ che con $dec(q, e, i+2^e)$; dalla proposizione 4.2.2 abbiamo

$$dec(q, e, i) \simeq_e dec(q, e, i + 2^e)$$

e quindi, essendo $\{\eta_d/a\}$ una sostituzione compatibile con entrambi i termini, dalla definizione di dec-corrispondenza abbiamo

$$dec((\nu a)q, e, i) = dec(q, e, i)\{\eta_d/a\} \simeq_e dec(q, e, i + 2^e)\{\eta_d/a\}$$

L'azione $\eta = \eta_d$ è quindi l'azione ristretta cercata (anche se $\eta = \eta_c$ sarebbe stata ugualmente accettabile come pure, si può dimostrare, qualunque $\eta = \eta_j$ con $j < i$).

□

Il corollario precedente non è particolarmente interessante di per sé, ma semplificherà notevolmente la casistica in varie dimostrazioni che seguiranno.

4.2.4 Congruenza strutturale e dec-equivalenza

Vediamo ora un lemma che ci garantisce che le decomposizioni di due processi tra loro congruenti sono dec-equivalenti.

Lemma 4.2.4. *Siano* $q_1, q_2 \in \mathcal{P}$ *due generici processi Multi-CCS; siano* $e, i \in \mathbb{N}$ *due generici valori interi; allora*

$$q_1 \equiv q_2 \implies dec(q_1, e, i) \simeq_e dec(q_2, e, i)$$

Dimostrazione. Dimostriamo l'assunto prima sui casi base su cui è definita la congruenza $q_1 \equiv q_2$, poi – per induzione strutturale – in tutte le possibili forme di q_1 e q_2.

Per quanto riguarda i casi base:

- se $q_1 = (r_1 \mid r_2) \mid r_3$ e $q_2 = r_1 \mid (r_2 \mid r_3)$, abbiamo[5]

$$\begin{aligned}
dec(q_1, e, i) &= dec((r_1 \mid r_2) \mid r_3), e, i) \\
&= dec(r_1 \mid r_2, e+1, i) \oplus dec(r_3, e+1, i+2^e) \\
&= dec(r_1, e+2, i) \oplus dec(r_2, e+2, i+2^{e+1}) \\
&\quad \oplus dec(r_3, e+1, i+2^e) \\
dec(q_2, e, i) &= dec(r_1 \mid (r_2 \mid r_3), e, i) \\
&= dec(r_1, e+1, i) \oplus dec(r_2 \mid r_3, e+1, i+2^e) \\
&= dec(r_1, e+1, i) \oplus dec(r_2, e+2, i+2^e) \\
&\quad \oplus dec(r_3, e+2, i+2^e+2^{e+1}) \\
&= dec(r_1, e+1, i) \oplus dec(r_2, e+2, i+2^e) \\
&\quad \oplus dec(r_3, e+2, i+3 \cdot 2^e)
\end{aligned}$$

Dalla proposizione 4.2.2 (pag. 176) abbiamo che

$$dec(r_1, e+2, i) \simeq_e dec(r_1, e+1, i)$$
$$dec(r_2, e+2, i+2^{e+1}) \simeq_e dec(r_2, e+2, i+2^e)$$

[5]in [7] – appendice B – la prova di questo caso e del seguente del corrispondente Lemma 2 è basata sull'assunto che non è fissata una regola per la scelta dei nomi ristretti e quindi "*implications (1) and (2) follow by the fact that it is possible to choose new restricted names during decomposition so that , e.g.,* $dec((p \mid q) \mid r) = dec(p \mid (q \mid r))$"; avendo però fissato la regola per la scelta dei nomi ristretti, siamo costretti a dimostrare questo caso, e il seguente, tenendo conto di questa regola; la prova, ovviamente, si complica

$$dec(r_3, e+1, i+2^e) \simeq_e dec(r_3, e+2, i+3 \cdot 2^e)$$

e quindi, se riusciamo a dimostrare che i nomi ristretti delle singole componenti non collidono, arriviamo – tenendo conto della definizione di dec-equivalenza – a dimostrare, per questo caso, la tesi.

Per prima cosa osserviamo che applicando il corollario 4.1.15 (pag. 147) a $dec(r_1 \mid r_2, e+1, i)$, otteniamo

$$r_a(dec(r_1, e+2, i)) \cap \overline{dec(r_2, e+2, i+2^{e+1})} = \emptyset$$

Osservando che i nomi ristretti di ognuna delle due componenti di $dec(r_2 \mid r_3, e+1, i+2^e)$ sono un sottoinsieme dei nomi ristretti dell'intera marcatura, e applicando il corollario 4.1.15 a $dec(r_1 \mid (r_2 \mid r_3), e, i)$ otteniamo

$$\begin{aligned} & r_a(dec(r_1, e+1, i)) \oplus \overline{r_a(dec(r_2, e+2, i+2^e))} \\ & \quad \subseteq r_a(dec(r_1, e+1, i)) \oplus \overline{r_a(dec(r_2 \mid r_3, e+1, i+2^e))} = \emptyset \end{aligned}$$

Avendo provato le condizioni di non collisione tra i nomi ristretti, dalla definizione di dec-equivalenza abbiamo

$$\begin{aligned} & dec(r_1, e+2, i) \oplus dec(r_2, e+2, i+2^{e+1}) \\ & \quad \simeq_e dec(r_1, e+1, i) \oplus dec(r_2, e+2, i+2^e) \end{aligned}$$

Vediamo ora di aggiungere la terza componente.

Applicando il corollario 4.1.15 a $dec((r_1 \mid r_2) \mid r_3, e, i)$, otteniamo

$$\begin{aligned} & r_a(dec(r_1, e+2, i) \oplus dec(r_2, e+2, i+2^{e+1})) \cap \overline{dec(r_2, e+1, i+2^e)} \\ & \quad = r_a(dec(r_1 \mid r_2, e+1, i)) \cap \overline{dec(r_2, e+1, i+2^e)} = \emptyset \end{aligned}$$

Per quanto riguarda la condizione di non collisione per $dec(r_3, e+2, i+3 \cdot 2^e)$, il ragionamento è un po' più complesso; per prima cosa osserviamo – anche se potrebbe essere considerato ovvio – che

$$r_a(dec(r_1, e+1, i) \oplus dec(r_2, e+2, i+2^e))$$

$$
\begin{aligned}
&= n(dec(r_1, e+1, i) \oplus dec(r_2, e+2, i+2^e)) \cap (\mathcal{N}_{\mathbb{N}} \cup \overline{\mathcal{N}_{\mathbb{N}}}) \\
&= (n(dec(r_1, e+1, i)) \cup n(dec(r_2, e+2, i+2^e))) \cap (\mathcal{N}_{\mathbb{N}} \cup \overline{\mathcal{N}_{\mathbb{N}}}) \\
&= (n(dec(r_1, e+1, i)) \cap (\mathcal{N}_{\mathbb{N}} \cup \overline{\mathcal{N}_{\mathbb{N}}})) \\
&\quad \cup (n(dec(r_2, e+2, i+2^e)) \cap (\mathcal{N}_{\mathbb{N}} \cup \overline{\mathcal{N}_{\mathbb{N}}})) \\
&= r_a(dec(r_1, e+1, i)) \cup r_a(dec(r_2, e+2, i+2^e))
\end{aligned}
$$

È evidente che l'insieme dei nomi ristretti di una marcatura è un sottoinsieme dei nomi ristretti di una marcatura che la contiene; applicando anche il corollario 4.1.15 a $dec(r_1 \mid r_2, e, i)$ abbiamo quindi che

$$
\begin{aligned}
&r_a(dec(r_1, e+1, i)) \oplus \overline{r_a(dec(r_3, e+2, i+3 \cdot 2^e))} \\
&\quad \subseteq r_a(dec(r_1, e+1, i)) \oplus \overline{r_a(dec(r_2 \mid r_3, e+1, i+2^e))} = \emptyset
\end{aligned}
$$

Applicando il corollario 4.1.15 a $dec(r_2 \mid r_3, e+1, i+2^e)$ otteniamo

$$r_a(dec(r_2, e+2, i+2^e)) \oplus \overline{r_a(dec(r_3, e+2, i+3 \cdot 2^e))} = \emptyset$$

Abbiamo quindi

$$
\begin{aligned}
&r_a(dec(r_1, e+1, i) \oplus dec(r_2, e+2, i+2^e)) \cap \overline{r_a(dec(r_3, e+2, i+3 \cdot 2^e))} \\
&\quad = (r_a(dec(r_1, e+1, i)) \cup r_a(dec(r_2, e+2, i+2^e))) \\
&\qquad \cap \overline{r_a(dec(r_3, e+2, i+3 \cdot 2^e))} \\
&\quad = (r_a(dec(r_1, e+1, i)) \cap \overline{r_a(dec(r_3, e+2, i+3 \cdot 2^e))}) \\
&\qquad \cup (r_a(dec(r_2, e+2, i+2^e)) \cap \overline{r_a(dec(r_3, e+2, i+3 \cdot 2^e))}) \\
&\quad = \emptyset \cup \emptyset = \emptyset
\end{aligned}
$$

Avendo provato le condizioni di non collisione dei nomi ristretti per le componenti basate su r_3 con la somme delle componenti basate su r_1 e r_2, e ricordando la definizione di dec-equivalenza abbiamo

$$dec(q_1, e, i) = dec(r_1, e+2, i) \oplus dec(r_2, e+2, i+2^{e+1})$$

$$\begin{aligned}
&\oplus\, dec(r_3, e+1, i+2^e)\\
&\simeq_e dec(r_1, e+1, i) \oplus dec(r_2, e+2, i+2^e)\\
&\quad \oplus\, dec(r_3, e+2, i+3\cdot 2^e)\\
&= dec(q_2, e, i)
\end{aligned}$$

- se $q_1 = (\nu a)(r_1 \mid r_2)$, $q_2 = r_1 \mid (\nu a)r_2$, dove a e il suo co-nome non sono liberi in r_1, avremo che $a, \overline{a} \notin fn(r_1)$, $\{a, \overline{a}\} \cap fn(r_1 \mid r_2) = \{a, \overline{a}\} \cap (fn(r_1) \cup fn(r_2)) = \{a, \overline{a}\} \cap fn(r_2)$; tenendo anche conto che la sostituzione di a in r_1 (nel caso q_1) sarà ininfluente, abbiamo quattro casi:
  - $a, \overline{a} \in fn(r_2)$, in tal caso il nome a dovrà essere sostituito da un nuovo nome ristretto non banale, diverso nei due casi; ricordando anche la proposizione 3.2.5 (pag. 83) avremo

$$\begin{aligned}
dec(q_1, e, i) &= dec((\nu a)(r_1 \mid r_2), e, i)\\
&= dec((r_1 \mid r_2)\{\eta_i/a\}, e, i+2^e)\\
&= dec(r_1\{\eta_i/a\} \mid r_2\{\eta_i/a\}, e, i+2^e)\\
&= dec(r_1 \mid r_2\{\eta_i/a\}, e, i+2^e)\\
&= dec(r_1, e+1, i+2^e)\\
&\quad \oplus\, dec(r_2\{\eta_i/a\}, e+1, i+2^e+2^e)\\
&= dec(r_1, e+1, i+2^e)\\
&\quad \oplus\, dec(r_2, e+1, i+2^{e+1})\{\eta_i/a\}\\
dec(q_2, e, i) &= dec(r_1 \mid (\nu a)r_2, e, i)\\
&= dec(r_1, e+1, i) \oplus dec((\nu a)r_2, e+1, i+2^e)\\
&= dec(r_1, e+1, i)\\
&\quad \oplus\, dec(r_2\{\eta_{i+2^e}/a\}, e+1, i+2^e+2^{e+1})\\
&= dec(r_1, e+1, i) \oplus dec(r_2, e+1, i+3\cdot 2^e)\{\eta_{i+2^e}/a\}
\end{aligned}$$

Dalla proposizione 4.2.2 abbiamo

$$dec(r_1, e+1, i+2^e) \simeq_e dec(r_1, e+1, i)$$

$$dec(r_2, e+1, i+2^{e+1}) \simeq_e dec(r_2, e+1, i+3\cdot 2^e)$$

Sapendo che $r_2 \in \mathcal{P}$, che $i < i+2^{e+1}$ e che $i+2^e < i+3\cdot 2^e$, possiamo applicare il corollario 4.1.18 (pag. 150) ottenendo che le sostituzioni $\{\eta_i/a\}$ e $\{\eta_{i+2^e}/a\}$ sono compatibili, rispettivamente, con $dec(r_2, e+1, i+2^{e+1})$ e con $dec(r_2, e+1, i+3\cdot 2^e)$; dalla definizione di dec-equivalenza abbiamo quindi anche che

$$dec(r_2, e+1, i+2^{e+1})\{\eta_i/a\} \simeq_e dec(r_2, e+1, i+3\cdot 2^e)\{\eta_{i+2^e}/a\}$$

Applicando poi il corollario 4.1.15 sia a $dec(r_1 \mid r_2\{\eta_i/a\}, e, i+2^e)$ che a $dec(r_1 \mid (\nu a)r_2, e, i)$, abbiamo

$$\begin{aligned} r_a(dec(r_1, e+1, i+2^e)) \cap \overline{r_a(dec(r_2, e+1, i+2^{e+1})\{\eta_i/a\})} &= \emptyset \\ r_a(dec(r_1, e+1, i)) \cap \overline{r_a(dec(r_2, e+1, i+3\cdot 2^e)\{\eta_{i+2^e}/a\})} &= \emptyset \end{aligned}$$

e quindi, avendo verificato le condizioni di non collisione dei nomi ristretti, ricordando la definizione di dec-equivalenza possiamo concludere che

$$\begin{aligned} dec(q_1, e, i) &= dec(r_1, e+1, i+2^e) \oplus dec(r_2, e+1, i+2^{e+1})\{\eta_i/a\} \\ &\simeq_e dec(r_1, e+1, i) \oplus dec(r_2, e+1, i+3\cdot 2^e)\{\eta_{i+2^e}/a\} \\ &= dec(q_2, e, i) \end{aligned}$$

– $a \in fn(r_2) \wedge \overline{a} \notin fn(r_2)$, in tal caso il nome a dovrà essere sostituito dal nome ristretto banale η_d; avremo

$$\begin{aligned} dec(q_1, e, i) = dec((\nu a)(r_1 \mid r_2), e, i) &= dec((r_1 \mid r_2)\{\eta_d/a\}, e, i) \\ &= dec(r_1\{\eta_d/a\} \mid r_2\{\eta_d/a\}, e, i) \\ &= dec(r_1 \mid r_2\{\eta_d/a\}, e, i) \\ &= dec(r_1, e+1, i) \oplus dec(r_2\{\eta_d/a\}, e+1, i+2^e) \\ &= dec(r_1, e+1, i) \oplus dec((\nu a)r_2, e+1, i+2^e) \\ &= dec(r_1 \mid (\nu a)r_2, e, i) = dec(q_2, e, i) \end{aligned}$$

Da $dec(q_1, e, i) = dec(q_2, e, i)$ deriva banalmente che $dec(q_1, e, i) \simeq_e dec(q_2, e, i)$;

– $a \notin fn(r_2) \wedge \overline{a} \in fn(r_2)$; caso del tutto analogo al precedente (usando il nome ristretto banale η_c al posto di η_d) e analogamente si deduce che $dec(q_1, e, i) = dec(q_2, e, i)$ e quindi, banalmente, $dec(q_1, e, i) \simeq_e dec(q_2, e, i)$;

– $a, \overline{a} \notin fn(r_2)$; in questo caso il nome a non viene sostituito da nulla; abbiamo

$$\begin{aligned} dec(q_1, e, i) &= dec((\nu a)(r_1 \mid r_2), e, i) = dec(r_1 \mid r_2, e, i) \\ &= dec(r_1, e+1, i) \oplus dec(r_2, e+1, i+2^e) \\ &= dec(r_1, e+1, i) \oplus dec((\nu a)r_2, e+1, i+2^e) \\ &= dec(r_1 \mid (\nu a)r_2, e, i) = dec(q_2, e, i) \end{aligned}$$

Da $dec(q_1, e, i) = dec(q_2, e, i)$ deriva banalmente che $dec(q_1, e, i) \simeq_e dec(q_2, e, i)$;

- se $q_1 = (\nu a)r$, $q_2 = (\nu b)(r\{b/a\})$ e b e il suo co-nome non sono presenti liberi in r, possiamo limitarci ad osservare che $dec((\nu a)r, e, i)$ e $dec((\nu b)(r\{b/a\}), e, i)$ sostituiranno ad a, nel primo caso, o a b, nel secondo caso, un medesimo nome ristretto η o nulla del tutto (a seconda del valore di $\{a, \overline{a}\} \cap fn(r)$); il caso della mancata sostituzione è banale; nel caso della sostituzione osserviamo che la doppia sostituzione $\{b/a\}\{\eta/b\}$ equivale (considerando che b e $\overline{b}$ non sono presenti in r) alla sostituzione semplice $\{\eta/a\}$; anche in questo caso abbiamo che $dec(q_1, e, i) = dec(q_2, e, i)$ e quindi, banalmente, che $dec(q_1, e, i) \simeq_e dec(q_2, e, i)$;

- se $q_1 = r_1 \mid r_2$ e $q_2 = r_2 \mid r_1$, abbiamo

$$dec(q_1, e, i) = dec(r_1 \mid r_2, e, i) = dec(r_1, e+1, i) \oplus dec(r_2, e+1, i+2^e)$$
$$dec(q_2, e, i) = dec(r_2 \mid r_1, e, i) = dec(r_2, e+1, i) \oplus dec(r_1, e+1, i+2^e)$$

Dalla proposizione 4.2.2 abbiamo

$$dec(r_1, e+1, i) \simeq_e dec(r_1, e+1, i+2^e)$$

$$dec(r_2, e+1, i+2^e) \simeq_e dec(r_1, e+1, i)$$

e dal momento che, applicando il corollario 4.1.15 sia a $dec(r_1 \mid r_2, e, i)$ che a $dec(r_2 \mid r_1, e, i)$, otteniamo

$$r_a(dec(r_1, e+1, i)) \cap \overline{r_a(dec(r_2, e+1, i+2^e))} = \emptyset$$
$$r_a(dec(r_2, e+1, i)) \cap \overline{r_a(dec(r_1, e+1, i+2^e))} = \emptyset$$

ovvero verifichiamo la condizione di non collisione dei nomi ristretti, dalla definizione di dec-equivalenza abbiamo

$$\begin{aligned} & dec(r_1, e+1, i) \oplus dec(r_2, e+1, i+2^e) \\ & \quad \simeq_e dec(r_2, e+1, i) \oplus dec(r_1, e+1, i+2^e) \end{aligned}$$

Riassumendo

$$\begin{aligned} dec(q_1, e, i) = dec(r_1 \mid r_2, e, i) &= dec(r_1, e+1, i) \oplus dec(r_2, e+1, i+2^e) \\ &\simeq_e dec(r_2, e+1, i) \oplus dec(r_1, e+1, i+2^e) \\ &= dec(r_2 \mid r_1, e, i) = dec(q_2, e, i) \end{aligned}$$

- se $q_1 = B$ e $q_2 = r$, con $B \stackrel{def}{=} r$, abbiamo banalmente

$$dec(q_1, e, i) = dec(B, e, i) = dec(r, e, i) = dec(q_2, e, i)$$

  e da $dec(q_1, e, i) = dec(q_2, e, i)$, banalmente abbiamo che $dec(q_1, e, i) \simeq_e dec(q_2, e, i)$.

Analizziamo ora nei casi in cui q_1 e q_2 sono congruenti induttivamente, ovvero in quanto definiti in base a sottotermini a loro volta congruenti:

- se q_1 e q_2 sono sequenziali, allora $dec(q_1, e, i) \simeq_e dec(q_2, e, i)$ in quanto i due termini sono, per definizione, sequenzialmente dec-equivalenti;

- se $q_1 = r_1 \mid r_2$ e $q_2 = s_1 \mid s_2$, per quattro opportuni processi a due a due congruenti $r_1, r_2, s_1, s_2 \in \mathcal{P}$, ovvero tali che con $r_1 \equiv s_1$ e $r_2 \equiv s_2$, per ipotesi induttiva abbiamo

$$dec(r_1, e+1, i) \simeq_e dec(s_1, e+1, i)$$

$$dec(r_2, e+1, i+2^e) \simeq_e dec(s_2, e+1, i+2^e)$$

Applicando il corollario 4.1.15 sia a $dec(r_1 \mid r_2, e, i)$ che a $dec(s_1 \mid s_2, e, i)$ otteniamo

$$r_a(dec(r_1, e+1, i)) \cap \overline{r_a(dec(r_2, e+1, i+2^e))} = \emptyset$$
$$r_a(dec(s_1, e+1, i)) \cap \overline{r_a(dec(s_2, e+1, i+2^e))} = \emptyset$$

e quindi, avendo verificato le condizioni di non collisione dei nomi ristretti, dalla definizione di dec-equivalenza abbiamo che

$$\begin{aligned} dec(q_1, e, i) = dec(r_1 \mid r_2, e, i) &= dec(r_1, e+1, i) \oplus dec(r_2, e+1, i+2^e) \\ &\simeq_e dec(s_1, e+1, i) \oplus dec(s_2, e+1, i+2^e) \\ &= dec(s_1 \mid s_2, e, i) = dec(q_2, e, i) \end{aligned}$$

- se $q_1 = (\nu a)q_1'$ e $q_2 = (\nu a)q_2'$, per opportuni processi $q_1', q_2' \in \mathcal{P}$, con $q_1' \equiv q_2'$, per ipotesi induttiva abbiamo che

$$dec(q_1', e, i+2^e) \simeq_e dec(q_2', e, i+2^e)$$

Applicando il corollario 4.2.3 (pag. 178) sia a $dec((\nu a)q_1', e, i)$ che a $dec((\nu a)q_2', e, i)$, abbiamo che esistono due azioni ristrette $\eta_x, \eta_y \in \mathcal{N}$ tali che

$$dec((\nu a)q_1', e, i) \simeq_e dec(q_1', e, i+2^e)\{\eta_x/a\}$$
$$dec((\nu a)q_2', e, i) \simeq_e dec(q_2', e, i+2^e)\{\eta_y/a\}$$

Tenendo conto del fatto che il corollario 4.2.3 ci garantisce anche che $\{\eta_x/a\}$ è una sostituzione compatibile con $dec(q_1', e, i+2^e)$ e che $\{\eta_y/a\}$ è una sostituzione compatibile con $dec(q_2', e, i+2^e)$, dalla definizione di dec-equivalenza abbiamo

$$\begin{aligned} dec(q_1, e, i) = dec((\nu a)q_1', e, i) &\simeq_e dec(q_1', e, i+2^e)\{\eta_x/a\} \\ &\simeq_e dec(q_2', e, i+2^e)\{\eta_y/a\} \simeq_e dec((\nu a)q_2', e, i) \end{aligned}$$

$$= dec(q_2, e, i)$$

e, per la proprietà transitiva della dec-equivalenza (che, come abbiamo dimostrato nella proposizione 4.2.1 (pag. 174), è una relazione di equivalenza), abbiamo

$$dec(q_1, e, i) \simeq_e dec(q_2, e, i)$$

- se $q_1 = B$ e $q_2 = C$, con $B \stackrel{def}{=} q'_1$, con $C \stackrel{def}{=} q'_2$, per opportuni processi $q'_1, q'_2 \in \mathcal{P}$ tali che $q'_1 \equiv q'_2$, per ipotesi induttiva

  $$dec(q'_1, e, i) \simeq_e dec(q'_2, e, i)$$

  e quindi, banalmente

  $$\begin{aligned} dec(q_1, e, i) &= dec(B, e, i) = dec(q'_1, e, i) \\ &\simeq_e dec(q'_2, e, i) = dec(C, e, i) = dec(q_2, e, i) \end{aligned}$$

□

Combinando quanto appena dimostrato con i precedenti risultati otteniamo il seguente

Corollario 4.2.5. *Siano $q_1, q_2 \in \mathcal{P}$ due generici processi Multi-CCS; siano $e, f, i, j \in \mathbb{N}$ quattro generici valori interi; allora*

$$q_1 \equiv q_2 \implies dec(q_1, e, i) \simeq_e dec(q_2, f, j)$$

Dimostrazione. Combinando la proposizione 4.2.2 (pag. 176) e il lemma 4.2.4 (pag. 181) abbiamo, in generale, che

$$q_1 \equiv q_2 \implies dec(q_1, e, i) \simeq_e dec(q_2, e, i) \simeq_e dec(q_2, f, j)$$

e, tenendo presente che la preposizione 4.2.1 (pag. 174) ci garantisce la transitività della dec-equivalenza, possiamo concludere che

$$q_1 \equiv q_2 \implies dec(q_1, e, i) \simeq_e dec(q_2, f, j)$$

□

4.2.5 Bisimulazione e dec-equivalenza

Sfruttando i risultati visti fino ad ora, dimostriamo una proposizione che sarà fondamentale per dimostrare che la dec-equivalenza implica la bisimulazione.

Per meglio motivarne la dimostrazione, osserviamo che quando $(m_1 \oplus m_2)[t\rangle m'$, quando $t \in T_{MCCS}$ è giustificata dalla regola (com) come sincronizzazione di $t_a, t_b \in T_{MCCS}$ e quando t non è totalmente supportata dalla sola m_1 o dalla sola m_2, nulla garantisce che t_a e/o t_b siano totalmente supportate da m_1 o da m_2.

Per fare un esempio concreto, siano

$$\begin{aligned} m_1 &= \{a.\mathbf{0}\} \oplus \{b.\mathbf{0}\} \\ m_2 &= \{\underline{a}.\overline{b}.\mathbf{0}\} \end{aligned}$$

e supponiamo che la transizione t sia ottenuta dal seguente albero di prova

$$\cfrac{\cfrac{\{a.\mathbf{0}\} \xrightarrow{a} \emptyset \quad \{\underline{a}.\overline{b}.\mathbf{0}\} \xrightarrow{\overline{a}\overline{b}} \emptyset \quad sync(a, \overline{a}\overline{b}, \overline{b})}{\{a.\mathbf{0}\} \oplus \{\underline{a}.\overline{b}.\mathbf{0}\} \xrightarrow{\overline{b}} \emptyset} \quad \{b.\mathbf{0}\} \xrightarrow{b} \emptyset \quad sync(\overline{b}, b, \tau)}{(\{a.\mathbf{0}\} \oplus \{\underline{a}.\overline{b}.\mathbf{0}\}) \oplus \{b.\mathbf{0}\} \xrightarrow{\tau} \emptyset}$$

In questo caso abbiamo che t_2 (la transizione con etichetta b) è supportata dalla sola marcatura m_1 ma anche che la transizione t_1 (con etichetta $\overline{b}$) non è pienamente supportata né dalla sola m_1 né dalla sola m_2 in quanto è basata su una componente della prima e sull'intera seconda marcatura.

Possiamo però rilevare, osservando le transizioni di tabella 3.2 (pag. 91), che le uniche transizioni che non sono supportata da un unico singoletto sono proprio quelle giustificate dalla regola (com). Dal momento che la regola (com) non genera ma combina transizioni, possiamo vedere una transizione generata dalla regola (com) come a un albero binario con almeno due foglie dove le foglie sono rappresentate da transizioni generate da regole diverse da (com) e quindi supportate da singoletti. E dal momento che i singoletti sono sempre contenuti in m_1 o in m_2, è sempre possibile "smontare" una transizione giustificata dalla regola (com) in maniera tale da ottenere un certo numero di transizioni contemporaneamente supportate dalla marcatura m_1 – ai sensi della definizione 1.2.5 (pag. 14) – e un altro numero di transizioni contemporaneamente supportate da m_2.

Nell'esempio di cui sopra, la marcatura m_1 supporta – contemporaneamente – le transizioni $t_{1a} = (\{a.\mathbf{0}\}, a, \emptyset)$ e $t_2 = (\{b.\mathbf{0}\}, b, \emptyset)$ mentre la marcatura m_2 supporta contemporaneamente la sola transizione $t_{1b} = (\{\underline{a}.\bar{b}.\mathbf{0}\}, \overline{a}\overline{b}, \emptyset)$

Si deve però tener presente che le transizioni in T_{MCCS} possono essere ottenute come sincronizzazione di transizioni non necessariamente contenute in T_{MCCS}, ovvero con etichette che comprendono azioni ristrette (che poi spariscono sincronizzandosi).

Il caso che incontreremo di frequente nelle dimostrazioni che seguiranno è quello di marcature m_1 ed m_2 tali che $r_a(m_1) \cap \overline{r_a(m_2)} = \emptyset$ – tipicamente $m_1 \oplus m_2 = dec(q_1 \mid q_2, e, i)$, ovvero $m_1 = dec(q_1, e+1, i)$ e $m_2 = dec(q_2, e+1, i+2^e)$ – e quindi tali che sincronizzazioni su azioni ristrette sono impossibili tra m_1 e m_2, pur non escludendo che possano esservi sincronizzazioni all'interno di m_1 e/o di m_2.

Ipotizzando quindi $(m_1 \oplus m_2)[t\rangle m'$, con $t \in T_{MCCS}$ e $r_a(m_1) \cap \overline{r_a(m_2)} = \emptyset$, possiamo sempre "smontare" t in due multiset di transizioni $u_1, u_2 \in \mathcal{M}_{fin}(T_{MCCS})$ tali che $m_1[[u_1\rangle\rangle m'_1$ e che $m_2[[u_2\rangle\rangle m'_2$, con $m' = m'_1 \oplus m'_2$, seguendo questo semplice algoritmo:

1 - inizializziamo il multiset di transizioni u_1, u_2 e u_s; i primi due col valore nullo ($\emptyset$), il terzo col singoletto $1 \cdot t$;

2 - fin quando u_s non è pari a $\emptyset$

  2.1 - individuiamo in u_s una qualunque transizione t_s e la sua molteplicità k;

  2.2 - sottraiamo $k \cdot t_s$ da u_s;

  2.3 - se t_s è totalmente supportata da m_1, aggiungiamo $k \cdot t_s$ a u_1;

  2.4 - altrimenti, se t_s è totalmente supportata da m_2, aggiungiamo $k \cdot t_s$ a u_2;

2.5 - altrimenti, individuiamo le due componenti t_1 e t_2 di t_s e aggiungiamo $k \cdot t_1 \oplus k \cdot t_2$ a u_s.

Osserviamo che l'algoritmo di cui sopra avrà termine dopo un numero di cicli (punto 2) non superiore al valore $\gamma(t)$ (il peso della transizione t), infatti abbiamo che il peso iniziale di u_s è pari a $\gamma(t)$ e che a ogni iterazione del ciclo si toglie da u_s un peso pari a $k \cdot \gamma(t_s)$ per aggiungere, nel caso peggiore, un peso $k(\gamma(t_1) + \gamma(t_2))$ ovvero – ricordando che $\gamma(t_s) = \gamma(t_1) + \gamma(t_2) + 1$ – un peso $k \cdot (\gamma(t_s) - 1)$. In altre parole, ad ogni iterazione del ciclo il peso $\gamma(u_s)$ diminuisce come minimo di 1 (il valore minimo per k) e quindi il numero di iterazioni non potrà essere superiore a $\gamma(t)$.

Osserviamo anche che la somma dei pesi dei multiset u_1 e u_2 sarà non superiore a $\gamma(t)$; infatti, ad ogni iterazione del ciclo o si toglie a u_s – il cui peso iniziale è proprio $\gamma(t)$ – un peso senza aggiungere nulla a u_1 e u_2, o si trasferisce del peso da u_s a u_1 oppure a u_2. Anzi: la somma $\gamma(u_1) + \gamma(u_2)$ potrà essere uguale a $\gamma(t)$ solo se $\gamma(u_1) = \gamma(t)$ e $\gamma(u_2) = 0$ o, al contrario, se $\gamma(u_2) = \gamma(t)$ e $\gamma(u_1) = 0$, ovvero quando t è supportata interamente o da m_1 o da m_2; se infatti t non è interamente supportata da una sola delle due marcature, l'algoritmo "spezzerà" t in due transizioni, perdendo quindi un peso pari a 1 che non potrà essere recuperato in seguito.

Ragionando – come sopra – sul supporto di $m_1 \oplus m_2$ non su una sola transizione ma sul supporto contemporaneo di un multiset u di transizioni in T_{MCCS}, utilizzando il medesimo algoritmo descritto sopra (ovviamente inizializzando u_s con u e non con t) possiamo concludere con la seguente

Osservazione 4.2.6. *Siano $m_1, m_2 \in \mathcal{M}_{fin}(S_{MCCS})$ tali che $r_a(m_1) \cap \overline{r_a(m_2)} = \emptyset$; se $m' \in \mathcal{M}_{fin}(S_{MCCS})$ è una marcatura e $u \in \mathcal{M}_{fin}(T_{MCCS})$ è un multiset di transizioni tale che $(m_1 \oplus m_2)[[u\rangle\rangle m'$, allora esistono due marcature $m'_1, m'_2 \in \mathcal{M}_{fin}(S_{MCCS})$ e due multiset $u_1, u_2 \in \mathcal{M}_{fin}(T_{MCCS})$ di transizioni che possono essere ricombinate ottenendo u e tali che:*

- $m_1[[u_1\rangle\rangle m'_1$
- $m_2[[u_2\rangle\rangle m'_2$

- $m' = m'_1 \oplus m'_2$
- $\gamma(u_1) + \gamma(u_2) \leqslant \gamma(u)$
- $\gamma(u_1) = \gamma(u) \implies (u_1 = u \wedge u_2 = \emptyset \wedge m'_2 = m_2)$
- $\gamma(u_2) = \gamma(u) \implies (u_2 = u \wedge u_1 = \emptyset \wedge m'_1 = m_1)$

Siamo quindi pronti per dimostrare la seguente

Proposizione 4.2.7. *Siano $m_a, m_b \in \mathcal{M}_{fin}(S_{MCCS})$ due generiche marcature; se $m_a \simeq_e m_b$, per ogni $u_a \in \mathcal{M}_{fin}(T_{MCCS})$ abbiamo*

$$\begin{aligned} m_a[[u_a\rangle\rangle m'_a \implies & \exists u_b \in \mathcal{M}_{fin}(T_{MCCS}).\exists m'_b \in \mathcal{M}_{fin}(S_{MCCS}). \\ & (m_b[[u_b\rangle\rangle m'_b \wedge mtl(u_a) = mtl(u_b) \wedge \gamma(u_a) = \gamma(u_b) \wedge m'_a \simeq_e m'_b) \end{aligned}$$

Inoltre, per ogni $u_b \in \mathcal{M}_{fin}(T_{MCCS})$ abbiamo

$$\begin{aligned} m_b[[u_b\rangle\rangle m'_b \implies & \exists u_a \in \mathcal{M}_{fin}(T_{MCCS}).\exists m'_a \in \mathcal{M}_{fin}(S_{MCCS}). \\ & (m_a[[u_a\rangle\rangle m'_a \wedge mtl(u_a) = mtl(u_b) \wedge \gamma(u_a) = \gamma(u_b) \wedge m'_a \simeq_e m'_b) \end{aligned}$$

Dimostrazione. Dimostriamo solo la prima implicazione poiché la seconda deriva immediatamente dal fatto che la dec-equivalenza è riflessiva.

Dimostriamo la prima implicazione per induzione numerica forte su $\gamma(u_a)$.

Se $\gamma(u_a) = 0$, è evidente che $u_a = \emptyset$ e quindi che $m'_a = m_a$; è sufficiente scegliere $u_b = \emptyset$ e otteniamo che $m'_b = m_b$, che $mtl(u_a) = mtl(\emptyset) = mtl(u_b)$ che $\gamma(u_a) = \gamma(\emptyset) = \gamma(u_b)$ e che $m'_a = m_a \simeq_e m_b = m'_b$; per il caso (degenere) $\gamma(u_a) = 0$ la tesi è banalmente verificata.

Se $\gamma(u_a) = 1$, dal momento che tutte le transizioni giustificate dalle altre regole di tabella 3.2 (pag. 91) hanno un peso maggiore di 1, abbiamo necessariamente che $u_a = 1 \cdot t_a$, dove $t_a \in T_{MCCS}$ è giustificata dalla regola (pref). Dimostriamo questo caso per induzione strutturale sulla definizione di dec-equivalenza:

- non può essere $m_a = dec(\mathbf{0}, e, i) = \emptyset = dec(\mathbf{0}, f, j) = m_b$, poiché m_a (cosi come m_b) non è in grado di supportare alcuna transizione;

- se $m_a = dec(\mu.q_a, e, i)$ e $m_b = dec(\mu.q_b, f, j)$ – per un'opportuna azione $\mu \in \mathcal{A}ct$, per opportuni numeri interi $e, f, i, j \in \mathbb{N}$ e per opportuni processi $q_a, q_b \in \mathcal{P}$ tali che $q_a \equiv q_b$ – l'ispezione delle regole di tabella 3.2 ci dice che $m_a = dec(\mu.q_a, e, i)$ supporta unicamente una transizione $t_a \in\longrightarrow$ – con etichetta $l(t_a) = \mu$, quindi anche $t_a \in T_{MCCS}$ – grazie alla regola (pref); abbiamo

$$m_a = dec(\mu.q_a, e, i)[t_a\rangle dec(q_a, e, i) = m'_a$$

  Applicando la medesima regola (pref) a m_b otteniamo una transizione $t_b \in\longrightarrow$ – con etichetta $l(t_b) = \mu$, quindi anche $t_b \in T_{MCCS}$ – tale che

$$m_b = dec(\mu.q_b, f, j)[t_b\rangle dec(q_b, f, j)$$

  Il multiset $u_b = 1 \cdot t_b$ e la marcatura $m'_b = dec(q_b, f, j)$ sono il multiset e la marcatura cercati; abbiamo infatti che $m_b[[u_b\rangle\rangle m'_b$, che

$$\begin{aligned} mtl(u_b) &= mtl(1 \cdot t_b) = 1 \cdot l(t_b) = 1 \cdot \mu = 1 \cdot l(t_a) = mtl(1 \cdot t_a) \\ &= mtl(u_a) \end{aligned}$$

  che

$$\gamma(u_b) = \gamma(1 \cdot t_b) = 1 \cdot \gamma(t_b) = 1 \cdot 1 = 1 \cdot \gamma(t_a) = \gamma(1 \cdot t_a) = \gamma(u_a)$$

  e, applicando il corollario 4.2.5 (pag. 189) a $q_a \equiv q_b$, anche che

$$m'_a = dec(q_a, e, i) \simeq_e dec(q_b, f, j) = m'_b$$

- non può essere $m_a = dec(\underline{\mu}.q_a, e, i)$ e $m_b = dec(\underline{\mu}.q_b, f, j)$ – per un'opportuna azione $\mu \in \mathcal{A}ct$, per opportuni numeri interi $e, f, i, j \in \mathbb{N}$ e per opportuni processi $q_a, q_b \in \mathcal{P}$ tali che $q_a \equiv q_b$ – poiché l'ispezione delle regole di tabella 3.2 ci dice che m_a (come m_b) non può supportare transizioni con la regola (pref), ovvero le uniche con peso 1;

- analogamente al caso precedente, e per lo stesso motivo, non può neppure essere $m_a = dec(p_{a1} + p_{a2}, e, i)$ e $m_b = dec(p_{b1} + p_{b2}, f, j)$ – per opportuni numeri interi $e, f, i, j \in \mathbb{N}$ e per opportuni processi sequenziali $p_{a1}, p_{a2}, p_{b1}, p_{b2} \in \mathcal{P}_{seq}$ tali che $p_{a1} \equiv p_{b1}$ e che $p_{a2} \equiv p_{b2}$;

- se $m_a = m''_a\{\eta_a/c\}$ e $m_b = m''_b\{\eta_b/c\}$ – per opportune marcature $m''_a, m''_b \in \mathcal{M}_{fin}(S_{MCCS})$, tali che $m''_a \simeq_e m''_b$, e per opportune sostituzioni $\{\eta_a/c\}$, compatibile con m''_a, e $\{\eta_b/c\}$, compatibile con m''_b – se $m_a = m''_a\{\eta_a/c\}[t_a\rangle m'_a$ (con, ricordiamo, $u_a = 1 \cdot t_a$ e $\gamma(t_a) = 1$), allora – applicando a $t_a \in T_{MCCS}$ il corollario 4.1.31 (pag. 171) – otteniamo che esistono una transizione $t'_a \in T_{MCCS}$ e una marcatura m'''_a tali che $l(t_a) = l(t'_a)$, che $c, \overline{c} \notin n(l(t_a)) = n(l(t'_a))$, che $m''_a[t'_a\rangle m'''_a$, che $m'_a = m'''_a\{\eta_a/c\}$ e che $\gamma(t'_a) = \gamma(t_a) = 1$.

  Applicando l'induzione strutturale su $m''_a \simeq_e m''_b$ e su $u'_a = 1 \cdot t'_a \in \mathcal{M}_{fin}(T_{MCCS})$ otteniamo che esistono un multiset $u'_b \in \mathcal{M}_{fin}(T_{MCCS})$ e una marcatura m'''_b tali che $mtl(u'_b) = mtl(u'_a) = mtl(1 \cdot t'_a) = 1 \cdot t'_a$, che $\gamma(u'_b) = \gamma(u'_a) = 1$, che $m''_b[[u'_b\rangle\rangle m'''_b$ e che $m'''_a \simeq_e m'''_b$.

  Dal momento che $mtl(u'_b) = mtl(1 \cdot t'_a)$ e che $\gamma(u'_b) = 1$, abbiamo che $u'_b = 1 \cdot t'_b$ con $\gamma(t'_b) = 1$ e con $l(t'_b) = l(t'_a)$. Dal momento che $c, \overline{c} \notin n(l(t'_a))$ e che $l(t'_b) = l(t'_a)$, è evidente che $c, \overline{c} \notin n(l(t'_b))$ e quindi che possiamo applicare a t'_b il corollario 4.1.25 (pag. 161) ottenendo che esiste una transizione $t_b = ({}^\bullet t'_b\{\eta_b/c\}, l(t'_b), t'^\bullet_b\{\eta_b/c\}) \in T_{MCCS}$ tale $\gamma(t_b) = \gamma(t'_b) = 1$ e che $m_b = m''_b\{\eta_b/c\}[t_b\rangle m'''_b\{\eta_b/c\}$; abbiamo

  $$l(t_b) = l(t'_b) = l(t'_a) = l(t_a)$$

  Il multiset $u_b = 1 \cdot t_b$ e la marcatura $m'_b = m'''_b\{\eta_b/c\}$ sono quindi il multiset e la marcatura cercati; abbiamo infatti $\gamma(u_b) = \gamma(1 \cdot t_b) = 1 \cdot \gamma(t_b) = 1 \cdot 1 = 1 = \gamma(u_a)$ e $mtl(u_b) = mtl(1 \cdot t_b) = 1 \cdot l(t_b) = 1 \cdot l(t_a) = mtl(1 \cdot t_a) = mtl(u_a)$; rimane solo da verificare che $m'_a \simeq_e m'_b$.

  Ricordando che $m''_a[t'_a\rangle m'''_a$, dalla proposizione 4.1.13 (pag. 144) e dal conseguente corollario 4.1.16 (pag. 148) abbiamo sia che $n(m'''_a) \subseteq$

$n(m''_a)$, sia che $r_a(m'''_a) \subseteq r_a(m''_a)$; ragionando sulla casistica della definizione di sostituzione compatibile è evidente che, dal momento che $\{\eta_a/c\}$ è una sostituzione compatibile con m''_a, non può che essere compatibile anche con m'''_a. Analogamente si prova anche che $\{\eta_b/c\}$ è una sostituzione compatibile con m'''_b e quindi, ricordando che $m'''_a \simeq_e m'''_b$, dalla definizione di dec-equivalenza abbiamo che

$$m'_a = m'''_a\{\eta_a/c\} \simeq_e m'''_b\{\eta_b/c\} = m'_b$$

- se $m_a = m_{a1} \oplus m_{a2}$ e $m_b = m_{b1} \oplus m_{b2}$ – per opportune marcature $m_{a1}, m_{a2}, m_{b1}, m_{b2} \in \mathcal{M}_{fin}(S_{MCCS})$, tali che $m_{a1} \simeq_e m_{b1}$, che $m_{a2} \simeq_e m_{b2}$ e per le quali valgono le condizioni di non collisione delle azioni ristrette $r_a(m_{a1}) \cap \overline{r_a(m_{a2})} = \emptyset$ e $r_a(m_{b1}) \cap \overline{r_a(m_{b2})} = \emptyset$ – riguardo alla transizione $t_a \in T_{MCCS}$ abbiamo solo due casi possibili: (a) è interamente supportata da m_{a1} oppure (b) è interamente supportata da m_{b1}; infatti, dal momento che $\gamma(t_a) = 1$, t_a è giustificata dalla regola (pref) e quindi supportata da un singoletto che è necessariamente contenuto o in m_{a1} o in m_{a2}; quindi
  - quando $u_a = 1 \cdot t_a$ è interamente supportata dalla marcatura m_{a1}, operando per induzione strutturale su $m_{a1} \simeq_e m_{b1}$ e su $m_{a1}[[u_a\rangle\rangle m'_{a1}$ (con $m'_a = m'_{a1} \oplus m_{a2}$) otteniamo che esistono un multiset $u_b \in \mathcal{M}_{fin}(T_{MCCS})$ e una marcatura m'_{b1} tali che $m_{b1}[[u_b\rangle\rangle m'_{b1}$, che $mtl(u_b) = mtl(u_a) = 1 \cdot l(t_a)$, che $\gamma(u_b) = \gamma(u_a) = 1$ e che $m'_{a1} \simeq_e m'_{b1}$; dal momento che $mtl(u_b) = 1 \cdot l(t_a)$ e che $\gamma(u_b) = \gamma(u_a) = 1$, abbiamo necessariamente che $u_b = 1 \cdot t_b$, per un'opportuna transizione $t_b \in T_{MCCS}$ tale che $l(t_b) = l(t_a)$; il multiset u_b e la marcatura $m'_b = m'_{b1} \oplus m_{b2}$ sono quindi il multiset e la marcatura cercati; rimane solo da dimostrare che $m'_a \simeq_e m'_b$; applicando il corollario 4.1.16 a $m_{a1}[t_a\rangle m'_{a1}$ e a $m_{b1}[t_b\rangle m'_{b1}$ otteniamo $r_a(m'_{a1}) \subseteq r_a(m_{a1})$ e $r_a(m'_{b1}) \subseteq r_a(m_{b1})$; ricordando anche le ipotesi abbiamo quindi

$$r_a(m'_{a1}) \cap \overline{r_a(m_{a2})} \subseteq r_a(m_{a1}) \cap \overline{r_a(m_{a2})} = \emptyset$$

$$r_a(m'_{b1}) \cap \overline{r_a(m_{b2})} \subseteq r_a(m_{b1}) \cap \overline{r_a(m_{b2})} = \emptyset$$

e ricordando che abbiamo $m'_{a1} \simeq_e m'_{b1}$ e $m_{a2} \simeq_e m_{b2}$, dal momento che sono soddisfatte le condizioni di non collisione delle azioni ristrette, dalla definizione di dec-equivalenza abbiamo

$$m'_a = m'_{a1} \oplus m_{a2} \simeq_e m'_{b1} \oplus m_{b2} = m'_b$$

– quando t_a è interamente supportata dalla marcatura m_{a2}, la dimostrazione è del tutto analoga a quella del sottocaso precedente.

Supponiamo ora che sia verificata la tesi per i multiset con peso inferiore a n – e supponendo anche che sia $n \geqslant 2$, dal momento che abbiamo dimostrato la tesi per $n = 0$ e per $n = 1$ – dimostriamo per induzione strutturale sulla definizione di dec-equivalenza che la tesi è verificata anche quando $\gamma(u_a) = n$:

- anche in questo caso, non può essere $m_a = dec(\mathbf{0}, e, i) = \emptyset = dec(\mathbf{0}, f, j) = m_b$, poiché m_a (cosi come m_b) non è in grado di supportare alcuna transizione;

- non può essere $m_a = dec(\mu.q_a, e, i)$ e $m_b = dec(\mu.q_b, f, j)$ – per un'opportuna azione $\mu \in \mathcal{A}ct$, per opportuni numeri interi $e, f, i, j \in \mathbb{N}$ e per opportuni processi $q_a, q_b \in \mathcal{P}$ tali che $q_a \equiv q_b$ – poiché abbiamo ipotizzato $\gamma(u_a) \geqslant 2$ e l'ispezione delle regole di tabella 3.2 ci dice che $m_a = dec(\mu.q_a, e, i)$ può supportare unicamente una transizione giustificata dalla regola (pref), ovvero con $\gamma(u_a) = 1$;

- se $m_a = dec(\underline{\mu}.q_a, e, i)$ e $m_b = dec(\underline{\mu}.q_b, f, j)$ – per un'opportuna azione $\mu \in \mathcal{A}ct$, per opportuni numeri interi $e, f, i, j \in \mathbb{N}$ e per opportuni processi $q_a, q_b \in \mathcal{P}$ tali che $q_a \equiv q_b$ – l'ispezione delle regole di tabella 3.2 ci dice che $m_a = dec(\mu.q_a, e, i)$ può supportare unicamente una transizione (per volta) giustificata dalla regola (s-pref); sia quindi $u_a = 1 \cdot t_a$, con $t_a \in T_{MCCS}$, con $l(t_a) = \sigma$, una di queste transizioni;

ricordando anche l'osservazione 3.2.6 (pag. 93), dalla premessa abbiamo che $dec(q_a, e, i)[t'_a\rangle m'_a$, dove $t'_a \in\longrightarrow$ è una transizione con etichetta $l(t'_a) = \sigma'$ tale che $conc(\mu, \sigma', \sigma)$; osservando le regole di concatenazione di tabella 3.4 (pag. 93) si può rilevare che *conc* non può annullare azioni ristrette eventualmente presenti in σ', quindi anche σ conterebbe azioni ristrette se σ' ne contenesse; da quanto osservato, e dal fatto che – per ipotesi – $t_a \in T_{MCCS}$, abbiamo che σ' non può contenere azioni ristrette e, quindi, anche $t'_a \in T_{MCCS}$.

Dal momento che $q_a \equiv q_b$, applicando il corollario 4.2.5 abbiamo che $dec(q_a, e, i) \simeq_e dec(q_b, f, j)$ e, applicando l'induzione numerica (sul peso dei multiset di transizioni) a questa dec-equivalenza e al supporto $dec(q_a, e, i)[[u'_a\rangle\rangle m'_a$ – dove $u'_a = 1 \cdot t'_a$, con $\gamma(t_a) = \gamma(t'_a) + 1$ (per definizione) e quindi $\gamma(t'_a) < \gamma(t_a)$ – otteniamo che esistono un multiset $u'_b \in \mathcal{M}_{fin}(T_{MCCS})$ e una marcatura m'_b tali che $dec(q_b, f, j)[[u'_b\rangle\rangle m'_b$, che $mtl(u'_b) = mtl(u'_a)$, che $\gamma(u'_b) = \gamma(u'_a)$ e che $m'_a \simeq_e m'_b$; dal momento che $mtl(u'_b) = mtl(u'_a) = 1 \cdot l(t'_a) = 1 \cdot \sigma'$, abbiamo che esiste una transizione $t'_b \in T_{MCCS}$ tale che $u'_b = 1 \cdot t'_b$ con $l(t'_b) = \sigma' = l(t'_a)$.

Applicando la regola (s-pref) a $dec(\underline{\mu}.q_b, f, j)$, usando $dec(q_b, f, j)[t'_b\rangle m'_b$ come premessa, otteniamo che esiste una transizione $t_b \in\longrightarrow$ tale che $conc(\mu, l(t'_b), l(t_b))$, ovvero $conc(\mu, \sigma', l(t_b))$, ovvero – anche per quanto visto in precedenza – $l(t_b) = \sigma = l(t_a)$ e, dal momento che $t_a \in T_{MCCS}$, è anche vero che $t_b \in T_{MCCS}$.

Il multiset $u_b = 1 \cdot t_b$ e la marcatura m'_b sono il multiset e la marcatura cercate; abbiamo infatti visto che $u_b \in \mathcal{M}_{fin}(T_{MCCS})$, che $mtl(u_a) = 1 \cdot l(t_a) = 1 \cdot \sigma = 1 \cdot l(t_b) = mtl(u_b)$, che $\gamma(u_a) = 1 \cdot \gamma(t_a) = 1 \cdot (\gamma(t'_a) + 1) = 1 \cdot (\gamma(t'_b) + 1) = 1 \cdot \gamma(t_b) = \gamma(u_b)$ e che $m'_a \simeq_e m'_b$;

- se $m_a = dec(p_{a1} + p_{a2}, e, i)$ e $m_b = dec(p_{b1} + p_{b2}, f, j)$ – per opportuni numeri interi $e, f, i, j \in \mathbb{N}$ e per opportuni processi sequenziali $p_{a1}, p_{a2}, p_{b1}, p_{b2} \in \mathcal{P}_{seq}$ tali che $p_{a1} \equiv p_{b1}$ e che $p_{a2} \equiv p_{b2}$ – l'ispezione delle regole di tabella 3.2 ci dice che $m_a = dec(p_{a1} + p_{a2}, e, i)$ supporta unicamente una transizione (per volta) giustificata dalla regola (sum-1) o dalla regola (sum-2); sia quindi $u_a = 1 \cdot t_a$, con $t_a \in T_{MCCS}$ e $\gamma(t_a) = n$:

- quando t_a è giustificata dalla regola (sum-1), la premessa ci garantisce che $dec(p_{a1}, e+1, i)[t'_a\rangle m'_a$, con $t'_a \in\longrightarrow$ e $l(t'_a) = l(t_a)$ e quindi, dal momento che $t_a \in T_{MCCS}$, vale anche $t'_a \in T_{MCCS}$; sia $u'_a = 1 \cdot t'_a \in \mathcal{M}_{fin}(T_{MCCS})$; osserviamo che $n = \gamma(u_a) = \gamma(t_a) = \gamma(t'_a) + 1 = \gamma(u'_a) + 1$, ovvero che $\gamma(u'_a) = n - 1 < n$; dal momento che $p_{a1} \equiv p_{b1}$, dalla definizione di dec-equivalenza sequenziale abbiamo che

  $$dec(p_{a1}, e+1, i) \simeq_e dec(p_{b1}, f+1, j)$$

  e applicando l'induzione numerica (dal momento che $\gamma(u'_a) < n$) a questa dec-equivalenza e al multiset $u'_a \in \mathcal{M}_{fin}(T_{MCCS})$ otteniamo che esistono un multiset di transizioni $u'_b \in \mathcal{M}_{fin}(T_{MCCS})$ e una marcatura m'_b tali che tali che $dec(p_{b1}, f+1, j)[[u'_b\rangle\rangle m'_b$, che $mtl(u'_b) = mtl(u'_a)$, che $\gamma(u'_b) = \gamma(u'_a)$ e che $m'_a \simeq_e m'_b$; dal momento che $mtl(u'_b) = mtl(u'_a) = 1 \cdot l(t'_a)$, abbiamo che esiste una transizione $t'_b \in T_{MCCS}$ tale che $u'_b = 1 \cdot t'_b$ con $l(t'_b) = l(t'_a)$.

  Applicando la regola (sum-1) a $dec(p_{b1} + p_{b2}, f, j)$, usando $dec(p_{b1}, f+1, j)[t'_b\rangle m'_b$ come premessa, otteniamo che esiste una transizione $t_b \in\longrightarrow$ tale che $l(t_b) = l(t'_b)$; dal momento che $l(t_b) = l(t'_b) = l(t'_a) = l(t_a)$ e che $t_a \in T_{MCCS}$, abbiamo anche che $t_b \in T_{MCCS}$.

  Il multiset $u_b = 1 \cdot t_b$ e la marcatura m'_b sono il multiset e la marcatura cercate; abbiamo infatti visto che $u_b \in \mathcal{M}_{fin}(T_{MCCS})$ e abbiamo che $mtl(u_a) = 1 \cdot l(t_a) = 1 \cdot l(t_b) = mtl(u_b)$, che $\gamma(u_a) = 1 \cdot \gamma(t_a) = 1 \cdot (\gamma(t'_a) + 1) = 1 \cdot (\gamma(t'_b) + 1) = 1 \cdot \gamma(t_b) = \gamma(u_b)$ e che $m'_a \simeq_e m'_b$;

- quando t_a è giustificata dalla regola (sum-2), la dimostrazione è analoga a quella del sottocaso precedente ma operando per induzione strutturale su $dec(p_{a2}, e+1, i+2^e)$ e su $dec(p_{b2}, f+1, j+2^f)$;

- se $m_a = m''_a\{\eta_a/c\}$ e $m_b = m''_b\{\eta_b/c\}$ – per opportune marcature $m''_a, m''_b \in \mathcal{M}_{fin}(S_{MCCS})$, tali che $m''_a \simeq_e m''_b$, e per opportune sostituzioni $\{\eta_a/c\}$, compatibile con m''_a, e $\{\eta_b/c\}$, compatibile con m''_b – se $m_a = m''_a\{\eta_a/c\}[[u_a\rangle\rangle m'_a$ con $u_a = k_1 t_{a1} \oplus k_2 t_{a2} \oplus \ldots \oplus k_r t_{ar}$ (per r opportuni numeri naturali $k_1, k_2, \ldots, k_r \in \mathbb{N}$ e r opportune transizioni $t_{a1}, t_{a2}, \ldots, t_{ar} \in T_{MCCS}$), allora – applicando a $u_a \in \mathcal{M}_{fin}(T_{MCCS})$ il corollario 4.1.30 (pag. 170) otteniamo che esistono un multiset di transizioni u'_a e una marcatura m'''_a tali che $u'_a = k_1 t'_{a1} \oplus k_2 t'_{a2} \oplus \ldots \oplus k_r t'_{ar}$ (per opportune transizioni $t'_{a1}, t'_{a2}, \ldots, t'_{ar} \in T_{MCCS}$), che $m''_a[[u'_a\rangle\rangle m'''_a$, che $m'_a = m'''_a\{\eta_a/c\}$, che – per ogni $i \in [1 \ldots r]$ – $t_{ai} = ({}^\bullet t'_{ai}\{\eta_a/c\}, l(t'_{ai}), t'^{\bullet}_{ai}\{\eta_a/c\})$, che $c, \overline{c} \notin n(l(t_{ai})) = n(l(t'_{ai}))$ e che $\gamma(t_{ai}) = \gamma(t'_{ai})$.

  Applicando l'induzione strutturale su $m''_a \simeq_e m''_b$ e su $u'_a \in \mathcal{M}_{fin}(T_{MCCS})$ otteniamo che esistono un multiset $u'_b \in \mathcal{M}_{fin}(T_{MCCS})$ e una marcatura m'''_b tali che $mtl(u'_b) = mtl(u'_a)$, che $\gamma(u'_b) = \gamma(u'_a)$, che $m''_b[[u'_b\rangle\rangle m'''_b$ e che $m'''_a \simeq_e m'''_b$.

  Sia $u'_b = l_1 \cdot t'_{b1} \oplus l_2 \cdot t'_{b2} \oplus \ldots \oplus l_n \cdot t'_{bs}$, per s opportuni numeri interi $l_1, l_2, \ldots, l_s \in \mathbb{N}$ e per s opportune transizioni $t'_{b1}, t'_{b2}, \ldots, t'_{bs} \in \mathcal{M}_{fin}(T_{MCCS})$; applicando il corollario 4.1.24 (pag. 160) a $m''_b[[u'_b\rangle\rangle m'''_b$ e a $\{\eta_b/c\}$ (ricordando che c e $\overline{c}$ non sono presenti nelle etichette di u'_b) otteniamo che esistono un multiset di transizioni u_b e una marcatura m'_b tali che $u_b = l_1 \cdot t_{b1} \oplus l_2 \cdot t_{b2} \oplus \ldots l_s \cdot t_{bs}$, per opportune transizioni $t_{b1}, t_{b2}, \ldots, t_{bs} \in \mathcal{M}_{fin}(T_{MCCS})$, che $m''_b\{\eta_b/c\}[[u_b\rangle\rangle m'''_b\{\eta_b/c\}$, che – per ogni $i \in [1 \ldots s]$ – $t_{bi} = ({}^\bullet t'_{bi}\{\eta_b/c\}, l(t'_{bi}), t'^{\bullet}_{bi}\{\eta_b/c\})$ e che $\gamma(t_{bi}) = \gamma(t'_{bi})$.

  Il multiset di transizioni u_b e la marcatura $m'_b = m'''_b\{\eta_b/a\}$ sono il multiset e la marcatura cercate; abbiamo infatti visto che $u_b \in$

$\mathcal{M}_{fin}(T_{MCCS})$; abbiamo anche che

$$\begin{aligned} mtl(u_a) &= \sum_{i=1}^{r} k_i \cdot l(t_{ai}) = \sum_{i=1}^{r} k_i \cdot l(t'_{ai}) = mtl(u'_a) \\ &= mtl(u'_b) = \sum_{i=1}^{s} l_i \cdot l(t'_{bi}) = \sum_{i=1}^{s} l_i \cdot l(t_{bi}) = mtl(u_b) \end{aligned}$$

e che

$$\begin{aligned} \gamma(u_a) &= \sum_{i=1}^{r} k_i \cdot \gamma(t_{ai}) = \sum_{i=1}^{r} k_i \cdot \gamma(t'_{ai}) = \gamma(u'_a) \\ &= \gamma(u'_b) = \sum_{i=1}^{s} l_i \cdot \gamma(t'_{bi}) = \sum_{i=1}^{s} l_i \cdot \gamma(t_{bi}) = \gamma(u_b) \end{aligned}$$

Rimane solo da dimostrare che $m'_a \simeq_e m'_b$; ricordando che $m''_a[[k_1 t_{a1} \oplus k_2 t_{a2} \oplus \ldots \oplus k_r t_{ar}\rangle\rangle m'''_a$, e che – in base all'osservazione 1.2.4 (pag. 17) – abbiamo

$$m''_a[\underbrace{t_{a1}\ldots t_{a1}}_{k_1 \text{ volte}}\underbrace{t_{a2}\ldots t_{a2}}_{k_2 \text{ volte}}\ \ldots\ \underbrace{t_{ar}\ldots t_{ar}}_{k_r \text{ volte}}\rangle m'''_a$$

e quindi, applicando $k_1 + k_2 + \ldots + k_r$ volte la proposizione 4.1.13 e il conseguente corollario 4.1.16 possiamo concludere che $n(m'''_a) \subseteq n(m''_a)$ e che $r_a(m'''_a) \subseteq r_a(m''_a)$; ragionando sulla casistica della definizione di sostituzione compatibile è evidente che, dal momento che $\{\eta_a/c\}$ è una sostituzione compatibile con m''_a, non può che essere compatibile anche con m'''_a. Analogamente si prova anche che $\{\eta_b/c\}$ è una sostituzione compatibile con m'''_b e quindi, ricordando che $m'''_a \simeq_e m'''_b$, dalla definizione di dec-equivalenza abbiamo che

$$m'_a = m'''_a\{\eta_a/c\} \simeq_e m'''_b\{\eta_b/c\} = m'_b$$

- se $m_a = m_{a1} \oplus m_{a2}$ e $m_b = m_{b1} \oplus m_{b2}$ – per opportune marcature $m_{a1}, m_{a2}, m_{b1}, m_{b2} \in \mathcal{M}_{fin}(S_{MCCS})$, tali che $m_{a1} \simeq_e m_{b1}$, che $m_{a2} \simeq_e m_{b2}$ e per le quali valgono le condizioni di non collisione delle azioni ristrette $r_a(m_{a1}) \cap \overline{r_a(m_{a2})} = \emptyset$ e $r_a(m_{b1}) \cap \overline{r_a(m_{b2})} = \emptyset$ – riguardo al multiset di transizioni u_a, ricordando l'osservazione 4.2.6 (pag. 192), abbiamo che u_a può essere "scomposto" nei multiset di transizioni u_{a1} e u_{a2} (con $\gamma(u_{a1})+\gamma(u_{a2}) \leqslant \gamma(u_a)$, e quindi con la possibilità di applicare o l'induzione numerica o l'induzione strutturale su u_{a1} e su u_{a2}) tali che $m_{a1}[[u_{a1}\rangle\rangle m'_{a1}$, che $m_{a2}[[u_{a2}\rangle\rangle m'_{a2}$, dove $m'_a = m'_{a1} \oplus m'_{a2}$; applicando l'induzione (numerica o strutturale) a u_{a1} otteniamo che esistono un multiset di transizioni u_{b1} e una marcatura m'_{b1} tali che $m_{b1}[[u_{b1}\rangle\rangle m'_{b1}$, che $mtl(u_{b1}) = mtl(u_{a1})$, che $\gamma(u_{b1}) = \gamma(u_{a1})$ e che $m'_{a1} \simeq_e m'_{b1}$; analogamente, applicando l'induzione (numerica o strutturale) a u_{a2} otteniamo che esistono un multiset di transizioni u_{b2} e una marcatura m'_{b2} tali che $m_{b2}[[u_{b2}\rangle\rangle m'_{b2}$, che $mtl(u_{b2}) = mtl(u_{a2})$, che $\gamma(u_{b2}) = \gamma(u_{a2})$ e che $m'_{a2} \simeq_e m'_{b2}$; "ricombinando" u_{b1} e u_{b2} nello stesso modo (sincronizzazione di etichetta per sincronizzazione di etichetta) con cui si sincronizzano u_{a1} e u_{a2} ottenendo u_a, si può costruire un multiset di transizioni u_b tale che $mtl(u_a) = mtl(u_b)$ e $\gamma(u_a) = \gamma(u_b)$; il multiset di transizioni u_b e la marcatura $m'_b = m'_{b1} \oplus m'_{b2}$ sono il multiset e la marcatura cercati; che $m_b[[u_b\rangle\rangle m'_{b1} \oplus m'_{b2} = m'_b$ è evidente (dal momento che le sincronizzazioni che combinano u_{b1} e u_{b2} in u_b non modifica il postset).

Rimane solo da provare che $m'_a \simeq_e m'_b$; sapendo che $m_{a1}[[u_{a1}\rangle\rangle m'_{a1}$, applicando sequenzialmente le singole transizioni (non importa in che ordine) e applicando sequenzialmente a queste transizioni il corollario 4.1.16 si può stabilire che $r_a(m'_{a1}) \subseteq r_a(m_{a1})$. Analogamente, ragionando su $m_{a2}[[u_{a2}\rangle\rangle m'_{a2}$, su $m_{b1}[[u_{b1}\rangle\rangle m'_{b1}$ e su $m_{b2}[[u_{b2}\rangle\rangle m'_{b2}$, si può stabilire che $r_a(m'_{a2}) \subseteq r_a(m_{a2})$, che $r_a(m'_{b1}) \subseteq r_a(m_{b1})$ e che $r_a(m'_{b2}) \subseteq r_a(m_{b2})$; abbiamo quindi, ricordando che – per ipotesi – $r_a(m_{a1}) \cap \overline{r_a(m_{a2})} = \emptyset$ e che $r_a(m_{b1}) \cap \overline{r_a(m_{b2})} = \emptyset$, abbiamo

$$r_a(m'_{a1}) \cap \overline{r_a(m'_{a2})} \subseteq r_a(m_{a1}) \cap \overline{r_a(m_{a2})} = \emptyset$$
$$r_a(m'_{b1}) \cap \overline{r_a(m'_{b2})} \subseteq r_a(m_{b1}) \cap \overline{r_a(m_{b2})} = \emptyset$$

e dal momento che $m'_{a1} \simeq_e m'_{b1}$ e che $m'_{a2} \simeq_e m'_{b2}$, visto che le clausole di non collisione dei nomi ristretti sono soddisfatte, per definizione di dec-equivalenza, abbiamo

$$m'_a = m'_{a1} \oplus m'_{a2} \simeq_e m'_{b1} \oplus m'_{b2} = m'_b$$

□

Data la proposizione appena dimostrata, abbiamo immediatamente il seguente

Corollario 4.2.8. *Siano $m_a, m_b \in \mathcal{M}_{fin}(S_{MCCS})$ due generiche marcature; se $m_a \simeq_e m_b$, per ogni $t_a \in T_{MCCS}$ abbiamo*

$$\begin{aligned} m_a[t_a\rangle m'_a \implies \exists t_b \in T_{MCCS}.\exists m'_b \in \mathcal{M}_{fin}(S_{MCCS}).(m_b[t_b\rangle m'_b \\ \wedge\ l(t_a) = l(t_b) \\ \wedge\ m'_a \simeq_e m'_b) \end{aligned}$$

Inoltre, per ogni $t_b \in T_{MCCS}$ abbiamo

$$\begin{aligned} m_b[t_b\rangle m'_b \implies \exists t_a \in T_{MCCS}.\exists m'_a \in \mathcal{M}_{fin}(S_{MCCS}).(m_a[t_a\rangle m'_a \\ \wedge\ l(t_a) = l(t_b) \\ \wedge\ m'_a \simeq_e m'_b) \end{aligned}$$

Dimostrazione. Ricordando anche le osservazioni 1.2.3 (pag. 15), la tesi è conseguenza immediata della proposizione 4.2.7 (pag. 193), applicata al multiset di transizioni $u_a = 1 \cdot t_a$. □

Osserviamo che non sarebbe stato possibile dimostrare direttamente, per induzione strutturale, il corollario 4.2.8 poiché – nel caso della somma di marcature dec-equivalenti – la transizione t_a (come abbiamo osservato in precedenza) non necessariamente viene "smontata" in una sola transizione supportata dalla prima componente e in una sola transizione supportata dalla seconda.

Possiamo finalmente dimostrare che la dec-equivalenza implica la bisimulazione.

Proposizione 4.2.9. *Siano* $m_1, m_2 \in \mathcal{M}_{fin}(S_{MCCS})$ *due generiche marcature; allora*

$$m_1 \simeq_e m_2 \implies m_1 \sim m_2$$

Dimostrazione. Sia

$$R_{deq} = \{(m_1, m_2) \mid m_1, m_2 \in \mathcal{M}_{fin}(S_{MCCS}) \wedge m_1 \simeq_e m_2\}$$

Date la proposizione 1.3.3 (pag. 24) – che, sostanzialmente, fornisce un criterio alternativo, sganciato dagli LTS, per individuare le bisimulazioni – e il corollario 4.2.8 (pagina precedente), abbiamo che R_{deq} è una bisimulazione e quindi che

$$m_1 \simeq_e m_2 \implies (m_1, m_2) \in R_{dec} \implies m_1 \sim m_2$$

□

4.3 Teorema di correttezza

Dimostriamo, in questa sezione, la "correttezza" della semantica introdotta in precedenza; ovvero dimostreremo che ogni processo Multi-CCS $q \in \mathcal{P}$ – inteso nella sua interpretazione di default, ovvero quella della semantica LTS – è bisimile alla sua immagine tramite *dec*, ovvero a $dec(q, e, i) \in \mathcal{M}_{fin}(S_{MCCS})$, a prescindere dai valori di supporto e ed i.

4.3.1 Replicabilità delle transizioni della semantica LTS

Con la seguente proposizione dimostriamo che la transizione di un generico processo $q \in \mathcal{P}$ può essere – al netto di dec-equivalenza – replicata nel corrispondente sistema P/T.

Proposizione 4.3.1. *Sia* $q \in \mathcal{P}$ *un generico processo Multi-CCS; se, nella semantica LTS,* $q \xrightarrow{\sigma} q'$ *– con* $\sigma \in \mathcal{A}ct^+ \cup \{\tau\}$ *e* $q' \in \mathcal{P}$ *– allora esistono una transizione* $t \in T_q$ *e una marcatura* $m \in \mathcal{M}_{fin}(S_q)$ *tali che* $dec(q, 0, 0)[t\rangle m$*, che* $l(t) = \sigma$ *e che, per ogni* $e, i \in \mathbb{N}$*,* $dec(q', e, i) \simeq_e m$

Dimostrazione. Per prima cosa osserviamo che grazie alla proposizione 4.2.2 (pag. 176) possiamo limitarci a trovare anche una sola coppia di valori $e, i \in \mathbb{N}$ tali che $dec(q', e, i) \simeq_e m$ e, grazie al fatto che la dec-equivalenza gode della proprietà transitiva – come dimostrato dalla proposizione 4.2.1 (pag. 174) –, la proprietà vale per ogni coppia di valori. La transitività della dec-equivalenza verrà comunque usata pesantemente, nel corso di questa dimostrazione.

La proposizione si dimostra per casistica sulle regole di transizione per $q \xrightarrow{\sigma} q'$ e per induzione sulle relative premesse:

- se la transizione è conseguenza dell'assioma (Pref)[6], allora $q = \mu.q'$, per un'opportuna azione $\mu \in \mathcal{A}ct$ con $\sigma = \mu$; dall'assioma (pref) otteniamo $dec(q, 0, 0) = dec(\mu.q', 0, 0) \xrightarrow{\mu} dec(q', 0, 0)$, ovvero l'esistenza di una transizione $t = (dec(q, 0, 0), \mu, dec(q', 0, 0))$; la transizione $t \in T_q$ e la marcatura $m = dec(q', 0, 0) \in \mathcal{M}_{fin}(S_q)$ sono la transizione e la marcatura cercate; abbiamo infatti $m = dec(q', 0, 0)$ e quindi, banalmente, $m \simeq_e dec(q', 0, 0)$;

- se la transizione è conseguenza della regola (S-pref), allora $q = \underline{\mu}.q_1$, per un'opportuna azione $\mu \in \mathcal{A}ct$ e per un opportuno processo $q_1 \in \mathcal{P}$, con $Conc(\mu, \sigma', \sigma)$, dove $q_1 \xrightarrow{\sigma'} q'$; per ipotesi induttiva sulla premessa di (S-pref), ovvero $q_1 \xrightarrow{\sigma'} q'$, abbiamo che esistono una transizione $t' \in T_{q_1}$ e una marcatura $m \in \mathcal{M}_{fin}(S_{q_1})$ tali che $dec(q_1, 0, 0)[t'\rangle m$, con $l(t') = \sigma'$ e $m \simeq_e dec(q', e, i)$, per generici e ed i; dalla regola (s-pref), usando come premessa $dec(q_1, 0, 0)[t'\rangle m$, otteniamo l'esistenza di una transizione $t \in T_q$ tale che $dec(q, 0, 0) = dec(\underline{\mu}.q_1, 0, 0)[t\rangle m$, con $conc(\mu, \sigma', l(t))$ ovvero, per quanto visto in precedenza e tenendo conto che le regole di concatenazione *Conc* di tabella 2.3 (pag. 39) (per la semantica LTS) sono identiche alle *conc* di tabella 3.4 (pag. 93) (per la semantica P/T), $l(t) = \sigma$; la transizione $t \in T_q$ e la marcatura $m \in \mathcal{M}_{fin}(S_q)$ sono quindi la transizione e la marcatura cercate;

[6]ovvero: se (Pref) è l'ultima regola usata per giustificare la transizione

- se la transizione è conseguenza della regola (Sum-1), allora $q = p_1 + p_2$, per opportuni processi sequenziali $p_1, p_2 \in \mathcal{P}_{seq}$, e $p_1 \stackrel{\sigma}{\longrightarrow} q'$; dall'ipotesi induttiva sulla premessa di (Sum-1), ovvero $p_1 \stackrel{\sigma}{\longrightarrow} q'$, otteniamo l'esistenza di una transizione $t_1 \in T_{p_1}$ tale che $dec(p_1, 0, 0)[t_1\rangle m_1$, con $l(t_1) = \sigma$ e $m_1 \simeq_e dec(q', e, i)$, per generici e ed i; dal momento p_1 è un processo sequenziale e che, banalmente, $p_1 \equiv p_1$, abbiamo, dalla definizione di dec-equivalenza, che $dec(p_1, 0, 0) \simeq_e dec(p_1, 1, 0)$; applicando il corollario 4.2.8 (pag. 203) a quest'ultima dec-equivalenza e a $dec(p_1, 0, 0)[t_1\rangle m_1$ otteniamo che esistono una transizione t' e una marcatura m tali che $dec(p_1, 1, 0)[t'\rangle m$, con $l(t') = l(t_1) = \sigma$ e con $m \simeq_e m_1 \simeq_e dec(q', e, i)$; usando la precedente come premessa della regola (sum-1), otteniamo l'esistenza di una transizione $t \in T_q$ tale che $dec(q, 0, 0) = dec(p_1 + p_2, 0, 0)[t\rangle m$, con $l(t) = l(t') = \sigma$; la transizione t e la marcatura $m \in \mathcal{M}_{fin}(S_q)$ sono la transizione e la marcatura cercate;

- se la transizione è conseguenza della regola (Sum-2), il ragionamento è analogo a quello del caso precedente ma applicando le ipotesi induttive a $p_2 \stackrel{\sigma}{\longrightarrow} q'$ e a $dec(p_2, 1, 1)$ anziché a $p_1 \stackrel{\sigma}{\longrightarrow} q'$ e a $dec(p_1, 1, 0)$;

- se la transizione è conseguenza della regola (Par-1), allora $q = q_1 \mid q_2$, per opportuni processi $q_1, q_2 \in \mathcal{P}$, e $q_1 \stackrel{\sigma}{\longrightarrow} q_1'$, per un opportuno processo $q_1' \in \mathcal{P}$ tale che $q' = q_1' \mid q_2$; per ipotesi induttiva sulla premessa di (Par-1), ovvero $q_1 \stackrel{\sigma}{\longrightarrow} q_1'$, abbiamo che esistono una transizione $t' \in T_{q_1}$ e una marcatura $m_1' \in \mathcal{M}_{fin}(S_{q_1})$ tali che $dec(q_1, 0, 0)[t'\rangle m_1'$, con $l(t') = \sigma$ e $m_1' \simeq_e dec(q_1', e, i)$, per generici e ed i e, in particolare, $m_1' \simeq_e dec(q_1', 1, 0)$; dalla proposizione 4.2.2 (pag. 176) abbiamo che $dec(q_1, 0, 0) \simeq_e dec(q_1, 1, 0)$; applicando il corollario 4.2.8 a quest'ultima dec-equivalenza e a $dec(q_1, 0, 0)[t'\rangle m_1'$ otteniamo che esistono una transizione t e una marcatura m_1 tali che $dec(q_1, 1, 0)[t\rangle m_1$, con $l(t) = l(t') = \sigma$ e con $m_1 \simeq_e m_1' \simeq_e dec(q_1', 1, 0)$; la transizione $t \in T_q$ e la marcatura $m = m_1 \oplus dec(q_2, 1, 1) \in \mathcal{M}_{fin}(S_q)$ sono la transizione e la marcatura cercate, infatti $dec(q_1, 1, 0)[t\rangle m_1$ implica $dec(q, 0, 0) = dec(q_1, 1, 0) \oplus dec(q_2, 1, 1)[t\rangle m_1 \oplus dec(q_2, 1, 1) = m$; rimane da provare che $m \simeq_e dec(q', 0, 0)$; a questo fine osserviamo che il corollario 4.1.16 (pag. 148), tenendo conto del fatto che $dec(q_1, 1, 0)[t\rangle m_1$, ci dice che $r_a(m_1) \subseteq r_a(dec(q_1, 1, 0))$ e, applicando il corollario 4.1.15 (pag. 147) a $dec(q_1 \mid q_2, 0, 0)$, otteniamo

$$r_a(m_1) \cap \overline{r_a(dec(q_2, 1, 1))} \subseteq r_a(dec(q_1, 1, 0)) \cap \overline{r_a(dec(q_2, 1, 1))} = \emptyset$$

Applicando invece il corollario 4.1.15 a $dec(q_1' \mid q_2, 0, 0)$, otteniamo

$$r_a(dec(q_1', 1, 0)) \cap \overline{r_a(dec(q_2, 1, 1))} = \emptyset$$

Avendo che $m_1 \simeq_e dec(q_1', 1, 0)$ (visto poc'anzi), che $dec(q_2, 1, 1) \simeq_e dec(q_2, 1, 1)$ (ovvio dall'uguaglianza) e avendo visto che le condizioni di non collisione sui nomi ristretti sono soddisfatte, dalla definizione di dec-equivalenza abbiamo che

$$\begin{aligned} m &= (m_1 \oplus dec(q_2, 1, 1)) \simeq_e (dec(q_1', 1, 0) \oplus dec(q_2, 1, 1)) \\ &= dec(q_1' \mid q_2, 0, 0) = dec(q', 0, 0) \end{aligned}$$

- se la transizione è conseguenza della regola (Par-2), il ragionamento è analogo a quello del caso precedente ma applicando l'ipotesi induttiva a $q_2 \xrightarrow{\sigma} q_2'$ e a $dec(q_2, 1, 1)$ anziché a $q_1 \xrightarrow{\sigma} q_1'$ e a $dec(q_1, 1, 0)$;

- se la transizione è conseguenza della regola (Com), allora $q = q_1 \mid q_2$, per opportuni processi $q_1, q_2 \in \mathcal{P}$, con $q_1 \xrightarrow{\sigma_1} q_1'$ e $q_2 \xrightarrow{\sigma_2} q_2'$, per opportuni processi $q_1', q_2' \in \mathcal{P}$ e per opportune etichette $\sigma_1, \sigma_2 \in \mathcal{A}$ tali che $Sync(\sigma_1, \sigma_2, \sigma)$ e che $q' = q_1' \mid q_2'$; per induzione sulla prima premessa di (Com), ovvero $q_1 \xrightarrow{\sigma_1} q_1'$, abbiamo che esistono una transizione $t_1' \in T_{q_1}$ e una marcatura $m_1' \in \mathcal{M}_{fin}(S_{q_1})$ tali che $dec(q_1, 0, 0)[t_1'\rangle m_1'$, con $l(t_1') = \sigma_1$ e $m_1' \simeq_e dec(q_1', e, i)$, per generici e ed i e, in particolare, $m_1' \simeq_e dec(q_1', 1, 0)$; dalla proposizione 4.2.2 abbiamo che

$dec(q_1, 0, 0) \simeq_e dec(q_1, 1, 0)$; applicando il corollario 4.2.8 a quest'ultima dec-equivalenza e a $dec(q_1, 0, 0)[t'_1\rangle m'_1$ otteniamo che esistono una transizione t_1 e una marcatura m_1 tali che $dec(q_1, 1, 0)[t_1\rangle m_1$, con $l(t_1) = l(t'_1) = \sigma_1$ e con $m_1 \simeq_e m'_1 \simeq_e dec(q'_1, 1, 0)$; analogamente, partendo dall'ipotesi induttiva sulla seconda premessa di (Com), si prova che esistono una transizione t_2 e una marcatura m_2 tali che $dec(q_2, 1, 1)[t_2\rangle m_2$, con $l(t_2) = \sigma_2$ e con $m_2 \simeq_e dec(q'_2, e, i)$ per e ed i generici e, in particolare, $m_2 \simeq_e dec(q'_2, 1, 1)$; tenendo conto che le regole di sincronizzazione *Sync* per nella semantica LTS in tabella 2.2 (pag. 39) sono sostanzialmente equivalenti alle *sync* per la semantica P/T in tabella 3.3 (pag. 92) (anzi: le prime sono un sottoinsieme delle seconde, non precedendo regole per le azioni ristrette) e che vale $Sync(\sigma_1, \sigma_2, \sigma)$, abbiamo anche che vale $sync(\sigma_1, \sigma_2, \sigma)$; la regola (com), combinando le transizioni t_1 e t_2, ci garantisce quindi che esiste una transizione $t \in T_q$, con $l(t) = \sigma$, tale che $dec(q, 0, 0) = dec(q_1, 1, 0) \oplus dec(q_2, 1, 1)[t\rangle m_1 \oplus m_2$; la transizione t e la marcatura $m = m_1 \oplus m_2 \in \mathcal{M}_{fin}(S_q)$ sono la transizione e la marcatura cercate; rimane solo da provare che $m \simeq_e dec(q', 0, 0)$; a tal fine osserviamo che il corollario 4.1.16, partendo da $dec(q_1, 1, 0)[t_1\rangle m_1$ e da $dec(q_2, 1, 1)[t_2\rangle m_2$, ci dice che $r_a(m_1) \subseteq r_a(dec(q_1, 1, 0))$ e che $r_a(m_2) \subseteq r_a(dec(q_2, 1, 1))$; abbiamo quindi, applicando anche il corollario 4.1.15 a $dec(q_1 \mid q_2, 0, 0)$, che

$$r_a(m_1) \cap \overline{r_a(m_2))} \subseteq r_a(dec(q_1, 1, 0)) \cap \overline{r_a(dec(q_2, 1, 1))} = \emptyset$$

Analogamente, applicando il corollario 4.1.15 a $dec(q'_1 \mid q'_2, 0, 0)$, abbiamo che

$$r_a(dec(q'_1, 1, 0)) \cap \overline{r_a(dec(q'_2, 1, 1))} = \emptyset$$

Dal momento che, come abbiamo visto, $m_1 \simeq_e dec(q'_1, 1, 0)$, che $m_2 \simeq_e dec(q'_2, 1, 1)$ e che abbiamo verificato le condizioni di non collisione dei nomi ristretti, dalla definizione di dec-equivalenza abbiamo che

$$\begin{aligned} m = (m_1 \oplus m_2) &\simeq_e (dec(q_1', 1, 0) \oplus dec(q_2', 1, 1)) \\ &= dec(q_1' \mid q_2', 0, 0) = dec(q', 0, 0) \end{aligned}$$

- se la transizione è conseguenza della regola (Res), allora $q = (\nu a)q_1$, per un opportuno nome $a \in \mathcal{L}$ e per un opportuno processo $q_1 \in \mathcal{P}$, tali che $a, \overline{a} \notin n(\sigma)$ e che $q_1 \xrightarrow{\sigma} q_1'$, per un opportuno processo $q_1' \in \mathcal{P}$ tale che $q' = (\nu a)q_1'$; dall'ipotesi induttiva sulla premessa, ovvero $q_1 \xrightarrow{\sigma} q_1'$, otteniamo che esistono una transizione $t' \in T_{q_1}$ e una marcatura $m' \in \mathcal{M}_{fin}(S_{q_1})$ tali che $dec(q_1, 0, 0)[t'\rangle m'$, con $l(t') = \sigma$ e con $m' \simeq_e dec(q_1', e, i)$, per e ed i generici e, in particolare, $m' \simeq_e dec(q_1', 0, 1)$; a questo punto possiamo distinguere i quattro casi relativi a $\{a, \overline{a}\} \cap fn(q_1)$:
  - quando $a, \overline{a} \in fn(q_1)$, applicando la proposizione 4.2.2 abbiamo che $dec(q_1, 0, 0) \simeq_e dec(q_1, 0, 1)$; applicando il corollario 4.2.8 a quest'ultima dec-equivalenza e a $dec(q_1, 0, 0)[t'\rangle m'$ otteniamo che esistono una transizione $t'' \in T_{MCCS}$ e una marcatura m'' tali che $dec(q_1, 0, 1)[t''\rangle m''$ con $l(t'') = l(t') = \sigma$ – e quindi, per quanto visto prima, con $a, \overline{a} \notin n(l(t'')) = n(\sigma)$ – e con $m'' \simeq_e m' \simeq_e dec(q_1', 0, 1)$; applicando il corollario 4.1.16 a $dec(q_1, 0, 1)[t''\rangle m''$, otteniamo che $r_a(m'') \subseteq r_a(dec(q_1, 0, 1))$; dal momento che $q_1 \in \mathcal{P}$ e $0 < 1$, possiamo applicare il corollario 4.1.18 (pag. 150) e concludere che la sostituzione $\{\eta_0/a\}$ è compatibile con $dec(q_1, 0, 1)$ e quindi, dalla definizione di sostituzione compatibile, che $\eta_0, \overline{\eta_0} \notin r_a(dec(q_1, 0, 1))$ e, ricordando che $r_a(m'') \subseteq r_a(dec(q_1, 0, 1))$, abbiamo banalmente che $\eta_0, \overline{\eta_0} \notin r_a(m'')$ e quindi possiamo concludere che $\{\eta_0/a\}$ è una sostituzione compatibile anche con m''; ricordando anche la proposizione 3.2.5 (pag. 83), abbiamo che $dec((\nu a)q_1, 0, 0) = dec(q_1\{\eta_0/a\}, 0, 1) = dec(q_1, 0, 1)\{\eta_0/a\}$, e quindi, ricordando che $t'' \in T_{MCCS}$ e che $a, \overline{a} \notin n(l(t''))$, possiamo

applicare a $dec(q_1, 0, 1)[t''\rangle m''$ il corollario 4.1.25 (pag. 161) ottenendo che esiste una transizione $t = ({}^\bullet t''\{\eta_0/a\}, l(t''), t''^\bullet\{\eta_0/a\}) \in T_{MCCS}$ tale che

$$dec(q, 0, 0) = dec(q_1, 0, 1)\{\eta_0/a\}[t\rangle m''\{\eta_0/a\}$$

Abbiamo quindi che la transizione $t \in T_q$ e la marcatura $m = m''\{\eta_0/a\} \in \mathcal{M}_{fin}(S_q)$ sono la transizione e la marcatura cercate; riguardo all'etichetta di t abbiamo che $l(t) = l(t'') = l(t') = \sigma$; rimane solo da provare che $m \simeq_e dec(q', 0, 0)$; dal corollario 4.2.3 (pag. 178) abbiamo che esiste un'azione ristretta $\eta \in \mathcal{N}$ tale che $\{\eta/a\}$ è una sostituzione compatibile con $dec(q_1', 0, 1)$ e tale che

$$dec((\nu a)q_1', 0, 0) \simeq_e dec(q_1', 0, 1)\{\eta/a\}$$

Dal momento che sappiamo che $\{\eta_0/a\}$ è una sostituzione compatibile con m'' e che $\{\eta/a\}$ è una sostituzione compatibile con $dec(q_1', 0, 1)$, da $m'' \simeq_e dec(q_1', 0, 1)$ e dalla definizione di dec-equivalenza abbiamo che

$$m''\{\eta_0/a\} \simeq_e dec(q_1', 0, 1)\{\eta/a\}$$

Combinando quanto visto possiamo concludere che

$$\begin{aligned} m = m''\{\eta_0/a\} &\simeq_e dec(q_1', 0, 1)\{\eta/a\} \simeq_e dec((\nu a)q_1', 0, 0) \\ &= dec(q', 0, 0) \end{aligned}$$

– quando $a \in fn(q_1) \wedge \overline{a} \notin fn(q_1)$, ricordando anche la proposizione 3.2.5 abbiamo che $dec((\nu a)q_1, 0, 0) = dec(q_1\{\eta_d/a\}, 0, 0) = dec(q_1, 0, 0)\{\eta_d/a\}$; ricordando che $t' \in T_{q_1} \subseteq T_{MCCS}$ e che $a, \overline{a} \notin n(\sigma) = n(l(t'))$, possiamo applicare a $dec(q_1, 0, 0)[t'\rangle m'$ il corollario 4.1.25 ottenendo che esiste una transizione $t = ({}^\bullet t'\{\eta_d/a\}, l(t'), t'^\bullet\{\eta_d/a\}) \in T_{MCCS}$ tale che

$$dec(q, 0, 0) = dec(q_1, 0, 0)\{\eta_d/a\}[t\rangle m'\{\eta_d/a\}$$

La transizione $t \in T_q$ e la marcatura $m = m'\{\eta_d/a\} \in \mathcal{M}_{fin}(S_q)$ sono la transizione e la marcatura cercata; riguardo all'etichetta di t abbiamo $l(t) = l(t') = \sigma$; rimane solo da provare che $m \simeq_e dec(q', 0, 0)$; dal corollario 4.2.3 abbiamo che esiste un'azione ristretta $\eta \in \mathcal{N}$ tale che $\{\eta/a\}$ è una sostituzione compatibile con $dec(q_1', 0, 1)$ e tale che

$$dec((\nu a)q_1', 0, 0) \simeq_e dec(q_1', 0, 1)\{\eta/a\}$$

Da $\overline{a} \notin fn(q_1)$ e dalla proposizione 4.1.9 (pag. 130) otteniamo che $\overline{a} \notin n(dec(q_1, 0, 0))$; applicando a $dec(q_1, 0, 0)[t'\rangle m'$ la proposizione 4.1.13 otteniamo che $n(m') \subseteq n(dec(q_1, 0, 0))$ e quindi, da $\overline{a} \notin n(dec(q_1, 0, 0))$ otteniamo, banalmente, $\overline{a} \notin n(m')$ ovvero, dalla definizione di sostituzione compatibile, che $\{\eta_d/a\}$ è una sostituzione compatibile con m'; dal fatto che $\{\eta/a\}$ è una sostituzione compatibile con $dec(q_1', 0, 1)$, dal fatto che $\{\eta_d/a\}$ è una sostituzione compatibile con m', dal fatto che $m' \simeq_e dec(q_1', 0, 1)$ e dalla definizione di dec-equivalenza abbiamo che

$$m'\{\eta_d/a\} \simeq_e dec(q_1', 0, 1)\{\eta/a\}$$

Combinando quanto visto possiamo concludere che

$$\begin{aligned} m = m'\{\eta_d/a\} &\simeq_e dec(q_1', 0, 1)\{\eta/a\} \simeq_e dec((\nu a)q_1', 0, 0) \\ &= dec(q', 0, 0) \end{aligned}$$

- quando $a \notin fn(q_1) \wedge \overline{a} \in fn(q_1)$, la prova è del tutto analoga a quella del sottocaso precedente, con l'unica differenza che η_c, e non η_d, sostituisce a in q_1;
- quando $a, \overline{a} \notin fn(q_1)$, il caso è degenere; abbiamo $q = (\nu a)q_1 = q_1 \xrightarrow{\sigma} q_1' = q'$; quindi

$$dec(q, 0, 0) = dec(q_1, 0, 0)[t'\rangle m'$$

La transizione $t = t' \in T_q$ e la marcatura $m = m' \in \mathcal{M}_{fin}(S_q)$ sono la transizione e la marcatura cercate; riguardo all'etichetta di t abbiamo $l(t) = l(t') = \sigma$; riguardo alla dec-equivalenza, ricordando che avevamo $m' \simeq_e dec(q_1', e, i)$, con e ed i generici, e quindi abbiamo

$$m = m' \simeq_e dec(q_1', e, i) = dec(q', e, i)$$

- se la regola (Cong) è l'ultima regola usata per giustificare la transizione, allora $q \equiv q_1$, $q_1 \xrightarrow{\sigma} q_1'$ e $q_1' \equiv q'$, per opportuni processi $q_1, q_1' \in \mathcal{P}$; per ipotesi induttiva sulla premessa di (Cong), ovvero $q_1 \xrightarrow{\sigma} q_1'$, abbiamo che esistono una transizione $t' \in T_{q_1}$ e una marcatura $m' \in \mathcal{M}_{fin}(S_{q_1})$ tali che $dec(q_1, 0, 0)[t'\rangle m'$, con $l(t') = \sigma$ e $m' \simeq_e dec(q_1', e, i)$, con e ed i generici; applicando a $q \equiv q_1$ e a $q_1' \equiv q'$ il lemma 4.2.4 (pag. 181) otteniamo che $dec(q, 0, 0) \simeq_e dec(q_1, 0, 0)$ e che $dec(q_1', e, i) \simeq_e dec(q', e, i)$; applicando il corollario 4.2.8 a $dec(q_1, 0, 0) \simeq_e dec(q, 0, 0)$ e a $dec(q_1, 0, 0)[t'\rangle m'$ otteniamo che esistono una transizione $t \in T_q$ e una marcatura $m \in \mathcal{M}_{fin}(S_q)$ tali che $dec(q, 0, 0)[t\rangle m$, con $l(t) = l(t') = \sigma$ e con $m \simeq_e m'$; la transizione t e la marcatura m sono la transizione e la marcatura cercate; riguardo alla dec-equivalenza di m abbiamo infatti

$$m \simeq_e m' \simeq_e dec(q_1', e, i) \simeq_e dec(q', e, i)$$

$\square$

4.3.2 Replicabilità delle transizioni della semantica P/T

Ora intendiamo dimostrare che a una transizione supportata dal generico processo q nella semantica P/T corrisponde – sempre al netto della dec-equivalenza – una transizione dello stesso processo proiettato nella semantica LTS.

Purtroppo una dimostrazione sulla singola transizione – per induzione sulle premesse, come fatto per la proposizione 4.3.1 (pag. 204) – fallirebbe perché, come evidenziato nella sottosezione 4.2.5, in caso di transizione giustificata dalla regola (com), le due transizioni della premessa non necessariamente graverebbero (ognuna) su una delle componenti del processo.

Possiamo comunque, così come fatto per il corollario 4.2.8 (pag. 203), dimostrare una proposizione simile sui multiset di transizioni – quindi un risultato "più forte", sul quale vale il ragionamento induttivo – che implica il più modesto risultato che ci interessa.

Quindi, per prima cosa, dimostriamo la seguente

Proposizione 4.3.2. *Dato un generico processo Multi-CCS $q \in \mathcal{P}$, se esistono un multiset di transizioni $u \in \mathcal{M}_{fin}(T_q)$ e una marcatura $m \in \mathcal{M}_{fin}(S_q)$ tali che $dec(q, 0, 0)[[u\rangle\rangle m$, allora esiste un processo $q' \in \mathcal{P}$ tale che, nella semantica LTS, $q[[mtl(u)\rangle\rangle q'$ e tale che, per ogni coppia di numeri interi $e, i \in \mathbb{N}$, $dec(q', e, i) \simeq_e m$.*

Dimostrazione. Per prima cosa osserviamo che grazie alla proposizione 4.2.2 (pag. 176) possiamo limitarci a trovare anche una sola coppia di valori $e, i \in \mathbb{N}$ tali che $dec(q', e, i) \simeq_e m$ e, grazie al fatto che la dec-equivalenza gode della proprietà transitiva – come dimostrato dalla proposizione 4.2.1 (pag. 174) –, la proprietà vale per ogni coppia di valori. La transitività della dec-equivalenza verrà comunque usata pesantemente, nel corso di questa dimostrazione.

Dimostriamo la prima implicazione, in primo luogo, per induzione numerica forte su $\gamma(u)$.

Se $\gamma(u) = 0$, è evidente che $u = \emptyset$, che $mtl(u) = mtl(\emptyset) = \emptyset$ e che $m = dec(q, 0, 0)$; è sufficiente scegliere $q' = q$ ottenendo, per definizione, che $q[[\emptyset\rangle\rangle q = q'$; dal momento che $dec(q', 0, 0) = dec(q, 0, 0) = m$, dall'uguaglianza abbiamo banalmente che $dec(q', 0, 0) \simeq_e m$.

Se $\gamma(u) = 1$, dal momento che tutte le transizioni giustificate dalle altre regole di tabella 3.2 (pag. 91) hanno un peso maggiore di 1, abbiamo necessariamente che $u = 1 \cdot t$, dove $t \in T_{MCCS}$ è giustificata dalla regola (pref). Dimostriamo questo caso per induzione strutturale su q:

- non può essere $q = \mathbf{0}$ poiché $dec(q, 0, 0) = dec(\mathbf{0}, 0, 0) = \emptyset$ non è in grado di supportare nessuna transizione;

- se $q = \mu.q'$, per un'opportuna azione $\mu \in \mathcal{A}ct$ e per un opportuno processo $q' \in \mathcal{P}$, l'ispezione delle regole di tabella 3.2 ci dice che $dec(\mu.q', 0, 0)$ supporta (solo) una transizione t, di peso pari a 1 in quanto giustificata dalla regola (pref), verso $m = dec(q', 0, 0)$ e con etichetta $l(t) = \mu$; la corrispondente regola (Pref) giustifica $q = \mu.q' \xrightarrow{\mu} q'$, ovvero – per definizione – $q[[1 \cdot \mu\rangle\rangle q'$; q' è il processo cercato; infatti $mtl(u) = mtl(1 \cdot t) = 1 \cdot l(t) = 1 \cdot \mu$ e, dal momento che $dec(q', 0, 0) = m$, dall'uguaglianza abbiamo banalmente che $dec(q', 0, 0) \simeq_e m$;

- non può essere $q = \underline{\mu}.q'$, per un'opportuna azione $\mu \in \mathcal{A}ct$ e un opportuno processo $q' \in \mathcal{P}$, poiché l'ispezione delle regole di tabella 3.2 ci dice che $dec(\underline{\mu}.q', 0, 0)$ non è in grado di supportare una transizione con la regola (pref), ovvero con peso 1;

- analogamente al caso precedente, e per lo stesso motivo, non può neppure essere $q = p_1 + p_2$, per opportuni processi sequenziali $p_1, p_2 \in \mathcal{P}_{seq}$;

- se $q = q_1 \mid q_2$, per opportuni processi $q_1, q_2 \in \mathcal{P}$, abbiamo che $dec(q, 0, 0) = dec(q_1 \mid q_2, 0, 0) = dec(q_1, 1, 0) \oplus dec(q_2, 1, 1)$; dal momento che la transizione t è giustificata – come abbiamo detto – dalla regola (pref) e quindi sostenuta da un singoletto, u (ovvero t) è interamente supportato o da $dec(q_1, 1, 0)$ o da $dec(q_2, 1, 1)$; vediamo quindi i due casi:

  – quando $dec(q_1, 1, 0)[[u\rangle\rangle m_1$, con $m = m_1 \oplus dec(q_2, 1, 1)$ – ovvero: quando u è supportato dalla sola prima componente di $dec(q, 0, 0)$ – dalla proposizione 4.2.2 (pag. 176) abbiamo che $dec(q_1, 1, 0) \simeq_e dec(q_1, 0, 0)$; applicando a quest'ultima dec-equivalenza e a $dec(q_1, 1, 0)[[u\rangle\rangle m_1$ la proposizione 4.2.7 (pag. 193), otteniamo che esistono un multiset di transizioni $u' \in \mathcal{M}_{fin}(T_{MCCS})$ e una marcatura m'_1 tali che $dec(q_1, 0, 0)[[u'\rangle\rangle m'_1$ (e quindi anche $u' \in \mathcal{M}_{fin}(T_{q_1})$), che $mtl(u') = mtl(u) = 1 \cdot l(t)$, che $\gamma(u') = \gamma(u) = 1$ e che $m'_1 \simeq_e m_1$; dal momento che $\gamma(u') = 1$, possiamo applicare l'induzione strutturale al processo q_1 e al multiset di transizioni u' ottenendo che esiste un processo $q'_1 \in \mathcal{P}$ tale che $q_1[[mtl(u')\rangle\rangle q'_1$ e che $dec(q'_1, e, i) \simeq_e m'_1 \simeq_e m_1$, per e ed i generici e, in particolare, $dec(q'_1, 1, 0) \simeq_e m_1$; applicando a $dec(q_1, 1, 0)[[u\rangle\rangle m_1$, ovvero a $dec(q_1, 1, 0)[t\rangle m_1$, il corollario 4.1.16 (pag. 148) otteniamo che $r_a(m_1) \subseteq r_a(dec(q_1, 1, 0))$ e quindi, applicando anche il corollario 4.1.15 (pag. 147) a $dec(q_1 \mid q_2, 0, 0)$, abbiamo

$$r_a(m_1) \cap \overline{r_a(dec(q_2, 1, 1)} \subseteq r_a(dec(q_1, 1, 0)) \cap \overline{r_a(dec(q_2, 1, 1)} = \emptyset$$

Applicando il corollario 4.1.15 anche a $dec(q'_1 \mid q_2, 0, 0)$ abbiamo anche che

$$r_a(dec(q'_1, 1, 0) \cap \overline{r_a(dec(q_2, 1, 1)} = \emptyset$$

Avendo che $dec(q'_1, 1, 0) \simeq_e m_1$ (dimostrato poc'anzi), che $dec(q_2, 1, 1) \simeq_e dec(q_2, 1, 1)$ (ovvio) e avendo verificato le condizioni di non collisione dei nomi ristretti, dalla definizione di dec-equivalenza abbiamo che

$$dec(q'_1, 1, 0) \oplus dec(q_2, 1, 1) \simeq_e m_1 \oplus dec(q_2, 1, 1)$$

Abbiamo quindi che $q' = q'_1 \mid q_2 \in \mathcal{P}$ è il processo cercato; infatti, ricordando che $q_1[[mtl(u)\rangle\rangle q'_1$, e avendo per definizione che $q_2[[\emptyset\rangle\rangle q_2$, applicando la regola (S-com-s) otteniamo che $q = q_1 \mid q_2[[mtl(u)\rangle\rangle q'_1 \mid q_2 = q'$ e, combinando quanto visto sopra, che

$$dec(q', 0, 0) = dec(q'_1 \mid q_2, 0, 0) = dec(q'_1, 1, 0) \oplus dec(q_2, 1, 1)$$

$$\simeq_e m_1 \oplus dec(q_2, 1, 1) = m$$

- quando $dec(q_2, 1, 1)[[u\rangle m_2$, con $m = dec(q_1, 1, 0) \oplus m_2$ – ovvero: quando la transizione è supportata dalla sola seconda componente di $dec(q, 0, 0)$ – la dimostrazione è del tutto analoga a quella del caso precedente ma operando induttivamente su q_2 anziché su q_1;

- se $q = (\nu a)q_1$, per un opportuno nome $a \in \mathcal{L}$ e per un opportuno processo $q_1 \in \mathcal{P}$, abbiamo quattro casi possibili:

  - quando $a, \overline{a} \in fn(q_1)$, ricordando anche la proposizione 3.2.5 (pag. 83), abbiamo

$$\begin{aligned} dec(q, 0, 0) &= dec((\nu a)q_1, 0, 0) = dec(q_1\{\eta_0/a\}, 0, 1) \\ &= dec(q_1, 0, 1)\{\eta_0/a\} \end{aligned}$$

  Ricordando che $u \in \mathcal{M}_{fin}(T_q) \subseteq \mathcal{M}_{fin}(T_{MCCS})$, con $u = 1 \cdot t$, possiamo applicare a $dec(q_1, 0, 1)\{\eta_0/a\}[[u\rangle\rangle m$ il corollario 4.1.30 (pag. 170) ottenendo che esistono un multiset di transizioni $u_1 \in \mathcal{M}_{fin}(T_{MCCS})$ – pari a $u_1 = 1 \cdot t_1$, dove $t_1 \in T_{MCCS}$ con $t = ({}^\bullet t_1\{\eta_0/a\}, l(t_1), t_1^\bullet\{\eta_0/a\})$, con $a, \overline{a} \notin n(l(t_1)) = n(l(t))$ e con $\gamma(t_1) = \gamma(t) = 1$ – e una marcatura m_1 tali che $dec(q_1, 0, 1)[[u_1\rangle\rangle m_1$ con $m = m_1\{\eta_0/a\}$; dalla proposizione 4.2.2 abbiamo che $dec(q_1, 0, 1) \simeq_e dec(q_1, 0, 0)$ e quindi, applicando a questa dec-equivalenza e a $dec(q_1, 1, 0)[[u_1\rangle\rangle m_1$ la proposizione 4.2.7, otteniamo che esistono un multiset di transizioni $u_1' \in \mathcal{M}_{fin}(T_{MCCS})$ e una marcatura m_1' tali che $dec(q_1, 0, 0)[[u_1'\rangle\rangle m_1'$ (e quindi anche $u_1' \in \mathcal{M}_{fin}(T_{q_1})$), con $mtl(u_1') = mtl(u_1) = 1 \cdot l(t_1) = 1 \cdot l(t) = mtl(u)$ e con $\gamma(u_1') = \gamma(u_1) = 1 \cdot \gamma(t_1) = 1 \cdot 1 = 1$, e che $m_1' \simeq_e m_1$; avendo $\gamma(u_1') = 1$, possiamo applicare l'induzione strutturale a q_1 e a $dec(q_1, 0, 0)[[u_1'\rangle\rangle m_1'$ e ottenere che esiste un processo $q_1' \in \mathcal{P}$ tale che $q_1[[mtl(u_1')\rangle\rangle q_1'$ e che $dec(q_1', e, i) \simeq_e m_1' \simeq_e m_1$, per e ed i generici e, in particolare, $dec(q_1', 0, 1) \simeq_e m_1$;

applicando il corollario 4.2.3 (pag. 178) otteniamo che esiste una sostituzione $\{\eta/a\}$, compatibile con $dec(q_1, 0, 1)$, tale che

$$dec((\nu a)q'_1, 0, 0) \simeq_e dec(q'_1, 0, 1)\{\eta/a\}$$

Applicando a $dec(q_1, 0, 1)[[u_1\rangle\rangle m_1$ – ovvero, ricordando l'osservazione 1.2.3 (pag. 15), a $dec(q_1, 0, 1)[t_1\rangle m_1$ – il corollario 4.1.16 otteniamo che $r_a(m_1) \subseteq r_a(dec(q_1, 0, 1))$; essendo ovviamente vero che $0 < 1$, dal corollario 4.1.18 (pag. 150) abbiamo che $\{\eta_0/a\}$ è una sostituzione compatibile con $dec(q_1, 0, 1)$ ovvero, dalla definizione di sostituzione compatibile, che $\eta_0, \overline{\eta_0} \notin r_a(dec(q_1, 0, 1))$; ricordando che $r_a(m_1) \subseteq r_a(dec(q_1, 0, 1))$, abbiamo ovviamente anche che $\eta_0, \overline{\eta_0} \notin r_a(m_1)$ e quindi, di nuovo dalla relativa definizione, che $\{\eta_0/a\}$ è una sostituzione compatibile anche con m_1; avendo che $dec(q'_1, 0, 1) \simeq_e m_1$, che $\{\eta/a\}$ è una sostituzione compatibile con $dec(q'_1, 0, 1)$ e che $\{\eta_0/a\}$ è una sostituzione compatibile con m_1, dalla definizione di dec-equivalenza abbiamo che

$$dec(q'_1, 0, 1)\{\eta/a\} \simeq_e m_1\{\eta_0/a\}$$

Il processo $q' = (\nu a)q'_1 \in \mathcal{P}$ è il processo cercato; infatti – avendo $mtl(u'_1) = mtl(u_1) = 1 \cdot l(t_1) = 1 \cdot l(t) = mtl(u)$ e avendo $a, \overline{a} \notin n(l(t))$ – possiamo applicare la regola (Res-s) a $q_1[[mtl(u'_1)\rangle\rangle q'_1$ ottenendo $q = (\nu a)q_1[[mtl(u)\rangle\rangle(\nu a)q'_1 = q'$; inoltre, riassumendo quanto visto fin'ora,

$$\begin{aligned} dec(q', 0, 0) = dec((\nu a)q'_1, 0, 0) &\simeq_e dec(q'_1, 0, 1)\{\eta/a\} \\ &\simeq_e m_1\{\eta_0/a\} = m \end{aligned}$$

– quando $a \in fn(q_1) \wedge \overline{a} \notin fn(q_1)$, ricordando anche la proposizione 3.2.5, abbiamo

$$\begin{aligned} dec(q, 0, 0) = dec((\nu a)q_1, 0, 0) &= dec(q_1\{\eta_d/a\}, 0, 0) \\ &= dec(q_1, 0, 0)\{\eta_d/a\} \end{aligned}$$

Ricordando che $u \in \mathcal{M}_{fin}(T_q) \subseteq \mathcal{M}_{fin}(T_{MCCS})$, con $u = 1 \cdot t$, possiamo applicare a $dec(q_1, 0, 0)\{\eta_d/a\}[[u\rangle\rangle m$ il corollario 4.1.30 ottenendo che esistono un multiset di transizioni $u_1 \in \mathcal{M}_{fin}(T_{MCCS})$ (ma anche $u_1 \in \mathcal{M}_{fin}(T_{q_1})$, visto che è supportato da $dec(q_1, 0, 0)$) – pari a $u_1 = 1 \cdot t_1$, dove $t_1 \in T_{MCCS}$ con $t = ({}^\bullet t_1\{\eta_d/a\}, l(t_1), t_1^\bullet\{\eta_d/a\})$, con $a, \overline{a} \notin n(l(t_1)) = n(l(t))$ e con $\gamma(t_1) = \gamma(t) = 1$ – e una marcatura m_1 tali che $dec(q_1, 0, 0)[[u_1\rangle\rangle m_1$ con $m = m_1\{\eta_d/a\}$; avendo $\gamma(u_1) = 1 \cdot \gamma(t_1) = 1 \cdot \gamma(t) = 1 \cdot 1 = 1$, possiamo applicare l'induzione strutturale a q_1 e a $dec(q_1, 0, 0)[[u_1\rangle\rangle m_1$ e ottenere che esiste un processo $q_1' \in \mathcal{P}$ tale che $q_1[[mtl(u_1)\rangle\rangle q_1'$ e che $dec(q_1', e, i) \simeq_e m_1$, per e ed i generici e, in particolare, $dec(q_1', 0, 1) \simeq_e m_1$; applicando il corollario 4.2.3 otteniamo che esiste una sostituzione $\{\eta/a\}$, compatibile con $dec(q_1, 0, 1)$, tale che

$$dec((\nu a)q_1', 0, 0) \simeq_e dec(q_1', 0, 1)\{\eta/a\}$$

Applicando a $dec(q_1, 0, 0)[[u_1\rangle\rangle m_1$ – ovvero, ricordando l'osservazione 1.2.3, a $dec(q_1, 0, 0)[t_1\rangle m_1$ – la proposizione 4.1.13 otteniamo che $n(m_1) \subseteq n(dec(q_1, 0, 0))$; essendo vero (per ipotesi di questo sottocaso) che $\overline{a} \notin fn(q_1)$, dalla proposizione 4.1.9 (pag. 130) abbiamo anche che $\overline{a} \notin n(dec(q_1, 0, 0))$ e quindi, per quanto appena visto, anche che $\overline{a} \notin n(m_1)$ ovvero, dalla definizione, che $\{\eta_d/a\}$ è una sostituzione compatibile con m_1; avendo che $dec(q_1', 0, 1) \simeq_e m_1$, che $\{\eta/a\}$ è una sostituzione compatibile con $dec(q_1', 0, 1)$ e che $\{\eta_d/a\}$ è una sostituzione compatibile con m_1, dalla definizione di dec-equivalenza abbiamo che

$$dec(q_1', 0, 1)\{\eta/a\} \simeq_e m_1\{\eta_d/a\}$$

Il processo $q' = (\nu a)q_1' \in \mathcal{P}$ è il processo cercato; infatti – avendo $mtl(u_1) = 1 \cdot l(t_1) = 1 \cdot l(t) = mtl(u)$ e avendo $a, \overline{a} \notin n(l(t))$ – possiamo applicare la regola (Res-s) a $q_1[[mtl(u_1)\rangle\rangle q_1'$ ottenendo

$q = (\nu a)q_1[[mtl(u)\rangle\rangle(\nu a)q'_1 = q'$; inoltre, riassumendo quanto visto fin'ora,

$$dec(q', 0, 0) = dec((\nu a)q'_1, 0, 0) \simeq_e dec(q'_1, 0, 1)\{\eta/a\}$$
$$\simeq_e m_1\{\eta_d/a\} = m$$

– quando $a \notin fn(q_1) \wedge \overline{a} \in fn(q_1)$, la dimostrazione è analoga a quella del sottocaso precedente ma operando con la sostituzione $\{\eta_c/a\}$ invece che con $\{\eta_d/a\}$;

– quando $a, \overline{a} \notin fn(q_1)$, il caso è degenere; abbiamo

$$dec(q, 0, 0) = dec((\nu a)q_1, 0, 0) = dec(q_1, 0, 0)$$

Applicando l'induzione strutturale a $q_1 \in \mathcal{P}$ e a $dec(q, 0, 0) = dec(q_1, 0, 0)[[u\rangle\rangle m$ otteniamo che esiste un processo $q'_1 \in \mathcal{P}$ tale che $q_1[[mtl(u)\rangle\rangle q'_1$ e che $dec(q'_1, e, i) \simeq_e m$ per e ed i generici e, in particolare, $dec(q'_1, 0, 1) \simeq_e m$; dal momento che $a, \overline{a} \notin fn(q_1)$ (ipotesi specifica del sottocaso), dalla proposizione 4.1.9 abbiamo $a, \overline{a} \notin n(dec(q_1, 0, 0))$ ed è quindi evidente che a e $\overline{a}$ non possono far parte dei nomi dell'etichetta di qualunque transizione supportata da $dec(q_1, 0, 0)$ e quindi $a, \overline{a} \notin n(l(t))$; $q' = (\nu a)q'_1 \in \mathcal{P}$ è il processo cercato; infatti da $q_1[[mtl(u)\rangle\rangle q'_1$ e da $a, \overline{a} \notin n(l(t))$, la regola (Res-s) ci permette di concludere che $q = (\nu a)q_1[[mtl(u)\rangle\rangle(\nu a)q'_1 = q'$; rimane da provare che $dec(q', 0, 0) \simeq_e m$; dal corollario 4.2.3 abbiamo che esiste una sostituzione $\{\eta/a\}$ compatibile con $dec(q'_1, 0, 1)$ e tale che

$$dec((\nu a)q'_1, 0, 0) \simeq_e dec(q'_1, 0, 1)\{\eta/a\}$$

Da $dec(q_1, 0, 0)[[u\rangle\rangle m$ – ovvero, ricordando l'osservazione 1.2.3, $dec(q_1, 0, 0)[t\rangle m$ – e dalla proposizione 4.1.13 otteniamo che $n(m) \subseteq n(dec(q_1, 0, 0))$; sappiamo che $a, \overline{a} \notin fn(q_1)$ quindi, dalla proposizione 4.1.9, abbiamo che $a, \overline{a} \notin n(dec(q_1, 0, 0))$ e quindi, per quanto appena visto, abbiamo anche che $a, \overline{a} \notin n(m)$; possiamo quindi dedurre sia che $\{\eta_d/a\}$[7] è una sostituzione compatibile con m, sia che $m = m\{\eta_d/a\}$; avendo che $dec(q_1', 0, 1) \simeq_e m$, che la sostituzione $\{\eta/a\}$ è compatibile con $dec(q_1', 0, 1)$ e che $\{\eta_d/a\}$ è compatibile con m, dalla definizione di dec-equivalenza abbiamo che

$$dec(q_1', 0, 1)\{\eta/a\} \simeq_e m\{\eta_d/a\}$$

e quindi, riassumendo,

$$\begin{aligned} dec(q', 0, 0) = dec((\nu a)q_1', 0, 0) &\simeq_e dec(q_1', 0, 1)\{\eta/a\} \\ &\simeq_e m\{\eta_d/a\} = m \end{aligned}$$

- se $q = B$, con $B \stackrel{def}{=} q_1$ per un opportuno processo $q_1 \in \mathcal{P}$, abbiamo

$$dec(q, 0, 0) = dec(B, 0, 0) = dec(q_1, 0, 0)$$

  Applicando l'induzione strutturale a $q_1 \in \mathcal{P}$ e a $dec(q, 0, 0) = dec(q_1, 0, 0)[[u\rangle\rangle m$, otteniamo che esiste un processo $q' \in \mathcal{P}$ tale che $q_1[[mtl(u)\rangle\rangle q'$ e che $dec(q', e, i) \simeq_e m$ per e ed i generici; q' è il processo cercato; infatti da $q_1[[mtl(u)\rangle\rangle q'$, la regola (Cong-s) – applicata a $B \equiv q_1$ – ci permette di concludere che $q = B[[mtl(u)\rangle\rangle q'$ e, come abbiamo visto,

$$dec(q', e, i) \simeq_e m$$

Supponiamo ora che sia verificata la tesi per generici multiset di transizioni con peso inferiore a n – e supponendo anche che sia $n \geqslant 2$, dal momento che abbiamo dimostrato la tesi per $n = 0$ e per $n = 1$ – dimostriamo per induzione strutturale su q che la tesi è verificata anche quando $\gamma(u) = n$; dal momento che alcuni casi sono molto simili (o persino identici) a quelli già trattati nel caso $\gamma(u) = 1$, a volte ci limiteremo a ricondurci alla dimostrazione già vista, indicando (se opportuno) eventuali differenze:

[7] più in generale, $\{\eta/a\}$ è una sostituzione compatibile con m, qualunque sia $\eta \in \mathcal{N}$

- anche con $\gamma(u) = n \geqslant 2$ non può essere $q = \mathbf{0}$ poiché $dec(q, 0, 0) = dec(\mathbf{0}, 0, 0) = \emptyset$ non è in grado di supportare nessuna transizione;

- non può essere $q = \mu.q'$, per un'opportuna azione $\mu \in \mathcal{A}ct$ e un opportuno processo $q' \in \mathcal{P}$, poiché l'ispezione delle regole di tabella 3.2 ci dice che $dec(\mu.q', 0, 0)$ supporta (solo) una transizione t, di peso pari a 1 in quanto giustificata dalla regola (pref), mentre ora stiamo supponendo che sia $\gamma(u) = n \geqslant 2$;

- se $q = \underline{\mu}.q_1$, per un'opportuna azione $\mu \in \mathcal{A}ct$ e per un opportuno processo $q_1 \in \mathcal{P}$, dall'ispezione delle regole di tabella 3.2 abbiamo che $dec(\underline{\mu}.q_1, 0, 0)$ è in grado di supportare solo una transizione (alla volta) con la regola (pref-s); se quindi $dec(\underline{\mu}.q_1, 0, 0)[[u\rangle\rangle m$, abbiamo inevitabilmente che $u = 1 \cdot t$ (con $\gamma(u) = 1 \cdot \gamma(t) = n \geqslant 2$) dove $t \in T_{MCCS}$ è giustificata dalla regola (pref-s), dove $t_1 \in \longrightarrow$ è la transizione della premessa (con $n = \gamma(t) = \gamma(t_1) + 1$ ovvero $\gamma(t_1) = n - 1$) con $conc(\mu, l(t_1), l(t))$ e dove – ricordando anche l'osservazione 3.2.6 (pag. 93) – $dec(q_1, 0, 0)[t_1\rangle m$; dal momento che le regole di concatenazione $conc$ non prevedono che le azioni ristrette, eventualmente presenti in $l(t_1)$, possano essere eliminate – ma, al contrario, verrebbero "trascritte" in $l(t)$ – e dal momento che sappiamo, dalle ipotesi, che $t \in T_q \subseteq T_{MCCS}$ e quindi che l'etichetta di t non può contenere azioni ristrette, neppure l'etichetta di t_1 può contenete azioni ristrette e quindi è anche vero che $t_1 \in T_{q_1} \subseteq T_{MCCS}$ e perciò che è possibile applicare a $u_1 = 1 \cdot t_1 \in \mathcal{M}_{fin}(T_{q_1})$, dal momento che $\gamma(u_1) = 1 \cdot \gamma(t_1) = 1 \cdot (n-1) = n - 1 < n$, l'ipotesi induttiva numerica; applicando quindi l'induzione numerica a u_1 e a $dec(q_1, 0, 0)[[u_1\rangle\rangle m$, otteniamo che esiste un processo $q' \in \mathcal{P}$ tale che $q_1[[mtl(u_1)\rangle\rangle q'$ e che $dec(q', e, i) \simeq_e m$, per e ed i generici; ma da $q_1[[mtl(u_1)\rangle\rangle q'$, ossia $q_1[[mtl(1 \cdot t_1\rangle\rangle q'$ e quindi $q_1[[1 \cdot l(t_1)\rangle\rangle q'$, ricordando la proposizione 2.3.1 (pag. 48) abbiamo che $q_1 \xrightarrow{l(t_1)} q'$; q' è il processo cercato; infatti, ricordando che vale $conc(\mu, l(t_1), l(t))$ e tenendo conto che le regole di concatenazione $Conc$ di tabella 2.3 (pag. 39) (per la semantica LTS) sono identiche alle $conc$ di tabella 3.4 (pag. 93) (per la semantica P/T), abbiamo anche che vale $Conc(\mu, l(t_1), l(t))$; possiamo quindi applicare la regola (Pref-s), avendo $q_1 \xrightarrow{l(t_1)} q'$ come premessa, ottenendo che $q = \underline{\mu}.q_1 \xrightarrow{l(t)} q'$, ovvero $q[[mtl(u)\rangle\rangle q'$;

- se $q = p_1 + p_2$, per opportuni processi sequenziali $p_1, p_2 \in \mathcal{P}_{seq}$, un'ispezione delle regole di tabella 3.2 ci mostra che $dec(p_1 + p_2, 0, 0)$ può supportare solo transizioni giustificate dalle regole (sum-1) e (sum-2):

  – quando $u = 1{\cdot}t$ e la transizione t è giustificata dalla regola (sum-1), la premessa della regola ci dice che esiste una transizione $t_1 \in\longrightarrow$ – con $l(t_1) = l(t)$, ovvero (tenendo conto che $t \in T_q \subseteq T_{MCCS}$) anche $t_1 \in T_{MCCS}$, e con $n = \gamma(t) = \gamma(t_1)+1$, ovvero $\gamma(t_1) = n-1$ – tale che $dec(p_1, 1, 0)[t_1\rangle m$, ovvero $dec(p_1, 1, 0)[[u_1\rangle\rangle m$, con $u_1 = 1 \cdot t_1$; dal momento che, banalmente, $p_1 \equiv p_1$ e che p_1 è un processo sequenziale, per definizione di dec-equivalenza sequenziale abbiamo che $dec(p_1, 1, 0) \simeq_e dec(p_1, 0, 0)$; applicando a quest'ultima dec-equivalenza e a $dec(p_1, 1, 0)[[u_1\rangle\rangle m$ la proposizione 4.2.7 (pag. 193), otteniamo che esistono un multiset $u_1' \in \mathcal{M}_{fin}(T_{MCCS})$ e una marcatura m' tali che $dec(p_1, 0, 0)[[u_1'\rangle\rangle m'$ (e quindi anche $u_1' \in \mathcal{M}_{fin}(T_{p_1})$), con $mtl(u_1') = mtl(u_1) = 1 \cdot l(t_1) = 1 \cdot l(t) = mtl(u)$ e con $\gamma(u_1') = \gamma(u_1) = 1 \cdot \gamma(t_1) = 1 \cdot (n-1) = n-1$, e che $m' \simeq_e m$; dal momento che $\gamma(u_1') = n-1 < n$, possiamo applicare l'induzione numerica a p_1 e a u_1' ottenendo che esiste un processo $q' \in \mathcal{P}$ tale che $p_1[[mtl(u_1')\rangle\rangle q'$ e tale che $dec(q', e, i) \simeq_e m'$, per e ed i generici; dalla definizione di supporto contemporaneo di transizioni per processi sequenziali, da $mtl(u_1') = 1 \cdot l(t)$ e da $p_1[[mtl(u_1')\rangle\rangle q'$, otteniamo che $p_1 \xrightarrow{l(t)} q'$ e, considerando quest'ultima transizione come premessa per la regola (Sum-1), otteniamo anche $q = p_1 + p_2 \xrightarrow{l(t)} q'$ e quindi – ricordando che $mtl(u) = 1 \cdot l(t)$ – $q[[mtl(u)\rangle\rangle q'$; q' è quindi il processo cercato e infatti abbiamo anche che $dec(q', e, i) \simeq_e m' \simeq_e m$;

- quando la transizione è giustificata dalla regola (sum-2), la dimostrazione è del tutto analoga a quella del sottocaso precedente ma operando induttivamente su p_2 anziché su p_1;

- se $q = q_1 \mid q_2$, per opportuni processi $q_1, q_2 \in \mathcal{P}$, abbiamo $dec(q, 0, 0) = dec(q_1 \mid q_2, 0, 0) = dec(q_1, 1, 0) \oplus dec(q_2, 1, 1)$; mentre nel caso $\gamma(u) = 1$ avevamo che l'unica transizione era supportata da una sola delle due componenti, questo non è generalmente vero con $\gamma(u) \geqslant 2$; ricordando però l'osservazione 4.2.6 (pag. 192) (e l'algoritmo che la precede e la giustifica) abbiamo che possiamo "suddividere" il multiset di transizioni u in due multiset di transizioni $u_1, u_2 \in \mathcal{M}_{fin}(T_{MCCS})$ tali, per opportune marcature m_1 e m_2, che $dec(q_1, 1, 0)[[u_1\rangle\rangle m_1$, che $dec(q_2, 1, 1)[[u_2\rangle\rangle m_2$, che $m = m_1 \oplus m_2$, che $\gamma(u_1) + \gamma(u_2) \leqslant \gamma(u) = n$ (e quindi $\gamma(u_1) \leqslant n \wedge \gamma(u_2) \leqslant n$) e che u_1 e u_2 possono essere "sincronizzate" (ai sensi della regola (com) e della relativa *sync*) per ottenere u.

  Si osservi che non è escluso che u possa essere supportato dalla sola $dec(q_1, 1, 0)$ (come anche dalla sola $dec(q_2, 1, 1)$); in questo caso avremo che $u_2 = \emptyset$ e che $m_2 = dec(q_2, 1, 1)$.

  Dalla proposizione 4.2.2 abbiamo $dec(q_1, 1, 0) \simeq_e dec(q_1, 0, 0)$; applicando a quest'ultima dec-equivalenza e a $dec(q_1, 1, 0)[[u_1\rangle\rangle m_1$ la proposizione 4.2.7 otteniamo che esistono un multiset di transizioni $u_1' \in \mathcal{M}_{fin}(T_{MCCS})$ e una marcatura m_1' tali che $dec(q_1, 0, 0)[[u_1'\rangle\rangle m_1'$ (e quindi anche $u_1' \in \mathcal{M}_{fin}(T_{q_1})$), che $mtl(u_1') = mtl(u_1)$, che $\gamma(u_1') = \gamma(u_1) \leqslant n$ e che $m_1 \simeq_e m_1'$; applicando l'induzione (numerica quando $\gamma(u_1') < n$, strutturale quando $\gamma(u_1') = n$) a $dec(q_1, 0, 0)[[u_1'\rangle\rangle m_1'$, otteniamo che esiste un processo $q_1' \in \mathcal{P}$ tale che (ricordando che $mtl(u_1') = mtl(u_1)$) $q_1[[mtl(u_1)\rangle\rangle q_1'$ e che $dec(q_1', e, i) \simeq_e m_1' \simeq_e m_1$ per e ed i generici e, in particolare, $dec(q_1', 1, 0) \simeq_e m_1$.

Analogamente, ragionando a partire da $dec(q_2, 1, 1)[[u_2\rangle\rangle m_2$, si può provare che esiste un processo $q'_2 \in \mathcal{P}$ tale che $q_2[[mtl(u_2)\rangle\rangle q'_2$ e, in particolare, $dec(q'_2, 1, 1) \simeq_e m_2$.

Il processo $q' = q'_1 \mid q'_2 \in \mathcal{P}$ è il processo cercato; infatti abbiamo che $q = q_1 \mid q_2[[mtl(u_1) \oplus mtl(u_2)\rangle\rangle q'_1 \mid q'_2 = q'$ e, tenendo conto che le regole di sincronizzazione *Sync* per nella semantica LTS in tabella 2.2 (pag. 39) sono sostanzialmente equivalenti alle *sync* per la semantica P/T in tabella 3.3 (pag. 92) (le prime sono un sottoinsieme delle seconde, non precedendo regole per le azioni ristrette, ma in questo caso sappiamo che non sono coinvolte azioni ristrette nelle sincronizzazioni tra u_1 e u_2), possiamo applicare la proposizione 2.3.2 (pag. 53) quante volte è necessario per replicare (alla rovescia) la scomposizione del multiset u, otteniamo che la somma tra $mtl(u_1)$ e $mtl(u_2)$ può essere sincronizzata per ottenere $mtl(u)$.

Rimane solo da provare che $dec(q', 0, 0) \simeq_e m$.

Partendo da $dec(q_1, 1, 0)[[u_1\rangle\rangle m_1$, e ricordando l'osservazione 1.2.4 (pag. 17), abbiamo che m_1 è ottenibile da $dec(q_1, 1, 0)$ dopo l'attivazione di una qualunque delle sequenze di scatto ottenibili da u_1; applicando quindi n volte (dove n è la lunghezza di queste sequenze di scatto) il corollario 4.1.16 otteniamo che $r_a(m_1) \subseteq r_a(dec(q_1, 1, 0))$. Analogamente, partendo da $dec(q_2, 1, 1)[[u_2\rangle\rangle m_2$ e dalle relative sequenze di scatto ottenibili, è possibile concludere che $r_a(m_2) \subseteq r_a(dec(q_2, 1, 1))$. Applicando quindi anche il corollario 4.1.15 a $dec(q_1 \mid q_2, 0, 0)$, otteniamo

$$r_a(m_1) \cap \overline{r_a(m_2)} \subseteq r_a(dec(q_1, 1, 0)) \cap \overline{r_a(dec(q_2, 1, 1))} = \emptyset$$

Applicando il corollario 4.1.15 anche a $dec(q'_1 \mid q'_2, 0, 0)$ otteniamo

$$r_a(dec(q'_1, 1, 0)) \cap \overline{r_a(dec(q'_2, 1, 1))} = \emptyset$$

Ricordando che $dec(q'_1, 1, 0) \simeq_e m_1$ e che $dec(q'_2, 1, 1) \simeq_e m_2$, e avendo appena dimostrato che le condizioni di non collisione dei nomi ristretti sono soddisfatte, dalla definizione di dec-equivalenza abbiamo

$$dec(q_1', 1, 0) \oplus dec(q_2', 1, 1) \simeq_e m_1 \oplus m_2$$

Riassumendo quando visto fin'ora, abbiamo quindi

$$\begin{aligned} dec(q', 0, 0) = dec(q_1' \mid q_2', 0, 0) &= dec(q_1', 1, 0) \oplus dec(q_2', 1, 1) \\ &\simeq_e m_1 \oplus m_2 \simeq_e m \end{aligned}$$

- se $q = (\nu a)q_1$, per un opportuno nome $a \in \mathcal{L}$ e per un opportuno processo $q_1 \in \mathcal{P}$, la dimostrazione (in tutti e quattro casi possibili) è del tutto analoga a quella del corrispondente caso per $\gamma(u) = 1$; l'unica differenza è che i multiset di transizioni u, u_1 e u_1' non sono (necessariamente) costituiti da un'unica transazione ma (potenzialmente) da una molteplicità di transizioni; i ragionamenti fatti sulle uniche transizioni t, t_1 e t_1' (che non hanno più necessariamente peso pari a 1 ma che mantengono, in corrispondenza, peso uguale) devono quindi essere estesi alla pluralità delle transizioni; in particolare, ragionamenti che permettono di concludere che $r_a(m_1) \subseteq r_a(dec(q_1, 0, 1))$ (primo sottocaso) o che $n(m_1) \subseteq n(dec(q_1, 0, 0))$ (secondo sottocaso) non possono più essere ottenuti dalla semplice applicazione, rispettivamente, del corollario 4.1.16 o della proposizione 4.1.13 ma devono essere ripetuti n volte dove n è la lunghezza delle sequenze di scatto previste dall'osservazione 1.2.4;
- se $q = B$, con $B \stackrel{def}{=} q_1$ per un opportuno processo $q_1 \in \mathcal{P}$, la dimostrazione è identica a quella del corrispondente caso per $\gamma(u) = 1$.

$\square$

Avendo a disposizione la proposizione appena dimostrata, abbiamo banalmente il seguente corollario che esprime quanto desiderato.

Corollario 4.3.3. *Dato un generico processo Multi-CCS $q \in \mathcal{P}$, se esistono una transizione $t \in T_q$, una marcatura $m \in \mathcal{M}_{fin}(S_q)$ e un'etichetta $\sigma \in \mathcal{A}$*

tali che $dec(q, 0, 0)[t\rangle m$, *con* $l(t) = \sigma$, *allora esiste un processo* $q' \in \mathcal{P}$ *tale che* $q \xrightarrow{\sigma} q'$ *e tale che, per ogni coppia di numeri interi* $e, i \in \mathbb{N}$, $dec(q', e, i) \simeq_e m$[8].

Dimostrazione. Dalla proposizione 4.3.2 (pag. 213), applicata al multiset di transizioni $u = 1 \cdot t$, otteniamo che esiste un processo $q' \in \mathcal{P}$ tale che $q[[mtl(u)\rangle\rangle q'$, ovvero $q[[1 \cdot \sigma\rangle\rangle q'$, e che $dec(q', e, i) \simeq_e m$, per e ed i generici.

Ricordando la proposizione 2.3.1 (pag. 48), possiamo concludere che $q \xrightarrow{\sigma} q'$. □

4.3.3 Correttezza

Grazie ai risultati precedenti, possiamo finalmente dimostrare il seguente

Teorema 4.3.4. *Per ogni processo Multi-CCS* $q \in \mathcal{P}$, *è vero che*

$$q \sim dec(q, 0, 0)$$

Dimostrazione. Sia $R \subseteq \mathcal{P} \times \mathcal{M}_{fin}(S_{MCCS})$ la relazione definita nel modo seguente

$$R = \{(q, m) \mid q \in \mathcal{P} \wedge m \in \mathcal{M}_{fin}(S_{MCCS}) \wedge m \simeq_e dec(q, 0, 0)\}$$

Osserviamo che è ovvio che, per ogni $q \in \mathcal{P}$, $(q, dec(q, 0, 0)) \in R$; se riusciamo quindi a dimostrare che R è una bisimulazione, la tesi è provata.

Sia $(q, m) \in R$ un generico elemento di R.

Vediamo, per prima cosa, che la seconda componente riesce sempre a replicare adeguatamente a una mossa della prima; ovvero vediamo che, quando $q \xrightarrow{\sigma} q'$, m è in grado di supportare una transizione tale che $m[t\rangle m'$, con $l(t) = \sigma$ e $(q', m') \in R$.

[8]nella corrispondente proposizione in [7] – la proposition 9 – Gorrieri e Versari hanno dimostrato, grazie alla maggiore flessibilità nella scelta dei nomi ristretti, che la marcature m e $dec(q', e, i)$ (K e $dec(p')$ nell'originale) sono uguali, non solo dec-equivalenti; fortunatamente, per dimostrare la correttezza ci è sufficiente questo risultato più debole

Supponiamo quindi che esista un $q' \in \mathcal{P}$ tale che $q \xrightarrow{\sigma} q'$; dalla proposizione 4.3.1 (pag. 204) abbiamo che esiste una transizione $t' \in T_q$ tale che $dec(q,0,0)[t'\rangle m''$, con $l(t') = \sigma$, e tale che $m'' \simeq_e dec(q',e,i)$, per e ed i generici e, in particolare, $m'' \simeq_e dec(q',0,0)$. Dal momento che, per definizione di R, abbiamo $m \simeq_e dec(q,0,0)$, applicando a $dec(q,0,0)[t'\rangle m''$ e a m il corollario 4.2.8 (pag. 203) otteniamo che esiste una transizione t tale che $m[t\rangle m'$, con $l(t) = l(t') = \sigma$, e che $m' \simeq_e m'' \simeq_e dec(q',0,0)$ ovvero, ricordando che la dec-equivalenza è una relazione di equivalenza e quindi anche transitiva – come da proposizione 4.2.1 (pag. 174) –, $m' \simeq_e dec(q',0,0)$; t è quindi la transizione cercata e $(q',m') \in R$.

Vediamo, per concludere, che la prima componente riesce sempre a replicare adeguatamente a una mossa della seconda; ovvero vediamo che, quando esiste una transizione t tale che $m[t\rangle m'$, esiste anche un processo $q' \in \mathcal{P}$ tale che $q \xrightarrow{\sigma} q'$, con $\sigma = l(t)$, e tale che $(q',m') \in R$.

Supponiamo quindi che $m[t\rangle m'$, con $l(t) = \sigma$; dalla definizione di R abbiamo $m \simeq_e dec(q,0,0)$; applicando a $m[t\rangle m'$ e a $dec(q,0,0)$ il corollario 4.2.8 otteniamo che esistono una transizione $t' \in T_{MCCS}$ e una marcatura m'' tali che $dec(q,0,0)[t'\rangle m''$ (e quindi anche $t' \in T_q$), con $l(t') = l(t) = \sigma$, e che $m'' \simeq_e m'$; applicando a $t' \in T_q$ e a q il corollario 4.3.3 (pag. 225), otteniamo che esiste un processo $q' \in \mathcal{P}$ tale che $q \xrightarrow{\sigma} q'$ e tale che $dec(q',e,i) \simeq_e m'' \simeq_e m'$, per e ed i generici e, in particolare, $dec(q',0,0) \simeq_e m'$; q' è quindi il processo cercato e $(q',m') \in R$.

Concludendo: R è una bisimulazione e per ogni processo $q \in \mathcal{P}$ vale che $q \sim dec(q,0,0)$ □

Utilizzando i risultati dimostrati in precedenza relativi alla dec-equivalenza, possiamo molto semplicemente estendere il risultato della precedente teorema e dimostrare il seguente

Corollario 4.3.5. *Per ogni processo Multi-CCS $q \in \mathcal{P}$ e per ogni coppia di numeri naturali $e, i \in \mathbb{N}$, è vero che*

$$q \sim dec(q,e,i).$$

Dimostrazione. Banale conseguenza del teorema 4.3.4 (pag. 226), della proposizione 4.2.2 (pag. 176), della proposizione 4.2.9 (pag. 204) e della transitività della bisimulazione; infatti dal teorema e dalla prima proposizione abbiamo

$$q \sim dec(q, 0, 0) \simeq_e dec(q, e, i)$$

Applicando anche la seconda proposizione la dec-equivalenza si riduce a bisimulazione, ovvero

$$q \sim dec(q, 0, 0) \sim dec(q, e, i)$$

e sfruttando la transitività della bisimulazione, otteniamo la tesi.

Si osservi anche, sempre alla luce della proposizione 4.2.2, che abbiamo

$$\forall e, i \in \mathbb{N}. \forall q \in \mathcal{P}.(q, dec(q, e, i)) \in R$$

dove R è la relazione usata nella dimostrazione del teorema 4.3.4; dimostrando che R è una bisimulazione, si è dimostrato – come effetto collaterale – anche il seguente corollario. $\square$

Capitolo 5

Processi Multi-CCS per reti P/T finite

Come accennavo nell'introduzione, il Multi-CCS aspira a essere un'algebra di processo in grado di rappresentare reti P/T finite così come il CCS è in grado di rappresentare LTS finiti.

Per completezza e molto rapidamente – dal momento che mi limito, quasi esclusivamente, a riportare quanto sviluppato da Gorrieri e Versari in [7], con le poche modifiche necessarie ad adeguarmi alle varianti che ho introdotto nella semantica P/T e a correggere un piccolo errore che ho rilevato nella versione originaria – riporto un sottoinsieme del Multi-CCS che genera sistemi P/T finiti e descrivo un modo, dato un sistema P/T finito, per costruire un processo – esprimibile con tale sottoinsieme del Multi-CCS – che, tramite *dec*, permette di generare un sistema P/T isomorfo.

5.1 Definizione del sottolinguaggio

I processi Multi-CCS, tramite la semantica che ho descritto nel capitolo 3, non necessariamente ma possono generare sistemi P/T infiniti.

Vediamo ora le possibili cause e come restringere la definizione del Multi-CCS per evitarlo.

5.1.1 Cause della generazione di un sistema P/T infinito

Le cause che possono portare alla generazione di un sistema P/T infinito sono tre:

- la definizione di una costante ricorsiva con generazione, ad ogni iterazione, di uno o più nomi ristretti; ad esempio, il processo $q = B$ – dove B è una costante definita come segue

  $$B \stackrel{def}{=} (\nu a)(a.B \mid \overline{a}.\mathbf{0})$$

  – richiede la generazione di un nome ristretto per sostituire a ad ogni iterazione e, quindi, un'infinità di posti e un'infinità di transizioni;

- la possibilità di usare un'infinità di costanti per definire un processo può portare a infiniti posti e a infinite transizioni come, ad esempio, nel processo $q = a.A_0$ dove abbiamo un'infinità di costanti A_i (con $i \in \mathbb{N}$) definite come segue

  $$A_i \stackrel{def}{=} a.A_{i+1}$$

- le regole di sincronizzazione definite in tabella 3.3 (pag. 92) – che sono sostanzialmente un'espansione (per tener conto delle azioni ristrette) delle regole di tabella 2.2 (pag. 39) – sono molto flessibili e consentono la generazione di infinite transizioni anche in presenza di un'insieme finito di posti; si osservi, ad esempio il processo $q = B$, dove B è una costante definita come segue

  $$B \stackrel{def}{=} \underline{a}.\overline{a}.(B \mid B)$$

  L'applicazione della funzione di decomposizione $dec(q, 0, 0)$ genera un unico posto e la marcatura iniziale $m_0 = 1 \cdot \{\underline{a}.\overline{a}.(B \mid B)\}$; l'applicazione delle regole di transizione di tabella 3.2 (pag. 91) ci mostra che esiste una transizione $t_1 \in T_q$ tale che $m_0 \xrightarrow{a\overline{a}} 2 \cdot m_0$, ovvero la transizione

t_1 consuma 1 token e produce 2 token nell'unico posto disponibile. Osservando che $sync(a\overline{a}, a\overline{a}, a\overline{a})$, abbiamo che è possibile sincronizzare due transizioni t_1 ottenendo una transizione $t_2 \in T_q$ pari a $2 \cdot m_0 \xrightarrow{a\overline{a}} 4 \cdot m_0$. Iterando il ragionamento è possibile dimostrare che l'esistenza di una transizione $t_n \in T_q$, che consuma n token e produce $2n$ token, implica l'esistenza di una transizione $t_{2n} \in T_q$ tale che $2n \cdot m_0 \xrightarrow{a\overline{a}} 4n \cdot m_0$. Il processo descritto supporta quindi un'infinità numerabile $t_{2i} \in T_q$ (con $i \in \mathbb{N}$), con la medesima etichetta e i medesimi posti (anzi: il medesimo posto) nei preset e nei postset ma che si differenziano per il numero di token consumati e prodotti[1].

5.1.2 Multi-CCS a reti finite

Definiamo ora un sottoinsieme del Multi-CCS che non è in grado di generare sistemi P/T infiniti.

Definizioni 5.1.1. *I* processi Multi-CCS a reti finite *sono i processi di tipo r generati dalla seguente grammatica*

$$\begin{array}{llll} p ::= \mathbf{0} & \mid\ \mu.q & \mid\ \underline{\mu}.q & \mid\ p+p \\ q ::= p & \mid\ q \mid q & \mid\ C & \\ r ::= q & \mid\ r \mid r & \mid\ (\nu a)r & \end{array}$$

con i seguenti ulteriori vincoli:

- *le costanti C possono essere definite pari a processi di tipo q (ovvero: non possono contenere restrizioni);*
- *il numero di costanti coinvolte nella definizione di un singolo processo deve essere finito.*

[1]altre transizioni, ovviamente, sono possibili: combinando, ad esempio, la transizione t_2 e la transizione t_4 è possibile ottenere la transizione $t_6 \in T_q$ che consuma $6 = 4+2$ token e produce $12 = 8+4$ token; sviluppando le possibili combinazioni, è possibile ottenere ogni transizione $t_i \in T_q$, con $i \in \mathbb{N}$, che consuma i token e ne produce $2 \cdot i$

La semantica delle reti finite Multi-CCS *è la medesima definita dalle tabelle 2.1 (pag. 37), 2.2 (pag. 39) e 2.3 (pag. 39) con l'ulteriore vincolo sulla regola (Com): Sync*$(\sigma_1, \sigma_2, \sigma)$ *è applicabile solo quando la lunghezza di* σ_1 *o di* σ_2 *è pari a* 1 *(ovvero* $\sigma_1 \in \mathcal{A}ct \vee \sigma_2 \in \mathcal{A}ct$*).*

Osserviamo che l'impossibilità di definire costanti che contengono restrizioni "disinnesca" la prima modalità – di quelle viste in precedenza – per generare sistemi infiniti; l'impossibilità di coinvolgere un numero infinito di costanti nella definizione di un processo disinnesca la seconda modalità mentre il vincolo sulla lunghezza di σ_1 (o di σ_2) per *Sync* disinnesca la terza modalità.

Per una dimostrazione formale del fatto che il sottolinguaggio appena definito permette solamente di costruire termini che generano sistemi P/T finiti, rimando a [7][2] in quanto le modifiche che ho assorbito (alla regola di sincronizzazione *Sync*) o introdotto (alle regole di congruenza strutturale) non ne cambiano la sostanza.

5.2 Generazione di un processo da un sistema P/T finito

Vediamo ora come generare un processo Multi-CCS, secondo la grammatica e compatibile con la semantica introdotte dalle definizioni 5.1.1 (pagina precedente), partendo da un sistema P/T finito, in maniera tale che il sistema P/T generato dalla funzione *dec*, applicata a tale processo, generi un sistema P/T isomorfo a quello di partenza.

5.2.1 Idea di fondo e definizione

Prima di dare una definizione formale, vediamo l'idea di fondo su cui si basa la creazione di un processo da parte di un sistema P/T.

[2] Teorema 3 (finiteness); dimostrazione in appendice C

Analogamente a quanto fece Milner in [8] tra CCS e LTS, l'idea di fondo è quella di associare una costante di processo ad ogni posto del sistema. Anche in questo caso, le costanti vengono definite come somma di processi che consentono le transizioni che devono essere supportate dai posti corrispondenti.

Nel nostro caso abbiamo però delle complicazioni:

- la funzione *dec* non si ferma alle costanti ma accede alla loro definizione; è quindi necessario impedire che a costanti diverse corrispondano posti uguali;

- alle transizioni possono corrispondere più token nel preset; è quindi necessario imporre la sincronizzazione di questi token per l'attivazione delle transizioni;

- alle transizioni possono corrispondere più token nel postset; è quindi necessario che la destinazione di una transizione non sia necessariamente una costante ma che possa essere la composizione parallela di più costanti.

Prima di procedere con l'opportuna definizione, osservo che supporremo che il sistema P/T finito di partenza sia costituito da un insieme $S = \{s_1, s_2, \ldots, s_n\}$ di posti, ovvero che i posti siano numerati da 1 a n; è un'assunzione di comodo che però non riduce in nessun modo la generalità. Analogamente supporremo che l'insieme delle transizioni $T = \{t_1, t_2, \ldots, t_k\}$ sia numerato e ordinato; anche questo: senza perdita di generalità.

Definizioni 5.2.1. *Supponiamo che $N(m_0) = (S, A, T, m_0)$ sia un sistema P/T finito, ovvero costituito da un insieme finito di n posti e un insieme finito di k transizioni. Supponiamo che sia $S = \{s_1, s_2, \ldots, s_n\}$ e che sia $T = \{t_1, t_2, \ldots, t_k\}$.*

La funzione $INet$ associa processi Multi-CCS per reti finite ai sistemi P/T finiti; è definita come segue:

$$INet(N(m_0)) = (\nu y_1 y_2 \ldots y_n x_1^1 x_1^2 \ldots x_1^k x_2^1 x_2^2 \ldots x_2^k \ldots x_n^1 x_n^2 \ldots x_n^k)$$

$$(\underbrace{C_1 \mid \ldots \mid C_1}_{m_0(s_1)\ volte} \mid \ldots \mid \underbrace{C_n \mid \ldots \mid C_n}_{m_0(s_n)\ volte})$$

dove gli x_i^j *e gli* y_i *(con* $i \in [1 \ldots n]$ *e con* $j \in [1 \ldots k]$*) sono nomi nuovi, ovvero*

$$\{x_i^j, y_i \mid i \in [1 \ldots n] \wedge j \in [1 \ldots k]\} \cap (A \cup \overline{A}) = \emptyset$$

dove le costanti di processo C_i *(con* $i = [1 \ldots n]$*) sono definite come segue*

$$C_i \stackrel{def}{=} y_i.\overline{y}_i.\mathbf{0} + c_i^1 + c_i^2 + \ldots + c_i^k$$

dove i termini c_i^j *(con* $i \in [1 \ldots n]$ *e con* $j \in [1 \ldots k]$*), che rappresentano le iterazioni tra i posti e le transizioni*[3]*, sono definiti in base alla seguente casistica:*

- *se* $1 \cdot s_i \not\subseteq {}^\bullet t_j$ *(ovvero se* s_i *non è nel preset della transizione* t_j*), allora*

$$c_i^j = \mathbf{0}$$

- *altrimenti, se* $1 \cdot s_i = {}^\bullet t_j$ *(ovvero se un token in* s_i *corrisponde esattamente al preset della transizione* t_j *che quindi non richiede sincronizzazione), allora*

$$c_i^j = a_j.\Pi_j$$

- *altrimenti, se* $1 \cdot s_i \subseteq {}^\bullet t_j$ *e se esiste* $r \in [1 \ldots i-1]$ *tale che* $1 \cdot s_r \subseteq {}^\bullet t_j$ *(ovvero se* s_i *non è il leader della transizione* t_j *che richiede sincronizzazione e almeno un token da* s_i *per attivarsi), allora*

$$c_i^j = \overline{x_i^j}.\mathbf{0}$$

- *altrimenti, se* $2 \cdot s_i \subseteq {}^\bullet t_j$ *(ovvero se* s_i *è il* leader della transizione t_j *che richiede sincronizzazione e che richiede almeno due token dal leader per attivarsi), allora*

$$c_i^j = \underbrace{\underline{x_i^j}. \ \ldots \ .\underline{x_i^j}}_{{}^\bullet t_j(s_i)-1\ volte} . \underbrace{\underline{x_{i+1}^j}. \ \ldots \ .\underline{x_{i+1}^j}}_{{}^\bullet t_j(s_{i+1})\ volte} . \ \ldots \ . \underbrace{\underline{x_n^j}. \ \ldots \ .\underline{x_n^j}}_{{}^\bullet t_j(s_n)\ volte} .a_j.\Pi_j + \overline{x_i^j}.\mathbf{0}$$

[3] e si tenga presente che il valore ad apice è un indice, non un esponente

- *altrimenti, se* $1 \cdot s_i \subseteq {}^\bullet t_j$ *(ovvero se* s_i *è il leader della transizione* t_j *che richiede sincronizzazione e che richiede esattamente un token dal leader per attivarsi), allora*

$$c_i^j = \underbrace{\underline{x_{i+1}^j}.\ \dots\ .\underline{x_{i+1}^j}}_{{}^\bullet t_j(s_{i+1})\ volte}.\ \dots\ .\underbrace{\underline{x_n^j}.\ \dots\ .\underline{x_n^j}}_{{}^\bullet t_j(s_n)\ volte}.a_j.\Pi_j$$

dove a_j *è l'etichetta della transizione* t_j *(con* $j \in [1 \dots k]$*) e dove* Π_j *rappresenta la destinazione della transizione* t_j *(ancora con* $j \in [1 \dots k]$*) ovvero è pari a*

$$\Pi_j = (\underbrace{C_1 \mid \dots \mid C_1}_{t_j^\bullet(s_1)\ volte} \mid \dots \mid \underbrace{C_n \mid \dots \mid C_n}_{t_j^\bullet(s_n)\ volte})$$

Riguardo ai problemi descritti prima della definizione, osserviamo che:

- per evitare che constanti diverse possano "collassare" nello stesso posto (una volta trasformate tramite *dec*) abbiamo introdotto il termine $y_i.\overline{y_i}.\mathbf{0}$, che è innocuo[4] ma distingue tra loro le varie costanti;

- per gestire la sincronizzazione dei preset della transizione t_j, eleggiamo un posto come leader della transizione (il posto con indice minore nel preset della transizione) e aggiungiamo alla definizione della costante il valore

$$\underbrace{\underline{x_i^j}.\ \dots\ .\underline{x_i^j}}_{{}^\bullet t_j(s_i)-1\ \text{volte}}.\underbrace{\underline{x_{i+1}^j}.\ \dots\ .\underline{x_{i+1}^j}}_{{}^\bullet t_j(s_{i+1})\ \text{volte}}.\ \dots\ .\underbrace{\underline{x_n^j}.\ \dots\ .\underline{x_n^j}}_{{}^\bullet t_j(s_n)\ \text{volte}}.a_j.\Pi_j$$

  dove $t_j(s_i) - 1$ può anche essere pari a zero (quando la transizione richiede un solo token dal leader) ovvero (tenendo conto che i nomi x_i^j sono soggetti a restrizione e quindi possono agire solo sincronizzandosi) richiediamo $r-1$ sincronizzazioni (dove r è il numero dei token nel preset di t_j) prima di attivare l'azione a_j della transizione t_j; alle costanti

[4]non potrà mai muovere in quanto il nome y_i è soggetto a restrizione e il nome ristretto che sostituirà y_i non sarà mai in grado di sincronizzarsi con il corrispondente co-nome ristretto, pure presente

dei posti che non sono leader della transizione – e anche dal posto leader, se la transizione richiede più di un token da questo – viene poi aggiunto

$$\overline{x_i^j}.\mathbf{0}$$

per consentire l'opportuna sincronizzazione al termine aggiunto nella costante del leader;

- per gestire la possibile molteplicità dei token generati da una transizione, il processo terminale di una transizione t_j (la parte dopo la relativa azione a_j) è diventato non una costante ma una combinazione parallela di processi

$$(\underbrace{C_1 \mid \ldots \mid C_1}_{t_j^\bullet(s_1) \text{ volte}} \mid \ldots \mid \underbrace{C_n \mid \ldots \mid C_n}_{t_j^\bullet(s_n) \text{ volte}})$$

Si osservi che ho apportato le seguenti modifiche alla definizione presente in [7][5]:

- il termine "identificativo" che ho aggiunto ad ogni costante C_i è ora $y_i.\overline{y_i}.\mathbf{0}$; in origine era semplicemente $y_i.\mathbf{0}$, ma con la modifica che ho introdotto nella semantica P/T relativa ai nomi ristretti banali, ogni y_i sarebbe stato sostituito da η_d in ogni costante (dal momento che non sarebbe stato presente il co-nome $\overline{y_i}$) perdendo la potenziale efficacia nella distinzione dei posti; con la modifica che ho introdotto, la sostituzione banale non è più applicabile, ogni nome y_i deve essere sostituito da uno specifico nome ristretto (non banale) η_{j_i}, la distinguibilità dei posti è ripristinata e, tenendo conto che le azioni ristrette possono agire solo sincronizzandosi e che la sincronizzazione di due azioni in sequenza è impossibile, il termine introdotto rimane concretamente impossibilitato a muoversi e, quindi, innocuo; si osservi che questo "trucco" è possibile grazie alla superficialità della regola che ho introdotto per decidere se

[5]sezione 6, definizione 11

usare o meno un nome ristretto banale: si chiede se sono presenti sia il nome da sostituire che il co-nome ma non arriva a stabilire se sono concretamente sincronizzabili; se tale criterio venisse migliorato, la definizione appena introdotta potrebbe dover essere rivista;

- nella versione originaria non sono presenti i nuovi nomi x_i^j (con $i \in [1 \dots n]$ e con $j \in [1 \dots k]$) ma solamente nuovi nomi x_j (con j che variava sugli indici delle transizioni); questo è un errore poiché in questo modo – per transizioni un minimo complesse – si può imporre che il numero dei token usati dalla transizione ma non l'esatta origine degli stessi; in altre parole, la funzione di conversione in [7] produce, potenzialmente, più transizioni in corrispondenza di una transizione $t_j \in T$; fortunatamente, individuato l'errore, la correzione è stata molto semplice: è bastato moltiplicare il numero delle costanti in maniera tale che x_i^j identifichi sia la transizione (indice j, ad apice) che il posto in cui il token è presente (indice i, a pedice);

- nella versione originaria le costanti C_i sono definite tramite termini c_i^j ma con l'indice j che non varia tra 1 e k, bensì tra 1 e p_i, dove p_i è il numero di transizioni che contengono s_i nel loro preset; questo comporta una complicazione nella gestione degli indici (aggravato dalla rilevata necessità di gestire il doppio indice nelle azioni x_i^j) che ho preferito evitare ipotizzando che c_i^j possa anche essere pari al processo nullo;

- nella versione originaria è presente una casistica per la definizione del termine leader delle transizioni per distinguere il caso in cui l'etichetta della transizione $l(t_j) = a_j$ è pari all'azione invisibile τ o meno; nel primo caso, dovremmo definire un termine del tipo

$$\underbrace{\underline{x_i^j}. \;\dots\; .\underline{x_i^j}}_{{}^\bullet t_j(s_i)-1 \text{ volte}} . \underbrace{\underline{x_{i+1}^j}. \;\dots\; .\underline{x_{i+1}^j}}_{{}^\bullet t_j(s_{i+1}) \text{ volte}} . \;\dots\; . \underbrace{\underline{x_n^j}. \;\dots\; .\underline{x_n^j}}_{{}^\bullet t_j(s_n)-1 \text{ volte}} .x_n^j.\Pi_j$$

ovvero l'ultima azione (destinata a essere ristretta) x_n^j (nel caso $1 \cdot s_n$ fosse compreso nel preset della transizione, altrimenti la casistica sarebbe diversa) non dovrebbe essere fortemente prefissa e non dovrebbe essere riportata l'azione della transizione (pari a τ); ho ritenuto questa casistica superflua dal momento che – tenendo conto che le regole di sincronizzazione avrebbero annullato la τ, anche con le regole di sincronizzazione originarie – la regola più generale genera comunque una transizione con la medesima etichetta della regola specifica.

Per una dimostrazione formale del fatto che il processo $INet(N(m_0))$ genera, tramite la funzione dec, un sistema P/T isomorfo a $N(m_0)$, rimando a [7][6] in quanto le modifiche e la correzione che ho assorbito non ne cambiano la sostanza.

5.2.2 Esempio

Vediamo ora un esempio concreto di trasformazione di un sistema P/T tramite la funzione $INet$ appena definita.

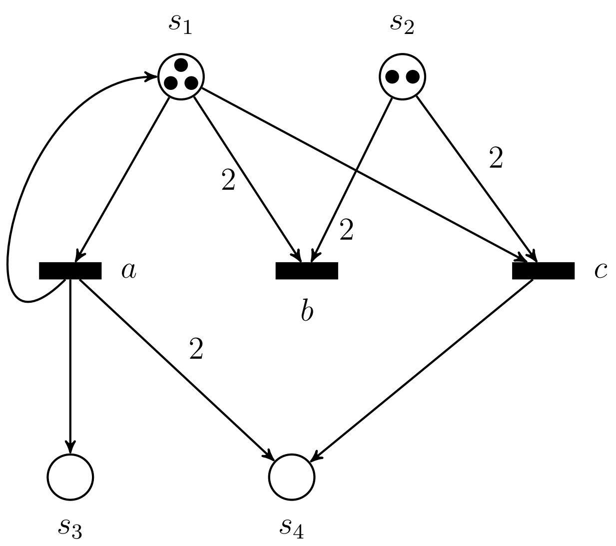

Figura 5.1: Semplice sistema P/T

[6]Teorema 4 (Representability Theorem); dimostrazione in appendice D

Sia dato quindi un sistema P/T, come da figura 5.1 (pagina a fronte), costituito dai quatto posti s_1, s_2, s_3 e s_4, e dalle tre transizioni

$$\begin{aligned} t_1 &= (1 \cdot s_1, a, 1 \cdot s_1 \oplus 1 \cdot s_3 \oplus 2 \cdot s_4) \\ t_2 &= (2 \cdot s_1 \oplus 2 \cdot s_2, b, \emptyset) \\ t_3 &= (1 \cdot s_1 \oplus 2 \cdot s_2, c, 1 \cdot s_4) \end{aligned}$$

Supponiamo anche che la marcatura iniziale sia $m_0 = 3 \cdot s_1 \oplus 2 \cdot s_2$.

Dal momento che abbiamo quattro posti, avremo quattro costanti di processo C_1, C_2, C_3 e C_4 e quattro nomi specifici y_1, y_2, y_3 e y_4; avendo anche tre transizioni, dobbiamo prevedere i 12 nomi x_i^j, con $i \in [1 \ldots 4]$ e $j \in [1 \ldots 3]$.

Premesso che ogni costante conterrà ovviamente il corrispondente $y_i.\overline{y_i}.\mathbf{0}$, vediamo quali sono i contributi derivanti dalle singole transizioni.

La prima transizione genera 1 token in s_1, 1 token in s_3 e 2 token in s_4; il processo di destinazione è quindi

$$\Pi_1 = (C_1 \mid C_3 \mid C_4 \mid C_4)$$

Il leader (per la prima transizione come per le altre due) è il posto s_1; per la prima transizione il preset richiede un solo token, proprio dal leader; tenendo conto che $l(t_1) = a$, abbiamo quindi

$$\begin{aligned} c_1^1 &= a.(C_1 \mid C_3 \mid C_4 \mid C_4) \\ c_i^1 &= \mathbf{0} && \forall i \in [2 \ldots 4] \end{aligned}$$

La seconda transizione ha il postset vuoto; abbiamo quindi, banalmente

$$\Pi_2 = \mathbf{0}$$

La seconda transizione richiede quattro token per essere attivata, ovvero 2 da s_1 (che è il leader) e 2 da s_2, e ha etichetta $l(t_2) = b$; abbiamo quindi

$$\begin{aligned} c_1^2 &= \underline{x_1^2}.\underline{x_2^2}.\underline{x_2^2}.b.\mathbf{0} + \overline{x_1^2}.\mathbf{0} \\ c_2^2 &= \overline{x_2^2}.\mathbf{0} \\ c_i^2 &= \mathbf{0} && \forall i \in [3 \ldots 4] \end{aligned}$$

La terza transizione ha il postset pari a un unico token in s_4; ovvero

$$\Pi_3 = C_4$$

La terza transizione richiede tre token per essere attivata, ovvero 1 da s_1 (che è il leader) e 2 da s_2, e ha etichetta $l(t_2) = c$; abbiamo quindi

$$\begin{aligned}
c_1^3 &= \underline{x_2^3}.\underline{x_2^3}.c.C_4 \\
c_2^3 &= \overline{x_2^3}.\mathbf{0} \\
c_i^3 &= \mathbf{0} & \forall i \in [3 \ldots 4]
\end{aligned}$$

Riassumendo, abbiamo

$$\begin{aligned}
C_1 &\stackrel{def}{=} y_1.\overline{y_1}.\mathbf{0} + a.(C_1 \mid C_3 \mid C_4 \mid C_4) + \underline{x_1^2}.\underline{x_2^2}.\underline{x_2^2}.b.\mathbf{0} + \overline{x_1^2}.\mathbf{0} + \underline{x_2^3}.\underline{x_2^3}.c.C_4 \\
C_2 &\stackrel{def}{=} y_2.\overline{y_2}.\mathbf{0} + \mathbf{0} + \overline{x_2^2}.\mathbf{0} + \overline{x_2^3}.\mathbf{0} \\
C_3 &\stackrel{def}{=} y_3.\overline{y_3}.\mathbf{0} + \mathbf{0} + \mathbf{0} + \mathbf{0} \\
C_4 &\stackrel{def}{=} y_4.\overline{y_4}.\mathbf{0} + \mathbf{0} + \mathbf{0} + \mathbf{0}
\end{aligned}$$

e, ricordando che $m_0 = 3 \cdot s_1 \oplus 2 \cdot s_2$, abbiamo

$$\begin{gathered}
INet(N(m_0)) = (\nu y_1 y_2 y_3 y_4 x_1^1 x_1^2 x_1^3 x_2^1 x_2^2 x_2^3 x_3^1 x_3^2 x_3^3 x_4^1 x_4^2 x_4^3) \\
(C_1 \mid C_1 \mid C_1 \mid C_2 \mid C_2)
\end{gathered}$$

Bibliografia

[1] N. Busi, R. Gorrieri: "*Distributed semantics for the π-calculus based on Petri nets with inhibitor arcs*", Journal of Logic and Alg. Prog. 78(3):138-162, 2009

[2] P. Degano, R. De Nicola, U. Montanari: "*A Distributed Operational Semantics for CCS based on C/E Systems*", Acta Informatica 26(1-2):59-91, 1988

[3] U. Goltz: "*On Representing CCS Programs by Finite Petri Nets*", dai lavori di MFCS'88, LNCS 324, 339-350, Springer-Verlag, 1988

[4] U. Goltz: "*CCS and Petri Nets*", LNCS 469, 334-357, Springer-Verlag, 1990

[5] Roberto Gorrieri: "*A Fully-Abstract Semantics for Atomicity*", nota tecnica

[6] Roberto Gorrieri, Cristian Versari: "*A Process Calculus for Expressing Finite Place/Transition Petri Nets*", dai lavori di EXPRESS'10, EPTCS, 2010. DOI 10.4204/EPTCS.41.6, disponibile presso arXiv:1011.6433v1

[7] Roberto Gorrieri, Cristian Versari: "*A Process Algebraic Representation of Finite Place/Transition Petri Nets*", evoluzione del precedente, in corso di pubblicazione

[8] R. Milner: "*Communication and Concurrency*", Prentice-Hall, 1989

[9] E. R. Olderog: "*Nets, Terms and Formulas*", Cambridge Tracts in Th. Comp. Science 23, Cambridge University Press, 1991

[10] J. L. Peterson: "*Petri Net Theory and the Modeling of Systems*", Prentice-Hall, 1981

[11] W. Reisig: "*Petri Nets: An Introduction*", EATCS Monographs on TCS, Springer-Verlag, 1985

Ringraziamenti

Intraprendere nuovi studi universitari, superati i quarant'anni, è un'esperienza interessante: ai ricordi e alle nostalgie per quello che è stato, si mescolano lo spaesamento per quello che è cambiato e, a causa del salto generazionale, la sgradevole sensazione di essere un intruso.

Al termine di questa esperienza non posso che riconoscere che, se questa è stata produttiva e piacevole oltre le più rosee aspettative, è stato anche per merito di molte persone.

Elencarle una per una sarebbe complicato, tedioso e ingeneroso nei confronti di coloro che, inevitabilmente, finirei per dimenticare.

Non posso però non ringraziare i miei genitori, che hanno avuto la pazienza di sopportare – per oltre due anni – un figlio che aveva la testa tra le nuvole persino più del solito.

Meritano un esplicito ringraziamento i miei compagni di corso che, superato lo sconcerto iniziale, hanno elegantemente fatto finta di ignorare la grossa differenza di età e mi hanno fatto sentire uno di loro.

Un ringraziamento complementare al corpo docente, che ha sorvolato sulla scarsa differenza di età e mi ha insegnato con pazienza anche quando il mio entusiasmo poteva essere fastidioso.

Un ringraziamento particolare al prof. Roberto Gorrieri, che mi accompagnato nell'ultima fase di questa avventura.

Un ringraziamento particolare anche al prof. Giulio Casciola, che mi ha dato un grosso aiuto per superare il più arduo ostacolo con cui ho dovuto confrontarmi: la burocrazia.

Il ringraziamento maggiore non va però a una singola persona ma un'istituzione, l'Università di Bologna, che – nonostante i tagli economici di una politica miope e autolesionista – continua a essere un faro per una cultura sempre più disprezzata da una società che esalta l'apparenza e trascura la qualità.