

ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA

SCUOLA DI INGEGNERIA E ARCHITETTURA

*Dipartimento di Ingegneria dell'Energia Elettrica e dell'Informazione
"Guglielmo Marconi" - DEI*

Corso di Laurea Magistrale in Ingegneria Elettronica

TESI DI LAUREA

in

Elaborazione dei Segnali nei Sistemi Elettronici M

**ALGORITMI PER LA REIEZIONE DEI DISTURBI NEI
SISTEMI DI ACQUISIZIONE DEI SEGNALI EEG BASATI
SULLA TECNICA DEL COMPRESSED SENSING**

CANDIDATA
Letizia Magenta

RELATORE:
Chiar.mo Prof. Mauro Mangia

CORRELATORE:
Prof. Riccardo Rovatti

Anno Accademico 2017/2018

Sessione III

Indice

1	Introduzione	4
1.1	Trend tecnologici	4
1.2	CS in smart sensor e IoT	7
2	Compressed Sensing	11
2.1	Compressed Sensing: definizione e metodo	11
2.2	Compressed Sensing basato sulla Rakeness	13
2.2.1	Toy case: risultati e analisi	15
3	La reiezione dei disturbi nel Compressed Sensing	16
3.1	Reiezione dei disturbi	17
3.2	Metodo della discesa del gradiente proiettato e approccio alla proiezione alternata	19
3.3	Toy Case: risultati e analisi	26
4	Caso di studio EEG	30
4.1	Segnali EEG	30
4.1.1	Genesi dei potenziali elettroencefalografici	31
4.1.2	Sistemi di posizionamento EEG	32
4.1.3	Potenziali evocati	34
4.2	Filtraggio	36
4.2.1	Filtraggio pre-acquisizione	37
4.2.2	Filtraggio post-acquisizione	38
4.3	Stima della statistica del segnale e dei disturbi	46
4.3.1	Rappresentazione sparsa	49

4.4	Risultati	52
4.4.1	Analisi per ogni singolo sensore	53
4.4.2	Analisi che considera il sistema complessivo	61
4.4.3	Analisi per cluster	65
4.4.4	Confronti finali	72
5	Conclusioni	87
5.1	Sviluppi futuri	88

Capitolo 1

Introduzione

1.1 Trend tecnologici

Nel corso del Gartner Symposium/ITxpo 2017 di Orlando, Gartner, la nota società di consulenza strategica, ricerca e analisi nel campo dell'Information Technology ha stilato una top 10 dei trend tecnologici che avranno un impatto notevole per la maggior parte delle imprese e organizzazioni. Questi dieci trend tecnologici sono correlati all'Intelligent Digital Mesh, ovvero quel mix digitale di persone, oggetti e servizi, ritenuti essere la base per i futuri business digitali ed ecosistemi. Queste tendenze tecnologiche per il 2018 comprendono tra le tante AI Foundation, Intelligent Apps & Analytics e Intelligent Things [2] nel 2019 si aggiungono anche gli Smart Spaces: un ambiente fisico o digitale in cui gli esseri umani e i sistemi tecnologici interagiscono in ecosistemi sempre più aperti, connessi, coordinati e intelligenti. Elementi multipli (persone, processi, servizi e oggetti) si uniscono in uno spazio intelligente per creare un'esperienza più coinvolgente, interattiva e automatizzata. Le applicazioni industriali dell'Internet of Things, ovvero tutto l'ambito IoT, fatta eccezione per il ramo consumer, sono considerate tra le aree più promettenti per lo sviluppo dell'economia globale. Secondo un'indagine condotta da Grand View Research e relativa al 2016, il mercato globale IoT in ambito industriale è risultato pari a 109,28 miliardi di dollari. L'industrial IoT ha decisamente preso piede in tutto il mondo, grazie a una maggiore disponibilità di processori, sensori e altre tecnologie, anche a prezzi più popolari, che facilitano il loro utilizzo

e di conseguenza l'accesso alle informazioni in tempo reale. Le aziende sentono sempre di più l'esigenza di migliorare la propria efficienza operativa, con una nuova alleanza tra i giocatori chiave del settore, e una ricerca di integrazione tra forza lavoro e digitale. Con tutti questi presupposti, l'IoT dovrebbe avere importanti opportunità di crescita nei prossimi otto anni. Secondo Grand View Research, infatti, il mercato IoT dovrebbe valere 933,62 miliardi di dollari nel 2025. [2]

In particolare il mercato dei dispositivi medicali indossabili gode di ottima salute. La crescita a due cifre si consolida e assicura anni estremamente positivi. Le tecnologie dei sensori, dell'elaborazione dei dati e della trasmissione wireless, ormai estremamente miniaturizzate, consentono di creare dispositivi sempre più performanti, oltre che accattivanti. Sotto il profilo strettamente medicale, il monitoraggio da remoto dell'attività cardiaca, della pressione del sangue e di altri valori si sta diffondendo per favorire la prevenzione di ricadute e di situazioni patologiche gravi. Secondo l'ultimo report di Market Data Forecast sul mercato dei dispositivi medicali indossabili, pubblicato il 4 luglio 2017, il mercato globale si svilupperà secondo un tasso medio di crescita del 19,58% nel periodo 2016-21, con una crescita del valore dai 4,57 miliardi di dollari nel 2016 agli 11,18 miliardi nel 2021. Gli ultimi sviluppi tecnologici, osserva il report, includono un feedback real-time, meccanismi di allarme, supporti alla decisione medica e accesso wireless alle informazioni raccolte. I dispositivi offrono l'hardware di supporto per trattare le situazioni di emergenza con collegamento diretto ai punti di assistenza, monitoraggio e supporto medico illimitato, assistenza nella gestione da remoto delle condizioni mediche del paziente in riabilitazione e del paziente cronico o disabile. I principali fattori che garantiscono *trend così positivi* sono tre:

- Vi è un fattore demografico: l'invecchiamento della popolazione e quindi l'aumentata incidenza delle patologie croniche tipiche dell'età;
- Una maggiore attenzione alla salute e al benessere nei paesi più avanzati, che è sostenuta dalla crescita della capacità di spesa;
- Una crescente facilità d'uso dei dispositivi. Un tema critico che rappresenta invece una sfida per i produttori, è quello dell'efficienza dei costi nel caso di

monitoraggio a distanza di alcune patologie croniche, ad esempio il cancro e le malattie del sangue.

Un tema tecnologico che può *rallentare* il mercato è quello della durata delle batterie nell'ulteriore miniaturizzazione dei dispositivi. Un argomento che resta sempre critico è quello della tutela della privacy nel momento in cui vengono trasmesse informazioni sensibili. Secondo gli analisti di Market Data Forecast, la miniaturizzazione è il tema dominante che orienta in senso molto positivo i trend per gli anni a venire. Si è già arrivati a prototipi di piccole capsule ingeribili, equipaggiate con sensori della dimensione di un granellino di sale e un altrettanto piccolo apparato di trasmissione, che consentono di temere monitorate le condizioni fisiologiche di un paziente, piuttosto che l'assorbimento di medicinali. I device medicali indossabili sono quindi destinati a esercitare un impatto molto positivo sul mercato dell'Internet delle cose dei prossimi anni, di per sé già orientato ad un continuo rialzo. Per quanto riguarda la distribuzione geografica del mercato, il Nord America detiene una quota del 40%, seguita dall'Europa e dalla regione Asia-Pacifico. Negli anni futuri è prevista un'accelerazione della crescita proprio in quest'ultima zona. Concreti sistemi di monitoraggio sono ad esempio il dispositivo proposto al CES 2017 da una partnership franco-statunitense che ha concepito e presentato sul mercato MOTIO HW, un braccialetto che integra tecnologie ottiche e cutanee; il dispositivo rileva i dati biometrici, li invia a qualsiasi tipo di piattaforma dove una app li processa con un algoritmo capace di diagnosticare le apnee e comprendere il profilo del sonno individuale, per dare indicazioni utili a migliorare la qualità della vita oppure il dispositivo della Royal Philips presentato a settembre del 2017 anche in Italia, a Milano, durante il Congresso Internazionale della Società Europea della Respirazione, con la sua innovativa gamma di soluzioni connesse per la respirazione e il sonno, progettate per offrire cure sempre più efficaci ed efficienti. *DreamStation Go*, il nuovo e più piccolo dispositivo di ventilazione meccanica a pressione positiva continua, è progettato per pazienti che soffrono di apnee ostruttive del sonno e che sono sempre in viaggio, al fine di semplificare la loro esperienza terapeutica e via dicendo. [3]

1.2 CS in smart sensor e IoT

IoT è quindi un recente paradigma di comunicazione in cui "Things" sta per dispositivi incorporati (anche detti smart o intelligenti) con sensori, attuatori, dispositivi elettronici, ricetrasmittitori per il monitoraggio remoto di diversi dispositivi fisici, che li rendono in grado di comunicare con altri dispositivi con l'obiettivo di migliorare gli standard di vita degli utenti. In generale i sensori sono di piccole dimensioni, non collegati e alimentati a batteria. La capacità finita della batteria limita però la durata del dispositivo, ricaricare o sostituire regolarmente le batterie è costoso e laborioso, quindi il consumo di energia dei nodi per l'aggregazione dei dati pone la principale preoccupazione in questo ambito. In uno scenario IoT tipico, viene raccolto un numero elevato di dati, i dispositivi intelligenti autoalimentati devono comunicare tra loro e con il cloud attraverso un collegamento wireless allo scopo di scambiarsi informazioni e fornire servizi specifici. Tuttavia, l'elevato consumo di energia associato alla trasmissione, limita le prestazioni di questi dispositivi IoT auto-alimentati in termini di capacità di calcolo e durata della batteria. Pertanto, per ottimizzare la trasmissione dei dati, diversi approcci devono essere esplorati come la trasmissione cooperativa, architetture di rete multi-hop e sofisticate tecniche di compressione. Per quest'ultimo, il compressed sensing (CS) è un paradigma interessante da incorporare nella progettazione di queste piattaforme. Il CS è una tecnica di acquisizione del segnale che integra una teoria di compressione che sfrutta il comportamento di sparsità della maggior parte dei segnali naturali e delle architetture IoT per realizzare piattaforme efficienti in tempo reale in grado di garantire applicazioni efficienti. [4]

Il Compressed Sensing (CS) mira ad acquisire direttamente una forma compressa dei segnali con un comportamento sparso nella fase di rilevamento e consente un'elevata qualità di ricostruzione nella fase di ricezione. Il CS presenta un paradigma alternativo al tradizionale teorema di acquisizione che afferma che il numero di campioni acquisiti deve essere almeno uguale al doppio della banda del segnale originale per garantire un corretto recupero. Tuttavia, queste condizioni non prendono in considerazione la struttura del segnale. Quindi, se il segnale di interesse è sparso, ovvero, può essere rappresentato da un numero minore di coefficienti diver-

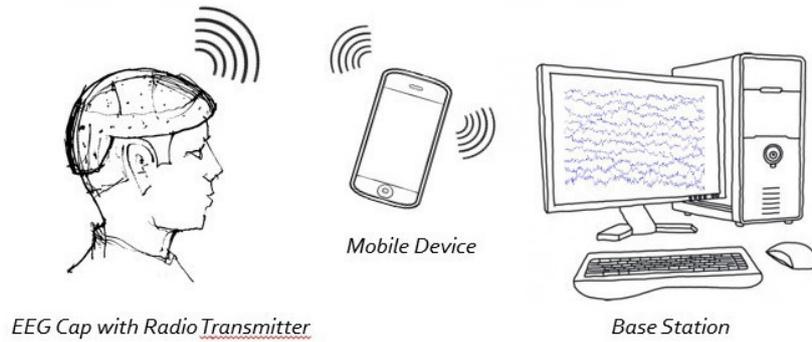


Figura 1.1: Sistema di trasmissione dei dati in cui si vuole spostare la parte computazionale più pesante dal lato della base station lasciando la cuffia per l'acquisizione del EEG il più semplice possibile, risparmiando così energia

si da zero rispetto la sua originale dimensione, il CS sostiene che prendendo solo poche misure (o proiezioni) del segnale sparso, questo è sufficiente per acquisire le informazioni salienti del segnale e per fornire un accettabile qualità nella ricostruzione. Il CS mira a spostare la complessità dai sensori che di solito sono risorse vincolate e autoalimentate, al lato ricevitore che di solito è installato su piattaforme di calcolo con vincoli rilassati. Per la maggior parte delle applicazioni del mondo reale, è sempre possibile trovare una rappresentazione sparsa o comprimibile per segnali di interesse usando la trasformazione appropriata. Pertanto, il CS si è diffuso ampiamente in vari applicazioni come radar, elaborazione di immagini, compressione di segnali biologici, comunicazioni wireless e molti altri. Il CS acquisisce un grande vantaggio anche dal punto di vista del *costo dei sensori* intelligenti, infatti questa teoria prevede che alcuni segnali possano essere recuperati da quelle che in precedenza si ritenevano fossero misure altamente incomplete. In questa teoria, il trasmettitore deve solo inviare i dati con un semplice metodo di misurazione alla terminazione anziché comprimere e codificare i dati. In questo caso, la terminazione può recuperare accuratamente i dati con un algoritmo di recupero efficace. Di conseguenza, il sistema di rilevamento della compressione può effettivamente ridurre il costo dei sensori nel processo di trasmissione delle informazioni e raccolta dei dati. Ad esempio, il settore sanitario ha visto un enorme sforzo per esplorare il CS in diversi campi di applicazioni. Gli esperti ritengono che questo algoritmo

sarebbe vantaggioso per un'ampia gamma di applicazioni in campo medico per test di laboratorio, in particolare quando vengono generati molti dati. Il CS può anche migliorare strumenti indossabili di monitoraggio della salute, rendendoli più piccoli, più economici e più efficienti dal punto di vista energetico. Infatti oltre che risparmio nel costo si parla anche di *risparmio di energia* in quanto questa tecnica può essere incorporata in sensori wireless intelligenti aumentando in modo efficace l'affidabilità della comunicazione wireless senza ritrasmettere i dati; la promessa di questo approccio è ridurre il traffico di dati risparmiando energia.1.1

In sintesi l'idea di base dell'approccio basato sul CS è che, invece di trasmettere il segnale grezzo acquisito dal sensore, viene trasmesso un segnale trasformato che viene generato proiettando il segnale grezzo su una matrice casuale. Durante la trasmissione di questo segnale trasformato potrebbe verificarsi una perdita di dati. Tuttavia, secondo la teoria di CS, il segnale grezzo può essere ricostruito efficacemente dal segnale trasformato incompleto ricevuto dato che il segnale grezzo è comprimibile in alcune basi e il rapporto di perdita di dati è basso [4]. Il CS si prevede che ottimizzi la potenza e l'energia utilizzata nei dispositivi ambulatoriali wireless, estendendo quindi la durata del sensore e semplificando significativamente la progettazione dell'hardware riducendo dimensioni e costi dell'intera piattaforma sanitaria. Le applicazioni sanitarie basate sul CS includono l'imaging medico, monitoraggio dell'elettrocardiogramma (ECG), compressione EEG, soluzioni biometriche, ecc. Successivamente, le piattaforme IoT hanno anche assistito all'integrazione del CS nelle sue varie applicazioni basate su due proprietà mostrate dalla maggior parte delle piattaforme IoT. Innanzitutto, una vasta gamma di dati del mondo reale può essere ben approssimata da un segnale sparso usando la trasformazione appropriata. Ad esempio, entrambi discrete cosine transform (DCT) e discrete wavelet transform (DWT) forniscono una buona rappresentazione sparsa per ECG, immagini, temperatura, dati di umidità, ecc. Inoltre, è sempre possibile formare una base di sparsificazione mediante metodi di apprendimento del dizionario. La seconda proprietà si basa sugli schemi di trasmissione sporadici che la maggior parte delle piattaforme IoT mostra. In questi schemi di trasmissione, non tutti i dispositivi trasmettono i loro dati contemporaneamente al nodo di fusione,

piuttosto solo un piccolo numero di dispositivi contribuisce al segnale aggregato a qualsiasi tempo a disposizione, ciò significa che la velocità dei dispositivi attivi a ciascuna fascia oraria di trasmissione è molto piccola. Così, la scarsità dell'architettura può essere esplorata e la conoscenza dell'attività dei nodi può essere sfruttata a livello di cloud tramite protocolli di rilevamento sparso e comuni per ottenere dati ad elevata affidabilità con un piccolo numero di dispositivi di trasmissione. [5]

Capitolo 2

Compressed Sensing

Compressed Sensing è una tecnica che permette la ricostruzione del segnale partendo da un numero limitato di misurazioni lineari che sono potenzialmente più piccole del numero di campioni raccolti alla frequenza di Nyquist. La possibilità di tale rappresentazione "sub-nyquista" dipende da un'ipotesi sulla classe considerata di segnali, vale a dire che siano *sparsi*.

2.1 Compressed Sensing: definizione e metodo

Usando una rappresentazione tempo-discreta, istanze e disturbi di un input in ogni finestra corrispondono ai vettori $x, v \in \mathbb{R}^n$ dove n è il numero di campioni alla frequenza di Nyquist f_s . Il CS proietta $x + v$ in m sequenze organizzate come righe della matrice di sensing $A \in \mathbb{R}^{m \times n}$ con $m < n$. Se questo accade allora le misure sono sufficienti per acquisire il necessario contenuto informativo del segnale, ottenendo così un apporto di compressione definito come $CR = \frac{n}{m}$. Questa procedura basa la sua efficacia nel fatto che l'istanza del segnale sia k -sparso, ovvero che esiste una base $D \in \mathbb{R}^{n \times n}$ tale che esista la rappresentazione $x = D\xi$ dove $\xi \in \mathbb{R}^n$ sia un vettore con non più di $k \leq n$ elementi diversi da zero. Se ciò si verifica allora il numero di gradi di libertà di x è minore di n , pertanto il contenuto informativo del segnale può essere rappresentato da un vettore di misurazione $y \in \mathbb{R}^m$ ottenuto mediante la proiezione su A .

$$y = A(x + v) = AD\xi + Av \quad (2.1)$$

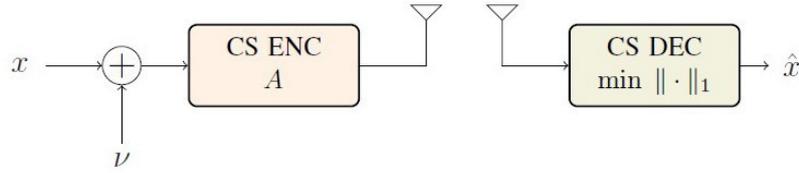


Figura 2.1: Schema a blocchi del sistema CS che include l'effetto del rumore.

Facendo riferimento alla fig. 2.1 entrambi i segnali di input e disturbo sono processati dall'encoder CS che effettua le proiezioni. L'uscita del blocco è appunto y che è anche l'ingresso del decoder che ricostruisce il segnale e la cui uscita è $\hat{x} = D\hat{\xi}$. Se la matrice AD soddisfa determinate condizioni, ξ (e quindi x) può essere recuperato da y nonostante il fatto che A (e quindi AD) sia una riduzione di dimensionalità, purché $m = O(k \log \frac{n}{k})$. Questo suggerisce che possono essere ottenuti valori non trascurabili di CR . In parole povere, le condizioni su AD richiedono che i vettori generici k -sparsi siano mappati quasi isometricamente nelle misurazioni. Tali condizioni sono comunemente soddisfatte impostando A come un'istanza di una matrice casuale con voci indipendenti e identicamente distribuite, le distribuzioni gaussiane sono le più comunemente adottate. Algoritmicamente, il recupero del segnale sparso può essere ottenuto risolvendo il seguente problema di ottimizzazione convessa che sfrutta l'assunzione di sparsità per selezionare, su un insieme infinito di vettori n -dimensionali mappato su y da AD , il vettore $\hat{\xi}$ più sparso.

$$\begin{aligned} \hat{\xi} &= \operatorname{argmin}_{\xi} \|\xi\|_1 \\ \text{subject to } &\|AD\xi - y\|_2^2 \leq \epsilon^2 \end{aligned} \tag{2.2}$$

dove $\|\xi\|_1$ è una norma che promuove la sparsità, e $\|AD\xi - y\|_2^2$ è la classica norma euclidea che misura l'accuratezza con cui le misure y sono adattate dalla soluzione. Sappiamo infatti da [24], [25] che l'errore di ricostruzione $\|\hat{x} - x\|_2$ svanisce con probabilità uno se gli elementi di \mathcal{A} sono istanze di variabili aleatorie Gaussiane (o sub-Gaussiane) indipendenti e identicamente distribuite (i.i.d.) e $m = O(k \log \frac{n}{k})$. Il parametro $\epsilon \geq 0$ viene scelto proporzionalmente alla quantità di rumore che

influenza y , $\epsilon = 0$ corrispondente al caso senza rumore*. [6]

2.2 Compressed Sensing basato sulla Rakeness

L'approccio sopra descritto è stato esteso in [26], [23] dall'introduzione di un'ulteriore proprietà sul segnale acquisito: la *localizzazione* ovvero la misura di quanto la distribuzione dell'energia del segnale è non uniforme sull'intero spazio del segnale stesso. Matematicamente, la localizzazione può essere valutata confrontando la deviazione della matrice di correlazione del segnale $C^x = \mathbf{E}[xx^T]$ con la matrice identità I_n che caratterizza perfettamente il segnale bianco, quindi avremo:

$$\mathcal{L}_x = \frac{\text{tr}((C^x)^2)}{\text{tr}(C^x)^2} - \frac{1}{n} \quad (2.3)$$

dove $\text{tr}()$ è la traccia della matrice e $C^x, I_n \in \mathbb{R}^{n \times n}$. Valori non nulli di \mathcal{L}_x corrispondono a matrici C^x che deviano da I_n , vale a dire che ci sono sottospazi lungo i quali la maggior parte dell'energia di x è concentrata. La proprietà della localizzazione è usata nel CS basato sulla *rakeness* (R-CS) che adatta il profilo di correlazione per costruire una generica riga di A a C^x e quindi a \mathcal{L}_x . L'idea dietro R-CS è quella di massimizzare l'energia media ρ raccolta dalle misure con il vincolo che la localizzazione delle sequenze di sensing (y) non sia troppo alta per evitare una sovradattazione. Questo problema viene risolto matematicamente come segue:

$$\rho = \max_{C^A} \text{tr}(C^A C^X) \quad (2.4)$$

$$s.t. \quad C^{AT} = C^A \quad (2.5)$$

$$s.t. \quad C^A \succeq 0 \quad (2.6)$$

$$s.t. \quad \text{tr}(C^A) = 1 \quad (2.7)$$

$$s.t. \quad \mathcal{L}_A \leq \tau \mathcal{L}_x \quad (2.8)$$

dove il primo e il secondo vincolo assicurano che C^A sia simmetrica e semidefinita positiva (ovvero che sia una matrice di correlazione), il terzo vincolo normalizza ogni singola sequenza di sensing mentre il quarto vincolo, una volta definito $0 <$

*Tutti i calcoli in questo lavoro vengono eseguiti utilizzando il software **Mathworks MATLAB® 2018b** con l'aggiunta di tool che vengono esaminati più avanti.

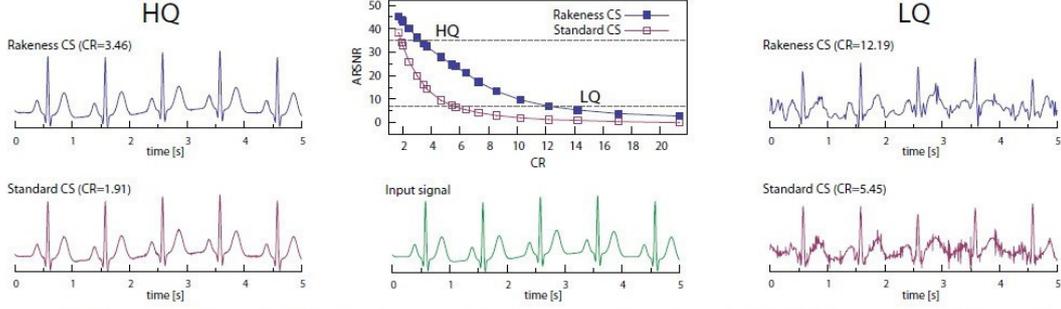


Figura 2.2: ARSNR in funzione delle rappresentazioni CR (centrale in alto) e visive del segnale di ingresso sintetico (centrale in basso) e qualità di ricostruzione per CS standard e CS basato su rakeness nei punti operativi LQ (a destra) e HQ (a sinistra).

$\tau < 1$, in genere fissato a $1/4$, stabilisce quanto la proiezione della forma d'onda deve essere localizzata rispetto \mathcal{L}_x . Questo problema è stato risolto analiticamente calcolando C^A per la generazione delle righe di A e il corrispondente valore massimo di ρ^* . [7]

$$C^A = \frac{1}{2} \left(\frac{C^x}{tr(C^x)} + \frac{I_n}{n} \right) \quad (2.9)$$

$$\rho^* = tr(C^x) \left(\frac{1}{2} \mathcal{L}_x + \frac{1}{n} \right) \quad (2.10)$$

Per mostrare l'efficacia di questo algoritmo prendiamo l'esempio riportato nell'articolo [33] che mostra l'implementazione in un caso di segnale ECG che proponiamo in figura 2.2. La figura mostra che a parità di un certo livello di ARSNR che si vuole ottenere, è necessario utilizzare fattori di ricostruzioni diversi in base al metodo utilizzato. Si pensi infatti a due casi di analisi in cui una ricostruzione debba essere accurata perchè a fini di registrazione per valutazione medica, dall'altra parte si potrebbe pensare ad un banale caso di "Wellness" in cui l'utente voglia solo vedere il segnale di Heart-Rate in tempo reale; nel primo caso avremo una ricostruzione di alta qualità, nel secondo una di bassa. In particolare nel caso in analisi, queste due soglie sono all'incirca di $35dB$ per una ricostruzione di alta qualità e $7dB$ per una di bassa. Si consideri ora il segnale verde centrale che rappresenta la vera informazione e confrontiamolo con i due casi sopra citati; a sinistra si ha la ricostruzione di alta qualità in alto con il CS in basso con R-CS

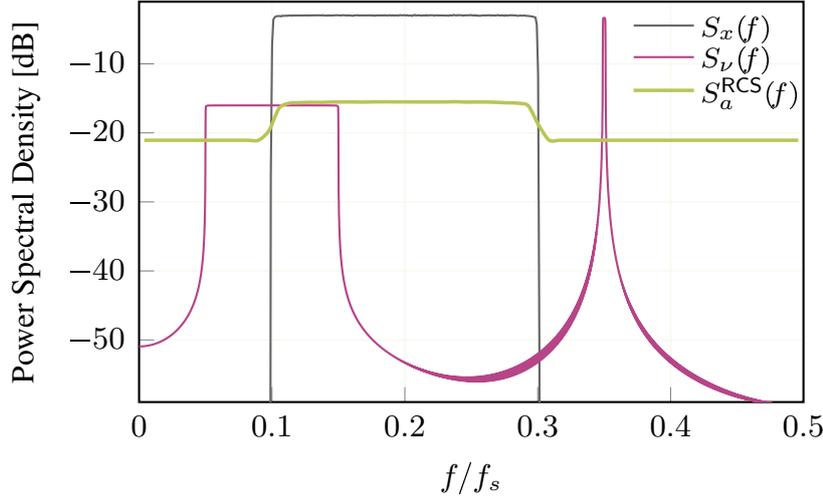


Figura 2.3: Con $n=64$, la densità spettrale di potenza della classe di segnale in input ($\mathcal{S}_x(f)$) e dei disturbi ($\mathcal{S}_v(f)$) insieme con i profili per entrambi CS ($\mathcal{S}_a^{CS}(f)$) e RCS ($\mathcal{S}_a^{RCS}(f)$) con $r=0.95$.

e si vede come effettivamente fissato un livello di ARSNR alto si hanno ottime ricostruzioni, con l'unica differenza che vengono sfruttati fattori di compressione diversi: infatti il CS per ottenere quel risultato usa un CR pari a 3.46 mentre il R-CS un $CR = 1.91$, quasi la metà. Stessa considerazione se valutiamo il caso a destra, ovvero una ricostruzione di bassa qualità dove di nuovo il CS necessita di un CR pari a 12.19 mentre R-CS 5.45, questa volta meno della metà.

2.2.1 Toy case: risultati e analisi

Prendiamo ora in esame l'esempio riportato nell'articolo [18], che porteremo per il confronto anche nel successivo sviluppo nel paragrafo seguente. La figura 2.3 mostra la densità spettrale di potenza prendendo in esame dei profili artificiali creati ad hoc con n pari a 64. Si ha un segnale passa banda perturbato da un disturbo composto da due contributi, un passabanda con un parziale overlap con il segnale di input, e un secondo a banda stretta fuori dalla banda del segnale. La figura riporta anche lo spettro per il problem R-CS. Si noti come il profilo risultante dalla risoluzione del problema di ottimizzazione tenda a seguire il segnale nell'intervallo in cui il segnale di input ha una banda importante trascurando però completamente l'andamento del disturbo.

Capitolo 3

La reiezione dei disturbi nel Compressed Sensing

Un altro importante problema nella realizzazione di un sistema di acquisizione è la possibilità di rigettare i disturbi che corrompono il segnale in ingresso al sistema. Indipendentemente dall'algoritmo di compressione adottato, quando il disturbo ha qualche caratteristica in frequenza sfruttabile, normalmente il design del sistema prevede dei filtri o nel dominio analogico o dopo la digitalizzazione. Queste soluzioni sono efficienti ma incrementano la complessità del sistema e quindi anche l'energia. In questo capitolo viene presentato un nuovo design del CS che, adattando lo stadio di sensing alla classe dei segnali acquisiti, permette di rigettare il disturbo sfruttando l'ipotesi che anche le loro caratteristiche statistiche di secondo ordine siano note. Chiaramente lo scopo è quello di evitare l'utilizzo di ulteriori filtri (vedi 3.1). Per questo il CS è un tool unico a bassa complessità per la codifica, compressione e reiezione dei disturbi. [7]

In altre parole, miriamo a sfruttare il CS per comprimere selettivamente solo le informazioni utili; in questo modo possiamo teoricamente raggiungere un livello di compressione più alto (cioè risparmiare più energia nel nodo wireless) e trasmettere contemporaneamente una versione del segnale con esclusione del rumore. Al lato della stazione di base l'obiettivo sarà quello di decomprimere / ricostruire la versione del segnale acquisita, eventualmente in tempo reale.

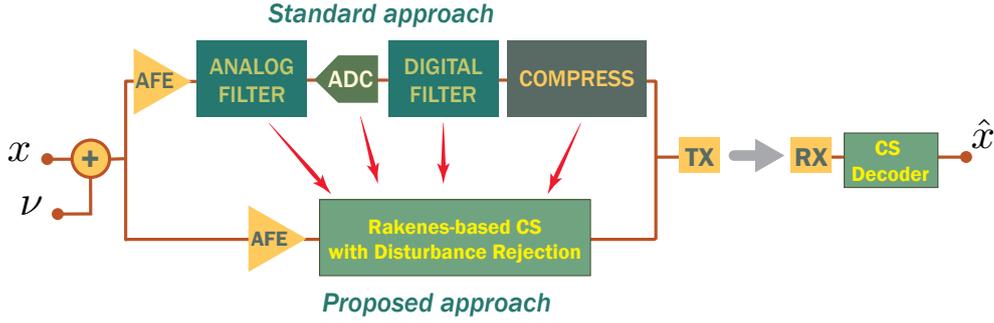


Figura 3.1: R-CS in basso a confronto con l’approccio standard che prevede l’utilizzo dei filtri.

3.1 Reiezione dei disturbi

Per raggiungere questo obiettivo, si procede mostrando un flusso algoritmico che segue due step in cui vengono considerate note le proprietà statistiche del secondo ordine sia del vero input C^x che dei disturbi $C^v = \mathbf{E}[vv^T]$. Questa procedura nel primo step risolve il problema di massimizzazione della rakeness (2.8) trascurando i disturbi, estraendo quindi la sequenza di sensing in grado di incrementare la quantità di energia media acquisita (ρ^*) nel segnale vero.

Sappiamo da [21], [22], [23] che ”rastrellare” una certa quantità di energia, implica un significativo miglioramento nella valutazione della compressione del segnale. Infatti, il numero di misure m necessario quando le righe della matrice di correlazione A disegnate in maniera random e con una correlazione C^A come (2.9), è molto minore rispetto al caso del CS standard. Considerando il caso in cui si abbia anche una certa quantità di disturbo, la matrice di sensing potrebbe anche estrarre una significativa quantità di energia da queste componenti indesiderate e che sono quindi implicitamente introdotte nell’algoritmo di ricostruzione, danneggiando così la qualità finale di \hat{x} . Al fine di ridurre questo effetto, viene introdotto il secondo step che cerca di minimizzare ciò che è stato raccolto dal

disturbo risolvendo il seguente problema di minimizzazione:

$$\min_{C^A} tr(C^A C^v) \quad (3.1)$$

$$s.t. \quad C^{AT} = C^A \quad (3.2)$$

$$s.t. \quad C^A \succeq 0 \quad (3.3)$$

$$s.t. \quad tr(C^A) = 1 \quad (3.4)$$

$$s.t. \quad \mathcal{L}_A \leq \tau \mathcal{L}_x \quad (3.5)$$

$$s.t. \quad tr(C^A C^X) \geq r \rho^* \quad (3.6)$$

dove appunto si cerca di minimizzare la quantità di energia estratta dal disturbo, i primi quattro vincoli su C^A sono gli stessi del primo step e che garantiscono quindi le proprietà essenziali affinché C sia una matrice di correlazione e il vincolo di localizzazione, mentre l'ultimo vincolo dove $0 \leq r \leq 1$, richiede che le sequenze di sensing, riducendo l'energia raccolta dal disturbo, non riducano troppo la quantità di energia raccolta dal segnale vero. Per limitare la differenza tra i due problemi di ottimizzazione il valore di r viene scelto il più vicino possibile a 1. [7]

Facendo un breve riassunto si è visto che le sequenze di sensing per l'approccio standard CS (S-CS) ha, come istanze della matrice, vettori n-dimensionali Gaussiani e incorrelati, per il R-CS le righe di A sono generate come istanze di un vettore Gaussiano con matrice di correlazione presa da 2.9 mentre per R-CSd, la matrice di correlazione che caratterizza il processo Gaussiano di generazione delle righe di A , è data risolvendo il problema di ottimizzazione 3.1 con il toolbox CVX o, in maniera alternativa e più efficiente, con il metodo che viene presentato nella prossima sezione.

Possiamo valutare graficamente il risultato che si ottiene da questo problema di ottimizzazione andando a richiamare lo stesso esempio riportato nel capitolo precedente 3.2. Oltre che allo spettro del segnale di input, del disturbo e lo spettro per il problema R-CS, questa volta si ha anche lo spettro per il problema R-CSd. Si noti come il profilo $(\mathcal{S}_a^{R-CSd}(f))$ sia alto nella stessa banda del segnale come nel caso di $(\mathcal{S}_a^{R-CS}(f))$ mentre però decresce anche rapidamente quando $(\mathcal{S}_v(f))$ è alto. Confrontando la traccia di R-CS con R-CSd quello che si nota è che le due sono estremamente simili quando i disturbi sono trascurabili, questo è dovuto al

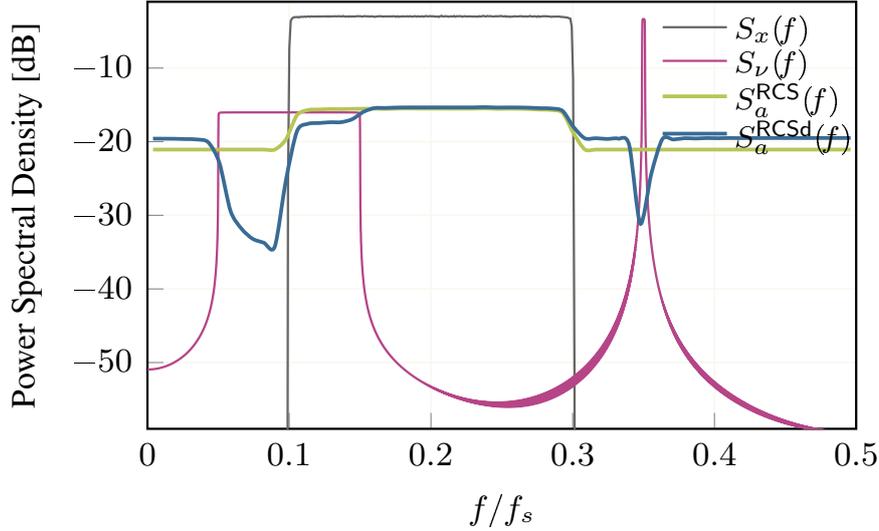


Figura 3.2: Con $n=64$, la densità spettrale di potenza della classe di segnale in input ($\mathcal{S}_x(f)$) e dei disturbi ($\mathcal{S}_v(f)$) insieme con i profili per entrambi R-CS ($\mathcal{S}_a^{RCS}(f)$) e R-CSd ($\mathcal{S}_a^{RCSd}(f)$) con $r=0.95$.

fatto che $r=0.95$ spinge la soluzione di 3.1 ad essere il più vicina possibile a 2.8 a causa del vincolo 3.6. Quando i disturbi non sono più trascurabili, il profilo R-CSd esibisce un'attenuazione che ha lo scopo di minimizzare la quantità di energia causata appunto dal rumore. Si noti che questa attenuazione è tanto più forte quanto il rapporto segnale-rumore (SNR) è basso (si veda per esempio nei due tratti prossimi alla frequenza $f/f_s = 0.1$ e $f/f_s = 0.35$) mentre è limitato quando entrambi i segnali e i rumori sono alti (si guardi in questo caso l'intervallo delle frequenze $f/f_s \in [0.2, 0.3]$).

3.2 Metodo della discesa del gradiente proiettato e approccio alla proiezione alternata

Il problema di ottimizzazione in 3.1 è ben formato in quanto comporta una funzione obiettivo lineare e vincoli convessi. Quindi, può essere risolto mediante solutori convessi standard come CVX *. Tuttavia, i risolutori generici tendono a soffrire

*Il tool **CVX** è un sistema di modellazione basato su Matlab per l'ottimizzazione convessa. CVX trasforma Matlab in un linguaggio di modellazione, consentendo di specificare vincoli e obiettivi utilizzando la sintassi di espressione standard di Matlab. Nella sua modalità predefinita,

dell'aumento della dimensionalità del problema che, nel nostro caso, è $O(n^2)$ poiché dobbiamo determinare le voci di una matrice di correlazione $n \times n$. Purtroppo, molto spesso l'applicazione del CS nell'acquisizione comporta dimensionalità del segnale molto grandi (centinaia o migliaia) e le dimensioni dei problemi risultanti possono essere completamente fuori dalla portata dei solutori non specializzati. Ecco perché viene sviluppata qui una procedura iterativa su misura per risolvere 3.1 in cui la complessità di ogni iterazione è $O(n^2)$ e che può essere facilmente eseguita su una workstation standard.

Lo scopo quindi della trattazione di questo capitolo è mostrare come il meccanismo del CS possa essere esteso oltre che garantendo l'abilità nella reiezione del disturbo anche adattando lo stadio di sensing alla classe del segnale; sfruttiamo quindi il metodo della discesa del gradiente proiettato. Il primo passo è la riformulazione di 3.1 in termini di sottoinsiemi convessi dello spazio di $n \times n$ matrici, chiamiamo quindi \mathbf{C}_{PSD} il sottoinsieme di tutte le matrici $n \times n$ simmetriche e semidefinite positive. In questo modo, i vincoli 3.2 e 3.3 del problema possono essere semplicemente scritti come $\mathbf{C}^a \in \mathbf{C}_{PSD}$. Sia anche \mathbf{C}_{En} la classe di matrici che soddisfa il vincolo di energia (Eq. 3.4), $\mathbf{C}_{\tau-Loc}$ l'insieme che soddisfa il vincolo in (Eq. 3.5 e 3.3), ed infine \mathbf{C}_{r-Rak} che soddisfa 3.6 e il secondo vincolo in 3.3. Con questo si traduce il problema R-CSd in 3.1 come:

$$\min_{\mathbf{C}^a} tr(\mathbf{C}^a \mathbf{C}^v) \quad (3.7)$$

$$s.t. \quad \mathbf{C}^a \in \mathbf{C}_{PSD} \cap \mathbf{C}_{En} \cap \mathbf{C}_{\tau-Loc} \cap \mathbf{C}_{r-Rak} \quad (3.8)$$

Osservando che il gradiente $\nabla_{\mathbf{C}^a} tr(\mathbf{C}^a \mathbf{C}^v) = \mathbf{C}^v$ è una matrice costante, possiamo trattare la funzione obiettivo come un classico caso a gradiente limitato soggetta a vincolo convesso. Un tale problema può essere affrontato efficacemente da un

CVX supporta un particolare approccio all'ottimizzazione convessa che chiamiamo programmazione convessa disciplinata. Secondo questo approccio, le funzioni e gli insiemi convessi sono costituiti da un piccolo insieme di regole dall'analisi convessa, a partire da una libreria di base di funzioni e insiemi convessi. I vincoli e gli obiettivi espressi utilizzando queste regole vengono automaticamente trasformati in una forma canonica e risolti. Per ulteriori informazioni sulla programmazione convessa disciplinata, consultare la risorsa [16]. All'interno di questo lavoro questa libreria è sfruttata per la risoluzione dell'eq. 3.1.

approccio con gradiente proiettato discendente [31]. Per applicare questo metodo, sia $\Pi_{\mathbf{C}}$ l'operatore di proiezione su un insieme di matrici \mathbf{C} , cioè l'operatore che prende una qualsiasi matrice $X_{n \times n}$ e la mappa nella matrice $Y = \Pi_{\mathbf{C}}(X)$ in modo tale che $Y \in \mathbf{C}$ e $\|X - Y\|$ sia minima. Quindi, partendo da una matrice iniziale $C_{(0)}^a$ che soddisfa i vincoli in 3.8, si può calcolare in modo iterativo:

$$C_{(t+1)}^a = \Pi_{C^a \in \mathbf{C}_{PSD} \cap \mathbf{C}_{En} \cap \mathbf{C}_{\tau-Loc} \cap \mathbf{C}_{r-Rak}} \left(C_{(t)}^a + \delta_{(t)} C^v \right) \quad (3.9)$$

per una sequenza opportunamente non crescente $\delta_{(0)} \geq \delta_{(1)} \geq \delta_{(2)} \geq \dots$ si può dimostrare che la sequenza di $C_a(t)$ converge alla soluzione ottimale come $t \rightarrow +\infty$. Una buona scelta per $C_a(0)$ è il C_a da 2.9, ovvero la soluzione del problema R-CS.

L'iterazione in 3.9 deve calcolare la proiezione su un insieme convesso che è l'intersezione di un numero di insiemi convessi elementari. Tale operatore di proiezione può essere calcolato mediante il metodo di proiezione alternata [19], [20], [32]. Per spiegare come funziona, facciamo finta di avere solo due insiemi convessi \mathbf{C}_0 e \mathbf{C}_1 e che vogliamo calcolare $\Pi_{\mathbf{C}_0 \cap \mathbf{C}_1}(C)$ per qualche matrice C . La proiezione è calcolata in modo iterativo producendo una sequenza di matrici $P_{(j)}$ e utilizzando due sequenze ausiliarie di matrici offset $\Delta P_{0,(j)}$ e $\Delta P_{1,(j)}$. Gli insiemi inizializzati valgono $P_{(0)} = C$ e $P_{0,(0)} = P_{1,(0)} = 0$. Quindi, il passo s -esimo della procedura calcola:

$$P_{(2s+1)} = \Pi_{\mathbf{C}_0} \left(P_{(2s)} - \Delta P_{(2s)} \right) \quad (3.10)$$

$$\Delta P_{0,(s+1)} = P_{(2s+1)} - P_{(2s)} - \Delta P_{0,(s)} \quad (3.11)$$

$$P_{(2s+2)} = \Pi_{\mathbf{C}_1} \left(P_{(2s+1)} - \Delta P_{1,(s)} \right) \quad (3.12)$$

$$\Delta P_{1,(s+1)} = P_{(2s+2)} - P_{(2s+1)} - \Delta P_{1,(s)} \quad (3.13)$$

Come $s \rightarrow +\infty$ allora $P_{(2s+1)} \rightarrow P_{(2s)} \rightarrow \Pi_{\mathbf{C}_0 \cap \mathbf{C}_1}(C)$.

Lo schema si generalizza direttamente a più di due insiemi convessi elementari nell'intersezione e rivela che la proiezione generale in 3.9 può essere calcolata a partire dai singoli operatori di proiezione su \mathbf{C}_{PSD} , \mathbf{C}_{En} , $\mathbf{C}_{\tau-Loc}$, \mathbf{C}_{r-Rak} . Intuitivamente le due sequenze di offset ausiliarie 3.11, 3.13 vengono introdotte a causa della complessità dell'unione di complicati insiemi convessi (come vediamo intuitivamente in fig.3.3) per cui si ha il rischio che la "classica" proiezione non sia

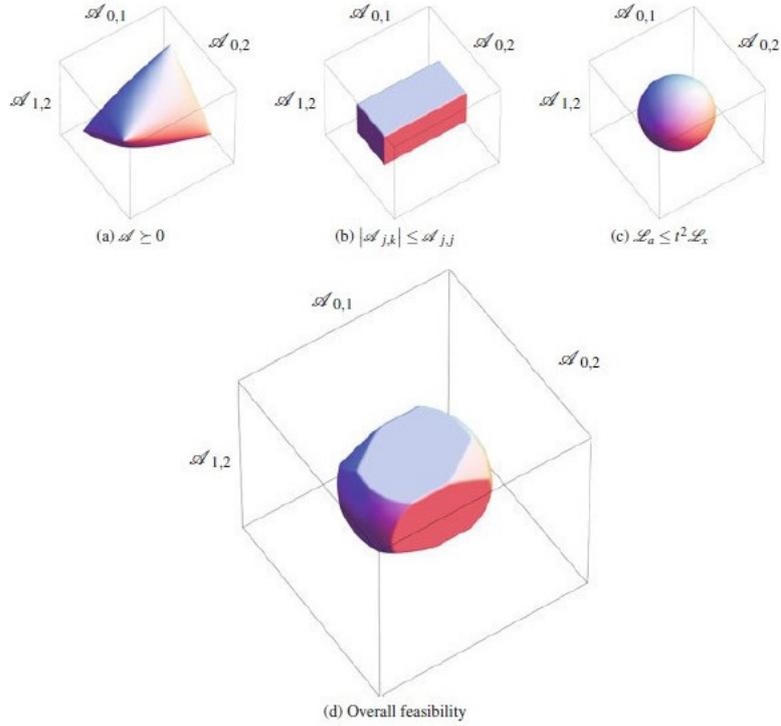


Figura 3.3: Lo spazio delle soluzioni come intersezione dei diversi vincoli.

esattamente quella vera. La Figura 3.4 mostra questo con un esempio. In questo caso C_0 è un disco e C_1 è un mezzo piano. A partire da $\zeta^{(t,0)}$ le prime due proiezioni sono sufficienti per produrre un punto in $C_0 \in C_1$ che è invariato da ulteriori proiezioni e quindi è il limite della sequenza. Purtroppo, un tale limite non è la vera proiezione. [8]

Supponendo che C possa essere spettralmente scomposto come $C = E\Lambda E$, dove E è la matrice degli autovettori ortonormali e Λ è la matrice diagonale dei corrispondenti autovalori, allora

$$\Pi_{C_{PSD}}(C) = E \max\{\Lambda, 0\} E^T \quad (3.14)$$

Il vincolo energetico 3.4, 3.5 e il vincolo minimo di rakeness 3.6 definiscono due sottospazi lineari e gli operatori di proiezione corrispondenti sono semplici trasfor-

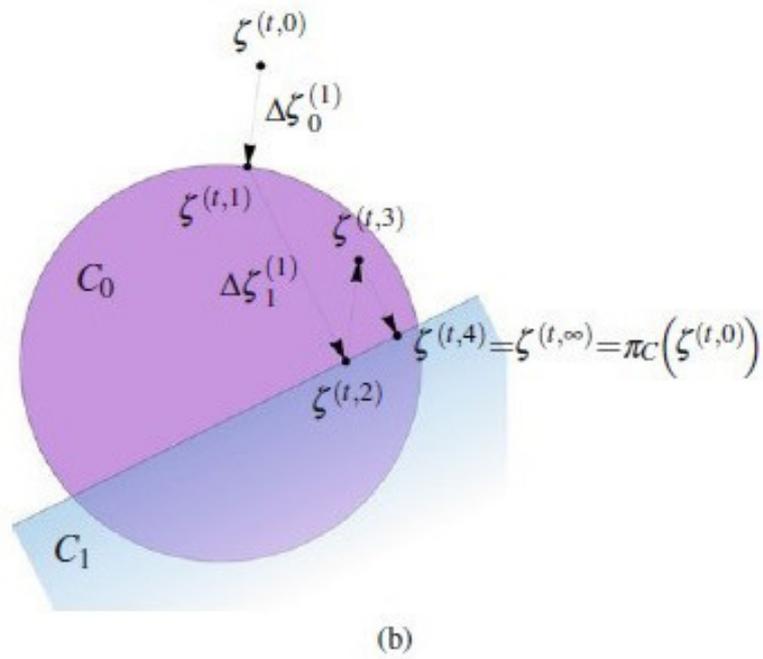
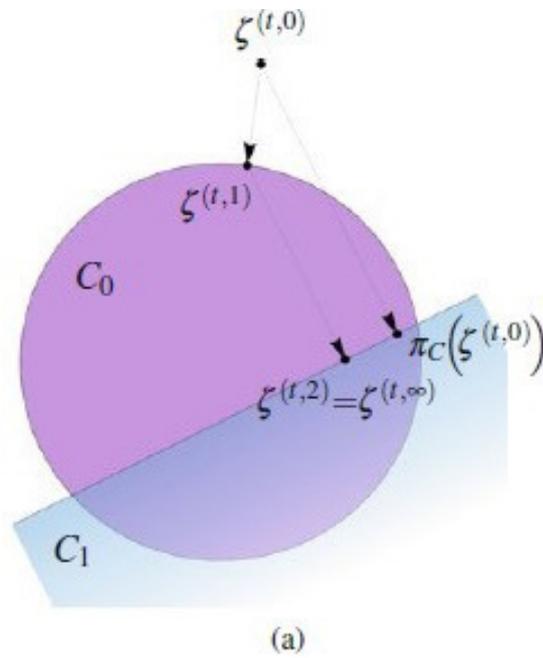


Figura 3.4: Il metodo di proiezione alternata (a) e la sua variante che garantiscono la convergenza alla proiezione vera (b).

mazioni affini che seguono la direzione del vincolo del gradiente.

$$\Pi_{\mathbf{C}_{En}}(C) = C + \frac{1 - \text{tr}(C)}{n} I_n \quad (3.15)$$

$$\Pi_{\mathbf{C}_{r-Rak}}(C) = C + \begin{cases} \frac{r\rho^* - \text{tr}(CC^x)]}{\text{tr}(C^x)} & \text{if } \text{tr}(CC^x) \leq r\rho^* \\ 0 & \text{otherwise} \end{cases} \quad (3.16)$$

Assumendo che $\text{tr}(C) = 1$, (il limite di localizzazione 3.5) sia un vincolo quadratico omogeneo e che la proiezione possa essere ottenuta mediante ridimensionamento:

$$\Pi_{\mathbf{C}_{\tau-Loc}}(C) = C \begin{cases} \frac{\tau\mathcal{L}_x + 1/n}{\text{tr}(C^2)} & \text{if } \text{tr}(C^2) \geq \tau\mathcal{L}_x + 1/n \\ 1 & \text{otherwise} \end{cases} \quad (3.17)$$

Approfondiamo questo concetto riportando il teorema che dimostra la convergenza del metodo appena descritto. IL teorema ci dice che data una certa dimensionalità p , sia $c : \mathfrak{R}^p \rightarrow \mathfrak{R}$ la funzione di costo convessa e $C \subset \mathfrak{R}^p$ l'insieme convesso delle soluzioni che definisce il problema di ottimizzazione

$$\min_{\psi} c(\psi) \quad \text{s.t.} \quad \psi \in C \quad (3.18)$$

la cui soluzione è $c(\psi^*)$ per qualche $\psi^* \in C$. Sia Π_C l'operatore che proietta su C , definito come:

$$\Pi_C(\psi) = \arg \min_{\xi} \|\xi - \psi\|_2 \quad \text{s.t.} \quad \xi \in C \quad (3.19)$$

Partendo da un qualsiasi $\psi^{(0)} \in C$ definito come

$$\psi^{(t+1)} = \Pi_C(\psi^{(t)} - \alpha^{(t)} \nabla_{\psi} c(\psi^{(t)})) \quad (3.20)$$

per alcuni coefficienti $\alpha^{(t)} > 0, t = 0, 1, \dots$, la cui sequenza è tale che $\sum_{t=0}^{\infty} \alpha^{(t)} = \infty$, ma $\sum_{t=0}^{\infty} (\alpha^{(t)})^2 < \infty$, se $\max_{0 \leq t < T} \|\nabla_{\psi} c(\psi^{(t)})\|_2 < \infty$, allora

$$\lim_{T \rightarrow \infty} c(\psi^*) - \min_{0 \leq t < T} c(\psi^{(t)}) = 0 \quad (3.21)$$

In termini meno formali, 3.20 consente di passare da una soluzione candidata alla successiva in due fasi. Innanzitutto, si fa un passo lungo la direzione del gradiente della funzione di costo per diminuirne il valore. In secondo luogo, poiché questo potrebbe fornire un punto fuori dallo spazio dei risultati, l'operatore di proiezione viene utilizzato per trovare la sua approssimazione più vicina ammissibile. La

convergenza è garantita se le lunghezze dei gradini non svaniscono troppo velocemente e le sfumature sono limitate, cosa che accade nel nostro caso dato che $\nabla_{\mathcal{A}tr}(\mathcal{A}\mathcal{X}) = \mathcal{X}$ è costante. Chiaramente, il punto critico qui è il calcolo di C la cui complessità dipende dalla sua struttura. In particolare, C è tipicamente dato come l'intersezione di un certo numero di sottoinsiemi convessi più semplici come abbiamo visto dall'analisi sopra descritta.

Per valutare le risorse computazionali necessarie per questo approccio, la Tabella I riporta il rapporto temporale CPU tra il risolutore CVX generico e il metodo proposto per n valori compresi tra 32 e 128. Come previsto, l'approccio su misura supera ampiamente un risolutore di uso generale, dove per $n = 128$ il tempo di CPU dell'approccio proposto è $5,31\text{secondi}^2$. Oltre a questo, per valori di n maggiori di 128 il risolutore CVX non è in grado di risolvere il problema [†]. [18]

n	32	64	96	128
CPU_{ratio}	3.3	5.9	5.8	15.9

Tabella 3.1: *Rapporto tempo computazionale, CPU_{ratio}, tra il solutore "general purpose" e l'algoritmo del gradiente proiettato discendente*

Un'ultima osservazione consta del fatto che varie analisi sono state effettuate per cercare di ottimizzare il valore di α , lo step che ci dice di quanto la funzione obiettivo deve scendere (perchè stiamo cercando un minimo 3.1) ad ogni iterazione (si osservi la fig 3.5 esplicativa di come lavora l'approccio al gradiente e di quanto sia importante riuscire a selezionare uno step il più coerente possibile al fine di velocizzare l'algoritmo). La maggior parte degli algoritmi standard imposta questo parametro in maniera proporzionale alla derivata seconda della funzione obiettivo o alla differenza delle derivate prime prese in istanti successivi di tempo, nel nostro caso però questo valore è nullo essendo la derivata prima di 3.1 una costante. Per questo motivo in questo lavoro si è inizializzato α con dei valori di tentativo in grado di dare una risposta il più coerente possibile al problema, comunque di valori all'incirca pari a $10^{-1}/10^{-2}$. Valore che viene decrementato di una certa

[†]Tutti i tempi della CPU sono valutati su un PC standard con un processore Intel i7-6820HQ e 16 GB di RAM

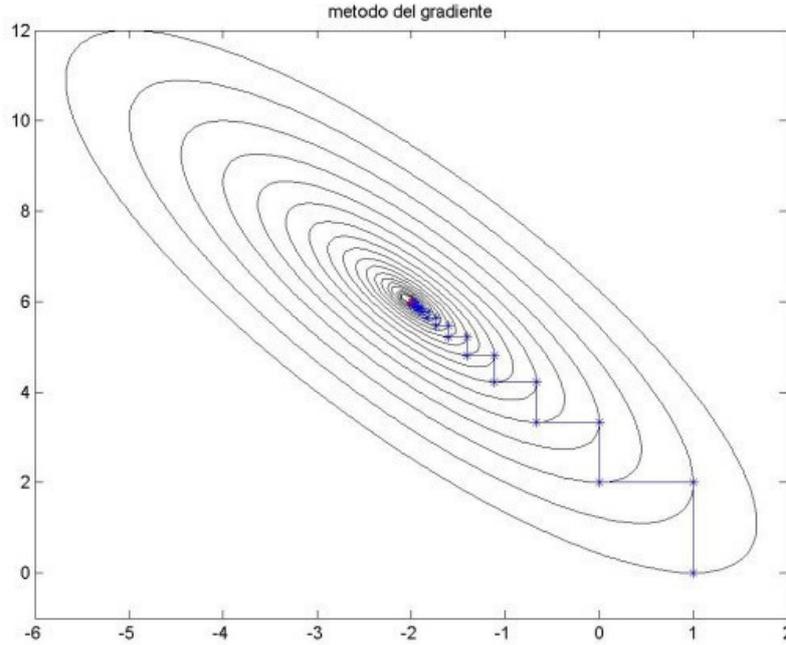


Figura 3.5: Figura semplificata che illustra la l’approccio della discesa del gradiente applicato ad una superficie

quantità, possibilmente variabile e scelta ad hoc in base alla classe del segnale ogni qualvolta la distanza matriciale, che tiene traccia di quanto ci siamo spostati in ogni iterazione, è maggiore di una certa soglia. Nel caso EEG in esame e sviluppato nell’ultimo capitolo di questo lavoro, il valore di discesa è stato stabilito essere pari a $1/2$ del valore precedente.

3.3 Toy Case: risultati e analisi

Per verificare l’efficacia del metodo proposto, consideriamo una finestra composta da $n = 512$ campioni di Nyquist, ovvero il caso dove il soluto CVX fallisce in quanto in grado di lavorare solo per finestre per un massimo di $n = 256$. I profili spettrali dei segnali in input e dei disturbi, sono gli stessi usati in Fig. 3.2 mentre la potenza spettrale per R-CS e R-CSd sono riportati in Fig.3.6. Confrontando i profili spettrali di $(\mathcal{S}_a^{CS}(f))$ e $(\mathcal{S}_a^{R-CS}(f))$ per $n = 64$ (Fig.3.2) e quelli in fig. 3.6, si vede come l’incremento della dimensionalità del segnale permetta di ottenere dei profili spettrali più netti e definiti. Come prima, usiamo un segnale di input

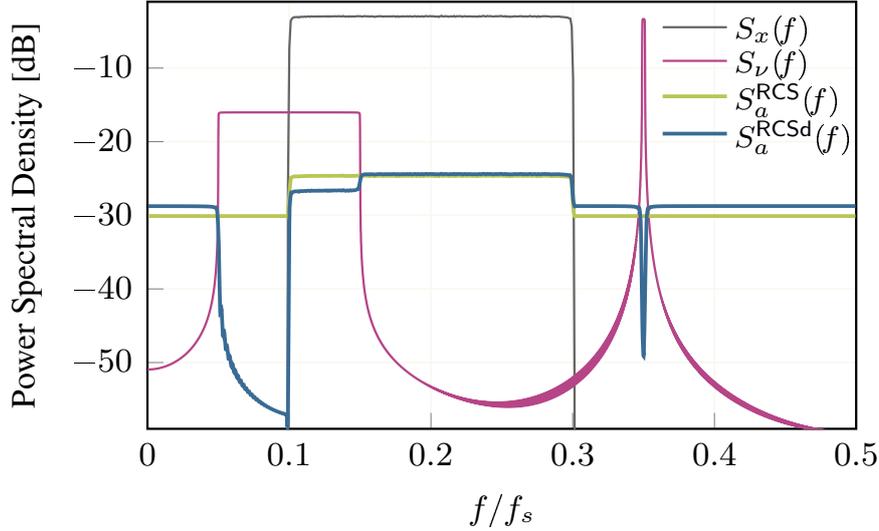


Figura 3.6: Con $n=512$, la densità spettrale di potenza della classe di segnale in input ($\mathcal{S}_x(f)$) e dei disturbi ($\mathcal{S}_v(f)$) insieme con i profili per entrambi R-CS ($\mathcal{S}_a^{RCS}(f)$) e R-CSd ($\mathcal{S}_a^{RCSd}(f)$) con $r=0.95$.

passa banda mentre per il profilo spettrale del disturbo è composto da un contributo passa banda, concentrato in una regione dove si ha un parziale overlap con la PSD del segnale di input, insieme con una banda stretta. Questo significa che entrambi, segnale e disturbo sono localizzati. In particolare ogni x è anche k -sparso rispetto alla base di Fourier con $k=51$, mentre ogni v è un'istanza di un vettore Gaussiano casuale a media nulla. Il profilo spettrale sopra menzionato determina i profili di correlazione \mathcal{C}^x e \mathcal{C}^v . Queste matrici sono utilizzate in 2.9 e nell'algoritmo proposto del gradiente per la risoluzione di 3.1.

I segnali sono ricostruiti risolvendo 2.2 per mezzo del toolbox SPGL1[‡] dove l'errore di ricostruzione per ogni prova viene calcolato per mezzo del Reconstructed Signal To Noise Ratio (RSNR)

$$RSNR = \frac{\|x\|_2}{\|x - \hat{x}\|_2} \quad (3.22)$$

Per le performance dell'intero sistema, due diverse figure di merito vengono considerate: la *Average Reconstructed Signal To Noise Ratio (ARSNR)*, che valuta la

[‡]Il tool **spgl1-1.9** è un solutore Matlab per problemi ai minimi quadrati a larga scala e con norma 1. È progettato per risolvere quindi problemi del tipo 2.2 ed è proprio a questo scopo che viene richiamato in questo lavoro. SPGL1 è adatto per problemi che si trovano nel dominio complesso. La teoria alla base di SPGL1 è descritta nei documenti [14] e [15].

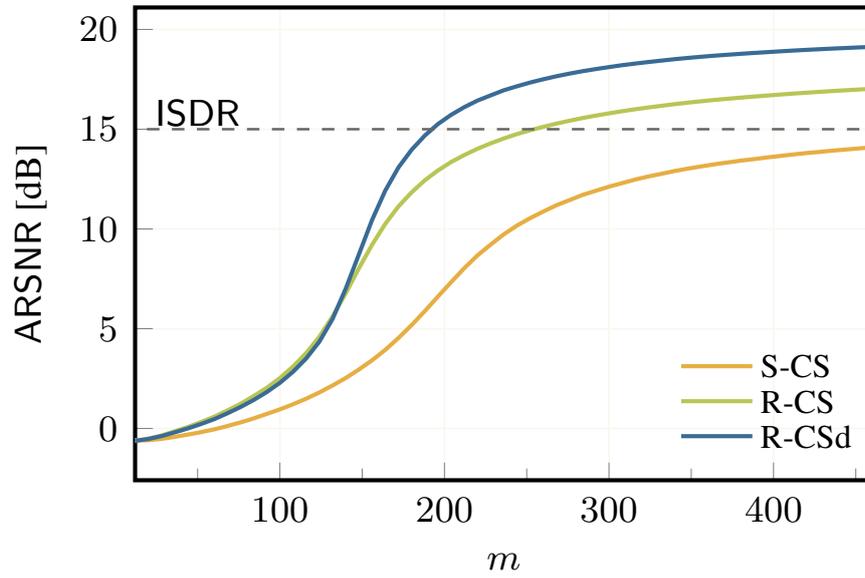


Figura 3.7: ARSNR in funzione del numero di misure m , per i tre approcci CS coosiderati con $n= 512$, $r=0.95$ e $ISDR= 15dB$.

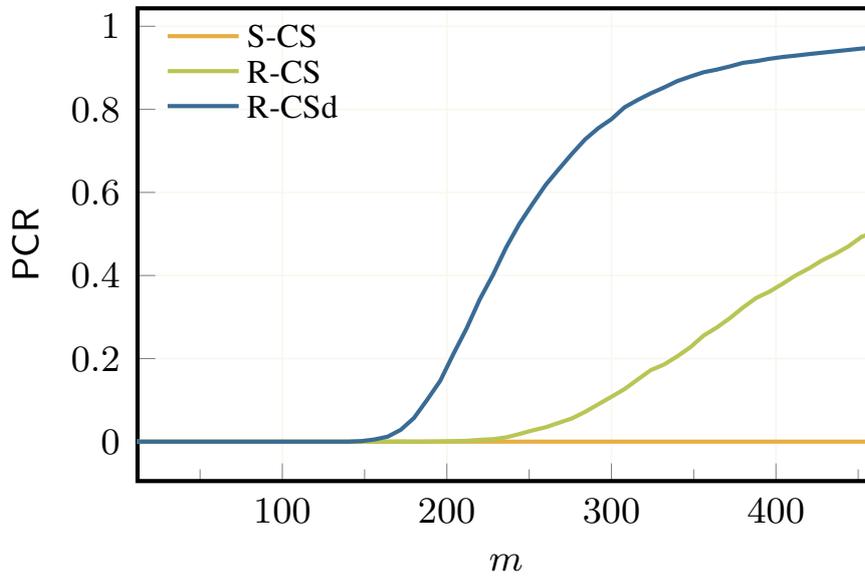


Figura 3.8: PCR in funzione del numero di misure m , per i tre approcci CS coosiderati con $n= 512$, $r=0.95$ e $RSNR_{min} = 17dB$.

media empirica dei valori osservati da $RSNR$, e la *Probability of Correct Reconstruction* (PCR), cioè la probabilità che i campioni osservati da $RSNR$ eccedano una soglia specifica $RSNR_{min}$.

$$PCR = Pr(RSNR > RSNR_{min}) \quad (3.23)$$

Entrambi sono stimati su 3000 prove Montecarlo usando diverse istanze di segnale e disturbo e dove la matrice di sensing viene generata in accordo agli algoritmi R-CS e R-CSd con $r=0.95$. Anche S-CS è stato incluso utilizzando una matrice di sensing le cui righe sono istanze di variabili aleatorie gaussiane indipendenti, a media nulla e con varianza unitaria. Vengono valutati intervalli di m a partire da intervalli molto piccoli fino a un numero pari a 400. Da notare che l'effetto di v nel calcolo delle misure in 2.1 è rilevato dalla figura di merito ISDR (Intrinsic Signal-To-Disturbance Ratio) definito come

$$ISDR = \left(\frac{\|x\|_2}{\|v\|_2} \right)_{dB} \quad (3.24)$$

dove $ISDR = 15dB$ e $RSNR_{min} = 17dB$. La figura 3.7 mostra le performance in termini di ARSNR mentre Fig. 3.8 quelli in termini di PCR. Per entrambe le figure di merito R-CSd è chiaramente quella con risultati migliori rispetto entrambi R-CS e S-CS. Questi risultati confermano l'abilità del metodo R-CSd di rigettare parzialmente i disturbi che affettano il segnale di input senza dover inserire stadi di filtri aggiuntivi. Evidenziamo come la dimensionalità del segnale in esame è tale che il tradizionale solutore general purpose non è in grado di risolvere il problema di ottimizzazione (Eq. 3.1), mentre l'algoritmo qui proposto converge alla soluzione in una quantità di tempo ragionevole.

Capitolo 4

Caso di studio EEG

I biopotenziali elettroencefalografici, la cui rilevazione prende il nome di elettroencefalografia (EEG), sono una espressione dell'attività elettrica spontanea della corteccia cerebrale. Essi possono essere misurati attraverso elettrodi posti sullo scalpo del soggetto sotto forma di pattern temporali di potenziali elettrici.

4.1 Segnali EEG

Il termine elettroencefalogramma è usato oggi comunemente per descrivere la registrazione dei potenziali elettrici cerebrali. Questi ultimi sono classificabili in tre categorie:

1. attività spontanea;
2. potenziali evocati;
3. eventi bioelettrici provocati da singoli neuroni.

L'attività spontanea determina propriamente l'elettroencefalogramma ed è misurata sullo scalpo o, in casi sperimentali, direttamente sulla corteccia cerebrale; nel secondo caso, essa è definita in modo più appropriato elettrocorticogramma. L'attività spontanea è continuamente presente nel cervello. L'ampiezza massima dei potenziali EEG può essere di circa $10 - 100\mu V$ nel caso di misura sullo scalpo, mentre ampiezze di circa $300\mu V$ (o superiori) possono essere possibili per registrazioni sulla superficie della corteccia cerebrale. La banda del segnale EEG si

estende da circa 1 a $50Hz$ anche se tipicamente la maggiore quantità del contenuto informativo si estende fino a $30Hz$.

I potenziali evocati, che sono poi quelli che vengono analizzati in questo lavoro, sono quelle componenti del segnale EEG che si presentano in risposta ad uno stimolo (visivo, tattile, uditivo, elettrico, ecc...). Questi segnali sono in genere di piccola ampiezza rispetto all'attività spontanea, divenendo confrontabili con il livello di rumore e risultando quindi non facilmente registrabili. Per indurre e analizzare i potenziali evocati si utilizzano in genere treni di impulsi di stimolazione, al fine di registrare segnali multipli e farne una media, così da aumentare il rapporto segnale-rumore.

4.1.1 Genesi dei potenziali elettroencefalografici

A differenza di quanto si possa pensare, i segnali EEG non derivano dalla somma dei potenziali di azione degli assoni neuronali, bensì da potenziali dendritici incrociati. I potenziali misurati, infatti, sono originati da medie nello spazio dei campi elettrici dendritici negli strati superficiali della corteccia. Questi campi sono generati dagli alberi dendritici delle cellule e sono composti dalla sommatoria di potenziali postsinaptici eccitatori (EPSP) e inibitori (IPSP). Dal momento che il campo di ogni cellula può avere una fase opposta a quello della cellula vicina, la sommatoria potrebbe tendere a zero. Questo non accade perché gruppi di cellule sono coinvolte in attività sincronizzate e il risultato di questa sincronizzazione è appunto il segnale EEG. Gli impulsi originati da stimolazioni degli organi periferici causano attività sincronizzata in particolari aree sensoriali riceventi: questi potenziali rappresentano le cosiddette *risposte evocate (ER)*. In condizioni normali, i potenziali d'azione trasmessi lungo gli assoni contribuiscono poco al segnale registrato sulla corteccia, poiché essi sono asincroni per la maggior parte degli assoni che corrono in molte direzioni rispetto alla superficie. Quindi il loro contributo al potenziale di superficie è trascurabile. Si ha un'eccezione nel caso delle risposte evocate da stimoli simultanei. In tutte le altre situazioni il segnale EEG è dovuto principalmente all'effetto dei potenziali postsinaptici locali (eccitatori o inibitori) delle cellule corticali. Ogni potenziale registrabile dalla superficie rappresenta la

misura di una differenza di potenziale netta (corrente elettrica moltiplicata per resistenza elettrica) tra l'elettrodo di misura e quello di riferimento. Se tutti i dendriti delle cellule corticali fossero disposti casualmente nella corteccia, la corrente totale sarebbe pressoché nulla. Se ne deduce che ogni variazione di potenziale sulla superficie corticale deve essere dovuta all'arrangiamento ordinato e simmetrico di alcune classi di cellule nella corteccia.

4.1.2 Sistemi di posizionamento EEG

Il sistema di registrazione dei segnali EEG, chiamato elettroencefalografo, è costituito da una unità di acquisizione dei segnali misurati sullo scalpo, da una unità di elaborazione dei segnali e da una unità di visualizzazione/memorizzazione dei dati. L'unità di acquisizione adotta elettrodi di misura tipicamente alloggiati su di una apposita cuffia, la quale viene adattata sulla testa del paziente. Gli elettrodi di superficie per EEG possono essere fissati alla cute con l'ausilio di collarini adesivi, cerotti o, come menzionato, di una cuffia apposita. Gli elettrodi possono essere di diversa tipologia, tra cui:

- coppetta, realizzati in stagno e in argento rivestito di cloruro d'argento (Ag/AgCl), del diametro di 10 mm;
- elettrodi in Ag/AgCl biadesivi monouso (2x3 cm e 1.5x2 cm);
- elettrodi ad anello in acciaio inossidabile.

Per eseguire registrazioni da un numero considerevole di punti dello scalpo, si ricorre spesso all'utilizzo di cuffie in tessuto elasticizzato (per garantire un'opportuna aderenza) con elettrodi incorporati. L'esatta disposizione degli elettrodi sullo scalpo è regolata dallo standard internazionale 10/20 (descritto nel seguito). Per quanto riguarda l'apposizione degli elettrodi, si ricorre a due derivazioni standard, a seconda degli specifici requisiti sperimentali: nella derivazione monopolare (o unipolare) un elettrodo viene posto in un sito attivo, mentre l'altro (elettrodo di riferimento) in un sito elettricamente neutro (ad es. il mastoide, la punta del naso, il lobo dell'orecchio, il mento).

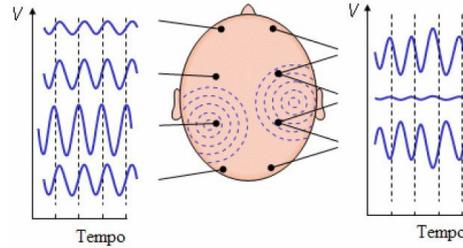


Figura 4.1: Esempi di derivazione unipolari (sinistra) e bipolari (destra)

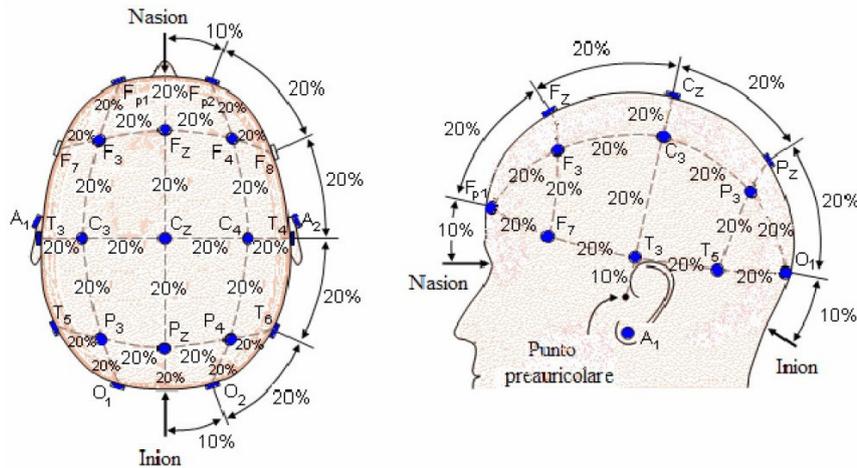


Figura 4.2: Standard internazionale 10/20 per il posizionamento degli elettrodi EEG sullo scalpo

Con una lettura unipolare il potenziale di ogni elettrodo viene misurato rispetto all'elettrodo neutro o alla media di tutti gli elettrodi (Fig. 4.1). Questo tipo di registrazione evidenzia pertanto il livello assoluto di attività elettrica sottostante il sito attivo. Nella derivazione bipolare, invece, entrambi gli elettrodi sono posti su siti attivi dell'area di interesse e il segnale rilevato corrisponde alla differenza che emerge tra le attività dei due siti. Convenzionalmente, gli elettrodi sono posizionati sulla cute avendo cura di rispettare alcune corrispondenze tra aree corticali e sito di apposizione; a questo scopo si segue la dislocazione standardizzata nota come Sistema Internazionale Standard 10/20. Esso prevede il posizionamento di 21 elettrodi sulla superficie dello scalpo, secondo la modalità convenzionale mostrata in figura 4.2 [10].

Le posizioni sono determinate come di seguito descritto: 10% oppure 20% si

riferisce al 100% della distanza tra due punti di repère cranici "inion" (prominenza alla base dell'osso occipitale) e "nasion" (attaccatura superiore del naso), questa distanza di solito va da 30 a 36 cm con grande variabilità interpersonale. Vengono collocati da 10 a 20 elettrodi e una massa, lungo cinque linee: 1) P1: longitudinale esterna e 2) P2: long. interna di destra, 3) centrale, 4) P1: longitudinale esterna e 5) P2: long. interna di sinistra. La linea trasversa T4-C4-Cz-C3-T3 (risultante delle precedenti) viene denominata montaggio P3, ed anch'essa deve seguire la regola del 10-20%. Gli elettrodi fronto-polari sono collocati al 10% (3-4 cm) della distanza I-N, sopra le sopracciglia, i frontali vengono collocati sulla stessa linea dei fronto-polari, più sopra del 20%, poi vengono i centrali (+ 20%), infine i parietali (+ 20%) e gli occipitali (+ 20%), con questi si arriva al 90% della distanza nasion-inion, ad una distanza del 10% dall'inion. Alla posizione che ogni elettrodo occupa sullo scalpo fa riferimento una sigla. Le sigle che individuano la posizione di un elettrodo sono formate da una/due lettere, che permettono di identificare la regione della corteccia esplorata (Fp: frontopolare; F: frontale; C: centrale; P: parietale; T: temporale; O: occipitale) e da un numero (o una z) che identifica l'emisfero (numeri dispari: sinistra; numeri pari: destra; z: linea mediana).

4.1.3 Potenziali evocati

I potenziali evocati o risposte evocate, vengono registrati come l'elettroencefalogramma con elettrodi di superficie posizionati sulla testa. Mentre l'EEG descrive l'attività elettrica cerebrale di base, un potenziale evocato consiste in una variazione specifica del segnale bioelettrico conseguente alla stimolazione di una via sensoriale o ad un evento motorio. Esso è costituito da oscillazioni del potenziale elettrico e ha una forma d'onda caratterizzata da una serie di deflessioni positive o negative, definite *componenti*. Ciascuna di queste componenti viene normalmente considerata come la presenza di potenziali post sinaptici (eccitatori o inibitori) sincroni di un gruppo di neuroni corticali, in grado di generare campi sufficientemente ampi da essere registrati in superficie. La localizzazione delle componenti permette quindi di identificare quale area corticale è attiva in seguito ad un particolare stimolo. Mentre le variazioni spontanee sono visibili direttamente nel tracciato

EEG, le variazioni evocate hanno un'ampiezza molto bassa (tipicamente intorno a $1 - 3\mu V$) e sono mascherate dall'attività di fondo. Per questo motivo, si provvede a registrare i potenziali mentre vengono presentati stimoli ripetitivi (visivi, uditivi, etc.). In seguito, i tracciati vengono sottoposti a procedure standard di elaborazione come la scomposizione in "epoche" discrete e sincronizzate con gli eventi stimolanti. I campioni del segnale vengono quindi mediati (*averaging*): tale tecnica costituisce l'essenza di base delle metodiche di elaborazione dei potenziali evocati. Mediante l'*averaging* è possibile ottenere la risposta media del cervello allo stimolo o all'evento, tramite la sommazione di numerose epoche sincronizzate con lo stimolo o l'evento stesso. In questo modo, l'attività evocata dall'evento si somma algebricamente con l'attività di fondo, la quale, essendo fondamentalmente casuale rispetto all'evento, tende a ridursi o annullarsi; in tal modo, l'attività evocata viene posta in risalto rispetto al rumore di fondo. Pertanto, la risposta media è la risposta evocata, le cui componenti (picchi positivi o negativi) sono riconducibili ai vari stadi di processamento dell'informazione sensoriale o evento-correlata nel cervello. Una volta isolata la risposta evocata, si procede con l'analisi delle sorgenti delle sue componenti, studiando la loro localizzazione. A tal scopo si sfruttano le variazioni nello spettro di potenza del segnale causate da uno stimolo sensoriale o evento. Il metodo consiste nel calcolo dello spettro dei segnali conseguenti lo stimolo e nel confronto con lo spettro dei segnali antecedenti lo stimolo. In base al tipo di stimolo (visivo, uditivo, ecc.) e al posizionamento dell'elettrodo, alcune bande di frequenza potranno mostrare delle variazioni, rispetto allo spettro pre-stimolo. I potenziali evocati possono essere distinti in due tipi fondamentali:

- *Stimolo-correlati*: dipendono dalle caratteristiche fisiche dello stimolo. La loro latenza è nell'ambito temporale della percezione. In generale, le caratteristiche di tali segnali dipendono dalle proprietà fisiche dello stimolo applicato (ad esempio tono e intensità per il sistema uditivo, contrasto, luminanza e frequenza spaziale per il sistema visivo, intensità e modalità di stimolazione per il sistema somatosensoriale, ecc.).
- *Evento-correlati o ERPs* (Event Related Potentials): la loro genesi è funzione del contesto psicologico (evento) in cui avviene la stimolazione. La loro

latenza è nell'ambito temporale dei fenomeni legati all'attenzione e alla cognizione. Tali potenziali, a differenza da quelli stimolo-correlati, dipendono dal contenuto informativo dello stimolo e compaiono solo quando il soggetto presta attenzione allo stimolo stesso e gli attribuisce un significato.

In diagnostica, i potenziali evocati più frequentemente utilizzati sono quelli somato-sensoriali (indotti da stimolazione elettrica tipicamente del nervo mediano del braccio o del nervo tibiale della gamba), quelli visivi (stimolazione tramite ad esempio un'immagine a scacchiera in movimento su un monitor) e quelli uditivi (stimolazione acustica ad esempio mediante "click" di basso volume applicati tramite una cuffia). I potenziali evocati esaminano l'integrità delle vie di conduzione nervosa periferiche e centrali. La forma e la latenza del potenziale possono rivelare alterazioni delle vie afferenti e permettono di evidenziare un difetto sensoriale, quantificandone anche l'entità. In particolare trattiamo *Potenziali evocati uditivi*. Lo stimolo utilizzato per i potenziali acustici è un "click" di breve durata con frequenze da 500 a 2000Hz ed intensità di 65 – 70dB; la registrazione viene fatta con un elettrodo indifferente posto sul padiglione auricolare dello stesso lato. I potenziali evocati acustici sono usati per diagnosticare le differenti sordità; in particolare nella sordità di conduzione (orecchio medio) la latenza aumenta proporzionalmente al deficit uditivo, mentre nella sordità sensoriale (orecchio interno) la latenza aumenta in misura minore e per livelli alti di deficit uditivo; in entrambi i casi le due latenze interpicco sono normali.

4.2 Filtraggio

Introduciamo in questo paragrafo le tecniche di filtraggio per mezzo di filtri classici ma anche filtri ad hoc per il segnale EEG. Si fa attenzione però nel distinguere i filtraggi che sono sostituibili con l'algoritmo da noi proposto e quelli che non lo sono. Infatti la Rakeness con Denoising è un algoritmo che sfrutta la correlazione nel tempo dei segnali mentre alcuni dei filtraggi avanzati proposti (vedi TSPCA, SNS e DSS descritti nei successivi paragrafi) sfruttano combinazioni su segnali di più sensori per determinare il disturbo da eliminare e prevedono tecniche complesse

di analisi quali ad esempio la PCA. Nel caso in esame quindi si analizza come sostituibile, il filtro passa alto utilizzato per l'eliminazione delle basse frequenze, in particolare per testimoniare l'effettivo funzionamento di questa parte di analisi, ciò che andremo a vedere è se l'algoritmo riesce a togliere la parte continua del segnale. I tre filtri complessi TSPCA, SNS e DSS vengono proposti a completamento della catena di analisi del segnale EEG.

4.2.1 Filtraggio pre-acquisizione

Le fasi di pre-elaborazione introdotte sui campioni acquisiti prima di ogni ulteriore analisi sono le seguenti:

- Filtraggio passa basso anti-aliasing a $200Hz$;
- sottocampionamento $\times 4$ (da $2048Hz$ fino a $512Hz$);

Successivamente il primo passo di analisi è l'utilizzo di un filtro passa alto per rimuovere il disturbo della componente continua con una frequenza di taglio pari a 1 Hz. Il processo del filtraggio è un'operazione offline solitamente eseguita dopo che il segnale è stato acquisito e memorizzato come fase di analisi obbligatoria al fine di eliminare le diverse componenti di rumore che influiscono sulle misurazioni. Successivamente si introducono filtri specifici per questo tipo di segnale, in particolare per eliminare: rumore ambientale (rumore esterno causato da interferenze elettromagnetiche e linee elettriche 50 – 60 Ripple Hz), rumore del sensore (problemi come la variazione momentanea del contatto con la pelle) e rumore fisiologico (componenti indesiderati di origine biologica). In questo lavoro vengono considerati gli algoritmi di filtraggio descritti in [11], [12], [13] come lo stato dell'arte per il denoising del segnale EEG, a cui faremo riferimento rispettivamente come Time-Shift PCA (**TSPCA**), Sensor Noise Suppression (**SNS**) e Denoising Source Separation (**DSS**). Il segnale originale, insieme alle sue versioni filtrate, dopo ogni fase di denoising, viene organizzato in epoche di lunghezza di un secondo e memorizzato in matrici 3D le cui dimensioni sono $N \times nChans \times nEpochs$ dove:

- N : numero di campioni (512);

- *nChans*: numero di canali ($32 + 2$);
- *nEpochs*: numero di epoche (699);

4.2.2 Filtraggio post-acquisizione

Time-shift PCA (TSPCA)

É disponibile una vasta gamma di strumenti per la riduzione del rumore: filtri spaziali, filtri spettrali, applicare la media etc. . . Insieme, consentono misurazioni di alta qualità dell'attività cervello. Tuttavia, alcuni hanno degli inconvenienti che interferiscono con l'osservazione della risposta morfologica del cervello. Per gli altri, la rimozione preventiva del rumore ambientale consentirebbe loro di essere ottimizzati ai fini dell'analisi della fonte cerebrale. Alcuni sistemi EEG sono dotati di sensori di riferimento che misurano i campi ambientali. La regressione dei segnali dei sensori cerebrali sul sottospazio esteso dai segnali dei sensori di riferimento consente di attenuare il contributo del rumore ambientale senza la necessità di filtraggio spettrale o filtraggio spaziale del vettore dei sensori cerebrali. Diversi metodi sono stati proposti a tale scopo. Supponendo che le sorgenti di rumore siano distanti e i loro campi omogenei, tre sensori dovrebbero essere sufficienti per catturare le tre componenti spaziali del campo di rumore, indipendentemente dal numero di sorgenti. Tuttavia un numero maggiore di sensori di riferimento può essere utile se i gradienti di campo differiscono tra le sorgenti di rumore. Supponendo la propagazione istantanea alle frequenze relativamente basse di interesse, le risposte di due sensori alla stessa componente di rumore differiscono solo da un fattore scalare. Ci si dovrebbe quindi aspettare che le tecniche di proiezione siano altamente efficaci. Tuttavia, la schermatura elettromagnetica è nota per essere dipendente dalla frequenza e l'elettronica (loop del flusso di flussaggio, filtri hardware) può introdurre disallineamento convolutivo tra i canali, nel qual caso la regressione scalare non funzionerebbe bene. Il metodo da descrivere estende queste tecniche aumentando la gamma di segnali di riferimento mediante versioni ritardate della stessa. La combinazione lineare di segnali ritardati costituisce, in effetti,

un filtro di risposta impulsiva (FIR) finita che viene applicato a ciascun segnale di riferimento prima della sottrazione da ciascun segnale del sensore cerebrale.

Osserviamo i segnali del sensore cerebrale K e i segnali del sensore di riferimento J . Denotando i vettori con lettere in grassetto, $\mathbf{S}(t) = [s_1(t), \dots, s_K(t)]^T$, i segnali del sensore cerebrale K riflettono una combinazione di attività cerebrale, rumore ambientale e rumore del sensore:

$$\mathbf{S}(t) = \mathbf{S}_b(t) + \mathbf{S}_e(t) + \mathbf{S}_s(t) \quad (4.1)$$

mentre i sensori J di riferimento $\mathbf{R}(t) = [r_1(t), \dots, r_J(t)]^T$ riflettono solo rumore. I segnali sono campionati e viene usato t per rappresentare l'indice delle serie temporali. Il rumore ambientale in entrambi i vettori di sensori proviene da sorgenti di rumore L all'interno dell'ambiente, $\mathbf{E}(t) = [e_1(t), \dots, e_L(t)]^T$. Se la relazione tra ciascuna fonte di rumore e ciascuno sensore fosse scalare (nessun filtro o ritardo), la dipendenza potrebbe essere descritta in notazione matriciale come:

$$\mathbf{S}_e(t) = \mathbf{A}\mathbf{E}(t) \quad (4.2)$$

$$\mathbf{R}(t) = \mathbf{B}\mathbf{E}(t) + \mathbf{R}_s(t)$$

dove $A = [a_{kl}]$ e $B = [b_{jl}]$ sono matrici di mescolamento con a_{kl} e b_{jl} scalari e $\mathbf{R}_s(t)$ è il rumore del sensore di riferimento. Il rumore del sensore è supposto trascurabile. Se la relazione tra rumore e segnali del sensore è convolutiva (filtraggio e / o ritardo) si può usare la stessa notazione supponendo che ogni elemento a_{kl} o b_{jl} delle matrici di mescolamento A o B rappresenti una risposta all'impulso e sostituisca la moltiplicazione per convoluzione nell'Eq. 4.2. Per esempio:

$$r_{jl}(t) = (b_{jl} * e_l)(t) \quad (4.3)$$

dove $r_{jl}(t)$ è il contributo della sorgente di rumore l al sensore j . Il termine di attività cerebrale $\mathbf{S}_b(t)$ in Eq. 4.1 presumibilmente riflette anche più fonti all'interno del cervello, tuttavia non abbiamo bisogno di dettagliare questa dipendenza. Per riassumere il modello del segnale, i sensori del cervello e i sensori di riferimento raccolgono le stesse fonti di rumore ambientale, ma la relazione tra rumore e sensore può avere proprietà convolute che differiscono tra il cervello e i sensori di riferimento.

L'algoritmo TSPCA è semplice. Innanzitutto, i canali di riferimento $\mathbf{R}(t)$ sono traslati nel tempo da una serie di multipli del periodo di campionamento, sia positivi che negativi: $\mathbf{R}(t + n), n = -N/(2 + 1), \dots, N/2$. In secondo luogo, l'insieme del riferimento traslato nel tempo viene ortogonalizzato applicando PCA, per ottenere una base di JN segnali ortogonali nel dominio del tempo. In terzo luogo, ogni sensore del cervello viene proiettato su questa base e la proiezione rimossa. Il risultato è il segnale "pulito". Per il sensore k del cervello il processo complessivo può essere descritto come:

$$\hat{s}_k(t) = s_k(t) - \sum_{j=1}^J \sum_{n=1}^N a_{kj}(n) r_j \left(t - n - \frac{N}{2} \right) \quad (4.4)$$

dove $s_k(t)$ è il segnale pulito e $a_{kj}(n)$ emerge dalla combinazione di ortogonalizzazione e proiezione. Il coefficiente $a_{kj}(n)$ può essere inteso come l' n -esimo coefficiente di un filtro a risposta impulsiva finita di ordine N applicato al segnale di riferimento j prima della sottrazione dal segnale cerebrale k . Questo filtro è ottimale, nel senso dei minimi quadrati, per minimizzare il contributo delle componenti di rumore al segnale del sensore cerebrale. Si noti che i segnali del sensore cerebrale $s_k(t)$ non vengono filtrati, e quindi non c'è distorsione spettrale dell'attività cerebrale $s_b(t)$. L'elaborazione può essere riassunta nella notazione matriciale come:

$$\hat{\mathbf{S}} = \mathbf{I}\mathbf{S} - \mathbf{A}\tilde{\mathbf{R}} \quad (4.5)$$

dove I è la matrice di identità, $A = a_{kj}$ la matrice dei coefficienti trovato per ortogonalizzazione e proiezione, e $\tilde{\mathbf{R}}$ rappresenta l'insieme dei segnali del canale di riferimento spostato nel tempo.

In sintesi questo algoritmo di filtraggio viene ottenuto ritardando i segnali di riferimento, ortogonalizzandoli per ottenere una base, proiettando i sensori del cervello sulla base derivata dal rumore e rimuovendo le proiezioni per ottenere dati puliti. Nel nostro caso, poiché i canali *VEOG* e *HEOG* sono scelti come riferimento del rumore, TSPCA è in grado di rimuovere bene i manufatti oculari. [11]

Sensor Noise Suppression (SNS)

Qui viene presentato un algoritmo per la soppressione del disturbo specifico del sensore. Questo metodo si basa sul presupposto che ogni fonte di interesse viene

rilevata da più di un sensore. Per ridurre il rumore, ciascun segnale del sensore viene proiettato sul sottospazio occupato dai suoi vicini e sostituito dalla sua proiezione. In questo processo vengono eliminati il rumore a banda larga e le "anomalie", che non vengono condivise dai vicini, mentre vengono mantenute le funzionalità condivise.

I segnali dei sensori $\mathbf{S}(t) = [s_1(t), \dots, s_K(t)]^T$ riflettono una combinazione di attività cerebrale e rumore del sensore:

$$\mathbf{S}(t) = \mathbf{B}(t) + \mathbf{N}(t) \quad (4.6)$$

dove $\mathbf{B}(t)$ rappresenta l'attività cerebrale nello spazio del sensore, e $\mathbf{N}(t)$ rappresenta il rumore del sensore. Altre fonti di rumore esistono ma non sono considerate qui. L'attività cerebrale riflette più sorgenti nel cervello:

$$\mathbf{B}(t) = \mathbf{A}\mathbf{X}(t) \quad (4.7)$$

dove $\mathbf{X}(t) = [x_1(t), \dots, x_J(t)]^T$ sono fonti cerebrali e $A = [a_{kj}]$ è la matrice di mescolamento sorgente-sensore.

Facciamo le seguenti due ipotesi:

1. Il rumore del sensore non è correlato con l'attività cerebrale e non è correlato tra i sensori.
2. L'attività cerebrale di qualsiasi sensore può essere ricostruita dai suoi vicini: per ogni sensore k esistono coefficienti a'_k tali che

$$b_k(t) = \sum_{k \neq k'}^K a'_k b'_k(t) \quad (4.8)$$

In altre parole per ogni sensore k , le componenti del cervello $b_k(t)$ appartengono allo span $[b_{k' \neq k}]$. Per ogni k , il rango dell'insieme complementare $[b_{k', k' \neq k}]$ dei segnali dei sensori è lo stesso di quello dell'intero insieme. Intuitivamente, ci aspettiamo che la seconda ipotesi venga soddisfatta se ogni sensore raccoglie un piccolo numero di sorgenti cerebrali, ciascuna delle quali viene rilevata anche da altri sensori. L'assunto implica che i segnali del sensore K siano linearmente dipendenti (Eq 4.8), ma la dipendenza lineare non è sufficiente affinché la proprietà sia vera. La proprietà è interessante perché i dati che la rispettano sono invarianti all'operazione

che consiste nel sostituire ogni canale con la sua regressione sul sottospazio formato dagli altri canali.

L'algoritmo di denoising è semplice: consiste nel sostituire ciascun canale rumoroso con la sua regressione sul sottospazio formato dagli altri canali. In pratica, per ogni canale k , l'insieme di segnali $[s_{k' \neq k}]$ è ortogonalizzato applicando PCA per ottenere una base ortogonale del sottospazio suddivisa dagli altri canali. Il canale s_k viene proiettato su questa base e sostituito dalla sua proiezione. Questi passaggi sono ripetuti per tutti i canali. Per il canale k :

$$\tilde{s}_k(t) = \sum_{k'=1, k' \neq k}^K a_{kk'} s_{k'}(t) \quad (4.9)$$

dove $\tilde{s}_k(t)$ rappresenta il segnale filtrato del sensore e $a_{kk'}$ minimizza $\|\tilde{s}_k(t) - s_k(t)\|$. L'algoritmo può essere formulato anche in maniera matriciale come segue:

$$\tilde{S}(t) = \mathcal{A}S(t) \quad (4.10)$$

dove $\mathcal{A} = [a_{kk'}]$ è una matrice con tutti zero nella diagonale principale. È facile intuire perché l'algoritmo potrebbe ridurre il rumore specifico dei sensori. La formula che definisce la proiezione del canale k sull'ampiezza degli altri canali non include il canale stesso, e quindi non è sensibile al rumore del sensore all'interno di quel canale mentre è sensibile alle componenti di rumore all'interno degli altri canali, ma probabilmente la sensibilità è indebolita dal fatto che si aggiungono incoerentemente tramite Eq. 4.9. Questa intuizione è confermata da simulazioni, che mostrano inoltre che il denoising ha scarso effetto sull'attività cerebrale come a lungo verifica l'assunzione 4.7 (Eq. 4.8). La mancanza di effetti sull'attività cerebrale implica che i modelli avanzati non debbano essere aggiustati, distinguendo l'algoritmo da altre forme di filtraggio spaziale. Anche l'algoritmo SNS è stato implementato in Matlab. Evidenziamo alcuni dettagli di implementazione che sono degni di nota:

1. Di solito è sufficiente proiettare il canale k su un sottoinsieme di canali complementari piuttosto che sull'insieme completo. Ciò consente di risparmiare calcoli, ma ha scarso effetto sul risultato, a patto che ci siano più canali di

fonti cerebrali indipendenti che contribuiscono al canale k . I canali sono selezionati in base alla correlazione con il canale k . La vicinanza fisica potrebbe essere sostituita dalla correlazione che sembra funzionare altrettanto bene e non richiede la conoscenza della disposizione dei sensori.

2. L'algoritmo può essere ripetuto più volte: ogni passo riduce la norma che è limitata, garantendo la convergenza. In pratica, il primo passaggio rimuove la maggior parte del rumore e le ripetizioni successive offrono un miglioramento minore.
3. Valori di segnale eccezionalmente ampi dovrebbero essere scartati dal calcolo dei parametri di proiezione. Ciò impedisce che le somme dei quadrati che determinano questi parametri siano dominate da quei grandi valori.
4. I file di grandi dimensioni possono essere trattati in più passaggi per ridurre i requisiti di memoria.

Tenendo conto di questi dettagli di implementazione, i dati possono essere neutralizzati in un tempo migliore rispetto a un normale PC. Il conteggio dei sottoinsiemi di canale (in, 1) e la soglia di outlier (in, 3) introducono parametri arbitrari in una procedura altrimenti parametrica, ma i loro valori esatti, se ragionevoli, hanno scarso effetto sul risultato. [12]

Denoising Source Separation (DSS)

L'ultimo algoritmo di denoising viene eseguito per rimuovere le fonti fisiologiche indesiderate. Viene progettato un filtro spaziale, utilizzando un metodo di separazione della sorgente cieco noto come *Separazione della sorgente di Denoising* per suddividere l'attività registrata in componenti correlati allo stimolo e non correlati allo stimolo, sulla base di un criterio di riproducibilità evocato dallo stimolo. Il passo finale, denominato *DSS1*, consiste semplicemente nel mantenere solo il primo componente **DSS**. [13]

Il segnale del sensore $\mathbf{S}(t) = [s_1(t), \dots, s_k(t)]^T$ include alcune attività "target" interessanti e altre attività di "rumore" che vogliamo eliminare:

$$\mathbf{S}(t) = \mathbf{S}_B(t) + \mathbf{S}_N(t). \quad (4.11)$$

Il primo termine è il risultato della sovrapposizione di più sorgenti di interesse $\mathbf{B}(t) = [b_1(t), \dots, b_J(t)]^T$ nel cervello:

$$\mathbf{S}_B(t) = \mathbf{A}_B \mathbf{B}(t) \quad (4.12)$$

dove \mathbf{A}_B è una matrice di mescolamento. Il secondo termine è il risultato della sovrapposizione di varie sorgenti di rumore $\mathbf{N}(t) = [n_1(t), \dots, n_{j'}(t)]^T$ provenienti dall'ambiente, dai sensori e dal corpo del soggetto:

$$\mathbf{S}_N = \mathbf{A}_N \mathbf{N}(t) \quad (4.13)$$

dove \mathbf{A}_N è una seconda matrice di mescolamento. Il nostro scopo è attenuare $\mathbf{S}_N(t)$ e quindi migliorare la nostra osservazione di $\mathbf{S}_B(t)$. Supponiamo che le fonti di rumore ambientale e del sensore (linea elettrica e macchinari) siano già state soppresse, e quindi ci occupiamo principalmente di sorgenti di rumore fisiologico all'interno del corpo e del cervello del soggetto, come attività del cuore, battiti degli occhi e altre attività cerebrale "non interessanti" in corso. La distinzione tra interessante e non interessante ovviamente dipende dall'esperimento o dall'applicazione. Qui, ci concentriamo sulle risposte per le quali "interessante" è definito come attività riproducibile innescata dalla presentazione di uno stimolo. Un tipico esperimento di stimolo-risposta può includere M distinte condizioni di stimolo, ciascuna delle quali implica N_m ripetizione di quello stimolo. Durante l'esperimento, gli stimoli di tutte le condizioni sono in genere raggruppati e presentati in ordine casuale, mentre i campi magnetici sono registrati continuamente dai sensori K attorno al cervello del soggetto. I dati del sensore $S(t)$ vengono quindi suddivisi temporalmente in segmenti di peristimolo (epoche) e le prove raggruppate per condizione, formando un insieme di M matrici tridimensionali $S_m(t)$, ciascuna con dimensioni N_m (prove), K (numero dei canali) e T (durata di un'epoca nei campioni).

La nostra definizione operativa di "interessante" è implementata come una funzione di bias utilizzabile da DSS. Dato che siamo interessati all'attività evocata, definiremo il bias come funzione \mathcal{F} che ai dati $S_m(t)$ associa l'insieme $\tilde{S}_m(t)$ delle

medie sulle epoche, una media per ogni condizione:

$$\mathcal{F}[\mathbf{S}^m(t)] = \left\{ \frac{1}{N_m} \sum_{n=1}^{N_m} \mathbf{S}^{mn}(t) \right\} \quad (4.14)$$

dove $S^{mn}(t)$ indica l'epoca n della condizione m . L'attività stimolata è rinforzata dalla media mentre l'attività non correlata allo stimolo e il rumore non lo sono, quindi la norma di $\mathcal{F}[\{S_m(t)\}]$ è maggiore per l'attività evocata che per il rumore.

Vengono eseguiti i seguenti passaggi:

1. Ogni canale di dati è normalizzato (diviso per la sua norma).
2. I dati vengono inviati alla PCA e le componenti con potenza trascurabile sono scartate.
3. Le serie temporali corrispondenti alle restanti componenti principali di L sono normalizzate per ottenere un insieme di vettori ortonormali, "spazialmente sbiancati".
4. La funzione di bias definita in Eq. 4.14 viene applicata a questi dati e i dati di bias vengono inviati a una seconda PCA.
5. La matrice di rotazione prodotta dalla seconda PCA viene applicata ai dati sbiancati della fase 3.
6. L'insieme delle serie temporali L prodotte dai passaggi 1 – 5, ordinato in base al punteggio di bias decrescente, viene suddiviso in componenti "segnale", che vengono mantenuti, e componenti "rumore", che vengono scartati.
7. I componenti del segnale vengono proiettati nello spazio del sensore per ottenere dati EEG "puliti".

Come ultima osservazione, dobbiamo dire che la frequenza minima di campionamento non può essere troppo bassa per consentire il corretto funzionamento degli algoritmi di riduzione del rumore. Infatti, se gli artefatti ad alta frequenza vengono filtrati, sarà più difficile trovare e rifiutare il rumore che si trova nella banda LF che si riassume e diventa indistinguibile dalla risposta uditiva. Pertanto, quando

il sistema EEG non è alimentato a batteria e non vi è alcun vincolo di energia, i segnali vengono solitamente sovracampionati per questo motivo.

L'effetto combinato di questi passaggi può essere ricapitolato in formato matriciale:

$$\tilde{\mathbf{S}}(t) = \mathbf{PQR}_2\mathbf{N}_2\mathbf{R}_1\mathbf{N}_1\mathbf{S}(t) \quad (4.15)$$

dove \mathbf{N}_1 rappresenta la normalizzazione iniziale, \mathbf{R}_1 la prima rotazione PCA, \mathbf{N}_2 la seconda normalizzazione (sbiancamento), \mathbf{R}_2 la seconda matrice di rotazione PCA, Q il selettore basato sui criteri visti prima e P è la matrice di proiezione che torna nello spazio del sensore. La normalizzazione iniziale non è critica: il suo scopo è quello di dare uguale peso a ciascun sensore indipendentemente dal suo guadagno. Nel passaggio 2, i componenti con potenza al di sotto di una soglia arbitraria (non critica) vengono scartati per risparmiare il calcolo ed evitare problemi numerici. Il criterio di selezione della fase 6 (numero di componenti mantenuti) costituisce l'unico parametro importante del metodo. Può essere impostato direttamente, o determinato indirettamente, ad esempio, in base a una soglia applicata al punteggio di bias. L'algoritmo produce un insieme di componenti che sono:

- reciprocamente ortogonali;
- ordinati in base al rapporto di potenza totale evocato diminuito.

Eq. 4.15 definisce un filtro spaziale nello spazio del sensore: ogni canale del sensore viene modificato aggiungendo una somma ponderata di altri canali. Il filtro può essere applicato ai dati registrati e quindi i dati filtrati vengono mediati su prove per produrre risposte evocate che sono più riproducibili e meno rumorose di quelle ottenute con la media semplice.

4.3 Stima della statistica del segnale e dei disturbi

I segnali EEG presi in considerazione sono le registrazioni del cervello di potenziali evocati (EP) come risposta a stimoli uditivi ripetitivi (sillabe vocali) separate da

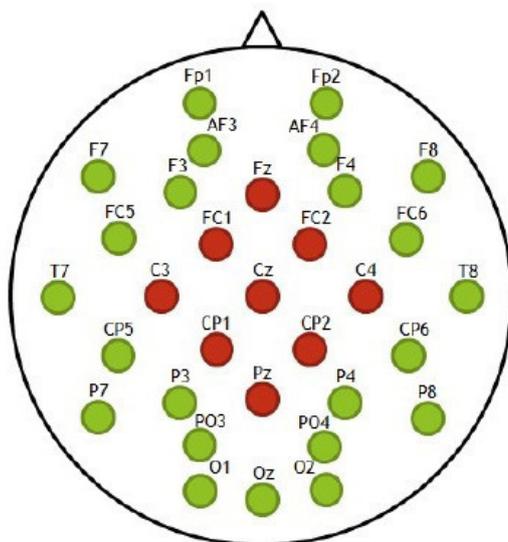


Figura 4.3: Posizionamento degli elettrodi EEG con canali uditivi evidenziati (rosso); HEOG e VEOG non sono mostrati

un intervallo di 1 secondo di tempo. Si è analizzato il set di dati contenente più di 10 minuti di registrazione di un soggetto udente in buona salute. Il segnale grezzo viene acquisito con una frequenza di campionamento di 2048 Hz da 32 sensori posizionati sul cuoio capelluto (vedere la Figura 4.3), più 2 canali differenziali aggiuntivi (denominati HEOG e VEOG) posizionati vicino agli occhi e utilizzati come riferimento di rumore (non contengono dati neurali ma tengono traccia degli artefatti oculari).

L'ampiezza del segnale EEG spontaneo è di solito molto più alta rispetto all'attività evocata e la media del segnale su epoche (intervallo di tempo di 1 secondo tra due stimoli) è il modo più semplice per far emergere la risposta uditiva dalle altre componenti indesiderati. Tuttavia, vi è un'ulteriore informazione data dalla distribuzione spaziale dei potenziali nel cuoio capelluto; considerando l'anatomia del cervello e gli studi effettuati riguardo le varie aree di risposta e di azione del cervello per le varie attività, si può prevedere che la risposta uditiva più forte sia situata nella parte superiore della testa, centrata attorno a Cz (si vedano i sensori rossi nella figura 4.3). Ciò è estremamente importante poiché consente di determinare a prima vista se in un determinato momento di tempo vi è un'attività evocativa in corso correlata a uno stimolo uditivo (FOI) o meno indipendentemen-

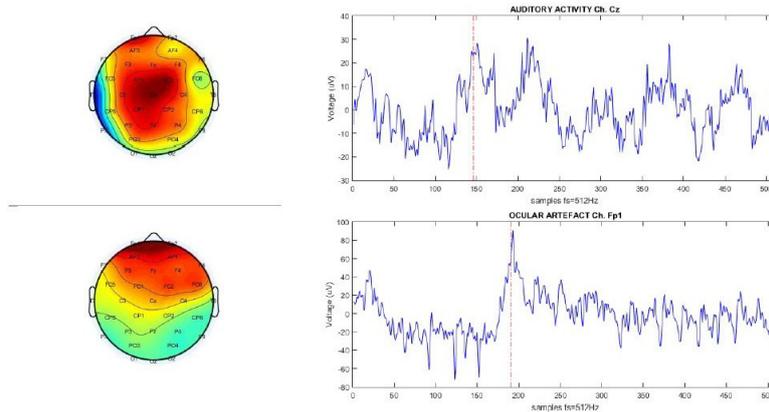


Figura 4.4: Esempio di un classico segnale uditivo sopra e di un tipico segnale dovuto principalmente da artefatti fisiologici (disturbo).

te dalla rappresentazione dei segnali tempo-frequenza. La Figura 4.4 sottolinea meglio questo aspetto, illustrando due forme d'onda esemplificative per il canale "uditivo" Cz (in alto) e il canale "oculare" Fp1 (in basso) e le relative mappe del cuoio capelluto (calcolate nell'istante temporale indicato dalle linee tratteggiate rosse). Mentre sembra davvero un compito arduo identificare una forma caratteristica per la risposta uditiva semplicemente nel dominio del tempo (i picchi di tensione in Cz sembrano non molto diversi da quelli in Fp1), guardando le mappe del cuoio capelluto (estraibili utilizzando il toolbox EEGLAB*) sulla sinistra

***EEGLAB** è un toolbox Matlab interattivo per l'elaborazione di EEG, MEG e altri dati elettrofisiologici correlati agli eventi e che includono analisi delle componenti indipendenti (ICA), analisi tempo / frequenza, rifiuto degli artefatti, statistiche relative agli eventi e varie modalità utili di visualizzazione della media e dati a prova singola. EEGLAB funziona con Linux, Unix, Windows e Mac OS X. EEGLAB fornisce un'interfaccia utente grafica interattiva (GUI) che consente agli utenti di elaborare in modo flessibile e interattivo il loro EEG ad alta densità e altri dati dinamici del cervello utilizzando analisi di componenti indipendenti (ICA) e / o analisi del tempo / frequenza (TFA), oltre alla media standard metodi. EEGLAB incorpora anche tutorial completi e finestre di aiuto, oltre a una funzione di cronologia dei comandi che facilita la transizione degli utenti dall'esplorazione dei dati basata sulla GUI alla creazione e all'esecuzione di script di analisi dei dati batch o personalizzati. EEGLAB offre una vasta gamma di metodi per visualizzare e modellare le dinamiche cerebrali legate agli eventi, sia a livello di singoli set di dati EEGLAB e / o attraverso una raccolta di set di dati riuniti in uno studio di EEGLAB. Per gli utenti esperti di Matlab, ed anche il metodo operativo in cui il tool viene usato in questo lavoro, EEGLAB offre un ambiente di programmazione strutturato per l'archiviazione, l'accesso,

è quasi immediato per scoprire la diversa natura dell'attività cerebrale in corso. Ovviamente, se siamo interessati ad ottimizzare il progetto di un sistema di monitoraggio EEG a bassa potenza, è estremamente importante concentrarsi sulle caratteristiche del segnale che vogliamo analizzare dal lato della stazione di base (cioè l'attività evocata), mentre le altre componenti non correlate dovrebbero essere idealmente tutte eliminate prima della trasmissione. Tuttavia, discernere cosa è da conservare da ciò che è da scartare è un compito sottile e deve essere affrontato guardando da diversi punti di vista. Dal punto di vista della frequenza temporale, si potrebbe desiderare di raccogliere tutte le informazioni contenute nella banda di frequenza occupata dall'attività uditiva (cioè intorno a 1-30 Hz) mentre si respingono i contributi di frequenza più alta. Dal punto di vista della distribuzione spaziale, si potrebbe voler mantenere il tipico schema uditivo, dove l'ampiezza massima si trova attorno al canale Cz, e scartare tutte le informazioni associate alla distribuzione del manufatto oculare. In altre parole, miriamo a sfruttare il CS per comprimere selettivamente solo le informazioni utili; in questo modo possiamo teoricamente raggiungere un livello di compressione più alto (cioè risparmiare più energia nel nodo wireless) e trasmettere contemporaneamente una versione del segnale con esclusione del rumore. Al lato della stazione di base l'obiettivo sarà quello di decomprimere / ricostruire la versione del segnale acquisita, eventualmente in tempo reale.

4.3.1 Rappresentazione sparsa

Il processo di campionamento digitale di un segnale naturale porta alla sua rappresentazione come la somma delle funzioni Delta nello spazio o nel tempo. Questa rappresentazione, sebbene conveniente ai fini della visualizzazione, è per lo più inefficiente per le attività di analisi. Le tecniche di elaborazione del segnale comunemente richiedono rappresentazioni più significative che catturano le caratteristiche utili del segnale per il riconoscimento, la rappresentazione dovrebbe evidenziare le caratteristiche salienti; per la ricostruzione, la rappresentazione dovrebbe separare efficacemente segnale e rumore; e per la compressione, la rappresentazione dovrebbe

la misurazione, la manipolazione e la visualizzazione dei dati EEG relativi agli eventi. [17]

be catturare gran parte del segnale con solo pochi coefficienti. È interessante notare che in molti casi questi obiettivi apparentemente diversi si allineano, condividendo un desiderio fondamentale di semplificazione. Per l'acquisizione di segnali biologici, e ai fini di questo lavoro, tutti e tre i compiti sopra menzionati sono particolarmente importanti e questo è il motivo per cui una corretta rappresentazione sparsa del segnale EEG è stata identificata come il primo, passaggio essenziale di questa ricerca.

La rappresentazione di un segnale implica la scelta di un dizionario \mathbf{D} , che è l'insieme di segnali elementari, chiamati atomi, usati per scomporre il segnale. Quando il dizionario costituisce una base, un segnale x , costituito da N campioni digitali, è rappresentato in modo univoco come combinazione lineare con i coefficienti s degli atomi del dizionario.

$$x_N = D_{N \times L} s_L \quad (4.16)$$

Nel caso più semplice si ha $L = N$ e il dizionario ortogonale, quindi i coefficienti della rappresentazione possono essere calcolati come prodotti interni del segnale e degli atomi mentre, nel caso non ortogonale, i coefficienti sono i prodotti interni del segnale e il dizionario inverso.

La limitata espressività di questa rappresentazione ha portato allo sviluppo di dizionari più nuovi e completi ($L > N$), con più atomi rispetto alle dimensioni del segnale, che ha permesso di rappresentare una gamma più ampia di segnale. Sfortunatamente, tali dizionari introducono un'ambiguità intrigante nella definizione della rappresentazione del segnale. Questo perché esiste un insieme infinitamente grande di vettori di coefficienti che soddisfano l'equazione di sintesi 4.16, per questo motivo la rappresentazione più informativa del segnale viene cercata rispetto ad alcune funzioni di costo $C(s)$.

$$s = \arg \min_s C(s) \text{ subject to } x = Ds \quad (4.17)$$

Le scelte pratiche di $C(s)$ promuovono la sparsità della rappresentazione, il che significa che vogliamo che i coefficienti ordinati decadano rapidamente. Risolvere 4.17 viene quindi comunemente indicato come "sparse coding" e rappresenta un metodo di analisi del segnale estremamente efficace. Infatti, guardando gli atomi

del dizionario relativi ai coefficienti di ampiezza più elevati siamo in grado di identificare le componenti di segnale più importanti e può anche essere visto come una sorta di filtraggio.

Ricordiamo che il segnale da rilevare deve essere scarso anche per soddisfare i requisiti CS per una corretta ricostruzione. Tuttavia, passando dalla teoria alla pratica, la sfida di scegliere il dizionario appropriato per un determinato compito deve essere affrontata in primo luogo. In generale, la scelta di un dizionario può essere effettuata utilizzando diversi modi:

- costruire un dizionario sparsificante basato su un **modello matematico** dei dati;
- **imparare un dizionario** per ottenere il meglio su un **set di allenamento**, che dovrebbe essere sufficientemente rappresentativo della maggior parte delle caratteristiche del segnale.

I dizionari del primo tipo sono caratterizzati da una formulazione analitica e un'implementazione implicita veloce, mentre i dizionari del secondo tipo offrono maggiore flessibilità e la capacità di adattarsi a specifici dati del segnale. Più recentemente, c'è un crescente interesse per i dizionari che possono mediare tra i due tipi e offrire i vantaggi di entrambi i mondi. [29] [30]

Da vari studi si prova che uno dei dizionari più adatti per la rappresentazione dei dati EEG è la base di Daubechies6[†]. Nel corso dell'algoritmo l'utilizzo del dizionario è reso possibile grazie al tool di Matlab **WaveLab850**[‡]. [27] [28]

[†]Le wavelet Daubechies sono una famiglia di wavelet ortogonali che definiscono una discreta trasformazione wavelet e caratterizzata da un numero massimo di momenti di fuga per un determinato supporto. Con ogni tipo di wavelet di questa classe, esiste una funzione di ridimensionamento (chiamata wavelet padre) che genera un'analisi di multirisoluzione ortogonale.

[‡]WaveLab is a library of MATLAB routines for wavelet analysis, wavelet-packet analysis, cosine-packet analysis and matching pursuit. The library is available free of charge over the Internet. Versions are provided for Macintosh, UNIX and Windows machines.

4.4 Risultati

Consideriamo quindi un caschetto di elettrodi come già visto in fig. 4.3. Si sono presi in considerazione tre canali posizionati in zone diverse del cervello di riferimento su cui confrontare le varie analisi: il canale Fp1, posto all'altezza dell'occhio sinistro (a cui a volte ci riferiremo come canale 1) e dove si rileva il massimo segnale di "blinking" ovvero il movimento delle palpebre durante la registrazione; il canale Cz posto al centro (a cui ci riferiremo come canale 32, considerando sempre la numerazione del sistema di riferimento citato nei paragrafi precedenti) su cui si prevede di rilevare la maggior parte del segnale di risposta all'impulso acustico; ed il canale PO4 che come dice la sigla stessa è in posizione "parietale-occipitale".

Il database utilizzato proviene da acquisizioni su pazienti sani sottoposti a stimoli acustici ripetuti ad intervalli di tempo regolari di 1 secondo dal dipartimento di Electrical and Computer Engineering dell'Università del Maryland College Park.

Da questo punto ci riferiremo al segnale che esce dal filtro anti-aliasing e dal sottocampionamento come *segnale grezzo* mentre parleremo di segnale filtrato in riferimento al segnale che attraversa il filtro passa alto che sarà poi il filtro che tenteremo di sostituire per mezzo del nostro algoritmo. Per concludere avremo il segnale in uscita alla catena che completa il filtraggio a valle del trasferimento con TSPCA, SNS e DSS. Successivamente valuteremo le performance della Rakeness con Denoising andando a valutare prima ogni singolo canale e quindi considerando segnale e disturbo proprio di quel sensore per diversi fattori di compressione (CR). Successivamente viene svolta la stessa analisi andando sia a considerare il sistema complessivo sia suddividendo la cuffia di sensori in due cluster. Ciò comporta un'importante diminuzione delle matrici di sensing che vengono utilizzate nell'algoritmo (da n matrici dove n sta per numero di sensori a 1 quando si considera il sistema complessivo mentre si hanno c matrici, numero di cluster nel segnale quando il disturbo non si clusterizza come nel nostro caso, quando si considera la suddivisione in macrogruppi). Anche qui le performance vengono valutate considerando diversi valori di CR.

Prima di introdurre i tre casi di analisi evidenziamo la differenza sull'uso del tool *spgl1*. Infatti nel primo caso si sfrutta il metodo classico e generico BPDN

(Basis Pursuit Denoise) che risolve il seguente problema:

$$\min \|x\|_1 \text{ subject to } \|Ax - b\|_2 \leq \sigma, \quad (4.18)$$

Per vedere come estendere questo concetto al secondo e terzo caso in esame vediamo che i risolutori SPG-BP e SPG-BPDN possono gestire problemi anche con incognite complesse:

$$\min \|z\|_1 \text{ subject to } A * z = b \quad (4.19)$$

Quando A è una matrice reale possiamo riscrivere il problema come

$$\min \|X\|_{1,2} \text{ subject to } A * X = B \quad (4.20)$$

con $X = [\text{real}(z), \text{imag}(z)]$, $B = [\text{real}(b), \text{imag}(b)]$, e con la norma (1,2)-mista definita come $\|x\|_{1,2} := \sum_i \|X(i, :)\|_2$. Quello che ne risulta è che questa formula è applicabile non solo a BP complesso, ma anche al problema MMV (Multiple Measurment Vector). In questi problemi ci sono misurazioni multiple date da $B = AX + E$ e, soprattutto, X si assume essere a righe sparse (molti zero nelle righe). Ciò dà origine al problema BPDN di MMV:

$$\min \|X\|_{1,2} \text{ subject to } \|A * X - B\|_{2,2} \leq \sigma, \quad (4.21)$$

con $\|X\|_{1,2} := \sqrt{\sum_i \|X(i, :)\|_2^2}$

4.4.1 Analisi per ogni singolo sensore

Come già accennato all'inizio del capitolo, quello che si cerca di fare è andare ad eliminare il filtro passa alto, lasciando all'algoritmo della R-CSd il compito di compensare a questa mancanza. La procedura inizia con il filtraggio del segnale per ogni canale, di modo che si riesca a separare nettamente ciò che è informazione da ciò che non lo è (ribadiamo che a questo punto del lavoro con filtraggio ci riferiamo al solo filtro passa alto). Successivamente vengono calcolate le matrici di correlazione dei due segnali trovati sopra (segnale vero e disturbo), questa informazione serve all'algoritmo della Rakeness con Denoising per il calcolo della

matrice di correlazione della matrice di sensing A (algoritmo presentato nel capitolo 3). Considerazione che facciamo è che i segnali analizzati, se presi dividendo il segnale in epoche divise in base al momento in cui agisce il trigger acustico, questi dipenderanno fortemente dai loro indici senza contare il fatto che il segnale non ha una banda di distribuzione di energia omogenea. La cosa diventa particolarmente rilevante se per esempio si volesse successivamente lavorare con delle matrici antipodali nella matrice di sensing A , formate cioè da $+1$ e -1 . Per questo si realizza la matrice di correlazione partendo dal segnale finestrato come prima dove però queste finestre vengono prese come intervalli della stessa lunghezza ma raccolti utilizzando l'estrazione di un indice random come primo campione evitando qualsiasi sorta di polarizzazione. Ci riferiremo a questo tipo di segnale come segnale stazionario e alle sue matrici come matrici Toeplitziane.

Una volta settati i parametri più appropriati per il caso in esame (step di α , σ etc..) si esegue la funzione che calcola la matrice di correlazione della matrice di sensing. Il passaggio alla sua matrice di sensing è un procedimento di facile risoluzione quando le righe di A sono istanze di un processo gaussiano.

Soffermiamoci un secondo su questo procedimento ed evidenziamo la differenza nelle tre diverse funzioni della Rakeness. La sequenza di sensing per l'approccio standard CS è composto da istanze di vettori n -dimensionali non correlati di un processo Gaussiano, per le righe di A del R-CS, la sequenza di sensing è generata come istanze di vettori Gaussiani ma questa volta con matrici di correlazione prese da 2.9, mentre per R-CSd, la matrice di correlazione che caratterizza il processo Gaussiano che genera le righe di A è data dalla risoluzione del problema di ottimizzazione 3.1.

Una volta trovata la sequenza di sensing A , elaborazione che viene eseguita offline, l'informazione viene data in pasto, insieme al dizionario con il quale è stato reso sparso il segnale, al solutore spgl al lato del ricevitore. Una volta estratto il segnale ricostruito la catena si conclude con l'applicazione dei tre filtri precedentemente analizzati.

Nelle analisi sotto riportate abbiamo fissato il parametro r presente nell'ultimo vincolo dell'equazione di minimizzazione 3.6 a 0.9 e τ di 3.5 pari a 0.25. I seguenti

grafici mostrano per i canali 5, 25, 32 le ricostruzioni risultanti per tutti gli approcci del CS confrontate con il segnale filtrato (inteso passa alto) ed il segnale grezzo. I grafici riportano il segnale mediato sulle epoche (che già di suo tende leggermente a filtrare il segnale) a cui togliamo la media per vederli sovrapposti, rispettivamente con fattore di compressione 2 ($m=256$), 4 ($m=128$) e 8 ($m=64$) in Fig.4.5 4.6 4.7

Si noti come variano le ricostruzioni al variare del fattore di compressione. In particolare, si vede che con $m=64$, ovvero con un importante fattore di compressione, per l'esattezza pari ad 8, le ricostruzioni seguono comunque abbastanza bene il segnale filtrato che prendiamo come Gold Standard a meno della ricostruzione con la Rakeness che in questo caso ha un po più di difficoltà. Già con un fattore di compressione un po più basso ma comunque importante ($CR=4$) si vedono degli andamenti già molto coerenti che diventano ottimi quando si passa ad un fattore di compressione 2. Per tutti e 3 i valori di m e per tutti i 3 canali la ricostruzione con la Rakeness è ottimale già per compressioni basse anche se il valore m pari a 256 sembra essere il minimo ottimale. Riportiamo infatti ora le tabelle che mostrano la media dei segnali presentati nei grafici appena descritti dove si vede che l'algoritmo della Rakeness riesce perfettamente a togliere la continua dal segnale esattamente come il segnale filtrato a differenza degli altri due metodi CS e R-CS.

	CR = 2	CR = 4	CR = 8
Grezzo	11.892	11.892	11.892
Filtrato	$-3.16e - 06$	$-3.16e - 06$	$-3.16e - 06$
CS	11.886	11.89	11.891
R - CS	11.87	11.887	11.891
R - CSd	$7.598e - 04$	$7.965e - 04$	$7.129e - 04$

Tabella 4.1: *Tabella che mette a confronto la media del segnale sulle epoche al variare del fattore di compressione CR e della tecnica di compressione per il canale 5.*

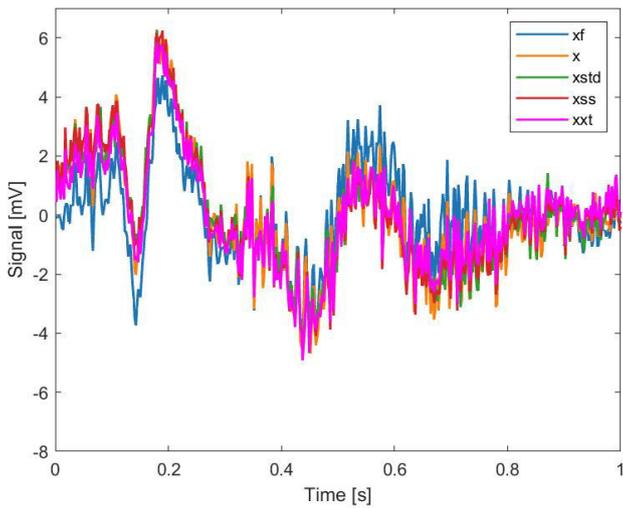
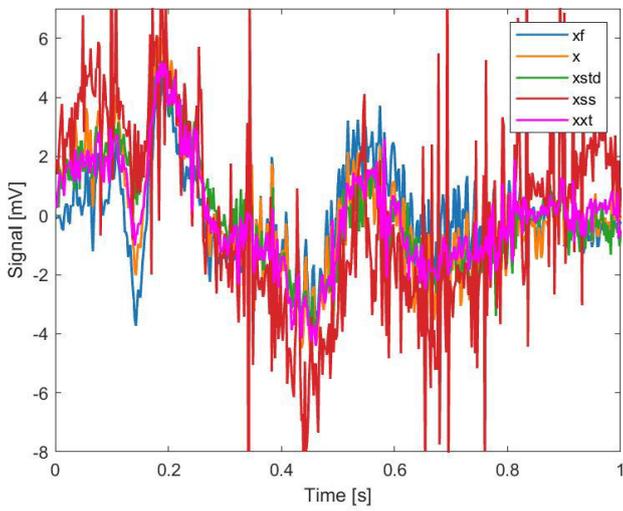
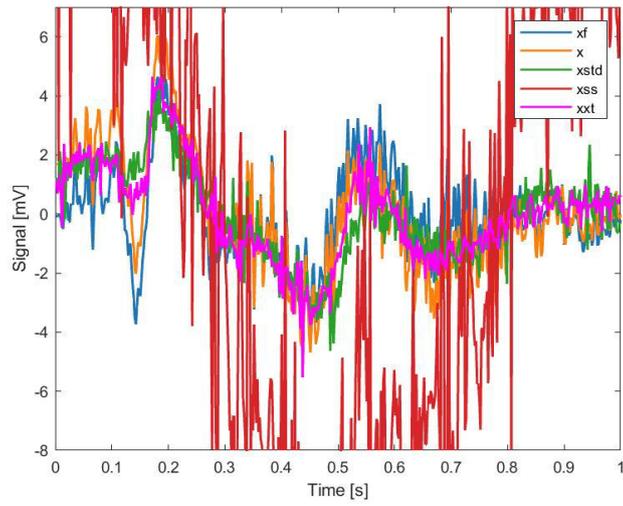


Figura 4.5: Rispettivamente si hanno le ricostruzioni del canale 5 usando 3 diversi valori di m , ovvero righe della matrice di sensing (64, 128, 256)

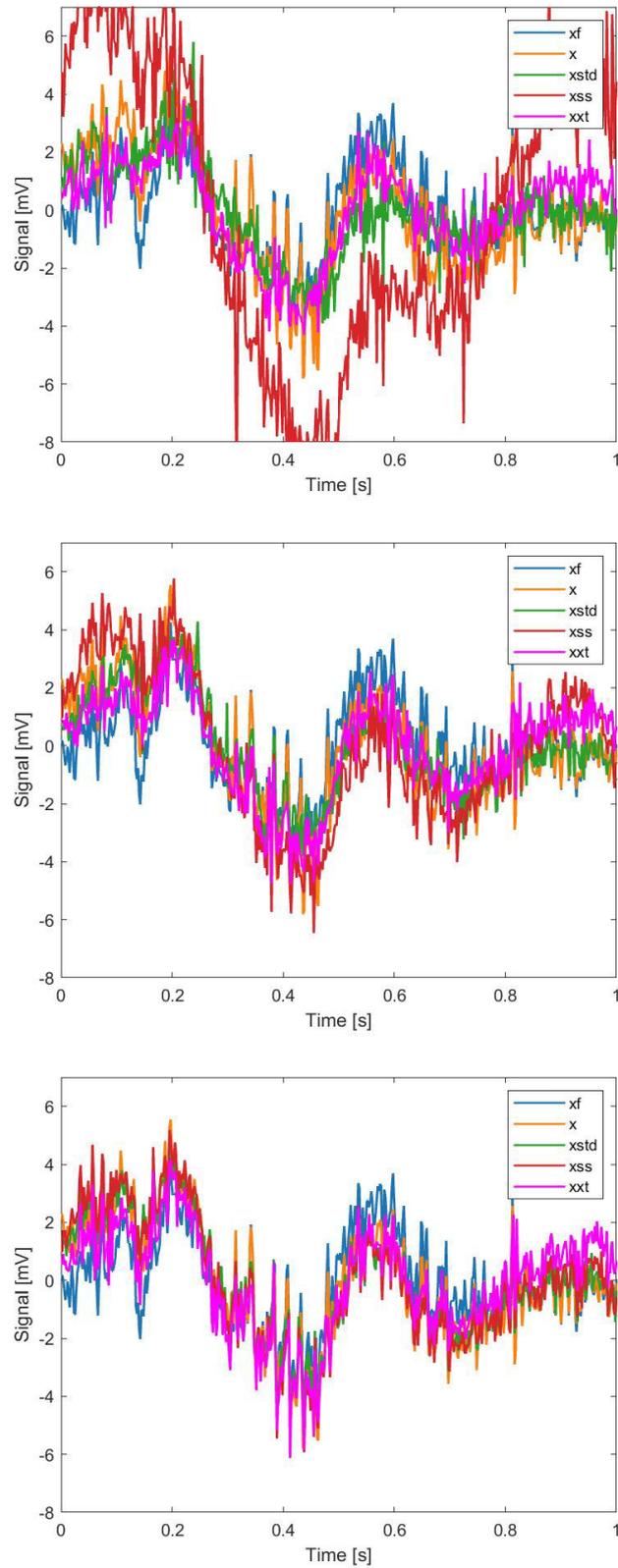


Figura 4.6: Rispettivamente si hanno le ricostruzioni del canale 25 usando 3 diversi valori di m , ovvero righe della matrice di sensing (64, 128, 256)

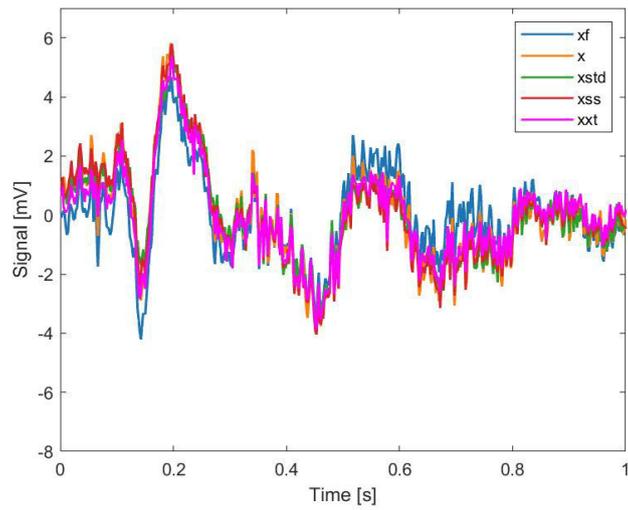
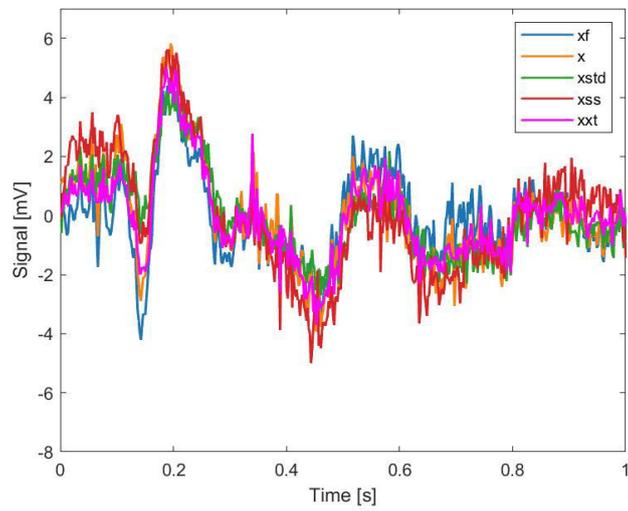
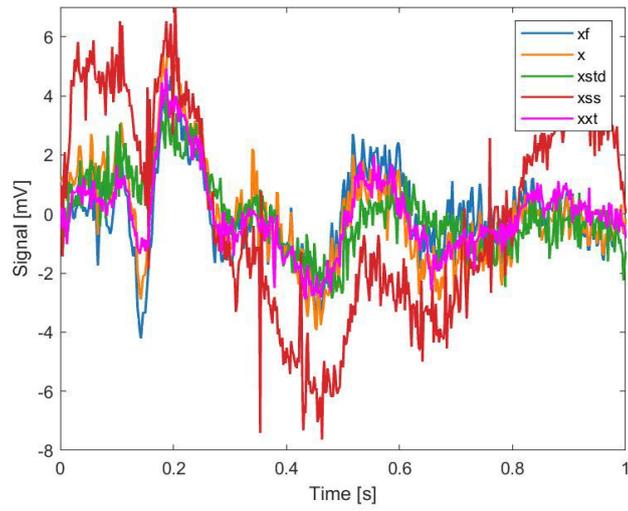


Figura 4.7: Rispettivamente si hanno le ricostruzioni del canale 32 usando 3 diversi valori di m , ovvero righe della matrice di sensing (64, 128, 256)

	CR = 2	CR = 4	CR = 8
Grezzo	4.959	4.959	4.959
Filtrato	$-2.348e - 06$	$-2.348e - 06$	$-2.348e - 06$
CS	4.953	4.957	4.958
R - CS	4.947	4.955	4.958
R - CSd	$4.803e - 04$	$4.318e - 04$	$4.653e - 04$

Tabella 4.2: *Tabella che mette a confronto la media del segnale sulle epoche al variare del fattore di compressione CR e della tecnica di compressione per il canale 25.*

	CR = 2	CR = 4	CR = 8
Grezzo	2.232	2.232	2.232
Filtrato	$-3.597e - 06$	$-3.597e - 06$	$-3.597e - 06$
CS	2.226	2.23	2.231
R - CS	2.224	2.23	2.232
R - CSd	$4.987e - 04$	$4.759e - 04$	$4.231e - 04$

Tabella 4.3: *Tabella che mette a confronto la media del segnale sulle epoche al variare del fattore di compressione CR e della tecnica di compressione per il canale 32.*

Anche dai valori di ARSNR dove si vede l'andamento dei tre segnali ricostruiti (CS R-CS e R-CSd) confrontato con quello filtrato per pochi punti di m , si ha la conferma di come l'algoritmo R-CSd riesca a guadagnare in dB l'informazione. Vedimo solo i due m significativi che abbiamo detto essere pari a 128 e 256.

ARSNR[dB]	CR = 2	CR = 4	CR = 8
CS	59.486	63.465	68.272
R – CS	59.463	65.156	69.783
R – CSd	$5.6e - 4$	$5.8e - 4$	$5.3e - 4$

Tabella 4.4: Tabella che mette a confronto la media del rapporto segnale-disturbo (ARSNR) sulle epoche rispetto al segnale filtrato nei tre diversi algoritmi per il canale 5 e $m=64,128$ e 256.

ARSNR[dB]	CR = 2	CR = 4	CR = 8
CS	52.044	55.108	58.841
R – CS	51.234	55.813	59.769
R – CSd	$8.4e - 4$	$7.7e - 4$	$8.4e - 4$

Tabella 4.5: Tabella che mette a confronto la media del rapporto segnale-disturbo (ARSNR) sulle epoche rispetto al segnale filtrato nei tre diversi algoritmi per il canale 25 e $m=64,128$ e 256.

ARSNR[dB]	CR = 2	CR = 4	CR = 8
CS	44.736	48.635	53.527
R – CS	45.783	51.101	55.556
R – CSd	0.0021	0.0021	0.0019

Tabella 4.6: Tabella che mette a confronto la media del rapporto segnale-disturbo (ARSNR) sulle epoche rispetto al segnale filtrato nei tre diversi algoritmi per il canale 32 e $m=64,128$ e 256.

Vediamo nello specifico il valore intrinseco ISDR dato dalla formula:

$$ISDR = 20 \log_{10} \frac{\|x_f\|_2}{\|x_f - x\|_2} \quad (4.22)$$

che vale nei tre canali 5, 25, 32 rispettivamente -54.93 dB, -46.93 dB, -42.8 dB. Notiamo infatti come questi valori siano all'incirca i valori di ARSNR che otteniamo dai metodi CS e R-CS.

4.4.2 Analisi che considera il sistema complessivo

L'analisi che viene qui descritta viene presentata come metodo alternativo a quello sopra riportato. Infatti quello che si cerca di fare è ridurre la complessità del sistema prendendo in considerazione l'insieme dei sensori invece che lavorare sul singolo canale. A questo proposito si utilizza il tool `spl mmv` sopra descritto che consente quindi di mantenere anche un'informazione spaziale tra i diversi canali. La differenza con l'approccio poco sopra descritto è che in questo caso le matrici di correlazione del segnale, così come le matrici di sensing che escono dall'algoritmo della R-CSd, non sono più tante quanto i canali ma saranno solo una. Vediamo come prima i grafici con le ricostruzioni per i canali selezionati e per i tre valori di m in Fig.4.8 4.9 4.10

Riportiamo in seguito anche le tabelle con le medie dei segnali delle figure sopra, ad evidenziare come la Rakenext anche in questo caso riesca ad eliminare la continua esattamente come il segnale filtrato. Si noti come i risultati siano leggermente discostati da quelli sopra ma sicuramente ancora pienamente validi ed accettabili.

	CR = 2	CR = 4	CR = 8
Grezzo	11.793	11.793	11.793
Filtrato	$-2.982e - 06$	$-2.982e - 06$	$-2.982e - 06$
CS	11.788	11.791	11.792
R – CS	11.783	11.791	11.792
R – CSd	$7.331e - 04$	$7.747e - 04$	$7.651e - 04$

Tabella 4.7: *Tabella che mette a confronto la media del segnale sulle epoche al variare del fattore di compressione CR e della tecnica di compressione per il canale 5.*

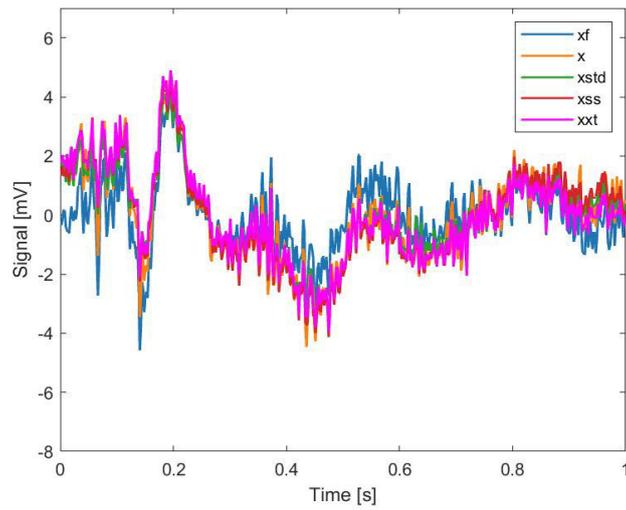
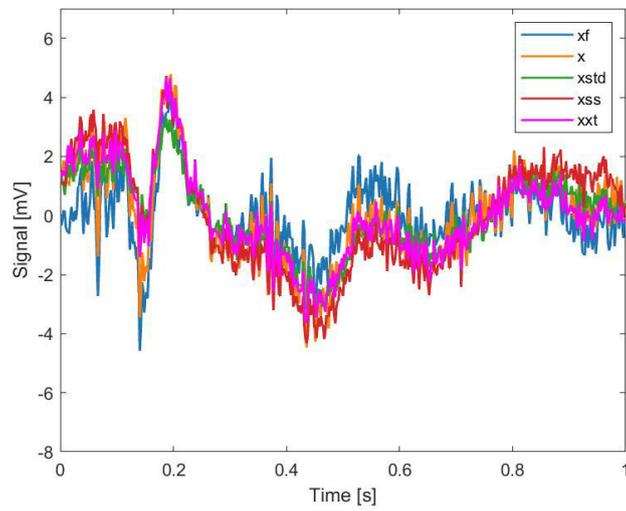
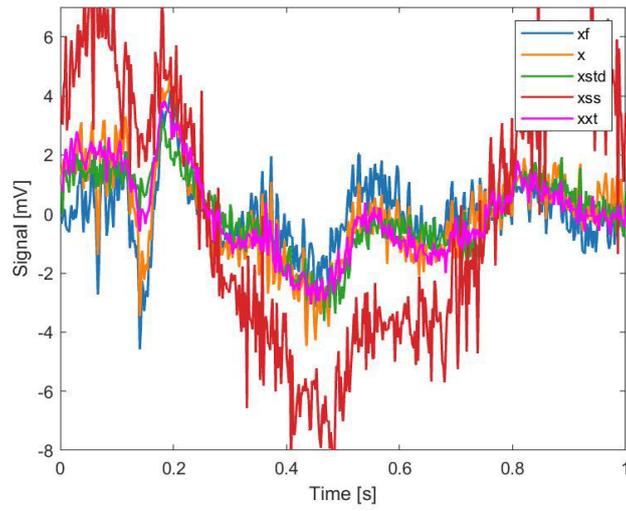


Figura 4.8: Rispettivamente si hanno le ricostruzioni del canale 5 usando 3 diversi valori di m , ovvero righe della matrice di sensing (64, 128, 256)

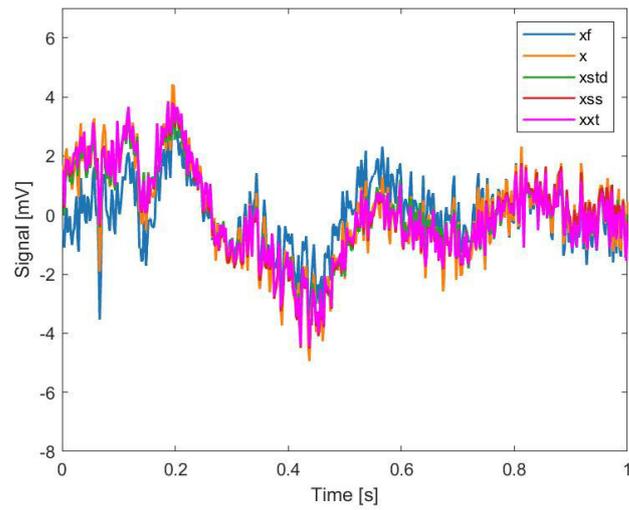
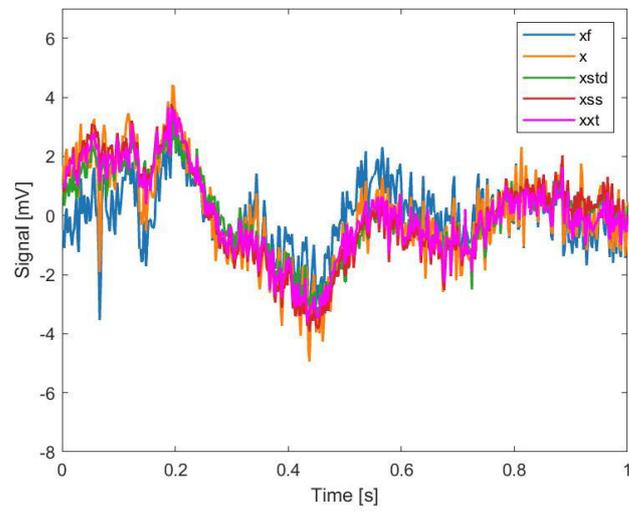
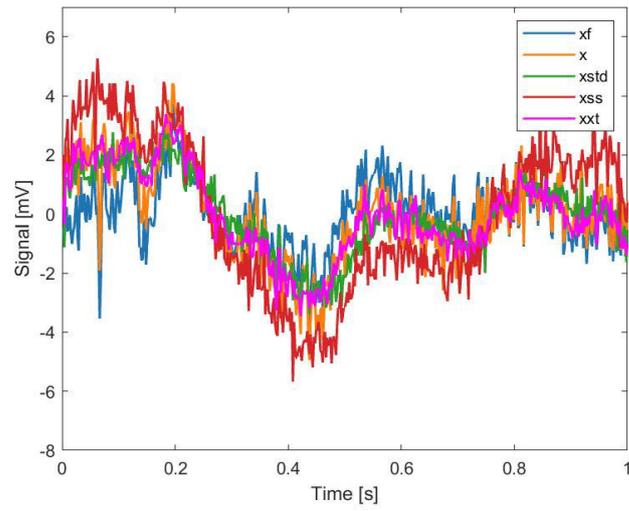


Figura 4.9: Rispettivamente si hanno le ricostruzioni del canale 25 usando 3 diversi valori di m , ovvero righe della matrice di sensing (64, 128, 256)

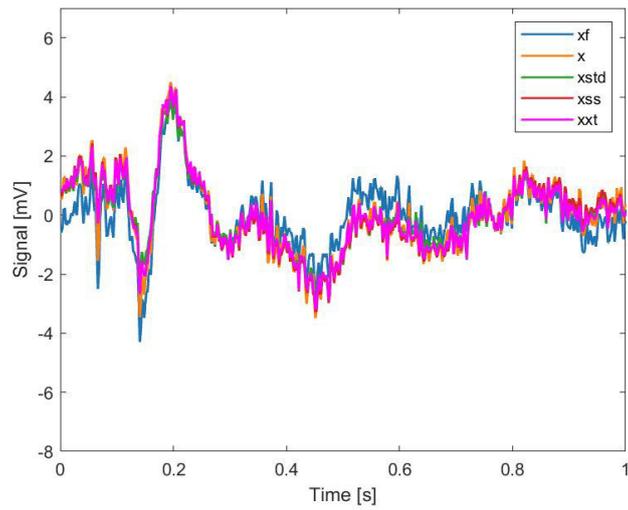
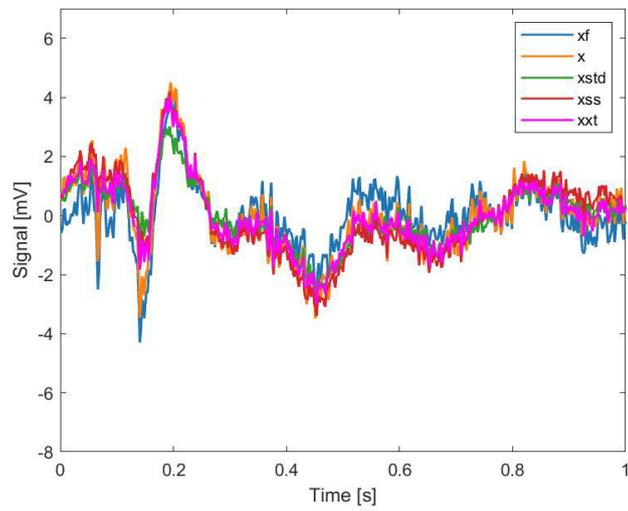
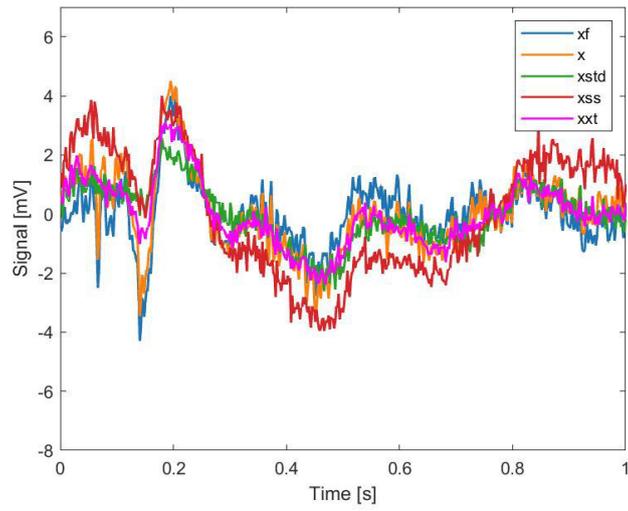


Figura 4.10: Rispettivamente si hanno le ricostruzioni del canale 32 usando 3 diversi valori di m , ovvero righe della matrice di sensing (64, 128, 256)

	CR = 2	CR = 4	CR = 8
Grezzo	4.612	4.612	4.612
Filtrato	$-2.171e - 06$	$-2.171e - 06$	$-2.171e - 06$
CS	4.61	4.611	4.612
R - CS	4.6	4.611	4.612
R - CSd	$8.401e - 04$	$8.54e - 04$	$8.569e - 04$

Tabella 4.8: *Tabella che mette a confronto la media del segnale sulle epoche al variare del fattore di compressione CR e della tecnica di compressione per il canale 25.*

	CR = 2	CR = 4	CR = 8
Grezzo	2.217	2.217	2.217
Filtrato	$-2.915e - 06$	$-2.915e - 06$	$-2.915e - 06$
CS	2.214	2.216	2.217
R - CS	2.213	2.216	2.217
R - CSd	$4.881e - 04$	$5.309e - 04$	$5.339e - 04$

Tabella 4.9: *Tabella che mette a confronto la media del segnale sulle epoche al variare del fattore di compressione CR e della tecnica di compressione per il canale 32.*

Anche nei casi che abbiamo qui riportato vale lo stesso discorso del paragrafo precedente, infatti si vede che la R-CSd già per valori di compressione molto bassi si assesta molto bene ai valori del segnale filtrato.

4.4.3 Analisi per cluster

Nel caso finale si cerca di trovare un approccio intermedio ai due descritti sopra cercando di prendere i vantaggi di entrambi. Quello che si fa è dividere il segnale vero in due cluster, calcolati per mezzo di matrici di similarità che sfruttano la

differenza tra matrici di correlazione del segnale e l'approccio kmeans [§] per la clusterizzazione. Il disturbo in questo caso specifico si è preferito elaborarlo come sopra dato che il solo segnale passa alto non rappresenta per questa classe di segnale una distinzione sostanziale tra sensori che permetta di distinguerli cosa che invece si potrebbe pensare di fare se per esempio si riuscisse a incorporare nell'algoritmo di compressione filtri che separano rumore fisiologico. Per il segnale vero, invece, la clusterizzazione è stata fatta a partire dalle matrici di correlazione del segnale filtrato a valle di tutti i filtri, quindi incorporando in questa sezione di codice, oltre al passa alto anche i tre filtri TSPCA, SNS e DSS. Quello che si ottiene è mostrato in figura 4.11

Evidenziamo che anche in questo caso si è sfruttato spgl come mmv per ciascuno dei due cluster. Vediamo le tre ricostruzioni per i tre valori di compressione, prendendo in considerazione i soliti tre canali che vediamo in Fig.4.12 4.13 4.14, ovvero 'FC1' e 'Cz' che appartengono al cluster centrale e 'FC6' che appartiene al cluster più esterno.

Valutiamo anche in questo caso le medie, dove ancora una volta si vede come l'approccio R-CSd riesca a togliere la media esattamente come il segnale filtrato, dimostrazione appunto di una effettiva eliminazione di rumore a bassa frequenza senza l'utilizzo di un filtro passa alto.

	CR = 2	CR = 4	CR = 8
Grezzo	11.793	11.793	11.793
Filtrato	$-2.982e - 06$	$-2.982e - 06$	$-2.982e - 06$
CS	11.787	11.791	11.792
R - CS	11.779	11.79	11.792
R - CSd	$5.691e - 04$	$6.197e - 04$	$6.147e - 04$

[§]L'algoritmo K-means è un algoritmo di clustering partizionale che permette di suddividere un insieme di oggetti in K gruppi sulla base dei loro attributi. È una variante dell'algoritmo di aspettativa-massimizzazione (EM) il cui obiettivo è determinare i K gruppi di dati generati da distribuzioni gaussiane. Si assume che gli attributi degli oggetti possano essere rappresentati come vettori, e che quindi formino uno spazio vettoriale.

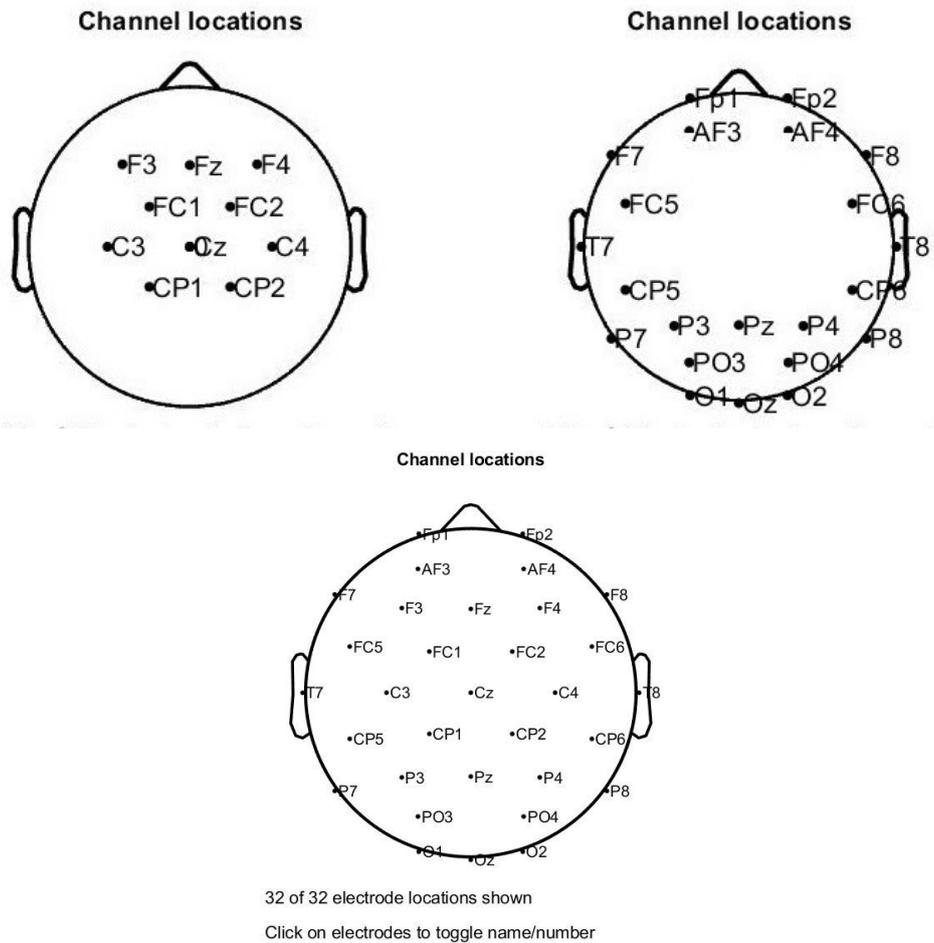


Figura 4.11: Nella figura sopra si ha la rappresentazione della clusterizzazione dei sensori in due zone. Nella seconda immagine sottostante si vede come invece per il disturbo venga elaborata una matrice per l'intero sistema.

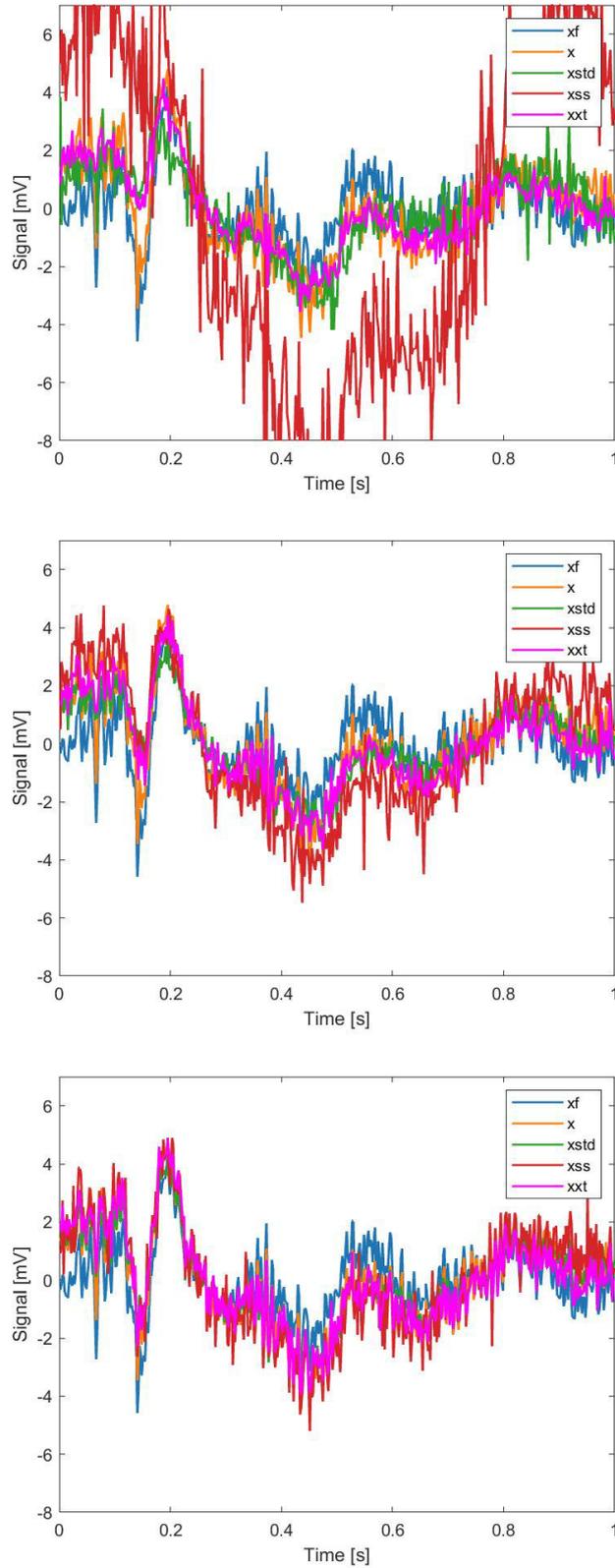


Figura 4.12: Rispettivamente si hanno le ricostruzioni del canale 5 usando 3 diversi valori di m , ovvero righe della matrice di sensing (64, 128, 256). Il canale 5 appartiene al secondo cluster e corrisponde a 'FC1'.

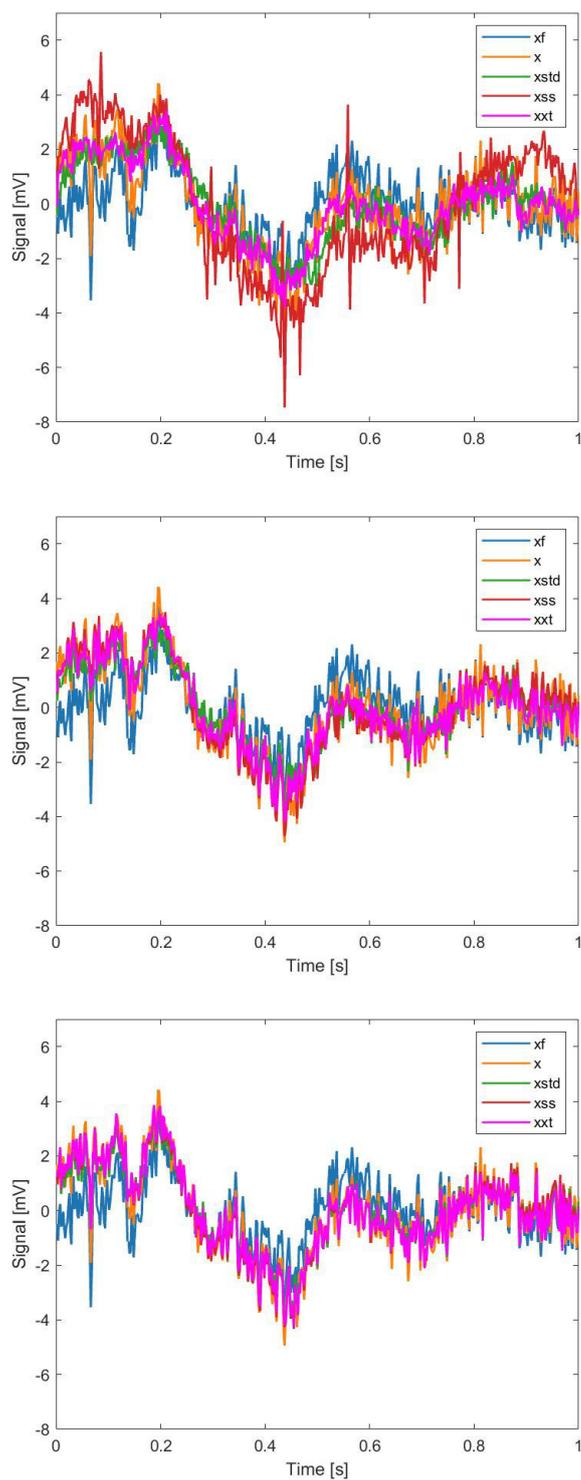


Figura 4.13: Rispettivamente si hanno le ricostruzioni del canale 25 usando 3 diversi valori di m , ovvero righe della matrice di sensing (64, 128, 256). Il canale 25 appartiene al primo cluster e corrisponde a 'FC6'.

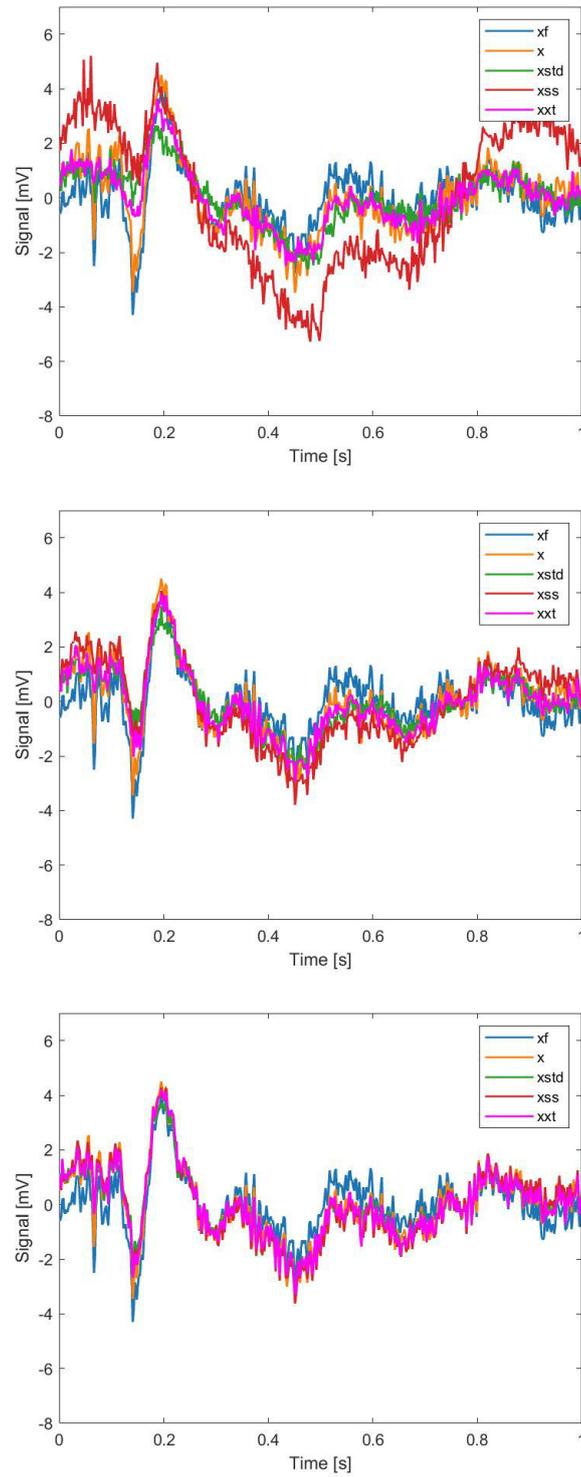


Figura 4.14: Rispettivamente si hanno le ricostruzioni del canale 32 usando 3 diversi valori di m , ovvero righe della matrice di sensing (64, 128, 256)- Il canale 32 appartiene al secondo cluster e corrisponde a 'Cz'.

Tabella 4.10: *Tabella che mette a confronto la media del segnale sulle epoche al variare del fattore di compressione CR e della tecnica di compressione per il canale 5.*

	CR = 2	CR = 4	CR = 8
Grezzo	4.612	4.612	4.612
Filtrato	$-2.171e - 06$	$-2.171e - 06$	$-2.171e - 06$
CS	4.61	4.611	4.612
R - CS	4.6	4.611	4.612
R - CSd	$8.15e - 04$	$8.685e - 04$	$8.757e - 04$

Tabella 4.11: *Tabella che mette a confronto la media del segnale sulle epoche al variare del fattore di compressione CR e della tecnica di compressione per il canale 25.*

	CR = 2	CR = 4	CR = 8
Grezzo	2.217	2.217	2.217
Filtrato	$-2.915e - 06$	$-2.915e - 06$	$-2.915e - 06$
CS	2.213	2.216	2.217
R - CS	2.212	2.216	2.217
R - CSd	$3.46e - 04$	$4.121e - 04$	$3.992e - 04$

Tabella 4.12: *Tabella che mette a confronto la media del segnale sulle epoche al variare del fattore di compressione CR e della tecnica di compressione per il canale 32.*

In questo approccio si vedono come valgono le stesse valutazioni precedenti facendo pensare che comunque tra i tre metodi, quello che tiene conto del sistema complessivo rimane il più vantaggioso, considerando che si ottengono gli stessi risultati utilizzando meno mantrici in memoria.

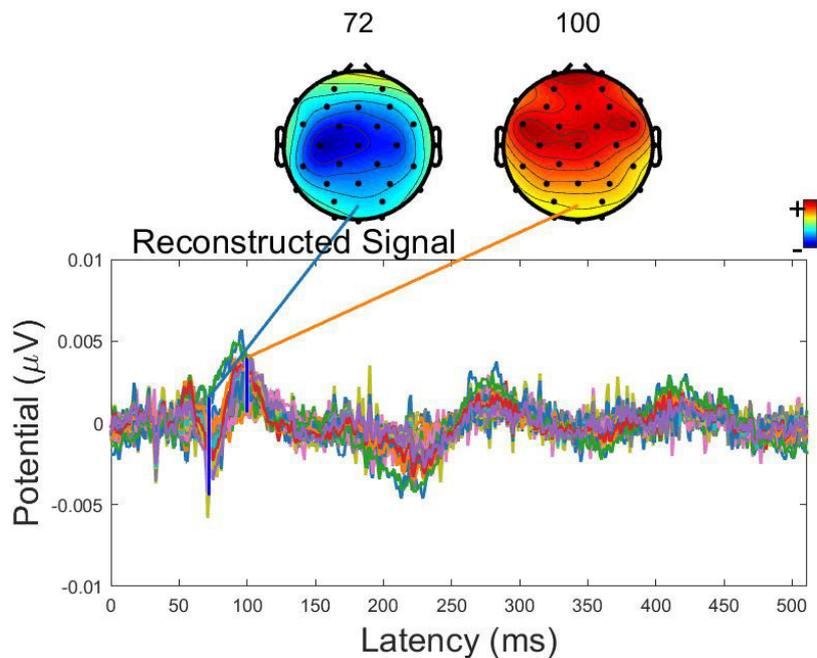
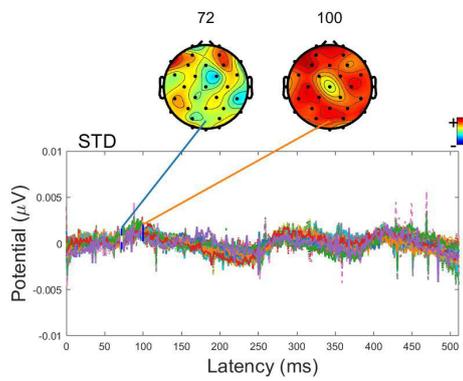


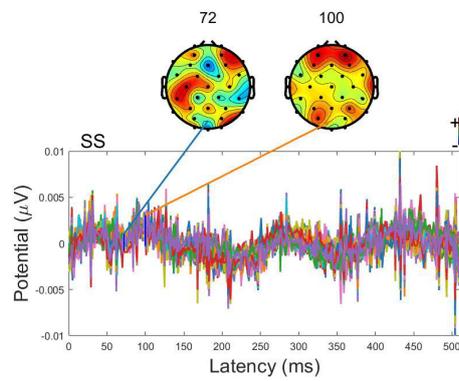
Figura 4.15: Segnale vero dei canali registrati a posteriori della ricostruzione e del filtro passa alto.

4.4.4 Confronti finali

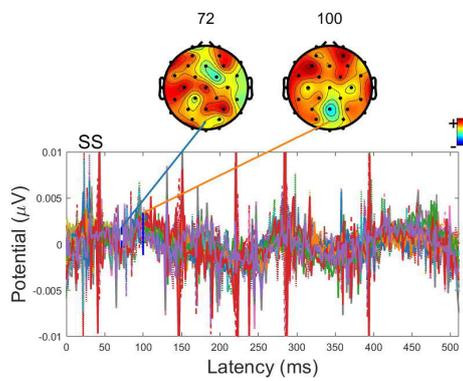
A questo punto concludiamo la catena raccogliendo i dati sopra elaborati e prima di proseguire con ulteriori step di elaborazione si mostra in seguito il contenuto informativo che si ottiene da tutti i canali appena il segnale è stato ricostruito (con l'aggiunta del passa alto nei casi CS e R-CS). In particolare abbiamo in fig 4.15 il contenuto informativo vero dei canali dopo il passa alto mentre nelle figure 4.16, 4.17 e 4.18 si vedono i 3 diversi algoritmi (CS, R-CS e R-CSd) nei 2 diversi metodi (2 cluster e sistema complessivo) al variare di m . Già da queste immagini possiamo intuire come la ricostruzione sia effettivamente buona in tutti e tre i metodi del CS ad eccezione di quando m assume il valore 64, dove a prima vista nessuno dei tre algoritmi sembrerebbe dare buoni risultati. Per l'esattezza quello che viene graficato sono tracce di più epoche di dati che mappano l'attività del cuoio capelluto in un dato momento, mostrando in rosso i picchi di massimo del segnale e in blu quelli di minimo. Considerando invece valori di compressione maggiori si nota che l'algoritmo R-CSd tende ad accentuare il segnale a livello frontale dove sappiamo esserci molto segnale rumoroso fisiologico causato dal movimento degli occhi; an-



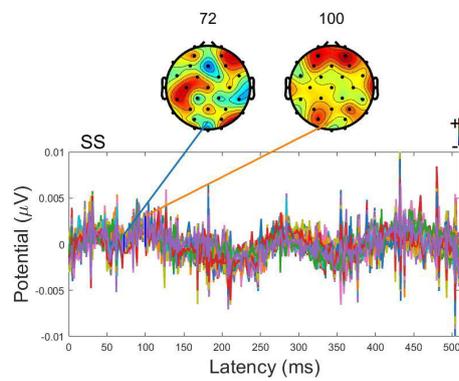
(a) *CS metodo 2 matrici*



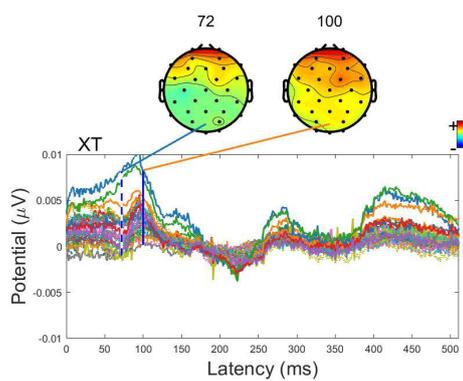
(b) *CS metodo 1 matrice*



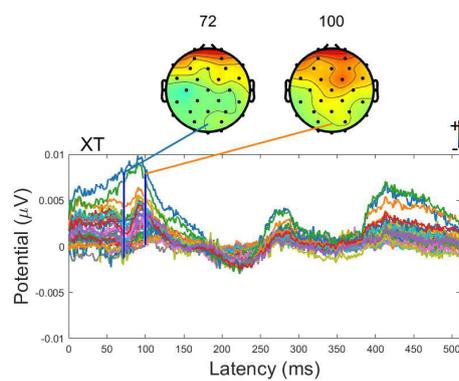
(c) *R-CS metodo 2 matrici*



(d) *R-CS metodo 1 matrice*

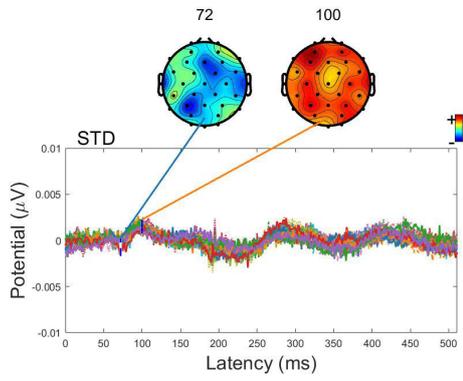


(e) *R-CSd metodo 2 matrici*

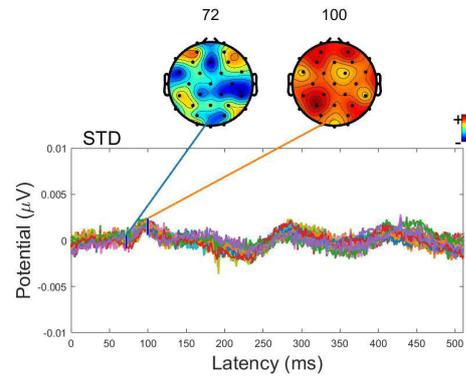


(f) *R-CSd metodo 1 matrice*

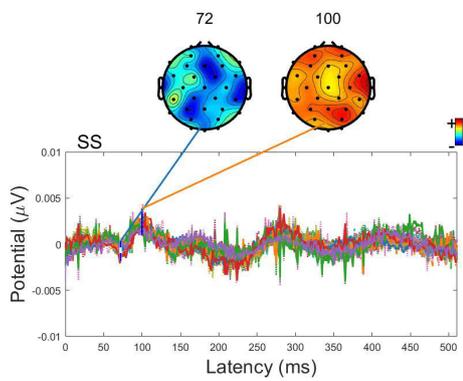
Figura 4.16: Rappresentazione informativa di tutti i canali con i 3 diversi metodi nei due algoritmi con $m=64$



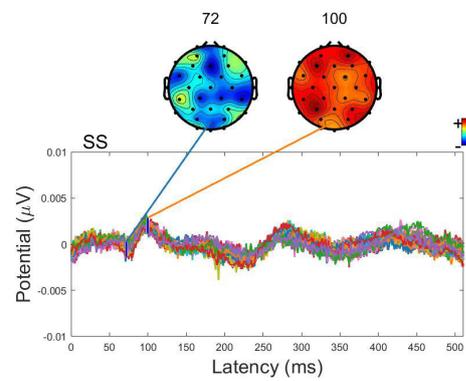
(a) *CS metodo 2 matrici*



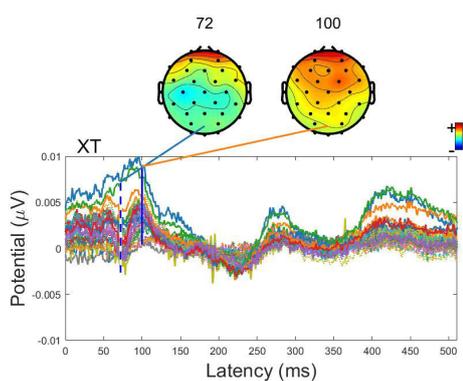
(b) *CS metodo 1 matrice*



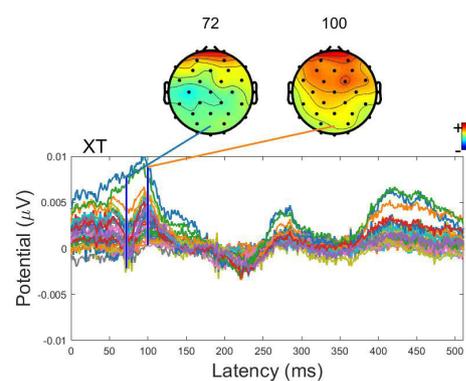
(c) *R-CS metodo 2 matrici*



(d) *R-CS metodo 1 matrice*

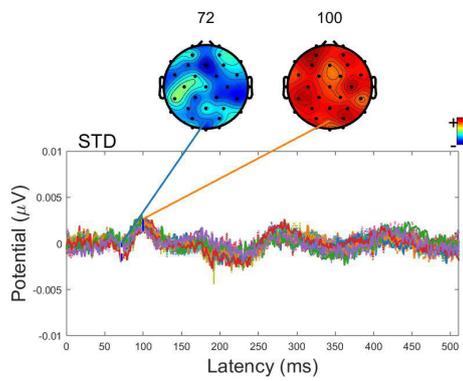


(e) *R-CSd metodo 2 matrici*

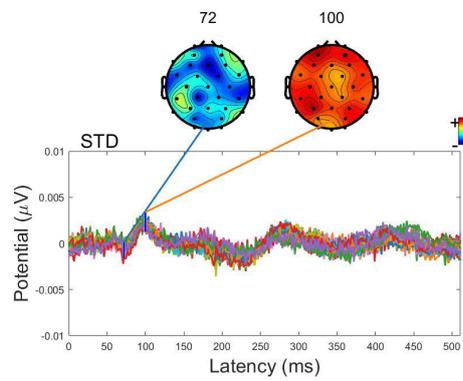


(f) *R-CSd metodo 1 matrice*

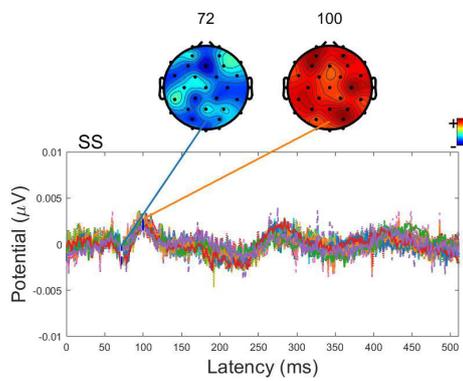
Figura 4.17: Rappresentazione informativa di tutti i canali con i 3 diversi metodi nei due algoritmi con $m=128$



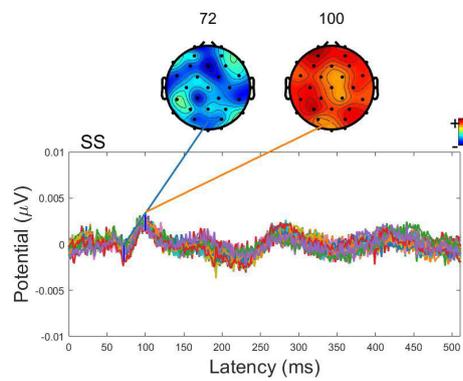
(a) *CS metodo 2 matrici*



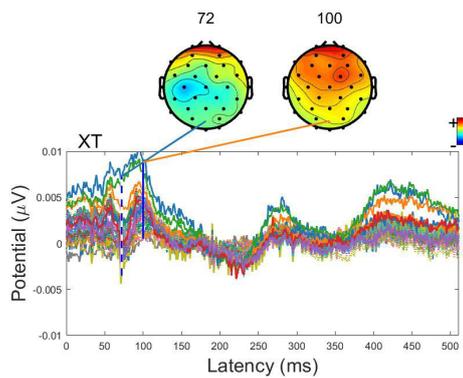
(b) *CS metodo 1 matrice*



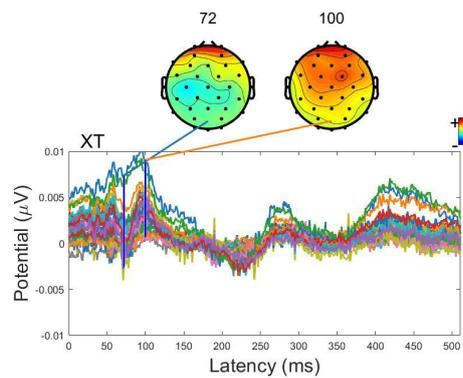
(c) *R-CS metodo 2 matrici*



(d) *R-CS metodo 1 matrice*



(e) *R-CSd metodo 2 matrici*



(f) *R-CSd metodo 1 matrice*

Figura 4.18: Rappresentazione informativa di tutti i canali con i 3 diversi metodi nei due algoritmi con $m=256$

che dai segnali se ne notano due che tendono a prevalere sugli altri, probabilmente hanno un elevato grado di sparsità.

Sfruttiamo ora i risultati dell'analisi che considera il sistema complessivo e i due cluster ed eseguiamo i 3 filtraggi sui segnali ricostruiti presentati nel paragrafo 4.2, avendo cura di aggiungere anche un passa alto per gli algoritmi CS e R-CS. Prendiamo in considerazione i canali 5 25 e 32 e mostriamo la ricostruzione dei 3 canali confrontando il risultato finale vero, ovvero applicando tutti i filtri al segnale originale, senza usufruire della ricostruzione, e l'approccio CS, R-CS e R-CSd per $m=64$, $m=128$ e $m=256$ in Fig. 4.19 4.20 4.21

Considerando il caso con unica matrice, si vede come la Rakeness con Denoising segue esattamente gli altri profili. Considerando che R-CSd non utilizza il filtro passa alto a monte della catena possiamo dire che riesce a mantenere comunque le stesse informazioni. La conferma la possiamo vedere dai grafici in fig 4.26 4.28 Quello che si vede è che con m pari a 256 la qualità del sistema è ottima in quanto siamo in grado di ricavare lo stesso contenuto informativo del segnale filtrato da tutti e tre gli algoritmi. Lo stesso vale per il caso di fattori di compressione più bassi dove tutti e tre i metodi del CS sembrano dare buoni risultati. Il R-CSd tende ad avere ampiezze meno evidenti in alcuni canali ma comunque mantenendo buona l'informazione.

Valutiamo ora le stesse performance andando a guardare i segnali ricostruiti presi dall'algoritmo che sfrutta la suddivisione in due cluster. Di seguito come prima la ricostruzione dei canali 5, 25, 32 confrontando il risultato finale classico, ovvero senza la ricostruzione, e l'approccio CS, R-CS e R-CSd per $m=64, m=128$ e $m=256$ in Fig. 4.22 4.23 4.24. In questo caso le performance sembrano peggiorare un po' rispetto al caso precedente andando a valutare il metodo classico del CS che sembra dar problemi principalmente per $m = 64$; al contrario dell'algoritmo R-CSd che sembra dare performance migliori rispetto al caso precedentemente trattato.

Visti i tre casi di analisi (per ogni canale, sistema complessivo e due cluster) facciamo un'ultimo confronto finale tra i segnali che si ottengono all'uscita della catena, quindi a posteriori di tutti i filtri citati al variare di m . Guardando le figure 4.26 che rappresentano l'informazione complessiva dei canali fissato il fattore

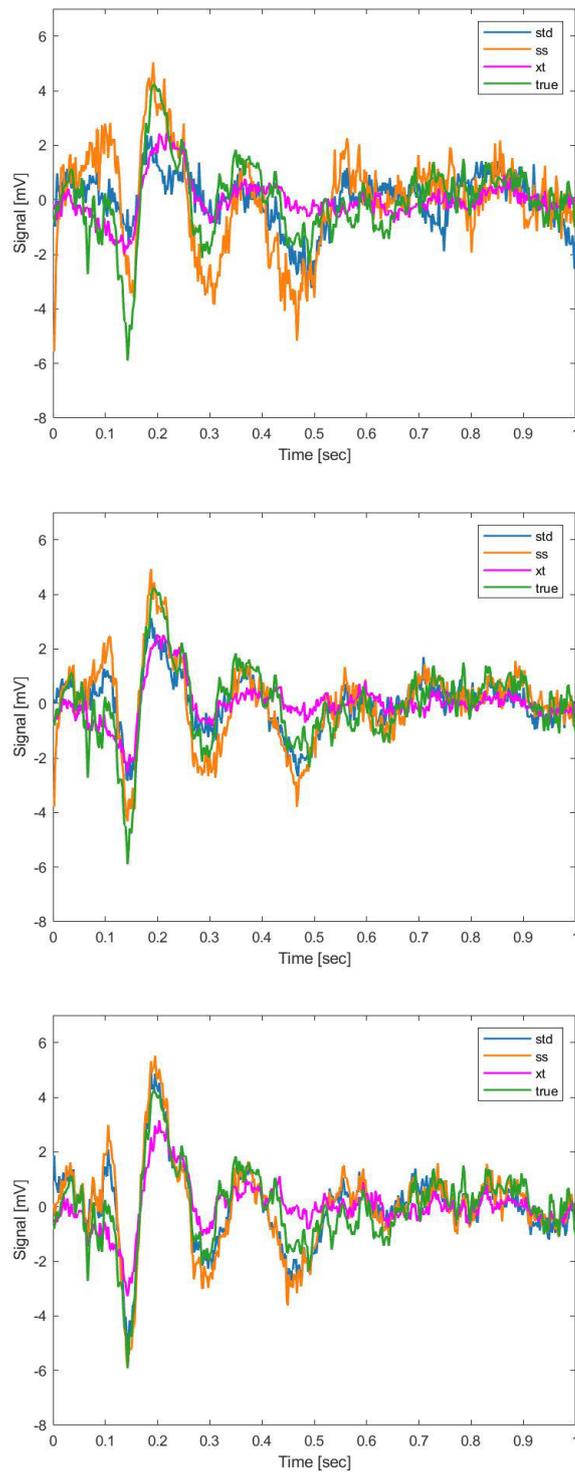


Figura 4.19: A confronto la ricostruzione finale del canale 5 con i tre metodi analizzati CS, R-CS e R-CSd rispetto al filtrato ($m=64, 128, 256$). Il canale 5 corrisponde a 'FC1'.

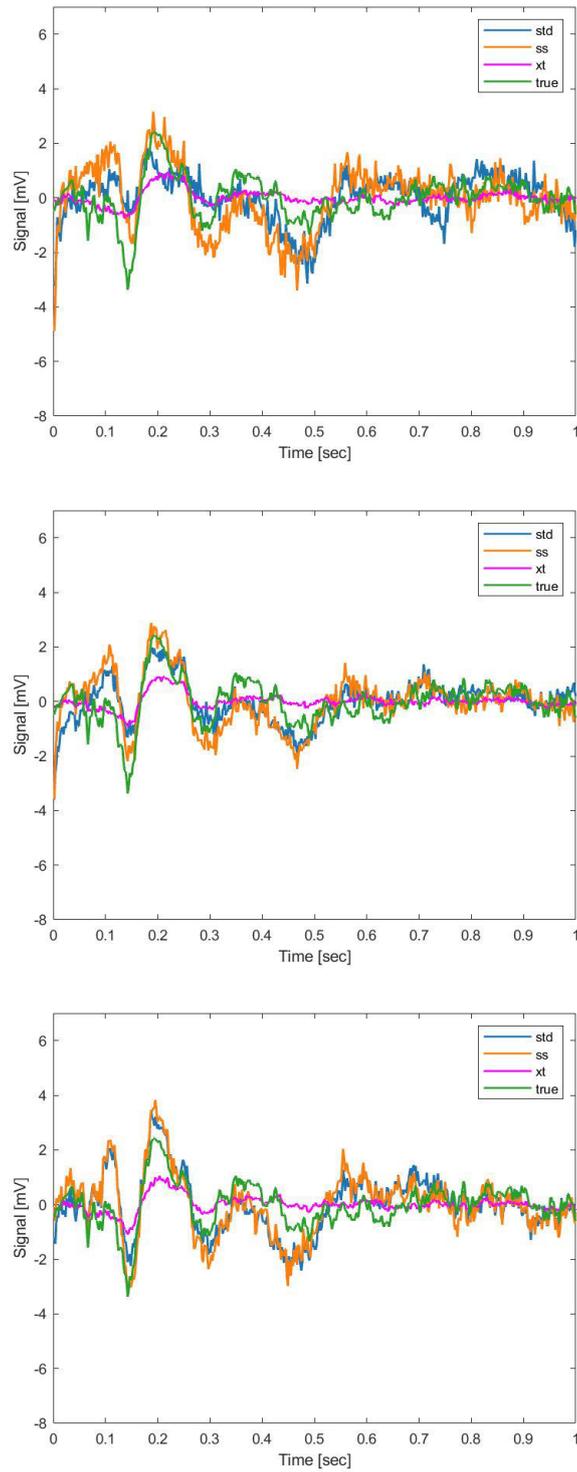


Figura 4.20: A confronto la ricostruzione finale del canale 25 con i tre metodi analizzati CS, R-CS e R-CSd rispetto al filtrato ($m=64, 128, 256$). Il canale 25 corrisponde a 'FC6'.

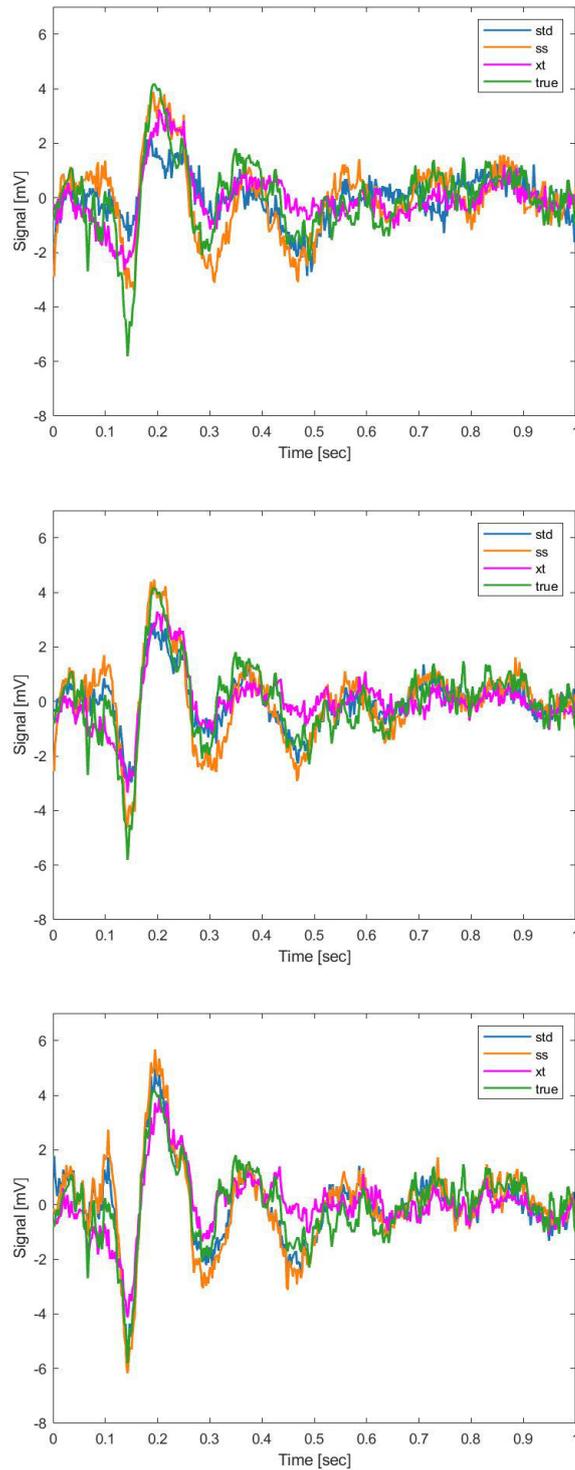


Figura 4.21: A confronto la ricostruzione finale del canale 32 con i tre metodi analizzati CS, R-CS e R-CSd rispetto al filtrato ($m=64, 128, 256$). Il canale 32 corrisponde a 'Cz'.

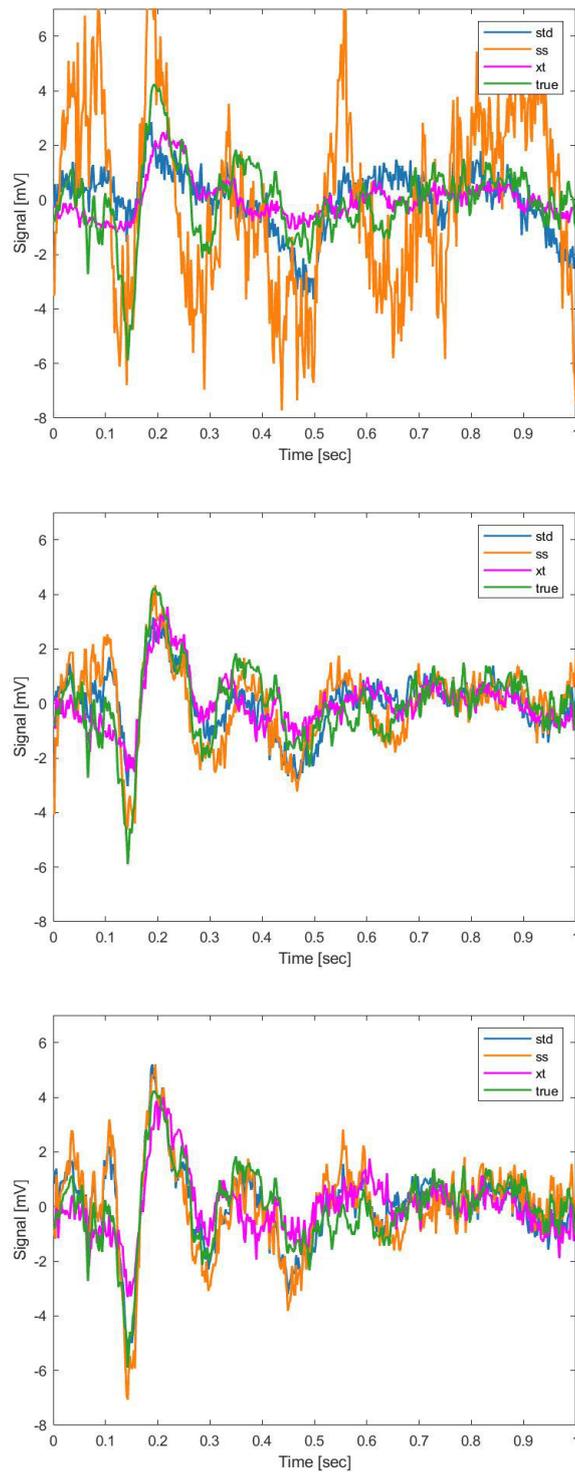


Figura 4.22: A confronto la ricostruzione finale del canale 5 con i tre metodi analizzati CS, R-CS e R-CSd rispetto al filtrato ($m=64, 128, 256$) utilizzando il metodo delle due matrici. Il canale 5 corrisponde a 'FC1'.

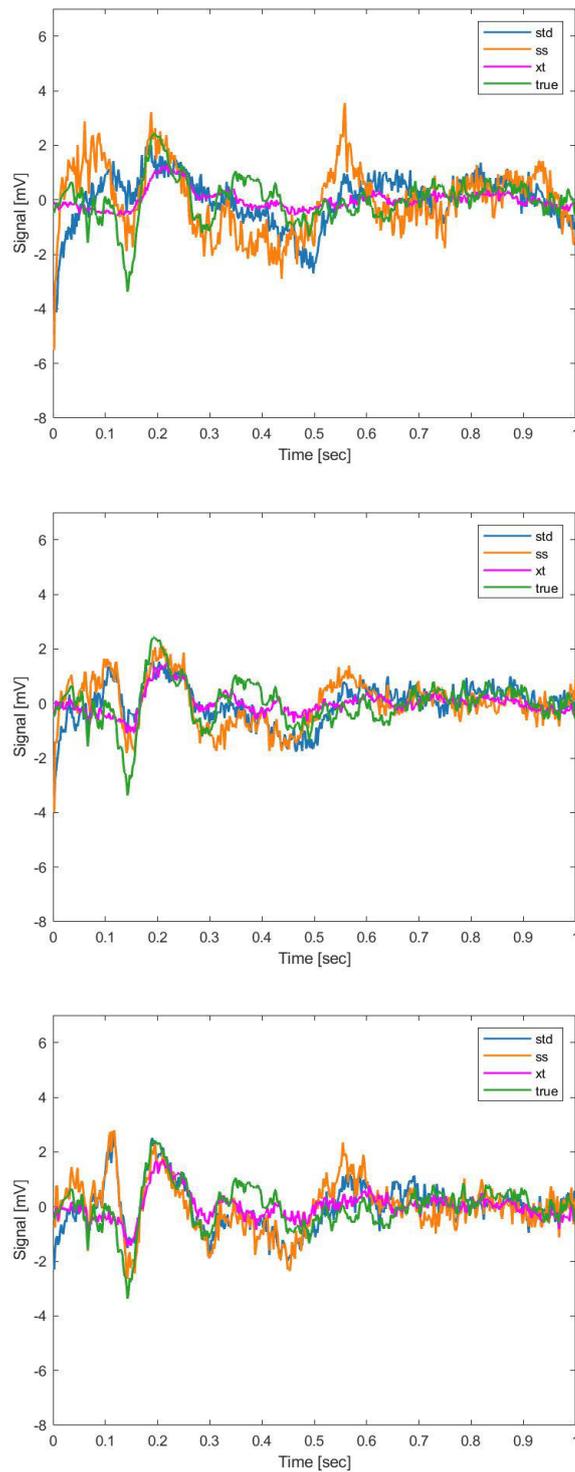


Figura 4.23: A confronto la ricostruzione finale del canale 25 con i tre metodi analizzati CS, R-CS e R-CSd rispetto al filtrato ($m=64, 128, 256$) utilizzando il metodo delle due matrici. Il canale 25 corrisponde a 'FC6'.

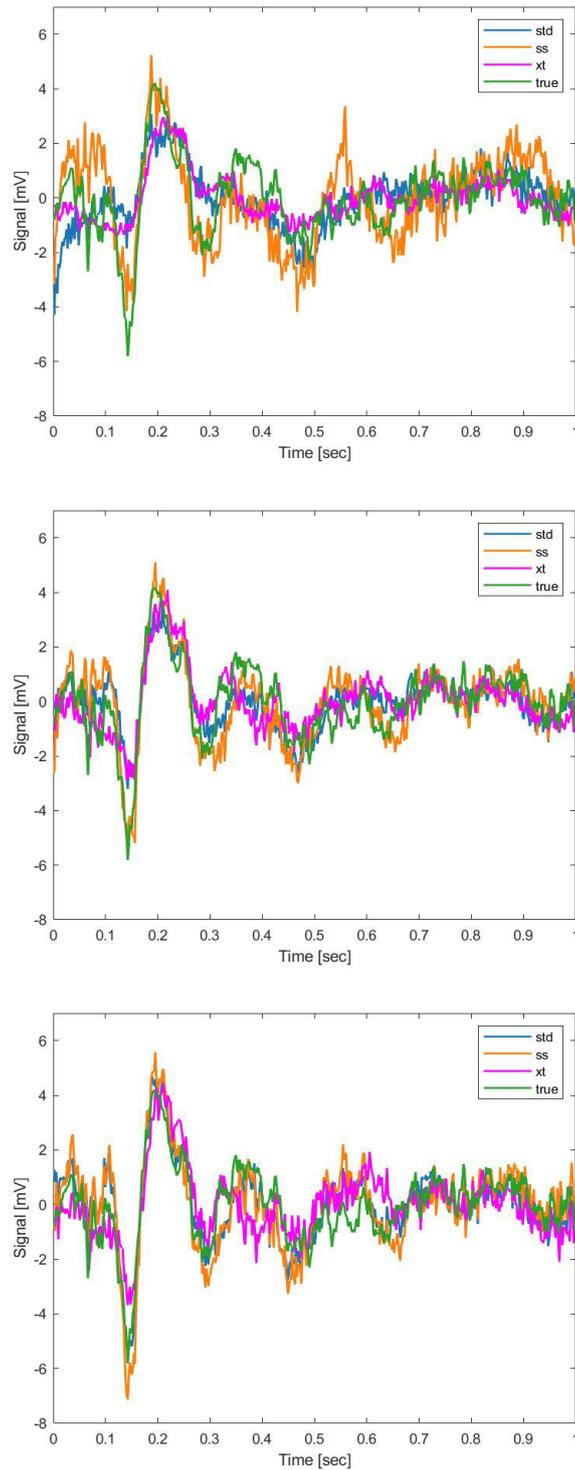


Figura 4.24: A confronto la ricostruzione finale del canale 32 con i tre metodi analizzati CS, R-CS e R-CSd rispetto al filtrato ($m=64, 128, 256$) utilizzando il metodo delle due matrici. Il canale 32 corrisponde a 'Cz'.

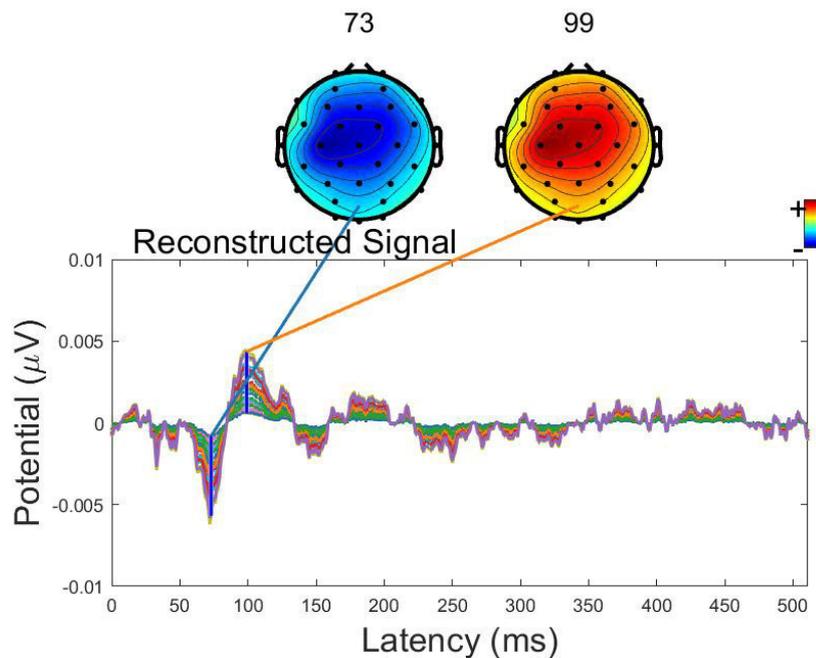
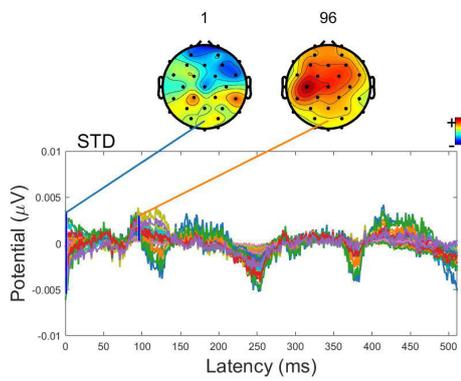
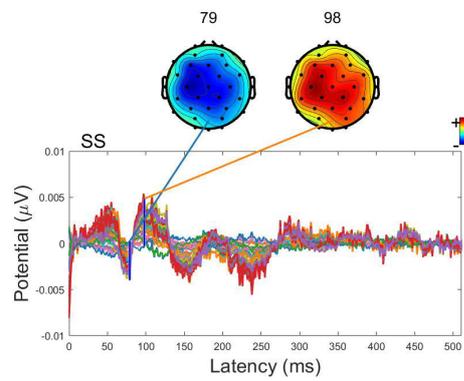


Figura 4.25: Informazione vera del segnale su tutti i canali.

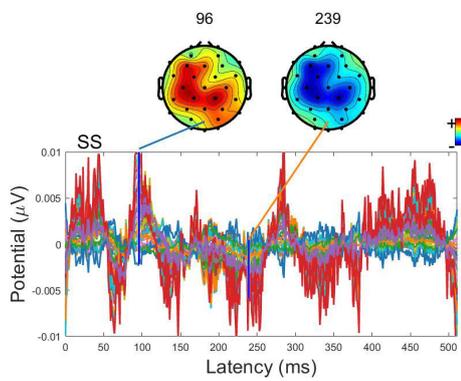
di compressione 8 e confrontandole con il riferimento 4.25 si vede che la R-CSd ha già delle buone performance sia considerando il calcolo con un'unica matrice sia considerando il caso con due; stessa cosa non si può dire del CS, mentre R-CS sembra essere migliore in questo caso nel metodo con unica matrice. Passando ad un m pari a 128 e quindi confrontando 4.27, si vede che in realtà entrambe le performance tendono ad ottimi risultati, lo stesso passando alle figure 4.28. Quello che si può vedere è anche come all'aumentare di m , gli istanti di tempo in cui si rilevano i picchi massimi e minimi del segnale si assestano a quelli veri. Altro particolare è il disturbo del movimento degli occhi che nell'algoritmo della Rakeness con Denoising sembra dare particolarmente fastidio causando una leggera deviazione verso la parte posteriore del cranio del contenuto informativo, mantenendo comunque performance ottimali. Si nota anche che in realtà l'algoritmo con la reiezione del disturbo ha già buone performance con m pari a 64 cosa che invece non sembrano avere il CS e R-CSd. Possiamo concludere dicendo che in realtà le performance con due cluster rispetto all'unico cluster non sembrano dare particolari miglioramenti, considerando sia le ricostruzioni che gli istanti di tempo in cui si rileva il massimo segnale.



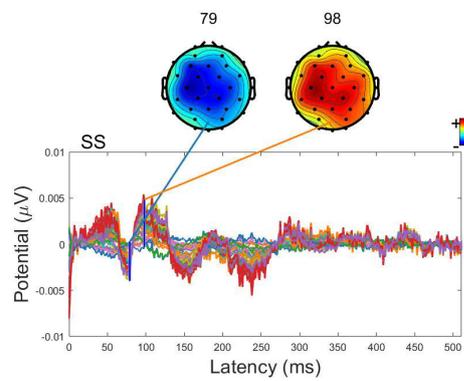
(a) *CS metodo 2 matrici*



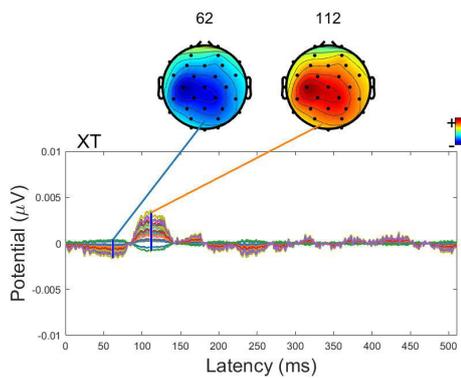
(b) *CS metodo 1 matrice*



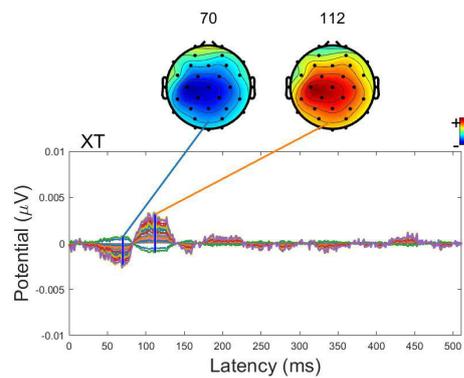
(c) *R-CS metodo 2 matrici*



(d) *R-CS metodo 1 matrice*

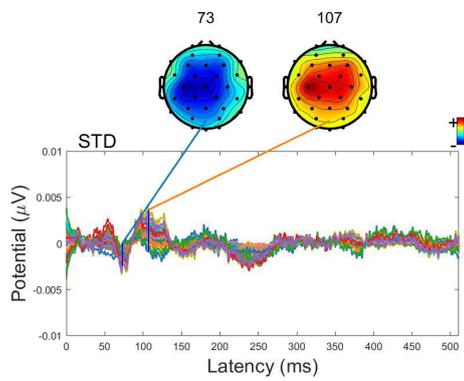


(e) *R-CSd metodo 2 matrici*

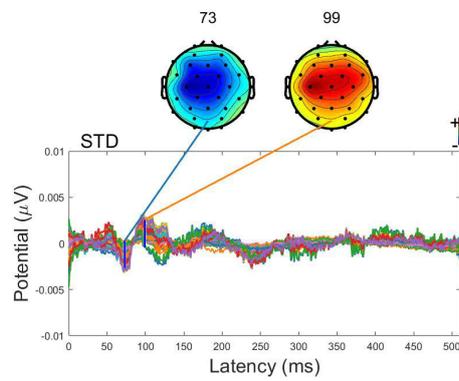


(f) *R-CSd metodo 1 matrice*

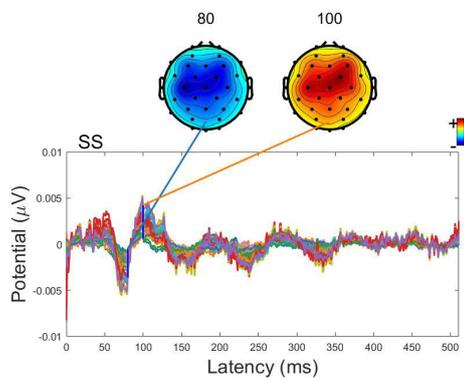
Figura 4.26: Rappresentazione informativa di tutti i canali con i 3 diversi metodi nei due algoritmi con $m=64$



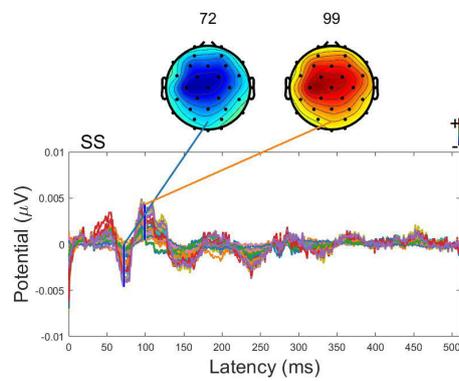
(a) *CS metodo 2 matrici*



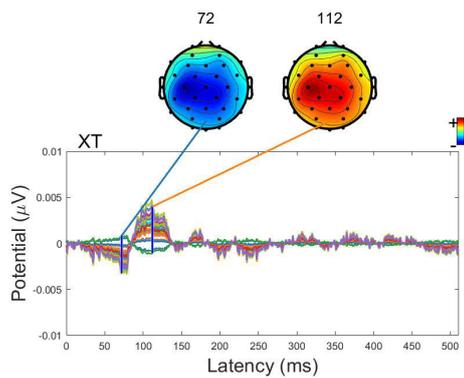
(b) *CS metodo 1 matrice*



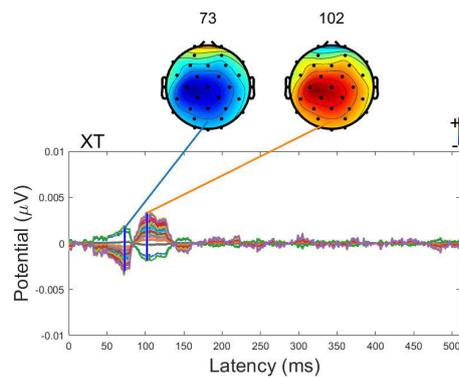
(c) *R-CS metodo 2 matrici*



(d) *R-CS metodo 1 matrice*

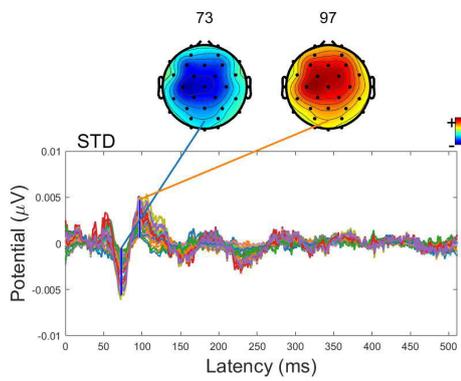


(e) *R-CSd metodo 2 matrici*

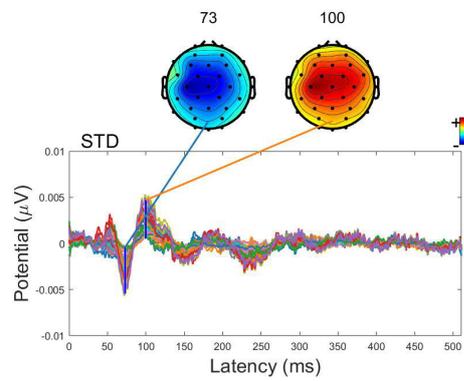


(f) *R-CSd metodo 1 matrice*

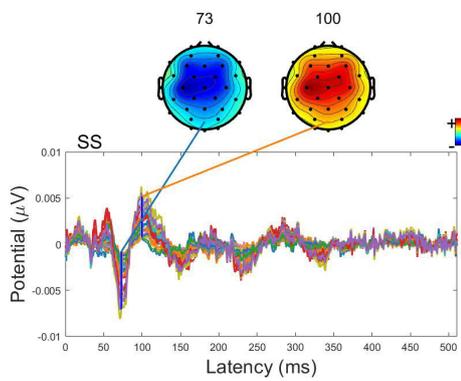
Figura 4.27: Rappresentazione informativa di tutti i canali con i 3 diversi metodi nei due algoritmi con $m=128$



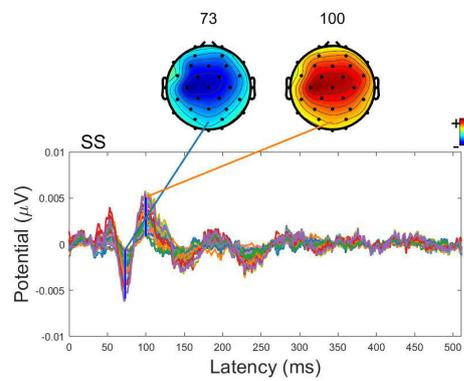
(a) *CS metodo 2 matrici*



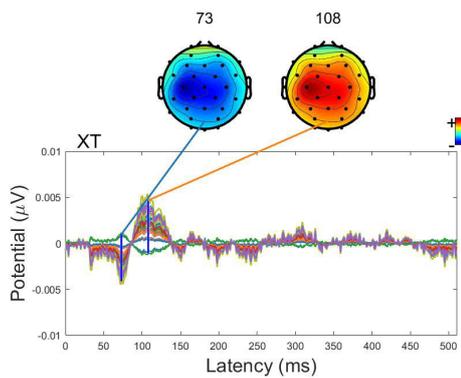
(b) *CS metodo 1 matrice*



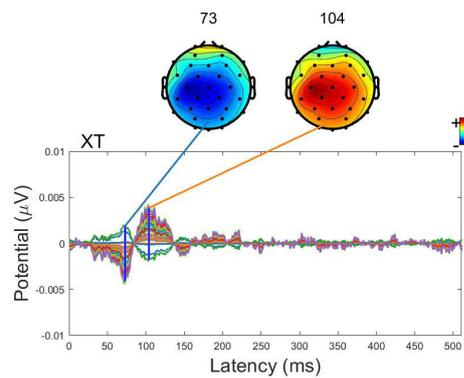
(c) *R-CS metodo 2 matrici*



(d) *R-CS metodo 1 matrice*



(e) *R-CSd metodo 2 matrici*



(f) *R-CSd metodo 1 matrice*

Figura 4.28: Rappresentazione informativa di tutti i canali con i 3 diversi metodi nei due algoritmi con $m=256$

Capitolo 5

Conclusioni

Sono stati presentati nel terzo capitolo gli algoritmi del Gradiente Proiettato Discendente e della proiezione alternata come metodi che risolvono il problema di minimizzazione della rakeness per la reiezione del disturbo. Come ci si aspettava, quest'ultima "sovraperforma" un solutore general purpose in termini di tempo di calcolo per la CPU e, più importante, è in grado di convergere al punto minimo anche per valori ragionevolmente alti di n .

In seguito abbiamo fatto un passo avanti nella realizzazione di un sistema di monitoraggio EEG a bassa potenza basato sul CS. In primo luogo, abbiamo identificato le caratteristiche salienti dei potenziali evocati, insieme alle componenti più importanti del rumore e alle attività fisiologiche indesiderate. Quindi, gli algoritmi di denoising allo stato dell'arte utilizzati per il filtraggio EP offline sono stati analizzati e applicati ai segnali EEG per ottenere un set di allenamento "pulito" per la stima dell'auto-correlazione del segnale EP. L'auto-correlazione EP calcolata sui dati filtrati DSS1 è stata utilizzata per sfruttare un nuovo concetto nella teoria CS chiamato rakeness che, allineando correttamente le proprietà statistiche delle sequenze di sensing rispetto alle caratteristiche di localizzazione dei segnali, è in grado di migliorare le prestazioni del sistema. Vengono indicati due risultati interessanti: il rapporto di compressione può aumentare ragionevolmente mantenendo una buona precisione di ricostruzione e alcuni componenti di rumore possono essere rimossi in modo efficiente senza bisogno di ulteriori livelli di filtraggio o di elaborazione. In particolare si è visto come attraverso l'algoritmo della Rakeness

con Denoising siamo stati in grado di togliere un filtro a valle della catena di elaborazione ricavando lo stesso contenuto informativo che si ha dalla Rakeness e quindi anche dal segnale filtrato senza ricostruzione.

5.1 Sviluppi futuri

Nel lavoro descritto in questo studio ci sono delle parti che potrebbero essere approfondite ed ulteriormente sviluppate. Si può pensare al parametro α , utile nel calcolo del Metodo del Gradiente alternato, che sta ad indicare di quanto la funzione obiettivo deve scendere ad ogni iterazione per cercare il minimo. Settare una precisa dipendenza dalla caratteristica del segnale in ingresso ($x + v$) a questo parametro permetterebbe forse un'ancora più veloce risoluzione del problema. Si può pensare anche ad r parametro che compare nell'ultimo vincolo della risoluzione del problema di minimizzazione della Rakeness con denoising. Questo parametro tiene conto di quanto il problema si deve allontanare dalla soluzione della Rakeness classica e quindi di quanto la soluzione del sistema è autorizzata a spostarsi dal segnale per compensare il disturbo. Quello che si potrebbe cercare di fare è parametrizzare il vincolo allo scopo di quantificare il disturbo presente nel segnale grezzo e quindi automaticamente valutare il valore più corretto che r debba assumere. Altra analisi che potrebbe essere sviluppata è la ricostruzione del segnale EEG con l'utilizzo di due cluster, settando diversamente i parametri degli ultimi tre filtri applicati si potrebbero ottenere forse performance migliori.

Punto debole della Rakeness con denoising è l'approccio a sostituzione di Filtri complessi o spaziali. Il problema è nato nel corso dell'approccio dell'algoritmo ai dati EEG; Avendo a disposizione filtri studiati ad hoc per questa classe di segnale che sfruttano tecniche avanzate di elaborazioni del segnale come per esempio la PCA, e combinano l'analisi su un'insieme di sensori (e quindi eseguono un'analisi spaziale), la Rakeness con denoising avendo un'algoritmo intrinseco lineare fa fatica a compensare questi tipi di filtraggi. Per questo motivo è stato possibile "sostituire" solo il filtro passa alto, lasciando a chiusura della catena i 3 filtraggi TSPCA, SNS e DSS.

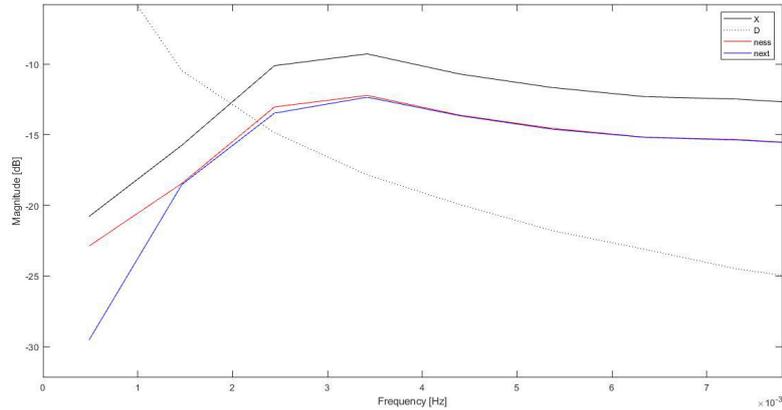


Figura 5.1: Dettaglio a bassa frequenza del canale 14.

Figura 5.2: Spettri in frequenza del segnale, del disturbo e degli algoritmi R-CS e R-CSd che mostrano come correggono il segnale. Nell'ultima figura in basso si vede il dettaglio a basse frequenze per il canale 14

Il dettaglio più particolare che differenzia la Rakeness classica da quella con il Denoising è la variazione che gli spettri hanno a bassa frequenza (vedi ultima figura in basso di Fig 5.2) dove si vede che la R-CSd si abbassa a bassissime frequenze dove infatti in questo caso si ha un elevato disturbo, cosa che invece non fa il R-CS.

Si noti che l'applicazione di questo algoritmo non si riduce all'eliminazione delle basse frequenze ma si può amplificare a bande di frequenza. Si prenda come esempio il caso della Brain Computer Interface: in questi casi è necessario rilevare l'informazione proveniente da una banda precisa in frequenza da alcuni canali.

Ringraziamenti

Vorrei esprimere gratitudine verso tutti coloro che mi hanno aiutata nella stesura della tesi con suggerimenti, osservazioni e critiche sottolineando che spetta solo a me la responsabilità per ogni errore contenuto in questo lavoro.

Un ringraziamento speciale va al Professor Mangia, che mi ha sempre seguita, supportata, sempre aiutata nel percorso della Tesi, ha sempre trovato del tempo da dedicarmi ma soprattutto ha sempre avuto fiducia in me. É a lui che devo questo lavoro. Grazie al Professore Rovatti, sempre chiaro e disponibile, basta una conversazione per trasmetterti la passione che mette nei suoi studi. Grazie anche a tutte le persone che nel dipartimento hanno reso questa esperienza unica e divertente.

Vorrei poi ringraziare tutte le persone che in qualche modo hanno contribuito più in generale al mio percorso universitario. A mio fratello Marco con il quale ho condiviso tutte le cose belle e brutte della nostra famiglia, non abbiamo bisogno di parlare, ci basta uno sguardo.

Ai miei genitori che mi hanno sempre supportata: a mio padre Mauro che non mi ha mai fatto mancare niente, che ha sempre sorretto economicamente tutte le spese di questo percorso e a mia madre Viviana che invece mi ha sorretto sempre dal punto di vista psicologico, mia prima fan e sostenitrice da sempre. Alle mie nonne che nonostante la distanza mi sono sempre vicine con il cuore e le preghiere.

Alle amiche di una vita con le quali sono cresciuta e che ho avuto la fortuna di avere sempre vicino in ogni difficoltà e a tutte le persone che anche se conosco da meno tempo non sono state da meno. Voi ci siete sempre state.

Al gruppo di Ancona senza il quale probabilmente non avrei ancora una laurea triennale.

Agli amici che ho conosciuto qua a Bologna: a quelli che sono stati di passaggio, agli amici dell'università e a quelli incontrati per sbaglio che hanno comunque deciso di restare. Grazie di aver reso questo cammino speciale.

Un ringraziamento anche a te, che nonostante la distanza, gli alti e i bassi e i litigi ci sei sempre stato.

Un ringraziamento anche a tutte le persone che volontariamente o involontariamente mi hanno ferito ma allo stesso tempo forgiato, senza le quali probabilmente non sarei la persona che sono oggi.

Bibliografia

- [1] *I dieci trend tecnologici secondo Gartner*, EONEWS n. 612 - OTTOBRE 2017.
- [2] *Avanti tutta con l'Industrial Internet of Things*, EONEWS n. 612 - OTTOBRE 2017.
- [3] *Dispositivi indossabili medicali*, EONEWS n. 612 - OTTOBRE 2017.
- [4] Amarlingam M., Pradeep Kumar Mishra, K. V. V. Durga Prasad, P. Rajalakshmi *Compressed Sensing for Different Sensors: A Real Scenario for WSN and IoT*, Department of Electrical Engineering, Indian Institute of Technology Hyderabad Hyderabad, India.
- [5] Hamza Djelouat, Abbas Amira e Faycal Bensaali *Compressive Sensing-Based IoT Applications: A Review*, Journal of Sensor and Actuator Networks, College of Engineering, Qatar University, Doha 2713, Qatar, 22 ottobre 2018.
- [6] Mauro Mangia, Fabio Pareschi, Valerio Cambareri, Riccardo Rovatti, Gianluca Setti, Javier Haboba, <http://cs.signalprocessing.it/>.
- [7] Alex Marchioni, Mauro Mangia, Fabio Pareschi, Riccardo Rovatti, Gianluca Setti, *Disturbance Rejection With Rakeness-based Compressed Sensing: Method and Application to Baseline/Powerline Mitigation in ECGs*.
- [8] Mauro Mangia, Fabio Pareschi, Valerio Cambareri, Riccardo Rovatti, Gianluca Setti, *Adapted Compressed Sensing for Effective Hardware Implementations. A Design Flow for Signal-Level Optimization of Compressed Sensing Stages*.
- [9] Federico Carpi, Danilo De Rossi, *Potenziali elettroencefalografici e potenziali evocati*, Fenomeni Bioelettrici.

- [10] Sharbrough F, Chatrian G-E, Lesser RP, Lüders H, Nuwer M, Picton TW, *American Electroencephalographic Society Guidelines for Standard Electrode Position Nomenclature*, J.Clin. Neurophysiol 8: 200-2, (1991).
- [11] Alain de Cheveigné, Jonathan Z. Simon, *Denoising based on time-shift PCA*, Journal of Neuroscience Methods, 3 Giugno 2007.
- [12] Alain de Cheveigné, Jonathan Z. Simon, *Sensor noise suppression*, Journal of Neuroscience Methods, 13 Settembre 2007.
- [13] Alain de Cheveigné, Jonathan Z. Simon, *Denoising based on spatial filtering*, Journal of Neuroscience Methods, 26 Marzo 2008.
- [14] E. van den Berg and M. P. Friedlander, *Probing the Pareto frontier for basis pursuit solutions*, SIAM J. on Scientific Computing, 31(2):890-912, November 2008
- [15] E. van den Berg and M. P. Friedlander, *Sparse optimization with least-squares constraints*, SIAM J. on Optimization, SIAM J. on Optimization, 21(4):1201–1229, 2011
- [16] Michael C. Grant, Stephen P. Boyd *The CVX Users*, Guide Release 2.1, CVX Research, Inc., December 15, 2018
- [17] EEGlab - Swartz Center for Computational Neuroscience
- [18] Mauro Mangia, Letizia Magenta, Alex Marchioni, Fabio Pareschix, Riccardo Rovatti, Gianluca Setti, *Projected-Gradient-Descent in Rakeness-based Compressed Sensing with Disturbance Rejection*,NGCAS 2018.
- [19] R.L. Dykstra, *An Algorithm for Restricted Least Squares Regression*, Journal of the American Statistical Association, vol. 78, n. 384, pp. 837–842, 1983
- [20] J.P. Boyle, R.L. Dykstra, *A method for finding projections onto the intersection of convex sets in Hilbert spaces*, Lecture Notes in Statistics, vol. 37, pp. 28—47, 1986

- [21] D. Bortolotti, M. Mangia, A. Bartolini, R. Rovatti, G. Setti, L. Benini, *An Ultra-Low Power Dual Mode ECG Monitor for Healthcare and Wellness*, proceeding of Design, Automation & Test in Europe, pp. 1611-1616, 2015
- [22] V. Cambareri, M. Mangia, F. Pareschi, R. Rovatti, G. Setti, *A Case Study in Low Complexity ECG Signal Encoding: How Compressing is Compressed Sensing?*, IEEE Signal Processing Letters, vol. 22, no. 10, pp.1743-11747, May 2015.
- [23] M. Mangia, R. Rovatti, G. Setti, *Rakeness in the Design of Analog-to-Information Conversion of Sparse and Localized Signals*, IEEE Transaction on Circuits and Systems I: Regular Papers, vol.59, no. 5, pp.1001-1014, May 2012.
- [24] D.L. Donoho, *Compressed Sensing*, IEEE Transaction on Information Theory, vol.54, no. 4, pp.1289-1306, May 2006.
- [25] E.J. Candes, T. Tao, *Decoding by linear programming*, IEEE Transaction on Information Theory, vol.51, no. 12, pp.4203-4215, Dec 2005.
- [26] M. Mangia, F. Parechi, V. Cambareri, R. Rovatti, G. Setti, *Rakeness-based Design of low-complexity compressed sensing*, IEEE Transaction on Circuits and Systems I: Regular Papers, vol.64, no. 5, pp.1201-1213, May 2017.
- [27] Jonathan Buckheit, Shaobing Chen, David Donoho, Iain Johnstone, Jeffrey Scargle, *About WaveLab*, Stanford University, NASA-Ames Research Center Version .850 December, 2005.
- [28] Jonathan Buckheit and David Donoho, Stanford University, *WaveLab Architecture*, Stanford University, Version .850, December, 2005.
- [29] Bathiya Senevirathna, Lauren Berman, Nicola Bertoni, Fabio Pareschi, Mauro Mangia, Riccardo Rovatti, Gianluca Setti, Jonathan Simon, Pamela Abshire, *Low Cost Mobile EEG for Characterization of Cortical Auditory Responses*, 2016 IEEE International Symposium on Circuits and Systems (ISCAS).

- [30] Nicola Bertoni, Bathiya Senevirathnax, Fabio Pareschi, Mauro Mangiay, Riccardo Rovatti, Pamela Abshirex, Jonathan Z. Simonx, Gianluca Setti, *Low-power EEG monitor based on Compressed Sensing with Compressed Domain Noise Rejection*, 2016 IEEE International Symposium on Circuits and Systems (ISCAS).
- [31] James P. BoyleRichard L. Dykstra, *A Method for Finding Projections onto the Intersection of Convex Sets in Hilbert Spaces*, Part of the Lecture Notes in Statistics book series (LNS, volume 37)
- [32] Sthephen Boyd e Jon Dattorro, *Alternating Projections*, EE392o, Stanford University. Autumn, 2003.
- [33] Daniele Bortolotti, Mauro Mangia, Andrea Bartolini, Riccardo Rovatti, Gianluca Setti e Luca Benini *An Ultra-Low Power Dual-mode ECG Monitor for Healthcare and Wellness*, 2015 Design, Automation & Test in Europe Conference & Exhibition (2015)