

ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA
SCUOLA DI INGEGNERIA

Laurea Magistrale in
INGEGNERIA PER L'AMBIENTE E IL TERRITORIO

Corso: Modelli per la geotermia e i fluidi nel sottosuolo

*Installazione, configurazione ed utilizzo di
Waiwera:
un free software open-source per simulazioni di
geotermia ad alta entalpia*

Candidata
Chiara Falcioni

Relatore
Prof. Villiam Bortolotti

Correlatore
Prof. Stefano Bonduà

Anno Accademico [2020/2021]

Sommario

La riduzione degli impatti dovuti al cambiamento climatico mediante una decarbonizzazione del settore energetico passa inevitabilmente attraverso ad una Transizione Energetica che si orienta sempre più all'utilizzo di fonti energetiche rinnovabili. In tale sfida la produzione energetica attraverso fonti geotermiche può giocare un ruolo importante. A rigore la geotermia non è una fonte di energia rinnovabile, diventa tale se correttamente coltivata, ossia utilizzata con piani di sfruttamento sostenibili nel lungo periodo. Per tale motivo l'attività di modellazione numerica diventa un fattore chiave nella coltivazione dell'energia geotermica. Nonostante aumenti significativi e generalizzati della potenza di calcolo, la necessità di modelli geotermici più grandi e complessi ha imposto al software di simulazione richieste sempre più severe, in particolare in termini di efficienza computazionale.

In tale contesto, il gruppo geotermico dell'Università di Auckland ha sviluppato un nuovo simulatore di flusso geotermico ad alta entalpia con la capacità di gestire i cambi di fase, chiamato "Waiwera" (che significa "acqua calda" in lingua Maori). In tale ambito, Waiwera è probabilmente il primo ad offrire le funzionalità e le prestazioni richieste per modelli di serbatoi geotermici grandi e complessi ed essere rilasciato open source e gratuitamente. Waiwera è stato pensato e realizzato con moderne tecniche di programmazione object-oriented per lavorare in modalità parallela su computer condivisi e cluster, con la possibilità di operare su diversi sistemi operativi. Questa versatilità si traduce però in una complessità iniziale di messa in funzione del software specialmente sotto Windows.

Nel presente lavoro di tesi Waiwera è stato installato, configurato e testato con un caso studio di letteratura, su di un server equipaggiato con Windows 10. I risultati di tale attività sono stati pienamente all'altezza delle aspettative, ottenendo un server per simulazioni geotermiche ad alta entalpia perfettamente funzionante e performante. Inoltre, come primo passo verso un confronto delle prestazioni di Waiwera con altri simulatori, è stata effettuata una simulazione del caso studio in esame con il codice TOUGH2-EWASG, ottenendo risultati in linea con quelli di Waiwera.

Indice

| | |
|---|----|
| Indice | 4 |
| Indice delle figure..... | 6 |
| Indice delle tabelle | 8 |
| 1 Introduzione alla Geotermia | 11 |
| 1.1 Struttura interna della Terra..... | 11 |
| 1.1.1 Crosta | 12 |
| 1.1.2 Mantello | 13 |
| 1.1.3 Il Nucleo..... | 13 |
| 1.2 Flusso geotermico terrestre..... | 14 |
| 1.3 Bacini geotermici..... | 15 |
| 1.3.1 Geotermia a bassa, media ed alta entalpia..... | 16 |
| 2 Concetti di simulazione..... | 17 |
| 2.1 Sistemi e modelli | 17 |
| 2.1.1 Tipi di sistema | 18 |
| 2.1.2 Modello del Sistema..... | 18 |
| 2.1.3 Generalità sulla simulazione di bacini geotermici | 20 |
| 2.1.4 Problematiche correlate all'impiego di modelli numerici | 21 |
| 3 Simulatori per bacini geotermici..... | 23 |
| 3.1 TOUGH2 versione 2.0..... | 24 |
| 3.2 T2Well..... | 26 |
| 4 Waiwera | 27 |
| 4.1 Caratteristiche di funzionamento..... | 27 |
| 4.1.1 Equazioni di conservazione della massa e dell'energia..... | 27 |
| 4.1.2 Discretizzazione a volumi finiti | 28 |
| 4.1.3 Variabili primarie | 28 |
| 4.1.4 Evoluzione nel tempo..... | 29 |
| 4.1.5 Valutazioni delle funzioni | 30 |
| 4.1.6 Soluzione di equazioni ad ogni passo temporale | 31 |
| 4.2 Struttura del file di Input | 32 |
| 4.2.1 Formato JSON..... | 32 |
| 4.2.2 Struttura del file di input | 33 |
| 4.3 Output di simulazione HDF5..... | 45 |
| 4.4 File di registro YAML..... | 48 |

| | | |
|--------|---|----|
| 5 | Installazione ed esecuzione di Waiwera | 50 |
| 5.1 | Procedura di istallazione..... | 50 |
| 5.1.1 | Docker Desktop per Windows | 50 |
| 5.1.2 | Libreria Python Pywaiwera | 53 |
| 5.2 | Esecuzione | 54 |
| 6 | Caso studio-flusso 1D con presenza di CO ₂ | 57 |
| 6.1 | Input “co2_column_1.json” | 58 |
| 6.1.1 | Title | 61 |
| 6.1.2 | Mesh..... | 62 |
| 6.1.3 | Rock | 63 |
| 6.1.4 | Eos..... | 64 |
| 6.1.5 | Boundaries..... | 64 |
| 6.1.6 | Sources | 65 |
| 6.1.7 | Initial | 65 |
| 6.1.8 | Gravity..... | 66 |
| 6.1.9 | Thermodynamics..... | 66 |
| 6.1.10 | Time | 66 |
| 6.1.11 | Outputfile | 67 |
| 6.2 | Elaborazione dati | 67 |
| 6.2.1 | Utilizzo di HDF5view | 67 |
| 6.2.2 | Utilizzo di ParaView | 69 |
| 6.2.3 | Costruzione grafici dai dati di output..... | 74 |
| 6.3 | Risultati..... | 78 |
| 6.3.1 | Risultati numerici | 78 |
| 6.3.2 | Risultati grafici..... | 79 |
| 6.4 | Simulazione del caso studio con TOUGH2-EWASG | 84 |
| 7 | Conclusioni | 94 |
| | Bibliografia..... | 95 |
| | Sitografia | 98 |

Indice delle figure

| | |
|---|----|
| Figura 1 Possibili schemi di come possa essere rappresentato l'interno della Terra. (immagine modificata dal sito http://www.pi.ingv.it/divulgazione/download/esperienze-di-sismologia/interno-della-terra/ [S35])..... | 12 |
| Figura 2 Flusso termico del globo terrestre (immagine modificata dal sito https://docplayer.it/52009763-La-struttura-della-terra.html [S36])..... | 14 |
| Figura 3 Sistema ingresso-uscita. Immagine estrapolata dalle dispense del corso MGFS (2019)..... | 18 |
| Figura 4 Schema di funzionamento di Tough2 versione 2.0. Immagine modificata dal manuale di Tough2 [10]..... | 25 |
| Figura 5 Ortogonalità della maglia. Immagine riportata dal manuale Waiwera..... | 35 |
| Figura 6 Regioni termodinamiche IAPWS-97. Immagine riportata dal manuale Waiwera..... | 37 |
| Figura 7 Gestione attività, stato di Virtualizzazione Abilitata..... | 51 |
| Figura 8 Utilizzo di Core info..... | 52 |
| Figura 9 Attivazione Piattaforma macchina virtuale, Sottosistema Windows per Linux, Contenitori e Hyper-V tramite finestra di funzionalità Windows tramite finestra di funzionalità Windows..... | 53 |
| Figura 10 Percorso eseguibile Waiwera - "waiwera-dkr" e Problemi di esempio..... | 55 |
| Figura 11 File di output generati con l'esecuzione di Waiwera per i 3 problemi di input co2_column.x.json | 56 |
| Figura 12 Modello concettuale "cos2_column_1.json". Immagine riportata da una pubblicazione scientifica [9]..... | 57 |
| Figura 13 Apertura file di input "cos_column_1.json"(a)..... | 59 |
| Figura 14 Apertura file di input "cos_column_1.json"(b)..... | 60 |
| Figura 15 Apertura file di input "cos_column_1.json"(c)..... | 61 |
| Figura 16 Descrizione mesh "gco2_column.msh"..... | 62 |
| Figura 17 Curve di permeabilità relativa. Andamento lineare "co2_column_1.json"..... | 63 |
| Figura 18 Eos “wce”, water, carbon dioxide and energy..... | 64 |
| Figura 19 Apertura primo output HDF5view..... | 68 |
| Figura 20 Visualizzazione output "problem1.h5" con apertura tabella “fluid_temperature” tramite HDFview..... | 68 |
| Figura 21 Inizializzazione ParaView..... | 69 |
| Figura 22 Visualizzazione "co2_column_1.h5" tramite ParaView..... | 70 |

| | |
|---|----|
| Figura 23 Modifica visualizzazione co2_column_1.h5 ParaView..... | 71 |
| Figura 24 Visualizzazione valori outputs tabellati tramite ParaView..... | 72 |
| Figura 25 Selezione ultimo step temporale ParaView..... | 73 |
| Figura 26 Visualizzazione ultimo step temporale ParaView..... | 73 |
| Figura 27 Script Python per ottenere Grafico temperatura/profondità all'ultimo step temporale riportato dalla guida Waiwera partendo da un file di output HDF ₅ | 74 |
| Figura 28 Script Python modificato per ottenere grafico temperatura-profondità all'ultimo step temporale da file di output "co2_column_1.h5"..... | 75 |
| Figura 29 Script Python creato per ottenere il grafico dell'andamento della pressione parziale di CO ₂ con la profondità. Da file "co2_column_0.1.h5", "co2_column_1.h5", "co2_column_5.h5" (ultimo step temporale)..... | 76 |
| Figura 30 Script Python per ottenere Grafico dell'andamento della dimensione del passo temporale riportato dalla guida Waiwera partendo da un file di registro YAML..... | 77 |
| Figura 31 Script Python modificato per ottenere grafico dell'andamento della dimensione del passo temporale da file "co2_column_1.yaml"..... | 77 |
| Figura 32 Visualizzazione output tramite Silx da manuale Waiwera..... | 78 |
| Figura 33 Visualizzazione output "co2_column_1.h5"..... | 79 |
| Figura 34 Grafico temperatura/profondità ottenuto da script Python e riportato nella guida Waiwera relativo all'ultimo step temporale del file di output "co2_column_1.h5"..... | 80 |
| Figura 35 Grafico temperatura/profondità ottenuto da script Python modificato relativo all'ultimo step temporale del file di output "co2_column_1.h5"..... | 80 |
| Figura 36 Grafico di riferimento per l'andamento pressione parziale di CO ₂ con la profondità. Da file "co2_column_0.1.h5", "co2_column_1.h5", "co2_column_5.h5"(ultimo step temporale)..... | 81 |
| Figura 37 Grafico dell'andamento della pressione parziale di CO ₂ con la profondità ottenuto da file "co2_column_0.1.h5", "co2_column_1.h5", "co2_column_5.h5" (ultimo step temporale)..... | 82 |
| Figura 38 Grafico dimensioni del passo temporale ottenuto da script Python e riportato dalla guida Waiwera. Da file di registro "co2_column_1.yaml"..... | 83 |
| Figura 39 Grafico ottenuto da Script Python con andamento della dimensione del passo temporale da file "co2_column_1.yaml"..... | 83 |
| Figura 40 Andamento pressione parziale di CO ₂ con la profondità, Croucher 2020[9]..... | 85 |
| Figura 41 Andamento pressione parziale di CO ₂ con la profondità ottenuto con TOUGH2 - EWASG..... | 85 |

Indice delle tabelle

| | |
|---|----|
| Tabella 1 Moduli EOS per TOUGH2 presenti nel sito https://tough.lbl.gov/licensing-download/itough2-price-sheet/ . Immagine estrapolata da Ambrogetti, L. 2020 [17]..... | 26 |
| Tabella 2 Struttura file JSON input Waiwera. Tabella riportata dal manuale Waiwera..... | 34 |
| Tabella 3 Modulo EOS "w". Tabella riportata dal manuale Waiwera..... | 37 |
| Tabella 4 Modulo EOS "we". Tabella riportata dal manuale Waiwera..... | 38 |
| Tabella 5 Modulo EOS "wae". Tabella riportata dal manuale Waiwera..... | 39 |
| Tabella 6 Modulo EOS "wce". Tabella riportata dal manuale Waiwera | 39 |
| Tabella 7 "rock" input Waiwera. Tabella riportata dal manuale Waiwera | 40 |
| Tabella 8 "rock_types" input Waiwera. Tabella riportata dal manuale Waiwera..... | 40 |
| Tabella 9 "rock_relative_permeability" input Waiwera, curva di permeabilità relativa lineare. Tabella riportata dal Manuale Waiwera | 41 |
| Tabella 10 "boundaries" input Waiwera. Tabella riportata dal manuale Waiwera..... | 42 |
| Tabella 11 "cells" input Waiwera. Tabella riportata dal manuale Waiwera..... | 42 |
| Tabella 12 "sources" input Waiwera. Tabella riportata dal manuale Waiwera..... | 43 |
| Tabella 13 "primary_condition" input Waiwera. Tabella riportata dal manuale Waiwera..... | 43 |
| Tabella 14 "time" input Waiwera. Tabella riportata dal manuale Waiwera..... | 44 |
| Tabella 15 Struttura file di input "co2_column_1.json"..... | 58 |
| Tabella 16 Problema "co2_column_1.json"- "types" input..... | 64 |

Simbologia

| <i>Simbolo</i> | <i>Significato</i> | <i>Unità di misura</i> |
|----------------|---|------------------------|
| A | - Superficie | m^2 |
| c | - Calore specifico della roccia | $J/kg/^\circ C$ |
| F | - Flusso (attraverso il confine ∂V) | $kg/m^2/s$ o $J/m^2/s$ |
| g | - Vettore gravità | m/s^2 |
| H | - Entalpia | J |
| J | - Matrice jacobiana | --- |
| k | - Permeabilità effettiva | m^2 |
| k (apice) | - Componente k-esimo | --- |
| k^r | - Permeabilità relativa di fase | [-] |
| K | - Conducibilità termica | $W/m/^\circ C$ |
| L | - Bilanci di massa e di energia medi delle celle | --- |
| M | - Massa per unità di volume o energia per unità di volume | kg/m^3 o J/m^3 |
| \hat{n} | - Vettore unitario normale | --- |
| p | - Pressione | Pa o bar |
| | - Fase p-esima | --- |
| q | -Portate per unità di volume | $kg/m^3/s$ o $J/m^3/s$ |
| S | - Saturazione | [-] |
| t | - Tempo | s |
| Δt | - Dimensione del passo temporale | s |
| T | - Temperatura | $^\circ C$ |
| u | - Densità di energia interna | J/kg |
| U | - Energia interna | J |
| V | - Volume | m^3 |
| X | - Frazione in massa | [-] |
| Y | - Vettore delle variabili primarie | --- |
| ρ | - Densità | kg/m^3 |
| v | - Viscosità cinematica | m^2/s |
| ϕ | -Porosità | [-] |

1 Introduzione alla Geotermia

La geotermia è stata utilizzata per la prima volta per la produzione di energia elettrica in Italia nel 1904, e negli ultimi anni ha rappresentato un settore in continua espansione. Infatti, secondo l'Agenzia internazionale per le energie rinnovabili [1], la produzione di energia da fonte geotermica è cresciuta in tutto il mondo da circa 10 GW nel 2010 a 13,3 GW nel 2018 in modo costante.

L'energia geotermica è di fatto calore proveniente dal sottosuolo e immagazzinato nelle rocce e nei fluidi sotto la crosta terrestre.

Per produrre questo tipo di energia, ad esempio, possono essere scavati dei pozzi fino a diverse centinaia o migliaia di metri di profondità dove si trovano i serbatoi sotterranei che consentono di accedere al vapore e all'acqua calda. Quest'ultimi possono essere utilizzati per azionare turbine collegate a generatori di elettricità. Le centrali geotermiche possono essere classificate in tre categorie principali: a vapore dominante (*dry stream*), ad acqua dominante (*flash*) e a ciclo binario [S37].

La tipologia *dry stream* è la forma più antica di tecnologia geotermica e consiste nel prelevare il vapore dal reservoir che viene utilizzato per azionare direttamente una turbina. Gli impianti *flash* utilizzano invece acqua calda ad alta pressione che viene raffreddata a bassa pressione, mentre gli impianti binari adoperano un secondo liquido, attraverso cui viene fatta passare l'acqua calda, con un temperatura di ebollizione più bassa dell'acqua che si trasforma in vapore per azionare la turbina.

1.1 Struttura interna della Terra

La Terra si è formata tra i 4,5 e i 4,6 miliardi di anni fa. In prima approssimazione si può paragonare la sua forma a quella di un ellissoide di rotazione con un leggero schiacciamento ai poli e caratterizzato da un raggio medio polare di 6356,7 km e un raggio medio equatoriale di 6378,2 km. Rifarsi a un ellissoide come approssimazione della Terra non è propriamente corretto, infatti, a causa di una distribuzione del materiale non omogenea all'interno della Terra e alla presenza di zone in rilievo e di depressione sulla sua superficie, risulta migliore l'approssimazione ad un geoide. Ad ogni modo, gli scostamenti tra le quote ellissoidiche e geodetiche assumono valori massimi dell'ordine dei 100 metri [S1].

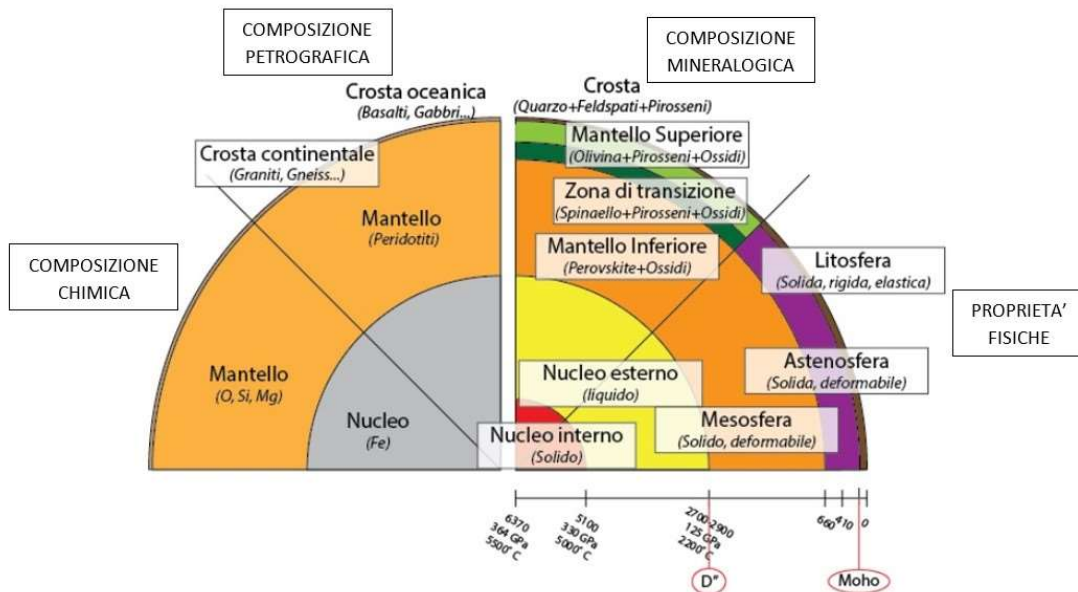


Figura 1 Possibili schemi di come possa essere rappresentato l'interno della Terra. (immagine modificata dal sito <http://www.pi.ingv.it/divulgazione/download/esperienze-di-sismologia/interno-della-terra/> [S35]).

Le nozioni che abbiamo sulla struttura interna della Terra non sono ancora esaustive. Le informazioni a disposizione provengono da osservazioni di tipo dirette e indirette. Le osservazioni dirette, che consistono nel prelievo di campioni della roccia interna terrestre, non possono essere effettuate ad elevate profondità, e il pozzo più profondo mai perforato raggiunge una profondità massima di circa 12 km. Per questo motivo, il maggior numero di informazioni arriva da osservazioni indirette (es. bilanci di massa del pianeta, studio delle variazioni dei campi gravimetrici e magnetici, etc..). Come riportato da C. Doglioni [2], gli studi sul tema hanno portato ad uno schema di suddivisione della Terra caratterizzato da tre gusci concentrici (Figura 1), che si differenziano per spessore, densità e composizione mineralogica. Questi tre gusci prendono i nomi di Crosta, Mantello e Nucleo.

1.1.1 Crosta

La crosta terrestre [S2] è lo strato più esterno e, con una massa pari all'1 % del pianeta, costituisce l'involucro della Terra. Il limite inferiore della crosta è rappresentato dalla discontinuità di Mohorovicic (o di Moho) come raffigurato in Figura 1. La crosta terrestre può essere divisa in crosta continentale, che costituisce appunto i continenti, e crosta oceanica, che forma i fondali oceanici; entrambe con diverse caratteristiche di spessore, densità e composizione. La crosta continentale ha valori medi di 35-40 km di spessore, con picchi fino a

più di 70 km in corrispondenza delle catene montuose più elevate. La sua densità è di circa 2800 kg/m^3 ed è composta in prevalenza da rocce granitiche. La crosta oceanica ha invece valori medi di 8-10 km di spessore e densità media di 3000 kg/m^3 ed è composta da rocce basaltiche ricche di alluminio, ferro e silicio.

1.1.2 Mantello

Il mantello [S2] occupa l'84% del volume terrestre con una massa pari al 68% di quella terrestre. È limitato superiormente dalla superficie di discontinuità di Mohorovicic e inferiormente dalla superficie di discontinuità di Gutenberg. La temperatura del mantello varia da alcune centinaia di gradi in prossimità della crosta fino a $3000 \text{ }^\circ\text{C}$ in prossimità del nucleo; anche la pressione aumenta con la profondità (da circa 9000 atm a circa 1,3 milioni atm). Il mantello viene suddiviso in tre macro-zone (*Figura 1*) in base alla loro composizione chimica e allo stato fisico del materiale costituente: il mantello superiore, la zona di transizione e il mantello inferiore. La parte più superficiale del mantello è solida e forma con la crosta terrestre la litosfera; quest'ultima è costituita dalla placche tettoniche, i cui movimenti sono all'origine dei terremoti e delle eruzioni vulcaniche. Nella parte al di sotto della litosfera si trova un'altra parte del mantello che viene chiamata astenosfera e che è formata da rocce parzialmente fuse. Sotto l'astenosfera, si trova l'inizio della mesosfera (astenosfera intermedia), dove le rocce tornano allo stato solido. La mesosfera arriva a circa 2900 km di profondità, dove si trova la discontinuità di Gutenberg.

1.1.3 Il Nucleo

Il nucleo [S2] rappresenta il 15% del volume terrestre con una massa pari al 31% di quella della Terra. In base alla fase delle componenti che lo compongono, viene suddiviso in due gusci concentrici: il nucleo esterno e il nucleo interno. Il primo è in fase liquida, e composto principalmente da ferro e nichel con una temperatura di circa $3000 \text{ }^\circ\text{C}$, una pressione di 130 GPa e una densità di 9300 kg/m^3 . Il nucleo interno è viscoso, composto in gran parte da ferro, con un raggio di circa 1250 km, una temperatura attorno ai 5800°C , una pressione di 330–365 GPa e una densità di 13000 kg/m^3 . Nonostante il nucleo interno abbia una temperatura maggiore di quello esterno, risulta viscoso inquanto caratterizzato da una pressione superiore, che porta a un innalzamento notevole del punto di fusione del ferro.

1.2 Flusso geotermico terrestre

Con l'aumentare della profondità dalla superficie terrestre, cresce anche la temperatura della Terra e, poiché il calore segue un principio di trasmissione da corpo caldo a corpo freddo, nel pianeta è presente un flusso geotermico, cioè un flusso di calore, diretto dall'interno verso l'esterno [3]. Tale flusso risulta comunque molto variabile, come è possibile intuire da *Figura 2*.

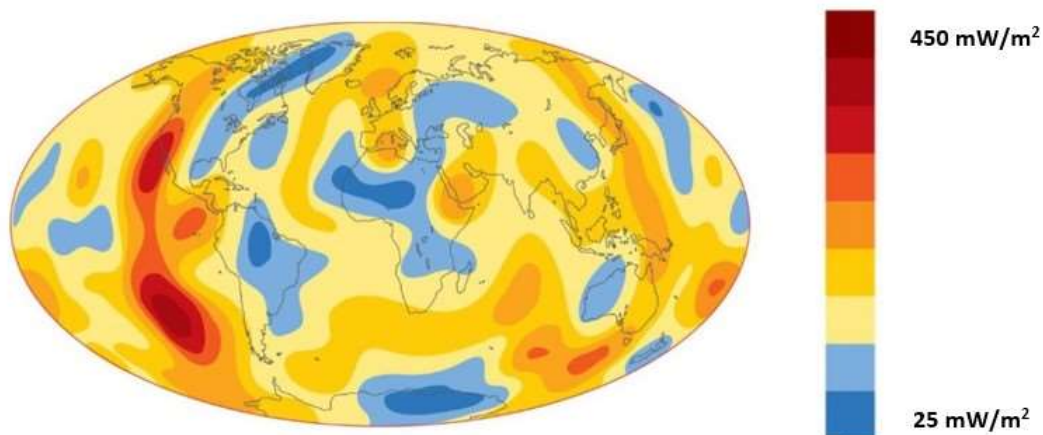


Figura 2 Flusso termico del globo terrestre (immagine modificata dal sito <https://docplayer.it/52009763-La-struttura-della-terra.html> [536]).

L'origine di questa forma di energia è attribuibile a due principali fattori: il calore primordiale del pianeta e il calore generato dal decadimento degli isotopi radioattivi di elementi come uranio, torio e potassio presenti nella crosta terrestre (calore radiogenico) [3]. Strettamente collegato alla temperatura interna della Terra vi è il gradiente termico, una quantità fisica che esprime il tasso di incremento della temperatura dalla crosta al nucleo. In condizioni normali esso assume un valore medio di 3 °C ogni 100 m di profondità con possibili variazioni da 1°C a 5 °C ogni 100 m.

In campo geotermico il gradiente di temperatura risulta essere un parametro di forte importanza in quanto la sua conoscenza è determinante per una corretta pianificazione dello sfruttamento delle fonti di calore del sottosuolo.

1.3 Bacini geotermici

Quando si parla di bacino o serbatoio geotermico [3] si fa riferimento ad una porzione del sottosuolo composta da rocce o formazioni geologiche di varia natura (spesso di tipo evaporitico come rocce carbonatiche) permeabili e caratterizzate da una discreta porosità efficace; questa parte del sottosuolo è situata al di sopra della fonte di calore. L'energia termica del sottosuolo è resa disponibile tramite vettori fluidi quali acqua o vapore, iniettati o naturali, che fluiscono spontaneamente dal serbatoio geotermico alla superficie (sorgenti termali, geysir, soffioni) o che vengono erogati artificialmente attraverso manufatti (pozzi geotermici) realizzati mediante perforazione meccanica. L'energia geotermica non è propriamente una fonte rinnovabile, ma può essere tale attraverso una corretta coltivazione, ossia con una coltivazione sostenibile in relazione ai suoi parametri fluido-termo dinamici. Per tanto, prima di sfruttare questo tipo di risorsa, deve essere effettuato un approfondito studio ingegneristico, in cui la simulazione numerica è una parte di fondamentale importanza ai fini di una migliore gestione. Le tipologie di sistema geotermico possono essere distinte come segue:

Sistemi geotermici idrotermali.

In questo tipo sistema l'acquifero è a contatto con una fonte di calore e la sua fonte di ricarica è meteorica. È presente una struttura geologica impermeabile che funge da copertura dell'acquifero impedendo ai fluidi caldi di risalire in superficie e mantenendoli in pressione. Possono comunque verificarsi delle manifestazioni superficiali; questi sistemi sono i più sfruttati per la produzione di energia sia termica che elettrica.

Sistemi geotermici in rocce secche e calde.

Questa tipologia è caratterizzata da porzioni di sottosuolo in cui le temperature sono superiori a quella dovuta ad un flusso geotermico normale, ma per via di bassi valori di permeabilità vi è assenza di acqua. Mediante una circolazione artificiale d'acqua, garantita da una fratturazione delle rocce in situ, è possibile sfruttare il flusso termico attraverso il riscaldamento dell'acqua iniettata o con la vaporizzazione della stessa; in questo caso si attiverebbe un sistema idrotermale artificiale.

Sistemi geotermici magmatici.

Sono sistemi artificiali che sfruttano il calore diretto del magma per il riscaldamento del fluido di lavoro.

Sistemi geotermici geo-pressurizzati.

Questi sistemi sono caratterizzati da acqua che si trova ad una pressione superiore rispetto alla pressione idrostatica. La temperatura dell'acqua varia fra i 120°C e i 170°C e la coltivazione di questi bacini garantisce lo sfruttamento dell'energia termica dell'acqua.

1.3.1 Geotermia a bassa, media ed alta entalpia

L'entalpia [3] è una proprietà estensiva, funzione di stato di un sistema termodinamico ed esprime la quantità di energia che un sistema può scambiare con l'ambiente. Viene definita dalla seguente relazione:

$$H=U + p V$$

Con:

H = entalpia (J),

U = energia interna (J)

p = pressione (Pa)

V=volume (m³)

L'entalpia di un fluido, che è contenuto e che circola nel serbatoio geotermico, esprime il valore energetico del fluido e del serbatoio. I bacini geotermici vengono quindi classificati anche in base a questa grandezza e si distinguono pertanto i sistemi a bassa, media o ad alta entalpia. Non esiste una classificazione standardizzata, ad ogni modo è comunemente accettato che i sistemi ad alta entalpia hanno temperature maggiori di 150 °C e i fluidi estratti dal sottosuolo sono utilizzati direttamente per la produzione di energia elettrica in impianti di elevata potenza. I sistemi a bassa entalpia hanno temperature minori di 90 °C e i fluidi vengono usati per la climatizzazione industriale o domestica. Mentre la media entalpia si colloca fra questi valori di temperatura e viene utilizzata sia per la produzione di energia elettrica in piccoli impianti sia per il teleriscaldamento.

2 Concetti di simulazione

Prima di affrontare il tema specifico del simulatore Waiwera è bene chiarire alcune nozioni fondamentali, necessarie alla comprensione dell'argomento "simulazione". In una simulazione numerica di un bacino geotermico vengono utilizzate terminologie specifiche quali simulazione, sistema, modello, processo, modello numerico, ecc...; il presente capitolo vuole quindi riportare le definizioni più importanti ai fini di una corretta comprensione. Si è fatto riferimento al materiale didattico fornito dal professor V. Bortolotti nel corso di "Modelli per la Geotermia e fluidi nel sottosuolo" (2019) [4].

2.1 Sistemi e modelli

Sistema: è un insieme di parti, coordinate in un unico "corpo", al fine di raggiungere uno scopo. Tuttavia, un sistema non è un semplice insieme dei suoi componenti, è richiesto infatti che le parti che lo compongono interagiscano fra di loro [S3]. Un esempio di interazione può essere lo scambio di informazioni piuttosto che lo scambio di energia. I sistemi possono assumere variegate forme, quali una squadra sportiva, un edificio, oppure, come in questo caso, un bacino geotermico.

Processo: è definito come l'insieme di interazioni, trasformazioni e trasmissioni di energia, informazioni e materiali finalizzati al raggiungimento di un determinato scopo.

Dalla definizione di processo è possibile dare una seconda definizione di sistema, ovvero, l'insieme di tutti i mezzi materiali attraverso i quali si realizza il processo. Ogni sistema viene definito da proprietà intrinseche che ne contraddistinguono il comportamento. Queste proprietà possono essere permanenti o condizionate e vengono espresse rispettivamente da due grandezze:

Parametri (grandezze costanti nel tempo che caratterizzano i sistemi tempo-invarianti) e **Variabili** (cambiano nel tempo e caratterizzano i sistemi tempo-varianti).

Una variabile rappresenta infatti la capacità di un sistema di evolvere nel tempo, mediante interazioni con l'ambiente esterno e con altri elementi del sistema. Ad esempio, un parametro di un sistema acquifero può essere la comprimibilità dell'acqua, mentre una sua variabile può essere la pressione dei fluidi contenuti al suo interno. Un sistema può essere definito schematicamente come un'entità, al cui interno avviene un processo, e che ha relazioni con l'ambiente esterno tramite ingressi e uscite:

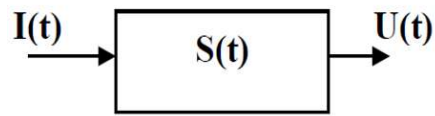


Figura 3 Sistema ingresso-uscita. Immagine estrapolata dalle dispense del corso MGFS (2019) [4].

L'uscita, anche detta risposta del sistema, dipenderà da ciò che il sistema “subisce” dal mondo esterno (azioni denominate anche ingressi o perturbazioni o sollecitazioni) e da ciò che avviene all'interno del sistema stesso (il suo stato interno). L'uscita del sistema $U(t)$ può essere rappresentata da un punto di vista matematico come una funzione che dipende da una funzione che descrive lo stato interno $S(t)$ e da una funzione che descrive le sollecitazioni $I(t)$:

$$U(t) = f [S(t), I(t)]$$

2.1.1 Tipi di sistema

I sistemi hanno diverse tipologie di classificazione, ad esempio, se si volesse classificare in base agli scambi di materia ed energia un sistema sarà detto aperto quando scambia sia materia che energia con l'ambiente esterno, isolato, se non è permesso scambio di energia e materia o chiuso se non è consentito scambio di materia (in generale i sistemi geotermici sono sistemi aperti). Un'altra classificazione può essere fatta in base alla variazione temporale dei parametri; in particolare, un sistema si dice invariante quando la stessa sollecitazione, ripetuta in momenti diversi, produce la stessa uscita mentre un sistema sarà variante quando i parametri variano nel tempo. Tuttavia, è bene precisare che a determinare l'invarianza di un sistema è la velocità di variazione dei parametri confrontata con la velocità d'evoluzione del sistema. I sistemi possono inoltre essere statici o dinamici (a seconda della variazione temporale delle variabili), a variabili continue o discrete (in base ai valori assunti da quest'ultime), deterministici o stocastici/probabilistici (in base alle relazioni tra variabili dipendenti e indipendenti).

2.1.2 Modello del Sistema

In generale un sistema reale richiede una complessa analisi per via dei numerosi aspetti da considerare e dell'impossibilità di effettuare valutazioni oggettive sulle proprietà dello stesso. Si può quindi adottare una rappresentazione semplificata detta *modello* e che sarà composto da tutti e solo quegli elementi del sistema originario ritenuti determinanti per la risoluzione del problema.

In generale, i modelli possono essere distinti come:

- modelli fisici: in scala o analogici;
- modelli simbolici: a simboli passivi o a simboli attivi.

In un modello fisico in scala ad ogni elemento considerato del modello originario viene fatto corrispondere un elemento che possa essere rappresentato dalla stessa grandezza fisica e viene mantenuto un rapporto di proporzionalità costante tra tutti gli elementi (es. un plastico tridimensionale di un edificio). In un modello fisico analogico si ha invece una rappresentazione delle proprietà del sistema originario attraverso grandezze fisiche differenti (es. un termometro che rappresenta la temperatura mediante la lunghezza della colonnina di mercurio).

Un modello simbolico a simboli passivi vuole rappresentare gli elementi del sistema tramite simboli non manipolabili, quali le icone, e le relazioni fra elementi viene mantenuta da una legenda (es. la mappa di un campo petrolifero, con icone che indicano la posizione dei pozzi). In un modello simbolico a simboli attivi è invece possibile una rielaborazione tramite regole definite (es. modello matematico che consente di elaborare simboli matematici tramite legami funzionali).

I modelli matematici possono essere divisi in modelli analitici e numerici: i primi permettono di ricavare soluzioni esatte, se esistenti, i secondi risolvono le equazioni che regolano il problema mediante approssimazioni, ricavando soluzioni non esatte ma sufficientemente vicine al valore esatto. L'utilizzo di modelli numerici nasce dall'impossibilità di risolvere analiticamente le equazioni nei casi più complessi (lo studio di un sistema reale come un bacino geotermico). Il termine modello numerico viene spesso associato all'insieme delle equazioni matematiche discretizzate e dei parametri necessari alla risoluzione di quest'ultime. I modelli numerici possono essere caratterizzati da una grande mole di dati (nel caso in cui il dominio spazio-temporale venga suddiviso in intervalli piccoli) che possono essere trattati in maniera automatizzata attraverso algoritmi eseguiti da computer.

2.1.2.1 Utilizzo dei Modelli

L'utilizzo dei modelli può essere sintetizzato in due aree principali: interpretazione e simulazione.

2.1.2.1.1 Interpretazione

Si prestano a questi utilizzi i modelli a simboli attivi matematici, utilizzati per ottenere informazioni aggiuntive del sistema, sulla base della risposta di quest'ultimo a seguito di una

perturbazione. Tramite tali procedimenti si vuole capire se e quanto il modello utilizzato sia adatto a riprodurre il sistema reale. Alcune operazioni eseguite a scopo interpretativo sono:

- Calibrazione o *History Matching* o *Model Calibration*;
- Analisi di sensitività o *Sensitivity Analysis*;
- Analisi di propagazione dell'errore.

La calibrazione dei parametri consente di determinare il miglior valore da attribuire all'insieme dei parametri di input di un modello tali che le variabili di output del modello riproducano più fedelmente possibile i valori misurati sul sistema reale. L'analisi di sensitività identifica le variabili di output del modello che sono maggiormente influenzate dai parametri di input. Tale funzionalità risulta importante per capire quali siano le misure da affinare (di laboratorio o in situ). L'analisi di propagazione dell'errore consente di studiare l'impatto dell'incertezza dei parametri sulle previsioni realizzate dai modelli, ad esempio, attraverso l'analisi di propagazione lineare degli errori. Quest'analisi fornisce correlazioni fra parametri [5].

2.1.2.1.2 Simulazione

L'attività di simulazione utilizza il modello per ottenere informazioni sul comportamento del sistema originario e per prevederne l'evoluzione futura, considerando diverse sollecitazioni esterne e diverse condizioni iniziali. L'attendibilità dei risultati è determinata dall'accuratezza con cui si è riprodotto il sistema originario, da caratteristiche del modello stesso e dal numero di dati a disposizione. Non è comunque possibile ottenere risultati con precisioni che siano superiori a quelle dei dati in ingresso; questo a causa della propagazione dell'incertezza e delle numerose semplificazioni adottate nello sviluppo dei modelli stessi: a questo proposito vale il detto “*garbage input, garbage output*”, (spazzatura entra, spazzatura esce) [6]. Perciò è assolutamente necessario avere cautela nell'analisi dei risultati di una simulazione, in quanto non si ha mai una certezza sull'affidabilità.

In seguito, verrà utilizzato il termine simulatore per indicare il software che esegue l'attività di simulazione su un modello numerico.

2.1.3 Generalità sulla simulazione di bacini geotermici

La simulazione mediante modelli numerici del sottosuolo si è evoluta molto velocemente negli ultimi anni, grazie a tecniche numeriche in costante evoluzione ed alla disponibilità di computer

con capacità di calcolo in continua crescita. È buona pratica procedere metodologicamente nella realizzazione dei modelli, consentendo di aumentare l'affidabilità dei risultati delle simulazioni. Infatti, stabilire nella fase iniziale gli obiettivi dell'operazione di modellazione e i possibili risultati, consente un utilizzo della simulazione sicuramente più efficace.

Uno schema tipico di progetto è il seguente [7]:

1. Definizione dell'obiettivo;
2. Sviluppo di un modello concettuale (geometrie, distribuzione spaziale delle caratteristiche petrofisiche, zonazioni, ecc....);
3. Individuazione dei parametri fluidodinamici e termo-fisici dei fluidi di strato e delle loro variazioni spaziali;
4. Individuazione delle equazioni che governano i processi e di un software di implementazione (simulatore);
5. Creazione della griglia (“*mesh*”) appropriata;
6. Inizializzazione del modello (assegnare ad ogni blocco i valori iniziali dei parametri termofisici e petrofisici, le variabili fluidodinamiche e termodinamiche delle rocce dei fluidi presenti). L'insieme delle fasi 5 e 6 rappresenta il “*pre-processing*” (creazione dei dati necessari a processare il modello numerico e che verranno inseriti nel file di input del simulatore);
7. Calibrazione del modello;
8. Analisi di sensitività;
9. Previsione del comportamento del sistema in funzione di programmi futuri di coltivazione. L'analisi dei risultati rappresenta la fase di “*post-processing*”;
10. Correzioni e modifiche del modello in base ai risultati ottenuti dalle simulazioni.

2.1.4 Problematiche correlate all'impiego di modelli numerici

Le simulazioni di sistemi geotermici necessitano di grandi quantità di dati in ingresso, spesso costosi e difficili da ottenere, e molte volte si ricorre ad ipotesi semplificative (o vengono adottati valori ricavati dalla letteratura), che possono non rappresentare correttamente le reali caratteristiche del bacino. Si pensi ai valori di parametri misurati in situ che sono intesi come valori medi (o complessivi) dell'area coinvolta [7]; o ancora ai valori ottenuti in laboratorio che sono invece rappresentativi solo della zona di campionamento, e che quindi avrà nella maggior parte dei casi caratteristiche molto diverse dalle zone circostanti. Ad ogni modo, modellare un sistema con pochi dati a disposizione risulta ugualmente un'operazione utile, in quanto successivamente sarà possibile identificare aree in cui sono necessari un maggior numero di informazioni [8].

Avere risultati in tempi ridotti richiede l'uso di potenti computer, quindi costosi, e nonostante l'adozione dei calcolatori più prestanti offerti dal mercato non si può aumentare a dismisura il dettaglio delle griglie, così come la precisione di calcolo, e questo a discapito della correttezza dei risultati finali. L'impiego di modelli numerici, originariamente in campo petrolifero, ha evidenziato la necessità di individuare il più probabile modello geologico del giacimento e la sua struttura interna e di suddividere, in funzione delle competenze individuali, i compiti all'interno della squadra di simulazione[6]. La procedura di calibrazione del modello, ottenuta mediante la soluzione del problema inverso, permette di ottenere un modello validato, tuttavia, anche se la fase di validazione del modello è obbligatoria, questa non garantisce che il modello validato sia capace di prevedere in modo corretto il comportamento futuro del giacimento [6]. In aggiunta, non si può mai provare che i risultati ottenuti da una simulazione siano corretti [8]. L'analisi di sensibilità consente di determinare l'influenza dell'incertezza dei parametri sulla risposta finale della simulazione e permette di determinare i valori dei parametri più probabili; uno studio di sensibilità risulta essere la maniera più realistica di impiego dei modelli numerici ai fini di prevedere il comportamento di un giacimento [6].

3 Simulatori per bacini geotermici

Questa introduzione ai simulatori per sistemi geotermici prende spunto da una pubblicazione del 2020 [9] e vuole dare un quadro generale sui simulatori più noti in tale ambito.

Da oltre 40 anni, i modelli numerici sono utilizzati per simulare i giacimenti geotermici al fine di gestirli in modo sostenibile. Le applicazioni geotermiche pongono requisiti speciali al software di simulazione, richiedendo in particolare metodi robusti di modellazione di fluidi multifase ad alta temperatura che bollono e si condensano.

Il software per la modellazione geotermica è stato dominato dai pochi codici che hanno questo tipo di capacità. O'Sullivan et al. (2001) [18] hanno condotto un'indagine internazionale sugli studi di simulazione dei giacimenti geotermici intrapresi dal 1990, mostrando che la maggior parte dei modelli geotermici è stato eseguito utilizzando il simulatore TOUGH2 [10] mentre pochi avevano utilizzato altri codici come TETRAD [19] e STAR [20]. Altri simulatori di flusso sotterraneo non isotermico come SHEMAT [21], FEFLOW [22], Open-GeoSys [23], COMSOL [24] e PFLOTTRAN [25] sono stati utilizzati per applicazioni geotermiche, ma la maggior parte non ha la capacità di simulare l'ebollizione e la condensazione necessarie per i modelli di serbatoi geotermici ad alta temperatura. Alcuni altri codici come FEHM [26], STARS [27] e più recentemente ECLIPSE [28] offrono questo tipo di funzionalità, ma non sono ancora stati ampiamente utilizzati per la modellazione numerica in geotermia. Quindi, TOUGH2 ha mantenuto la sua posizione di standard industriale de facto per la simulazione di serbatoi geotermici [29].

L'Università di Auckland mantiene un fork del codice TOUGH2, denominato AUTOUGH2 [30], che include funzionalità aggiuntive progettate specificatamente per la modellazione di serbatoi geotermici, ad esempio, un comportamento di transizione di fase migliorato e una gamma di tipi di generatori extra per la modellazione delle numerose variazioni nel comportamento di produzione e reiniezione riscontrate nei giacimenti di vapore geotermici.

Nonostante aumenti significativi della potenza di calcolo, la necessità di modelli geotermici più grandi e complessi ha imposto al software richieste sempre più severe, in particolare in termini di efficienza computazionale. Sebbene siano stati compiuti alcuni progressi per migliorare l'efficienza di TOUGH2, ad esempio tramite risolutori lineari migliorati e termodinamica più rapida, attualmente l'unico percorso pratico per ulteriori guadagni di efficienza risiede nella parallelizzazione. Le varianti TOUGH2-MP [31] e più recenti TOUGH3 [32] del codice sono parallelizzate, ma non includono tutte le caratteristiche geotermiche specifiche di AUTOUGH2.

Per i modelli molto grandi ($\geq 10^6$ celle), sono auspicabili anche alternative più efficienti ai file di testo formattati utilizzati per l'input e l'output di TOUGH2 e TOUGH3.

I modelli di giacimenti geotermici vengono in genere prima eseguiti per ottenere quella che si chiama “soluzione allo stato stazionario” (o stato naturale) che si avvicina allo stato pre-sfruttamento. L'elaborazione di una soluzione geotermica allo stato stazionario spesso richiede una grande quantità di tempo di calcolo e potrebbe non riuscire se il simulatore incontra problemi di convergenza. AUTOUGH2 ha introdotto alcune misure per migliorare il comportamento di convergenza ma ha comunque lasciato spazio per ulteriori miglioramenti [33]. Un codice grande e complesso come un simulatore di flusso può trarre notevoli vantaggi dai moderni principi di progettazione orientati agli oggetti, risultando in un codice modulare chiaramente strutturato e più facile da estendere e mantenere, con opportunità di riutilizzo del codice tramite l'ereditarietà. TOUGH2 ha le sue radici nei precedenti codici TOUGH e MULKOM [34], con una parte sostanziale del suo codice Fortran e del design del software che è rimasto poco cambiato dagli anni '80. TOUGH3 ha introdotto alcune funzionalità del Fortran 95 come i moduli e l'allocazione dinamica della memoria, ma ancora una volta ha lasciato in gran parte invariato il design generale del software. Fino a poco tempo fa lo sviluppo di codice orientato agli oggetti significava abbandonare il linguaggio Fortran, ma con i compilatori che ora supportano gli standard Fortran 2003 e 2008 è possibile mantenere i tradizionali vantaggi numerici di Fortran in un contesto orientato agli oggetti. Come risultato di queste considerazioni, il gruppo dell'Università di Auckland ha sviluppato un nuovo simulatore di flusso geotermico, basandosi sull'esperienza nello sviluppo e nell'utilizzo dei codici TOUGH2 e AUTOUGH2, ma con una base di codice Fortran 2003 parallelizzata e orientata agli oggetti completamente nuova. Il nuovo simulatore, chiamato "Waiwera" (che significa "acqua calda" in lingua Maori), è il primo ad offrire le funzionalità e le prestazioni richieste per modelli di serbatoi geotermici grandi e complessi e ad essere rilasciato open source e gratuitamente. Poiché Waiwera si basa sull'esperienza maturata dagli autori su TOUGH2 e poiché iTOUGH2, che utilizza TOUGH2 come routine, è utilizzato per verificare le prestazioni di Waiwera, diventa importante introdurre anche una breve descrizione di TOUGH2.

3.1 TOUGH2 versione 2.0

TOUGH2 [10] (*Transport Of Unsaturated Groundwater and Heat*) è un simulatore numerico che permette di studiare flussi di fluidi non isotermi, multicomponente e multifase in mezzi porosi continui o fratturati in tutte e tre le dimensioni. TOUGH2 è stato sviluppato a partire

dagli anni '80 presso il *Lawrence Berkeley National Laboratory* (LBNL, Berkeley, California, USA). È scritto in linguaggio Fortran e può essere quindi compilato e utilizzato sotto diversi sistemi operativi. Oggi trova molti impieghi a livello universitario, industriale ma anche governativo, in svariati ambiti applicativi come nello studio e nello stoccaggio dei rifiuti radioattivi, nell'ingegneria degli acquiferi, nel risanamento del sottosuolo e nello stoccaggio della CO₂. Ma il suo principale impiego ad oggi è nell'ambito dell'ingegneria dei reservoir geotermici. Il simulatore adotta un approccio modulare mediante moduli intercambiabili detti EOS (*Equation Of State*) per descrivere i diversi fenomeni. Queste, identificano l'eventuale comparsa o scomparsa delle fasi nei blocchi della griglia, definiscono le componenti e le proprietà dipendenti dalle condizioni termodinamiche (es. viscosità, entalpia, densità) (*Figura 4*). Ogni modulo EOS ha uno determinato campo di applicazione e viene pertanto scelto in base al problema da simulare (*Tabella 1*). Il simulatore può trattare un fluido in entrambe le fasi (liquida e gassosa) utilizzando la legge di generalizzata Darcy in cui, mediante le funzioni di permeabilità relativa, viene valutata l'interferenza delle fasi durante il flusso nel mezzo poroso. La discretizzazione spaziale, per tutti i codici della famiglia TOUGH, avviene nella forma integrale delle equazioni di conservazione di massa e energia, utilizzando il metodo IFDM (*Integral Finite Difference Method*) che permette una discretizzazione del dominio anche con griglie non strutturate. L'approccio che utilizza TOUGH2 è quindi di tipo *cell-based*, con la creazione di una griglia composta da celle, alle quali vengono associate le variabili fluido-termodinamiche e le velocità dei fluidi.

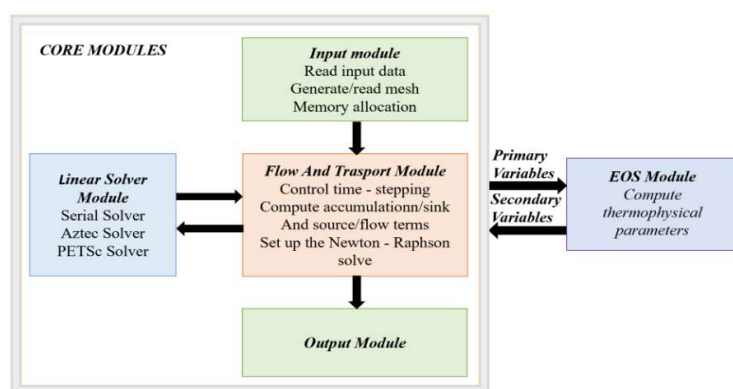


Figura 4 Schema di funzionamento di Tough2 versione 2.0. Immagine modificata dal manuale di Tough2 [10].

Tabella 1 Moduli EOS per TOUGH2 presenti nel sito <https://tough.lbl.gov/licensing-download/itough2-price-sheet/>. Immagine estrapolata da Ambrogetti, L. 2020 [17].

| Moduli | Descrizione |
|-------------------|--|
| <i>CORE</i> | Framework di ottimizzazione della simulazione per modellazione predittiva, modellazione inversa, analisi di sensibilità, analisi del valore dei dati e analisi della propagazione dell'incertezza per il simulatore di flusso multifase non isotermico TOUGH e codici esterni. |
| <i>EOS1</i> | Acqua, acqua con tracciante, calore |
| <i>EOS1SC</i> | Acqua, acqua con tracciante, calore in condizioni sub- e supercritiche |
| <i>EOS2</i> | Acqua, CO2, calore |
| <i>EOS3</i> | Acqua, aria, calore |
| <i>EOS4</i> | Acqua, aria, calore, con abbassamento della pressione del vapore |
| <i>EOS5</i> | Acqua, idrogeno, calore |
| <i>EOS7</i> | Acqua, salamoia, aria, calore |
| <i>EOS7R</i> | Acqua, salamoia, aria, radionuclide1, radionuclide2, calore |
| <i>EOS7C</i> | Acqua, salamoia, CO2 o N2, CH4, tracciante, calore |
| <i>EOS7CA</i> | Acqua, salamoia, NCG (CO2, N2 o CH4), gas tracciante, aria, calore |
| <i>EOS8</i> | Acqua, aria, olio, calore |
| <i>EOS9</i> | Acqua (equazione di Richards) |
| <i>EOSGEL</i> | Acqua, aria, liquido gelificante miscibile, calore |
| <i>ECO2N</i> | Acqua, brine, CO2, calore |
| <i>ECO2M</i> | Acqua, brine, CO2, calore (multifase) |
| <i>EWASG</i> | Acqua, aria, NCG, calore |
| <i>T2VOC</i> | Acqua, aria, COV, calore |
| <i>TMVOC</i> | Acqua, COV, NCG, calore |
| <i>GSLIB</i> | Crea internamente campi di proprietà spazialmente correlati basati su GSLIB |
| <i>Hysteresis</i> | Funzioni di Permeabilità relativa isteretica e di pressione capillare |
| <i>PEST</i> | Interfaccia PEST per analisi iTOUGH2 indipendenti dal modello |
| <i>PVM</i> | Esecuzione parallela su cluster Unix / Linux (richiede PVM) |
| <i>PARALLEL</i> | Esecuzione parallela su macchine multicore |

3.2 T2Well

Per una corretta e completa simulazione di un reservoir anche il pozzo diventa parte integrante delle performance del reservoir. Quindi è utile avere simulatori che includono il comportamento del pozzo. A questo proposito, T2Well [11] è un simulatore numerico accoppiato pozzo-serbatoio per flussi multifase, multicomponente e non isoterma nato come estensione del simulatore TOUGH2 ai fini di calcolare in modo completo l'iniezione o la produzione di fluidi in giacimento ma considerando gli effetti di pozzo con l'introduzione nella griglia numerica di un apposito sottodominio dedicato al pozzo. Le equazioni vengono risolte numericamente con il metodo IFDM. T2Well viene accoppiato con un'opportuna EOS, che gestisce le proprietà termofisiche dei fluidi presenti e può essere utilizzato, per esempio, nelle simulazioni di produzione di fluidi ad alta entalpia.

Questa estensione risulta sicuramente molto utile per altri simulatori come Waiwera. Una futura attività di sicuro interesse e utilità sarebbe estendere le funzionalità di Waiwera anche al flusso in pozzo.

4 Waiwera

Waiwera [S4] è un recente simulatore numerico parallelo di flusso geotermico ad alta entalpia, con licenza gratuita e *open-source* la cui prima versione è stata rilasciata nel 2019 dopo cinque anni di sviluppo grazie a una collaborazione tra l'Università di Auckland e il *GNS Science* [S5]. L'ultima versione è *Waiwera 1.2.1* rilasciata il 20 luglio 2020 [12]. Questo simulatore trova al suo interno diverse innovazioni per migliorare il comportamento nelle transizioni di fase e la convergenza di modelli in stato stazionario. Come detto precedentemente il software è caratterizzato da un'esecuzione parallela che può avvenire su computer e cluster a memoria condivisa o distribuita inoltre, fa un ampio uso della libreria di calcolo scientifico open-source *PETSc (Portable Extensible Toolkit for Scientific Computation)* [13][S6] utilizzata per la soluzione delle equazioni differenziali alle derivate parziali. Il codice, scritto in *Fortran 2003* [S7], è strutturato e si basa su un paradigma di programmazione orientato ad oggetti [S8]. Vengono utilizzati formati di file standard sia per gli inputs (JSON [S9]) che per gli outputs (HDF5 [S10], YAML [S11]).

4.1 Caratteristiche di funzionamento

In questo sotto-capitolo sono riportate nel dettaglio le caratteristiche di funzionamento di *Waiwera* seguendo quanto riportato nel suo manuale.

4.1.1 Equazioni di conservazione della massa e dell'energia

Waiwera funziona risolvendo equazioni di conservazione dipendenti dal tempo per la massa (e solitamente l'energia) nel dominio di simulazione. In generale possono essere presenti diversi 'componenti' di massa nel problema, ad esempio acqua e CO₂. Viene risolta un'equazione di conservazione della massa separata per ogni componente di massa presente. Per problemi non isotermici, viene risolta anche un'ulteriore equazione di conservazione dell'energia, contemporaneamente alle equazioni di conservazione della massa. Le equazioni di conservazione della massa e dell'energia su un volume arbitrario V sono:

$$\frac{d}{dt} \int_V M^\kappa dV = \int_{\partial V} \mathbf{F}^\kappa \cdot \hat{n} dA + \int_V q^\kappa dV \quad (1)$$

Dove $\kappa=1, \dots, N+1$ (con N il numero di componenti di massa, ad esempio 1 per l'acqua pura). Il componente $N+1$ rappresenta la componente energetica. Per ogni componente κ , M^κ è la massa o densità di energia in V , F^κ è il flusso di quel componente attraverso il confine ∂V e q^κ rappresenta i termini source o sink (per unità di volume).

4.1.2 Discretizzazione a volumi finiti

Il dominio di flusso è discretizzato utilizzando una mesh a volume finito. Nell' i -esima cella con volume V_i , per ogni componente si definiscono la massa media delle celle o le densità di energia M_i^κ per ogni componente e termini di origine mediati dalle celle q_i^κ come:

$$M_i^\kappa = \frac{1}{V_i} \int_{V_i} M^\kappa dV \quad (2)$$

$$q_i^\kappa = \frac{1}{V_i} \int_{V_i} q^\kappa dV \quad (3)$$

Possiamo anche rappresentare l'integrale di flusso della (1) come:

$$\int_{\partial V_i} \mathbf{F}^\kappa \cdot \hat{n} dA = \sum_j A_{ij} F_{ij}^\kappa \quad (4)$$

Dove A_{ij} è l'area della faccia che collega le cellule i e j , e F_{ij}^κ è la componente normale del flusso che lo attraversa. Quindi le equazioni di conservazione discretizzate per cella V_i possono essere scritte come:

$$\frac{d}{dt} M_i^\kappa = \frac{1}{V_i} \sum_j A_{ij} F_{ij}^\kappa + q_i^\kappa \quad (5)$$

4.1.3 Variabili primarie

Waiwera risolve l'equazione (5) per lo stato termodinamico in ogni cella nella mesh di simulazione. Lo stato termodinamico in ogni cella è rappresentato da un piccolo insieme di variabili primarie, una per ogni equazione di conservazione. Le variabili primarie dipendono dall'equazione di stato utilizzata e dalle condizioni di fase nella cella. Ad esempio, per le simulazioni di acqua pura non isoterma, c'è solo una componente di massa (acqua) e una componente energetica; quindi, sono necessarie due variabili primarie per descrivere lo stato

termodinamico. Per condizioni monofase possiamo usare pressione e temperatura come variabili primarie. Tutte le altre proprietà del fluido (ad es. densità, viscosità, ecc.) possono essere calcolate dalle variabili primarie. Tuttavia, per le condizioni a due fasi, la pressione e la temperatura non sono indipendenti, poiché sono correlate tramite la curva di saturazione. Pertanto, non possono essere utilizzate come variabili primarie per descrivere lo stato termodinamico. Per le condizioni a due fasi, Waiwera utilizza la pressione e la saturazione del vapore come variabili primarie. Poiché la scelta delle variabili primarie dipende dalle condizioni di fase, quando il fluido in una cella cambia fase, le variabili primarie devono essere modificate. In pratica, Waiwera non risolve per le variabili primarie grezze in ogni cella, ma per le variabili primarie non dimensionalizzate. Ciò migliora il comportamento numerico, soprattutto perché le variabili primarie grezze in genere hanno grandezze molto diverse l'una dall'altra. Le variabili non dimensionalizzate sono correlate alle corrispondenti variabili primarie grezze nella maggior parte dei casi tramite un semplice ridimensionamento di una costante fissa. Questi parametri di ridimensionamento hanno valori predefiniti ma possono anche essere personalizzati per problemi particolari.

4.1.4 Evoluzione nel tempo

Le equazioni di conservazione discretizzate (5) sono della forma:

$$\frac{d}{dt}\mathbf{L}(t, \mathbf{Y}) = \mathbf{R}(t, \mathbf{Y}) \quad (6)$$

Dove t è il tempo e \mathbf{Y} è il vettore delle variabili primarie per tutte le celle nella mesh di simulazione. \mathbf{L} rappresenta i bilanci di massa ed energia medi delle celle, in funzione del tempo e delle variabili termodinamiche primarie. Allo stesso modo, \mathbf{R} rappresenta gli afflussi (per unità di volume) nelle celle dai flussi attraverso le facce delle celle, insieme a sorgenti e pozzi all'interno della cella.

Risolvendo l'insieme delle equazioni differenziali ordinarie (6) rispetto al tempo, possiamo calcolare l'evoluzione temporale di \mathbf{Y} , lo stato termodinamico dell'intero dominio di simulazione discretizzato. Quando si risolvono le equazioni di conservazione (5), \mathbf{L} e \mathbf{R} sono funzioni complicate e non lineari delle variabili primarie \mathbf{Y} . Quindi l'equazione (6) deve essere risolta numericamente, calcolando la soluzione \mathbf{Y} a tempi discreti. Waiwera contiene un modulo per la soluzione numerica di equazioni differenziali ordinarie della forma (6), utilizzando

diversi metodi numerici. Il più semplice di questi è il metodo di "Eulero all'indietro", che discretizza l'equazione (6) come segue:

$$\frac{\mathbf{L}(t^{n+1}, \mathbf{Y}^{n+1}) - \mathbf{L}(t^n, \mathbf{Y}^n)}{\Delta t} = \mathbf{R}(t^{n+1}, \mathbf{Y}^{n+1}) \quad (7)$$

Dove t^n è il n^{th} tempo discretizzato, e Δt è il passo temporale, in modo che $t^{n+1} = t^n + \Delta t$. Per il metodo di Eulero all'indietro, ad ogni passo di tempo dobbiamo risolvere l'equazione (7) per la nuova soluzione sconosciuta \mathbf{Y}^{n+1} .

4.1.5 Valutazioni delle funzioni

Waiwera deve valutare le funzioni L e R per ogni dato insieme di variabili primarie (e tempo). La funzione L, che rappresenta le densità di massa ed energia M_i^κ nelle cellule, è relativamente semplice da valutare, sommando i contributi delle diverse fasi. Considerando una cella particolare:

$$M_i^\kappa = \begin{cases} \phi_i \sum_p S_{i,p} \rho_{i,p} X_{i,p}^\kappa & \kappa \leq N \\ (1 - \phi_i) \rho_i^r c_i T_i + \phi_i \sum_p S_{i,p} \rho_{i,p} u_{i,p} & \kappa = N + 1 \end{cases} \quad (8)$$

dove per ogni cella i , ϕ_i è la porosità della roccia, c_i è il calore specifico della roccia e T_i è la temperatura. Per ogni fase p , $S_{i,p}$ è la saturazione di fase, $\rho_{i,p}$ è la densità di fase, $X_{i,p}^\kappa$ è la frazione di massa di fase per il componente κ e $u_{i,p}$ è la densità di energia interna di fase. La funzione R, che rappresenta i flussi nelle celle, ha contributi dai termini source e sink (che sono facilmente valutabili) e dai flussi attraverso le facce. Quest'ultimo contributo è calcolato sommando i flussi facciali del componente in ciascuna fase:

$$F_{ij}^\kappa = \sum_p F_{ij,p}^\kappa \quad (9)$$

dove i flussi di fase sono dati da:

$$F_{ij,p}^\kappa = \begin{cases} -\frac{k \cdot k_p^r}{\nu_p} X_p^\kappa \left(\frac{\partial P}{\partial n} - \rho_{ij,p} \mathbf{g} \cdot \hat{n} \right) & \kappa \leq N \\ -K \frac{\partial T}{\partial n} + \sum_{m=1}^N \sum_p h_p^m F_{ij,p}^m & \kappa = N + 1 \end{cases} \quad (10)$$

Qui k è la permeabilità effettiva della roccia normale alla faccia, K è la conduttività termica della roccia, P è la pressione, g è il vettore gravità e \hat{n} è il vettore unitario normale alla faccia. Per fase p , k_p^r è la permeabilità relativa di fase, ν_p è la viscosità cinematica di fase, h_p^m è l'entalpia di fase per il componente m e $\rho_{ij,p}$ è la densità di fase effettiva sulla faccia. I normali gradienti di pressione e temperatura sono valutati mediante differenziazione finita lungo la fase, cioè prendendo la differenza tra i valori nelle celle su entrambi i lati della faccia e dividendo per la distanza tra i centri delle celle. Questa “approssimazione del flusso a due punti” si basa sul presupposto che la mesh soddisfi il “criterio di ortogonalità” (par. 4.2.2.2), cioè che la linea che unisce i centri delle celle sia ortogonale alla faccia. Quando si valutano i flussi di fase utilizzando l'equazione (10), le quantità di flusso k_p^r , ρ_p , ν_p , X_p^k e h_p^m sono “*upstream weighted*”, cioè i loro valori sono presi dalla cella a monte della faccia. Questo è necessario per la stabilità numerica. La permeabilità della roccia k e la conducibilità termica K sulla faccia vengono valutate utilizzando la ponderazione armonica dei valori nelle celle su entrambi i lati della faccia. Per il termine gravità, Waiwera calcola la densità di fase effettiva sulla faccia come media ponderata per la saturazione delle densità di fase nelle celle su entrambi i lati:

$$\rho_{ij,p} = \frac{S_{i,p}\rho_{i,p} + S_{j,p}\rho_{j,p}}{S_{i,p} + S_{j,p}} \quad (11)$$

Questa formulazione può essere derivata considerando un equilibrio di forza sulle due celle e garantisce una variazione graduale della densità di fase effettiva sulla faccia quando le celle adiacenti cambiano fase. Se entrambe le celle adiacenti hanno la stessa saturazione (ad esempio in condizioni monofase), questa media pesata si riduce a una semplice media aritmetica di $\rho_{i,p}$ e $\rho_{j,p}$.

4.1.6 Soluzione di equazioni ad ogni passo temporale

Indipendentemente dal metodo di time step utilizzato, le equazioni discretizzate da risolvere ad ogni time step sono non lineari. Scritte in una forma generica vale la seguente:

$$\mathbf{f}(\mathbf{Y}) = \mathbf{0} \quad (12)$$

quindi ad ogni passo dobbiamo risolverlo per la soluzione $\mathbf{Y}=\mathbf{Y}^{n+1}$. A causa della non linearità, deve essere risolto numericamente utilizzando una tecnica di soluzione non lineare come il metodo di Newton. Il metodo di Newton è un metodo iterativo che parte da una stima iniziale

della soluzione (qui assunta come $Y=Y^n$) e adegua la soluzione provvisoria Y ad ogni iterazione finché l'equazione (12) non è soddisfatta, entro una tolleranza prestabilita. Ad ogni iterazione, il metodo di Newton aggiunge un aggiornamento ΔY alla soluzione provvisoria Y secondo:

$$\mathbf{J}\Delta\mathbf{Y} = -\mathbf{f} \quad (13)$$

Dove J è la matrice jacobiana della funzione f , ovvero la matrice delle derivate parziali di f riguardo a Y . Ad ogni iterazione, l'equazione di aggiornamento di Newton (13) rappresenta un ampio e sparso sistema di equazioni lineari da risolvere numericamente. I metodi iterativi del "sottospazio di Krylov" (ad es. metodi a gradiente coniugato) sono appropriati per risolvere tali sistemi. Per le tipiche simulazioni di grandi problemi di flusso, la maggior parte del tempo di calcolo viene impiegato nella soluzione delle equazioni lineari. Waiwera utilizza il solutore non lineare "SNES" fornito dalla libreria PETSc per risolvere l'equazione (12) ad ogni passaggio temporale. Per problemi in cui la matrice jacobiana J è difficile da calcolare, il risolutore SNES offre un'opzione per calcolarlo automaticamente utilizzando le differenze finite. In questo caso le derivate parziali jacobiane vengono valutate approssimativamente aggiungendo piccoli incrementi sul vettore della variabile primaria y e rivalutando la funzione f . Waiwera utilizza questo approccio per calcolare la matrice jacobiana. Il solutore SNES a sua volta utilizza la suite di solutori lineari "KSP", fornita anche da PETSc, per risolvere il sistema lineare (13) ad ogni iterazione di Newton.

4.2 Struttura del file di Input

Diversamente da TOUGH2, Waiwera fa uso per i suoi file di input output di diverse tipologie di formato, con l'obiettivo di rendere il più efficiente possibile le operazioni di pre- e post-processing.

4.2.1 Formato JSON

Waiwera gestisce il suo input principale mediante un file in formato JSON, un formato di scambio dati leggero, progettato per essere facilmente leggibile e scrivibile dall'uomo, oltre ad essere facile da generare e analizzare su PC. Infatti, un file in formato JSON è solo un file di testo, quindi, può essere letto e modificato utilizzando anche un semplice editor di testo. Molti editor di testo (soprattutto quelli progettati per i programmatori) hanno modalità speciali per

aiutare l'utente a preparare e leggere i file JSON. Questi evidenzieranno caratteri JSON speciali e parole chiave in diversi colori, per renderli più facili da leggere. La sintassi JSON fa molto uso di parentesi e parentesi graffe, quindi, un'altra utile funzione dell'editor di testo è la corrispondenza delle parentesi. Ciò consente all'utente di identificare coppie di parentesi corrispondenti all'interno del file, ad esempio per trovare la parentesi di chiusura che corrisponde a una parentesi di apertura e verificare che la parentesi di chiusura non sia mancante. I file JSON possono essere strutturati in modo gerarchico, in modo che le strutture di dati più grandi nel file possano contenere quelle più piccole, che a loro volta possono contenere le proprie strutture di dati più piccole e così via. I file JSON di input di Waiwera utilizzano questa struttura gerarchica per raggruppare i dati correlati. JSON consente una certa flessibilità nel modo in cui i dati vengono specificati. Ad esempio, i dati possono talvolta essere specificati in modi diversi, a seconda del problema. Inoltre, potrebbe non essere necessario che alcuni dati siano presenti e possono essere forniti valori predefiniti se sono assenti. Le strutture dati rappresentabili in JSON sono definite in dettaglio sul sito JSON . Non ce ne sono molti, ma possono essere combinati insieme per creare complesse strutture di dati composite. Le due principali strutture di dati JSON sono l'array e l'oggetto. Un array è una raccolta ordinata di valori, che inizia e termina con parentesi quadre e con i valori delimitati da virgole. I valori nell'array possono essere stringhe di caratteri (racchiuse tra virgolette), numeri (interi o in virgola mobile), valori booleani (`true` o `false`) o valori nulli (`null`). Inoltre (e soprattutto), i valori possono essere a loro volta array o oggetti. I valori in un array non devono essere tutti dello stesso tipo. Un oggetto è una raccolta non ordinata di valori denominati, ovvero coppie di nomi (o "*key*") e i loro valori corrispondenti. L'oggetto è racchiuso tra parentesi graffe. Ogni nome (una stringa di caratteri) è seguito da due punti e quindi dal valore. Le coppie di nomi e valori sono separate da virgole. I valori in un oggetto possono essere dello stesso tipo di un array (stringhe, numeri, ecc.). Ancora una volta i valori possono essere essi stessi array o oggetti. Le posizioni esatte delle strutture dati in un file JSON non sono importanti. Le matrici e gli oggetti sono delimitati da parentesi o parentesi graffe come descritto sopra e i valori sono separati da virgole, quindi, i dati possono essere posizionati ovunque su ciascuna riga. Di solito sono disposti in modo facile da leggere per l'utente.

4.2.2 Struttura del file di input

Tutti i dati di input per una simulazione con Waiwera sono contenuti in un oggetto nel file JSON di input. I vari aspetti della simulazione sono specificati da diversi valori denominati

all'interno di quell'oggetto, come elencato di seguito. Questi valori sono di vario tipo, e molti sono oggetti stessi, con una loro struttura interna (*Tabella 2*). I paragrafi successivi riportano una descrizione più o meno dettagliata dei valori che compongono l'input di Waiwera.

Tabella 2 Struttura file JSON input Waiwera. Tabella riportata dal manuale Waiwera.

| name | type | value |
|------------------|-----------------------|---------------------------|
| "title" | string | simulation title |
| "mesh" | string object | simulation mesh |
| "rock" | object | rock properties |
| "boundaries" | array | boundary conditions |
| "source" | array | source terms |
| "initial" | object | initial conditions |
| "gravity" | number array null | gravity |
| "thermodynamics" | string | thermodynamic formulation |
| "eos" | object | equation of state |
| "time" | object | time stepping |
| "logfile" | boolean object | output log file |
| "output" | boolean object | output results file |

4.2.2.1 Titolo

Alla simulazione può essere assegnato un titolo descrittivo utilizzando il valore "title" nel file di input JSON. Questo è un semplice valore di stringa ed è facoltativo.

4.2.2.2 Mesh (o griglia)

Waiwera utilizza una discretizzazione a volume finito del dominio di simulazione per risolvere le equazioni di conservazione della massa e dell'energia al suo interno. Tale discretizzazione richiede rigorosamente solo la specificazione dei volumi delle celle della griglia, insieme alle aree delle facce della mesh e alle loro distanze dai centri delle celle su entrambi i lati, poiché queste sono le uniche quantità di mesh utilizzate nella soluzione delle equazioni. Tuttavia, Waiwera richiede che venga fornita una mesh geometrica completa per la simulazione, con la specifica delle posizioni di tutti i vertici delle celle e la definizione dell'elenco dei vertici per ciascuna cella. Questo è esattamente lo stesso tipo di definizione di mesh utilizzata, ad esempio, per i modelli agli elementi finiti. La discretizzazione del volume finito è generata internamente da Waiwera dalla mesh geometrica. In pratica è spesso più conveniente generare una mesh geometrica che una pura discretizzazione a volume finito. Una mesh geometrica è necessaria

anche per la post-elaborazione dei risultati della simulazione. La mesh di simulazione non è memorizzata nel file di input JSON principale di Waiwera. Viene invece memorizzato in un file ausiliario. Il file di input JSON specifica solo la posizione di questo file mesh ausiliario. Tutti i dati relativi alla mesh nel file JSON di input di Waiwera sono contenuti nel valore "mesh". Questo può essere un valore stringa che specifica il nome del file mesh o un oggetto, con un valore di stringa "filename" (e anche altri valori opzionali).

Formati della mesh

La gestione della mesh in Waiwera utilizza la libreria PETSc, in particolare utilizza la funzionalità mesh non strutturata DMplex di PETSc. Di conseguenza, le mesh Waiwera possono essere archiviate in uno qualsiasi dei formati di file supportati da DMplex. Attualmente questi formati includono: ExodusII [S12], GMSH [S13] e Salome MED [S14].

Sistemi di coordinate della mesh

È possibile utilizzare mesh di tipo cartesiane 2D e 3D o radiali 2D.

Ortogonalità della maglia

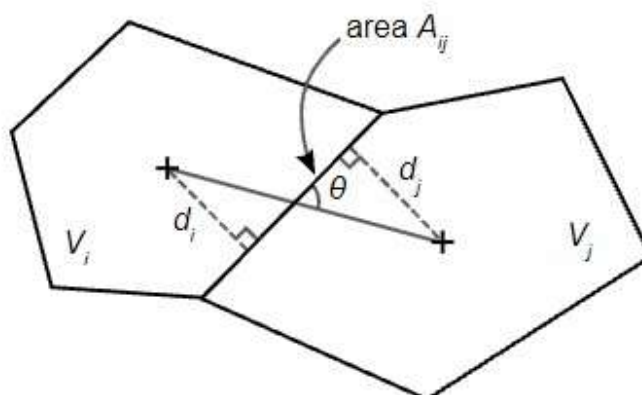


Figura 5 Ortogonalità della maglia. Immagine riportata dal manuale Waiwera.

Poiché Waiwera utilizza una discretizzazione a volume finito per risolvere le equazioni di conservazione della massa e dell'energia, con un'approssimazione di flusso a due punti per valutare i gradienti di pressione e temperatura alle facce della mesh, la mesh deve soddisfare il "criterio di ortogonalità", cioè la linea che unisce due centri di celle deve essere ortogonale per essere rivolta tra di loro. In altre parole, l'angolo θ in Figura 5 dovrebbe essere sempre 90° . Occorre pertanto prestare attenzione affinché la rete soddisfi questo criterio. Alcuni tipi più semplici di mesh lo soddisfano banalmente, ad esempio tutte le mesh rettangolari regolari o irregolari.

4.2.2.3 Termodinamica dell'acqua

Waiwera include implementazioni di due differenti “formulazioni termodinamiche” per l'acqua, ovvero insiemi di equazioni per il calcolo delle proprietà termodinamiche dell'acqua pura in funzione delle variabili primarie. Questi sono: la formulazione IFC-67 [S15] e la formulazione IAPWS-97 [S16]. Il file di input JSON Waiwera ha un valore di stringa "*thermodynamics*", che può essere impostato su "IFC67" o "IAPWS" (maiuscolo o minuscolo) per selezionare la formulazione termodinamica. Se non specificato, il valore predefinito è "*iapws*".

Regioni termodinamiche

Entrambe le formulazioni termodinamiche IFC-67 e IAPWS-97 dividono lo spazio variabile primario in regioni distinte, la maggior parte delle quali rappresentano diverse condizioni di fase. Le quattro regioni termodinamiche utilizzate dalla formulazione IAPWS-97 sono:

1. Acqua liquida;
2. Vapore;
3. Supercritico;
4. Bifase.

Questa numerazione della regione termodinamica viene utilizzata da Waiwera quando riporta le condizioni di fase (indipendentemente dalla formulazione termodinamica utilizzata), ad esempio quando si verificano cambiamenti di fase. Le regioni IAPWS-97 sono mostrate su un diagramma pressione-temperatura in *Figura 6*. Il diagramma si estende nel range di validità della formulazione IAPWS-97 (pressione ≤ 100 MPa, 0 °C \leq temperatura ≤ 800 °C).

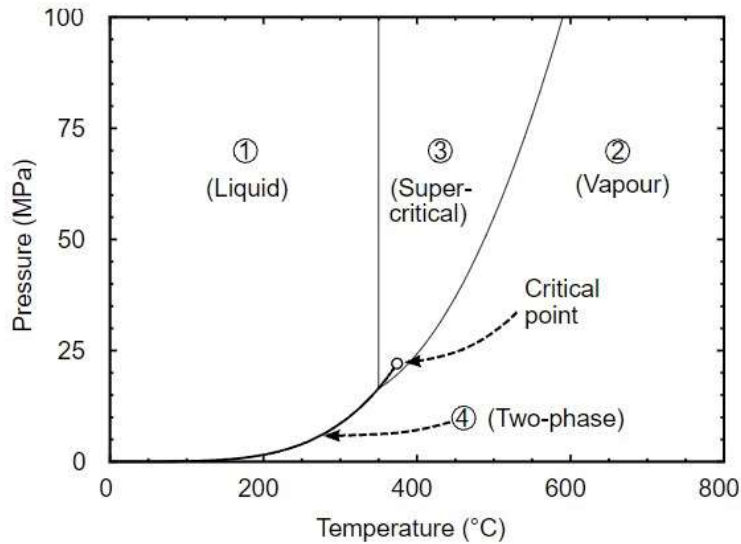


Figura 6 Regioni termodinamiche IAPWS-97. Immagine riportata dal manuale Waiwera

4.2.2.4 Eos

Il valore "eos" specifica il modulo dell'equazione di stato da utilizzare per la simulazione. Può essere una stringa contenente il nome abbreviato di EOS o un oggetto contenente un valore "name". In generale non è necessario specificare EOS come oggetto a meno che non sia necessario impostare altri parametri EOS oltre al nome. Ogni modulo EOS ha un particolare insieme di variabili termodinamiche primarie che determinano lo stato del fluido in ogni cella.

Moduli EOS per acqua:

Acqua "w"

Tabella 3 Modulo EOS "w". Tabella riportata dal manuale Waiwera.

| | |
|------------------------------|------------------------|
| abbreviated name: | "w" |
| component names: | ["water"] |
| phase names: | ["liquid"] |
| primary variable names: | ["pressure"] |
| default primary variables: | [10 ⁵ Pa] |
| default region: | 1 (liquid) |
| default eos.primary.scale: | {"pressure": 1e6} |
| default output fluid fields: | ["pressure", "region"] |

Questa è l'equazione di stato più semplice del modulo, che simula il comportamento dell'acqua liquida isoterma monofase. Esiste una sola variabile termodinamica primaria: la pressione.

Acqua ed energia “we”

Tabella 4 Modulo EOS “we”. Tabella riportata dal manuale Waiwera.

| | |
|-------------------------------------|---|
| abbreviated name: | “we” |
| component names: | [“water”, “energy”] |
| phase names: | [“liquid”, “vapour”] |
| primary variable names: | single-phase: [“pressure”, “temperature”] two-phase: [“pressure”, “vapour_saturation”] |
| default primary variables: | [10 ⁵ Pa, 20 °C] |
| default region: | 1 (liquid) |
| default eos.primary.scale: | {“pressure”: 1e6, “temperature”: 100} |
| default output fluid fields: | [“pressure”, “temperature”, “region”, “vapour_saturation”] |

Questa è la più semplice equazione non isoterma del modulo di stato, con una sola componente di massa (acqua) ma includendo anche la componente energetica. L'acqua può trovarsi in condizioni liquide, di vapore o bifase e può passare tra questi stati. Le variabili primarie sono la pressione e la temperatura per le condizioni monofase, ma passano alla pressione e alla saturazione del vapore in condizioni bifase. Le proprietà del fluido sono calcolate direttamente dalla formulazione termodinamica dell'acqua.

Moduli EOS acqua/NCG

Questi moduli EOS simulano miscele di acqua e gas incondensabili (NCG), insieme all'energia. Funzionano più o meno allo stesso modo degli EOS acqua/energia (“we”) a parte le modifiche alle proprietà del fluido dovute alla presenza del gas incondensabile. Le variabili primarie per questi moduli EOS sono come per l'EOS acqua/energia, ma con una terza variabile aggiunta, la pressione parziale del gas non condensabile.

-Acqua, aria ed energia “*wae*”

Tabella 5 Modulo EOS “*wae*”. Tabella riportata dal manuale Waiwera.

| | |
|-------------------------------------|---|
| abbreviated name: | “wae” |
| component names: | [“water”, “air”, “energy”] |
| phase names: | [“liquid”, “vapour”] |
| primary variable names: | single-phase: [“pressure”, “temperature”, “air_partial_pressure”] two-phase: [“pressure”, “vapour_saturation”, “air_partial_pressure”] |
| default primary variables: | [10 ⁵ Pa, 20 °C, 0 Pa] |
| default region: | 1 (liquid) |
| default eos.primary.scale: | {“pressure”: 1e6, “temperature”: 100, “partial_pressure”: “pressure”} |
| default output fluid fields: | [“pressure”, “temperature”, “region”, “air_partial_pressure”, “vapour_saturation”] |

-Acqua, CO2 e energia “*wce*”

Tabella 6 Modulo EOS “*wce*”. Tabella riportata dal manuale Waiwera.

| | |
|-------------------------------------|---|
| abbreviated name: | “wce” |
| component names: | [“water”, “CO2”, “energy”] |
| phase names: | [“liquid”, “vapour”] |
| primary variable names: | single-phase: [“pressure”, “temperature”, “CO2_partial_pressure”] two-phase: [“pressure”, “vapour_saturation”, “CO2_partial_pressure”] |
| default primary variables: | [10 ⁵ Pa, 20 °C, 0 Pa] |
| default region: | 1 (liquid) |
| default eos.primary.scale: | {“pressure”: 1e6, “temperature”: 100, “partial_pressure”: “pressure”} |
| default output fluid fields: | [“pressure”, “temperature”, “region”, “CO2_partial_pressure”, “vapour_saturation”] |

4.2.2.5 Gravità

L'entità (e la direzione rispetto alla griglia) dell'accelerazione di gravità per la simulazione può essere specificata nel file di input JSON Waiwera tramite il valore “*gravity*”. Può essere un array, un numero o null. Tutti i valori numerici della gravità hanno le unità m/s². Specificando il valore di gravità come una matrice si imposta l'intero vettore di gravità per intero. Può essere specificato come array a due elementi per le simulazioni 2D e come array a tre elementi per le simulazioni 3D.

4.2.2.6 Parametri della roccia

I parametri relativi ai dettagli del supporto rock per la simulazione sono specificati nel file di input JSON Waiwera tramite il valore "rock". Questo valore è un oggetto e può contenere i valori mostrati di seguito.

Tabella 7 "rock" input Waiwera. Tabella riportata dal manuale Waiwera.

| name | type | default | value |
|-------------------------|---------------|--------------------|-------------------------------------|
| "types" | array | [] | <i>Rock types</i> |
| "relative_permeability" | object | {"type": "linear"} | <i>Relative permeability curves</i> |
| "capillary_pressure" | object null | {"type": "zero"} | <i>Capillary pressure functions</i> |

I tipi di roccia possono essere specificati nel file di input JSON Waiwera tramite il valore "rock.types". Questa è una serie di oggetti, ogni oggetto corrispondente a una definizione del tipo di roccia.

Tabella 8 "rock_types" input Waiwera. Tabella riportata dal manuale Waiwera

| name | type | default | value |
|--------------------|----------------|------------------------|--|
| "name" | string | "" | optional rock type name |
| "permeability" | number array | 10^{-13} m^2 | permeability (m^2) |
| "wet_conductivity" | number | 2.5 W / (m °C) | heat conductivity of fully saturated rock (W / (m °C)) |
| "dry_conductivity" | number | = wet conductivity | heat conductivity of unsaturated rock (W / (m °C)) |
| "porosity" | number | 0.1 | porosity |
| "density" | number | 2200 kg / m^3 | grain density (kg / m^3) |
| "specific_heat" | number | 1000 J / (kg °C) | specific heat (J / (kg °C)) |
| "cells" | array | [] | cells with this rock type |
| "zones" | string array | [] | <i>Mesh zones</i> with this rock type |

Sono stati proposti diversi tipi di curve di permeabilità relativa. Waiwera ne offre molte e consente di specificare le curve desiderate nel file di input JSON tramite il valore "rock.relative_permeability". Questo valore è un oggetto, contenente un valore di stringa "type" che seleziona il tipo di curva, insieme ad altri parametri che dipendono dal tipo di curva. Di seguito sono riportati i diversi tipi di curve di permeabilità relativa disponibili in Waiwera:

- Completamente mobile;
- Lineare;
- Tipo Pickens;
- Tipo Corey;
- Van Genuchten.

Tabella 9 "rock_relative_permeability" input Waiwera, curva di permeabilità relativa lineare. Tabella riportata dal Manuale Waiwera.

| name | type | default | value |
|----------|--------|----------|----------------------------------|
| "type" | string | "linear" | relative permeability curve type |
| "liquid" | array | [0, 1] | liquid saturation limits |
| "vapour" | array | [0, 1] | vapour saturation limits |

La funzione di pressione capillare desiderata è specificata nel file di input JSON Waiwera tramite il valore *"rock.capillary_pressure"*. Questo valore è un oggetto (o `null`), contenente un valore stringa *"type"* che seleziona il tipo di funzione, insieme ad altri parametri che dipendono dal tipo di funzione. I tipi di funzioni di pressione capillare disponibili in Waiwera sono: Zero, lineare e Van Genuchten.

4.2.2.7 Condizioni iniziali

Nel file di input JSON Waiwera, le condizioni al contorno di Dirichlet vengono impostate tramite il valore *"boundaries"*, che è una matrice di oggetti. Ciascun oggetto contiene una specifica delle condizioni al contorno, che può essere applicata a una o più facce mesh. In ogni condizione al contorno, le variabili primarie sono specificate tramite il valore dell'array *"primary"*. La lunghezza e il valore predefinito di questa matrice dipendono entrambi dall'equazione di stato (EOS) utilizzata. Oltre alle variabili primarie, la regione termodinamica deve essere specificata come parte della condizione al contorno, tramite il valore intero *"region"*. Ciò è necessario affinché le variabili primarie possano essere interpretate correttamente per le condizioni di fase desiderate al contorno.

Tabella 10 "boundaries" input Waiwera. Tabella riportata dal manuale Waiwera.

| name | type | default | value |
|-----------|----------------|----------------|--|
| "primary" | array | depends on EOS | prescribed primary variables |
| "region" | integer | depends on EOS | prescribed thermodynamic region |
| "faces" | object array | { } | faces on which to apply boundary condition |

Gli indici delle celle, specificati dal valore dell'array "*cells*", sono gli indici delle celle appena all'interno del confine è anche necessario specificare il vettore normale esterno del confine, tramite il valore dell'array "*normal*".

Tabella 11 "cells" input Waiwera. Tabella riportata dal manuale Waiwera.

| name | type | default | value |
|----------|-------|-----------|-----------------------------------|
| "cells" | array | [] | boundary cell indices |
| "normal" | array | [0, 0, 1] | outward normal vector of boundary |

4.2.2.8 Sorgenti

Una sorgente inietta o produce (cioè estrae) massa o energia con una particolare portata (kg/s o J/s) in una data cella. Per convenzione, la portata è positiva per l'iniezione e negativa per la produzione. (Qui usiamo il termine "*sources*" per includere "*levels*", che sono semplicemente considerati sorgenti con portata negativa.) Le sorgenti possono essere utilizzate per rappresentare sorgenti fisiche o pozzi come pozzi o sorgenti. Possono anche essere usati per implementare le condizioni al contorno di Neumann, cioè per modellare i confini dove sono prescritte portate di massa o di energia. Le sorgenti vengono impostate nel file di input JSON Waiwera tramite il valore "*sources*". Questa è una serie di oggetti. Ciascun oggetto nell'array contiene una specifica di origine, che può impostare una singola origine o più origini con parametri simili in celle diverse. La portata per ciascuna sorgente può essere specificata utilizzando il suo valore di "*rate*".

Tabella 12 "sources" input Waiwera. Tabella riportata dal manuale Waiwera.

| name | type | default | value |
|------------------------|-------------------------|---|---|
| "name" | string | "" | optional source name |
| "component" | integer string | "water" (injection) 0 (production) | mass or energy component |
| "production_component" | integer string | "heat" if "component" = "heat", otherwise 0 | mass or energy component for <i>Mixed flow</i> production |
| "rate" | number array object | 0 | flow rate (kg/s or J/s) |
| "enthalpy" | number array object | 83.9×10 ³ J/kg | injection enthalpy (J/kg) |
| "cell" "cells" | integer array | [] | indices of cells with this source specification |
| "zones" | array | [] | <i>Mesh zones</i> containing cells with this source specification |
| "interpolation" | string | "linear" | interpolation method for data tables |
| "averaging" | string | "integrate" | averaging method for data tables |
| "deliverability" | object | { } | <i>Deliverability</i> source control |
| "recharge" | object | { } | <i>Recharge</i> source control |
| "limiter" | object | { } | <i>Limiter</i> source control |
| "direction" | string | "both" | <i>Direction</i> source control |
| "factor" | number array object | { } | <i>Factor</i> source control |

4.2.2.9 Condizioni iniziali

Prima che la simulazione possa essere eseguita, è necessario specificare inizialmente le variabili primarie in ogni cella. Inoltre, devono essere specificate anche le regioni termodinamiche nelle celle, in modo che le variabili primarie iniziali possano essere interpretate correttamente. Ci sono due modi per farlo. Tuttavia, entrambi i modi specificano le condizioni iniziali tramite il valore "*initial*" nel file di input JSON Waiwera e in entrambi i casi questo valore è un oggetto (sebbene alcuni dei suoi valori siano diversi in ciascun caso).

Tabella 13 "primary_condition" input Waiwera. Tabella riportata dal manuale Waiwera

| name | type | default | value |
|-----------|-----------------|----------------|--|
| "primary" | number array | (no default) | initial <i>Primary variables</i> |
| "region" | integer array | depends on EOS | initial <i>Thermodynamic regions</i> |
| "minc" | boolean | false | whether initial conditions include values for MINC cells (see <i>MINC initial conditions</i>) |

4.2.2.10 Parametri temporali

Waiwera risolve l'evoluzione temporale del sistema simulato utilizzando un metodo di time-step numerico. Il tempo viene discretizzato, procedendo dal tempo iniziale della simulazione fino al tempo finale per gradi. Ad ogni time step, il modello si compone di un sistema di equazioni non lineari da risolvere utilizzando il solutore iterativo non lineare PETSc "SNES". Questo a sua volta porta a una sequenza di sistemi di equazioni lineari da risolvere ad ogni iterazione del solutore non lineare, utilizzando i solutori lineari PETSc "KSP". Tutti i parametri relativi al tempo sono specificati nel file di input JSON Waiwera tramite il valore "time", che è un oggetto.

Tabella 14 "time" input Waiwera. Tabella riportata dal manuale Waiwera

| name | type | default | value |
|---------|---------------|-----------|---------------------------|
| "start" | number | 0 | simulation start time (s) |
| "stop" | number null | null | simulation stop time (s) |
| "step" | object | see below | time stepping parameters |

I valori "start" e "stop" nell'oggetto "time" sono numeri che specificano gli orari di inizio e fine della simulazione. Tutti i parametri relativi al time-step sono specificati tramite il valore "step" nell'oggetto "time". Il valore "step" è esso stesso un oggetto. Per le specifiche sui parametri degli step temporali si riporta alla guida Waiwera.

4.2.2.11 Modulo Output

Il modulo "output" consente di gestire gli outputs della simulazione. Con "filename" viene assegnato il nome del file di output in formato HDF5 che verrà creato dopo aver eseguito la simulazione, Gli altri valori nell'oggetto "output" controllano i tempi in cui i risultati vengono emessi e quali proprietà fluido e sorgente vengono scritte.

4.2.2.12 Modulo file di registro

L'output del log può essere controllato impostando il valore "logfile" nel file di input JSON. Il valore "logfile" può assumere un valore booleano ed essere utilizzato semplicemente per

attivare o disattivare l'output del registro. In alternativa, il valore "*logfile*" può essere specificato come oggetto, con un valore di stringa "*filename*" per specificare il nome del file (può essere impostato su "" per disabilitare l'output del registro). Ha anche un valore booleano "*echo*" che controlla se l'output del registro viene inviato al display della console durante l'esecuzione della simulazione.

4.3 Output di simulazione HDF5

Waiwera trascrive i risultati della simulazione (non i messaggi di registro, che vengono scritti nell'output del registro) in un file di output nel formato di file HDF5.

HDF5 è un formato di file binario e un modello di dati progettato per archiviare e gestire in modo efficiente set di dati grandi e complessi. Sono disponibili numerosi strumenti software per la gestione, la visualizzazione e l'analisi dei file HDF5. I risultati della simulazione Waiwera consistono principalmente in:

- proprietà del fluido selezionato (ad es. pressioni, temperature) in ciascuna cella;
- proprietà di flusso selezionato (ad es. flussi di massa) attraverso ciascuna faccia;
- proprietà della sorgente selezionata (es. portate, entalpie) in corrispondenza di ciascuna sorgente;
- proprietà della geometria della mesh selezionata (ad es. centroidi e volumi delle celle e aree delle facce);

che vengono scritti nel file HDF5 in momenti specificati. L'output della simulazione può essere controllato tramite il valore "*output*" nel file di input JSON Waiwera. Questo può essere specificato come un valore booleano, in modo che sia possibile (sebbene insolito) disabilitare l'output della simulazione impostandolo su *false*. Nella maggior parte dei casi, tuttavia, viene specificato come oggetto. Questo oggetto ha un valore di stringa "*filename*" per specificare il nome del file dell'output della simulazione. Se questo non è specificato, verrà utilizzato un nome file predefinito, formato dal nome file di input JSON Waiwera ma con l'estensione modificata in ".*h5*". Gli altri valori nell'oggetto "*output*" controllano i tempi in cui i risultati vengono emessi e quali proprietà fluido e sorgente vengono scritte.

Struttura output

I dati nei file HDF5 sono generalmente organizzati in gruppi, che possono essere considerati analoghi alle directory in un file system. Un gruppo può contenere uno o più set di dati. Tutti i

gruppi e i set di dati in un file HDF5 sono contenuti in un gruppo di livello superiore chiamato gruppo "root". Per ulteriori informazioni sul modello dati HDF5, fare riferimento alla documentazione HDF5 .

Sotto il gruppo radice, un file di output Waiwera HDF5 contiene solitamente due gruppi, "cell_fields" e "source_fields". Possono essere presenti anche altri due gruppi, "face_fields" e "minc", a seconda di come è configurata la simulazione. Il contenuto di questi quattro gruppi è descritto di seguito, insieme ad altri set di dati che possono essere presenti nel gruppo radice.

Stampa delle celle

Il gruppo "cell_fields" in un file di output Waiwera HDF5 contiene i dati di output definiti nelle celle. I principali dataset di questo gruppo contengono le proprietà del fluido calcolate nelle celle. Questi set di dati hanno nomi che iniziano con "fluid_" (es. "fluid_liquid_density"). I set di dati sui fluidi specifici inclusi qui possono essere selezionati nel file di input JSON Waiwera. Poiché i set di dati del fluido sono generalmente dipendenti dal tempo, è presente una riga nel set di dati per ogni tempo di output e su ogni riga è presente una colonna per cella. Il gruppo "cell_fields" contiene anche set di dati relativi alla geometria della cella, i cui nomi iniziano con "cell_geometry". Ad esempio, il set di dati "cell_geometry_volume" contiene i volumi di tutte le celle. Questi set di dati non dipendono dal tempo, quindi c'è solo una riga per cella. Il set di dati "cell_geometry_centroid" contiene un array di tre elementi per ogni cella, quindi, ci sono tre colonne su ogni riga.

Stampa delle sorgenti

Il gruppo "source_fields" in un file di output Waiwera HDF5 contiene i dati di output definiti alle sorgenti, ad esempio la portata e l'entalpia a ciascuna sorgente. Questi set di dati hanno nomi che iniziano con "source_", ad esempio "source_rate". I set di dati qui inclusi possono essere selezionati nel file di input JSON Waiwera. Come i set di dati fluidi, la maggior parte dei set di dati di origine dipende dal tempo, con una riga per tempo di output, con ogni riga con una colonna per origine.

Stampa delle facce

Il gruppo "face_fields" in un file di output Waiwera HDF5 contiene i dati di output definiti sulle facce della mesh. I principali set di dati in questo gruppo contengono flussi di fluido calcolati attraverso le facce. Questi dataset hanno nomi che iniziano con "flux_" (es. "flux_water", "flux_energy" o "flux_vapour"). I set di dati di flusso specifici inclusi qui possono essere

selezionati nel file di input JSON Waiwera. Si noti che se non vengono specificati campi di flusso per l'output (che è l'impostazione predefinita), il gruppo "*face_fields*" non sarà presente. Per quanto riguarda i campi cella, i set di dati del flusso sono generalmente dipendenti dal tempo, quindi, c'è una riga per ogni tempo di output, e su ogni riga c'è una colonna per faccia. Il gruppo "*face_fields*" contiene anche set di dati relativi alla geometria della faccia, i cui nomi iniziano con "*face_geometry*". Ad esempio, il set di dati "*face_geometry_area*" contiene le aree delle facce. Questi set di dati non dipendono dal tempo, quindi c'è solo una riga per faccia. Alcuni dei set di dati della geometria della faccia (es. "*face_geometry_distance*", "*face_geometry_normal*", "*face_geometry_centroid*") contengono array per ogni faccia, quindi ci sono più colonne su ogni riga.

Stampa del dataset temporale

Il gruppo radice in un file di output Waiwera HDF5 contiene anche un set di dati "*time*". Questo è un semplice array contenente tutti i tempi di output, uno per riga. Queste righe corrispondono alle righe nei set di dati dipendenti dal tempo nei gruppi "*cell_fields*", "*face_fields*" e "*source_fields*". Indicizza set di dati e ordinamento dei dati. Quando PETSc scrive i dati della cella da una simulazione parallela all'output HDF5, per impostazione predefinita i dati non vengono scritti nell'ordine originale o "naturale" che si verificherebbe in una simulazione seriale. Questo perché in una simulazione parallela, la mesh è distribuita tra i diversi processi paralleli e riassemblare i dati distribuiti nel suo ordinamento naturale richiederebbe un'operazione di "*scattering*" parallela ogni volta che i dati devono essere emessi. Le operazioni che richiedono la comunicazione parallela devono essere ridotte al minimo se il codice deve essere adattato bene a un numero elevato di processi paralleli. Invece, i dati vengono scritti in quello che è noto come ordinamento "globale". Qui i dati vengono scritti in ordine di processo, quindi tutti i dati dal processo parallelo 0 vengono scritti per primi, seguiti da tutti i dati dal processo 1 e così via. Su ogni processo, i dati vengono scritti secondo un ordinamento "locale" su quel processo, che generalmente non è correlato all'ordinamento naturale.

Il file di output Waiwera HDF5 contiene un set di dati (nel gruppo radice) chiamato "*cell_index*" che è una mappatura dall'ordinamento naturale delle celle all'ordinamento globale delle celle utilizzato nell'output. Quindi, se il set di dati "*cell_index*" è rappresentato dall'array, l'indice dei dati della cella globale corrispondente all'indice naturale i . Questo array di indici può essere utilizzato per riordinare l'output nell'ordinamento dell'output globale nell'ordinamento naturale, per la post-elaborazione. Viene anche utilizzato internamente da Waiwera per riordinare i dati del fluido quando una simulazione viene riavviata dall'output di una corsa precedente. Allo

stesso modo, esiste un altro set di dati chiamato "*source_index*" che mappa l'ordinamento della sorgente naturale sull'ordinamento della sorgente globale nell'output. Se è presente un output su facce (nel gruppo HDF5 "*face_fields*"), nel gruppo radice saranno presenti altri due set di dati indice: "*face_cell_1*" e "*face_cell_2*". Questi contengono gli indici naturali delle celle su entrambi i lati di ciascuna faccia e possono essere utilizzati per identificare i dati corretti del campo facciale per una data faccia nella mesh di simulazione. Oltre alle facce della mesh interna, i set di dati del campo delle facce contengono anche dati per le facce al contorno su cui vengono applicate le condizioni al contorno di Dirichlet. Queste facce non possono essere identificate da una coppia di indici di celle naturali, perché non c'è una cella mesh all'esterno del confine. Le condizioni al contorno di Dirichlet sono specificate tramite i "*boundaries*" valore dell'array nel file di input JSON Waiwera. Ogni elemento di questo array specifica una diversa condizione al contorno. Per le facce di confine nell'output, il set di dati "*face_cell_1*" contiene l'indice naturale della cella all'interno del confine, ma il set di dati "*face_cell_2*" contiene il negativo dell'indice (basato su 1) della specifica della condizione al contorno nell'ingresso.

4.4 File di registro YAML

Tutti i messaggi scritti nel file di registro Waiwera hanno la stessa struttura, formata da quattro parti:

1. "*level*": il livello del messaggio di log, che può essere:
 - "*info*", per messaggi con informazioni sul normale funzionamento della simulazione
 - "*warn*", per gli avvertimenti (condizioni che sono insolite ma non causano l'arresto della simulazione)
 - "*err*", per errori, che causeranno l'arresto della simulazione
2. "*source*": quale parte del codice ha generato il messaggio
3. "*event*": l'evento che il messaggio sta segnalando
4. dati: dati supplementari che forniscono ulteriori dettagli sull'evento

Ogni messaggio di registro è formattato nel file YAML come una singola riga.

In YAML le righe che iniziano con un trattino (-) indicano gli elementi dell'array, quindi, le righe nel file di registro rappresentano collettivamente un array di messaggi. Gli array possono anche essere scritti in YAML in forma abbreviata di una riga all'interno di coppie di parentesi quadre (come in JSON), quindi ogni messaggio rappresenta anche un array, con quattro elementi (livello, origine, evento e dati).

Gli elementi *"level"*, *"source"* e *"event"* sono semplici stringhe. L'elemento *"data"* è un oggetto (cioè una raccolta di elementi nominati) che può essere scritto in YAML (come in JSON) all'interno di parentesi graffe, con nomi accoppiati e i loro valori corrispondenti (vedi strutture dati JSON).

In generale, i messaggi di registro vengono scritti nel file su righe consecutive, ma vengono inserite righe vuote dopo alcuni gruppi di messaggi (ad es. messaggi per un particolare passaggio temporale) per renderli più facilmente leggibili. Queste righe vuote vengono ignorate quando il file viene analizzato utilizzando uno script o un altro programma.

5 Installazione ed esecuzione di Waiwera

Il presente capitolo vuole riportare nel dettaglio quelle che sono le operazioni da svolgere per l'installazione e l'esecuzione di Waiwera. Questa attività, tutt'altro che semplice, ha comportato l'installazione di programmi aggiuntivi a corredo di Waiwera, nonché diverse consultazioni a siti web dedicati per ottenere la corretta configurazione dei software utilizzati.

5.1 Procedura di istallazione

Waiwera può essere eseguito in *Linux*, *Windows* e *MacOS* utilizzando *Docker* [S17]. *Docker* è una tecnologia di “containerizzazione” che utilizza il kernel di Linux proprio per isolare i processi e poterli eseguire in modo indipendente. È stato quindi necessario procedere all'installazione di *Docker*.

5.1.1 Docker Desktop per Windows

Per installare *Docker Desktop* su *Windows* sono necessari diversi requisiti [S18], tra cui *Windows* 10 a 64 bit: *Home* o *Pro* 2004 (build 19041) o versioni successive, oppure *Enterprise* o *Education* 1909 (build 18363) o versioni successive. Inoltre, è necessario abilitare la funzione *WSL2* [S19] su *Windows 10*; questo può essere fatto con i seguenti requisiti hardware:

- Processore a 64 bit con *Second Level Address Translation (SLAT)* [S20]
- *RAM* di sistema da 4 GB;
- Il supporto per la virtualizzazione hardware a livello di *BIOS* deve essere abilitato.

Infine, è necessario abilitare le funzionalità di *Windows Hyper-V* e *Container*. Pertanto, come prima cosa, è necessario scaricare e installare il pacchetto del *kernel Linux WSL2* [S21] [S22]. A installazione completata è stato impostato *WSL2* come versione predefinita durante l'installazione di una nuova distribuzione *Linux*. Questo può essere fatto attraverso l'esecuzione del seguente comando in un terminale di *Windows*:

```
wsl --set-default-version 2
```

Per eseguire correttamente *WSL2* è necessario che la macchina sia dotata di supporto hardware alla virtualizzazione e *SLAT*. Accedendo a “*Gestione attività*” di Windows e cliccando sulla scheda “*Prestazioni*” è possibile verificare se il supporto alla virtualizzazione è attivo. In *Figura 7* è riportata la finestra di interesse con evidenziato lo stato della virtualizzazione che, come si vede, risulta abilitata.

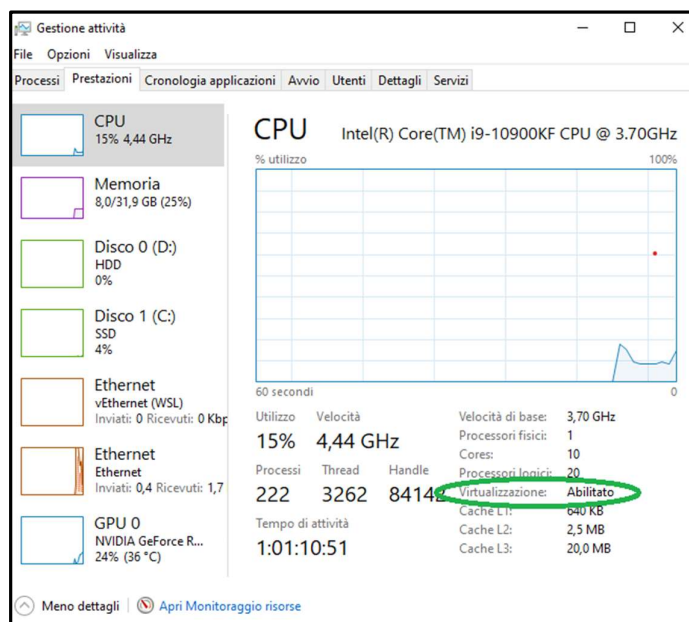


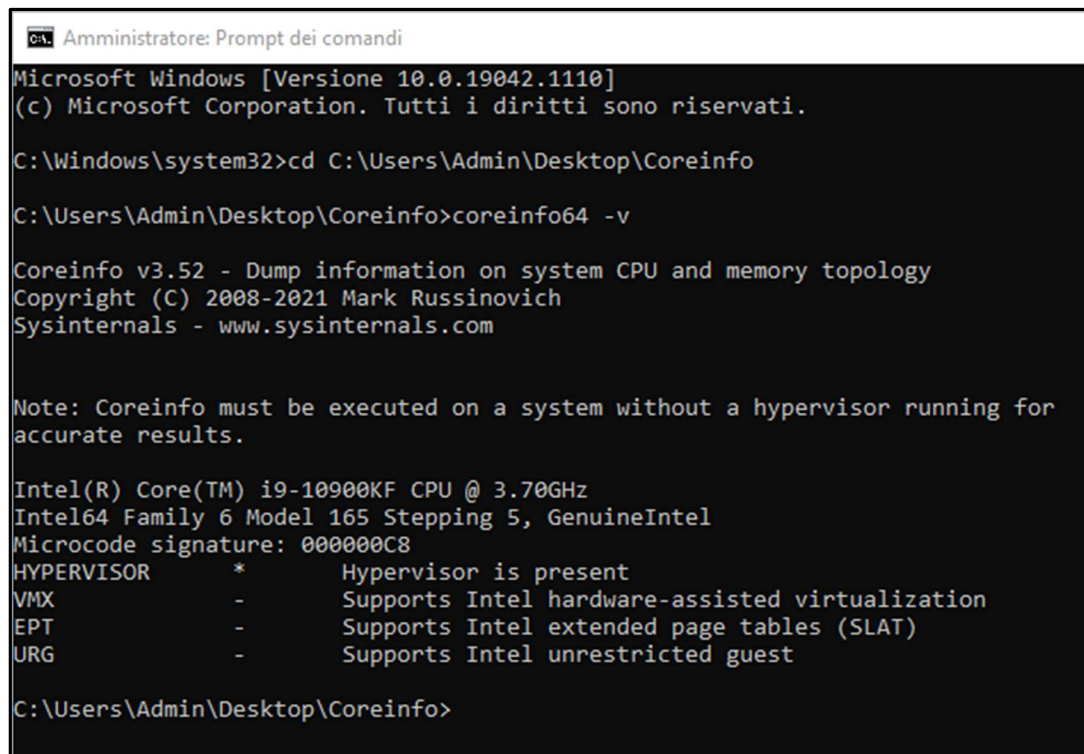
Figura 7 Gestione attività, stato di Virtualizzazione Abilitata.

Se così non fosse è possibile che la virtualizzazione sia disabilitata nel *BIOS* [S23] /*UEFI* [S24] oppure che la *CPU* non abbia il supporto hardware alla virtualizzazione. In particolare, nel PC usato per installare Waiwera, è stato necessario abilitare la virtualizzazione tramite il *BIOS* della scheda madre, accessibile attraverso l’opzione offerta nella schermata che compare all’avvio del PC. Una volta dentro al *BIOS* si è seguito il seguente percorso: *Advanced mode* → *CPU Configuration* → *Intel (VMX) Virtualization Technology*, dove è stata abilitata quest’ultima voce (ma *BIOS* di differenti computer possono avere un percorso differente). A questo punto è stato necessario avere informazioni in merito allo *SLAT*. Si può utilizzare l’eseguibile *Coreinfo*, scaricabile gratuitamente [S25], e accedendo al percorso in cui è stata estratta la cartella zippata può essere usato il seguente comando da terminale:

```
coreinfo64 -v
```

Se il PC avesse montato *Windows 10 a 32 bit* allora si sarebbe dovuto usare il seguente comando:

```
coreinfo -v
```



```
Amministratore: Prompt dei comandi
Microsoft Windows [Versione 10.0.19042.1110]
(c) Microsoft Corporation. Tutti i diritti sono riservati.

C:\Windows\system32>cd C:\Users\Admin\Desktop\Coreinfo

C:\Users\Admin\Desktop\Coreinfo>coreinfo64 -v

Coreinfo v3.52 - Dump information on system CPU and memory topology
Copyright (C) 2008-2021 Mark Russinovich
Sysinternals - www.sysinternals.com

Note: Coreinfo must be executed on a system without a hypervisor running for
accurate results.

Intel(R) Core(TM) i9-10900KF CPU @ 3.70GHz
Intel64 Family 6 Model 165 Stepping 5, GenuineIntel
Microcode signature: 000000C8
HYPERVISOR      *      Hypervisor is present
VMX             -      Supports Intel hardware-assisted virtualization
EPT            -      Supports Intel extended page tables (SLAT)
URG            -      Supports Intel unrestricted guest

C:\Users\Admin\Desktop\Coreinfo>
```

Figura 8 Utilizzo di Core info.

L'output di coreinfo è riportato in *Figura 8* dove il simbolo “*” indica che la caratteristica è rispettata, al contrario, il simbolo “-“ indica che la CPU non offre la specifica funzione. In questo caso non è supportato lo *SLAT /EPT* ma è supportato l'*HyperVisor* (caratteristica comunque richiesta per l'installazione di *Docker*). Nonostante la mancanza di *SLAT* si è scelto di proseguire in quanto è stato comunque possibile installare *Docker* e *Waiwera* risultava girare correttamente. Bisognerà tuttavia definire successivamente se il software funziona correttamente attraverso dei confronti con problemi tipo o con dei test. Come ultimo si è proceduto ad accertare che le funzionalità di *Windows Hyper-V*, *Containers*, Virtualizzazione e piattaforma della macchina virtuale fossero abilitate. Questo può essere fatto attraverso la

finestra “Funzionalità di Windows” (accessibile da “Pannello di Controllo->Programmi e funzionalità”) spuntando le caselle desiderate come riportato in *Figura 9*.

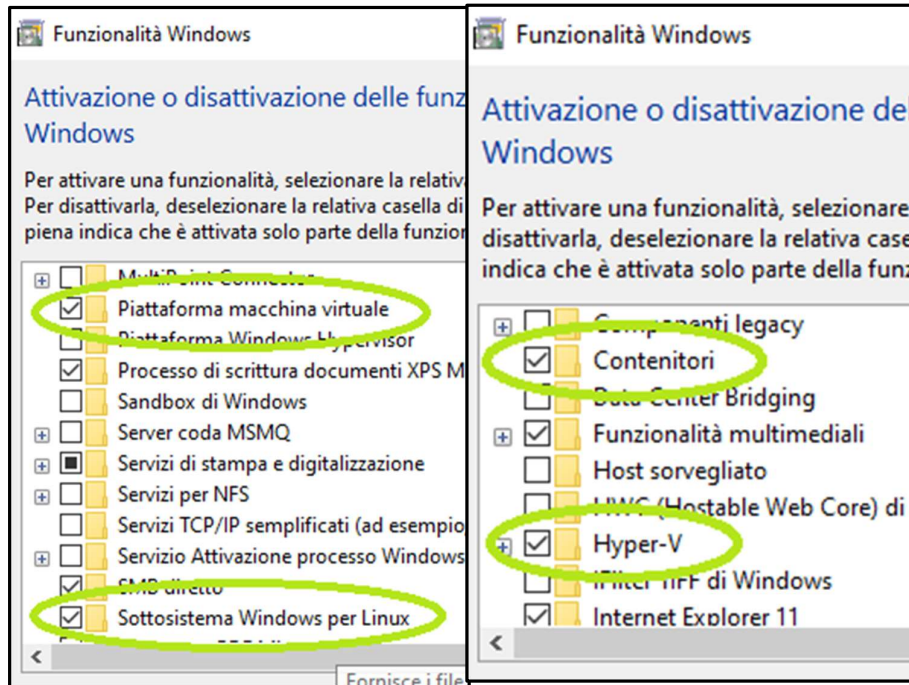


Figura 9 Attivazione Piattaforma macchina virtuale, Sottosistema Windows per Linux, Contenitori e Hyper-V tramite finestra di funzionalità Windows tramite finestra di funzionalità Windows.

5.1.1.1 Installazione Docker Desktop

Docker Desktop Installer.exe può essere scaricato molto semplicemente tramite le pagine ufficiali di *Docker* o tramite i link rilasciati dal manuale di installazione di *Waiwera* [S18] Per eseguire l’installazione si effettua doppio clic sul file .exe scaricato assicurandosi di spuntare “Abilita funzionalità Windows Hyper-V” e “Installa i componenti Windows richiesti per WSL2” quando richiesto.

5.1.2 Libreria Python Pywaiwera

Oltre a *Docker* è necessaria l’installazione della libreria *Python PyWaiwera*. Questa libreria Python semplifica le attività relative al lavoro con *Waiwera* ed è quindi necessaria l’installazione degli strumenti per il supporto del linguaggio di scripting Python e della libreria *PyWaiwera*.

Per installare Python è stato sufficiente collegarsi al sito ufficiale ed effettuare il download [S26] dell’installatore ed eseguirlo. Per installare la libreria *PyWaiwera* si è utilizzato il tool “*pip*” (*Pip Installs Packages*); questo permette di accedere ai pacchetti del *Python Package Index*. Attraverso il prompt dei comandi si è utilizzato il seguente comando:

```
pip install pywaiwera
```

In questo caso, come specificato nella guida all’installazione, è stato necessario aggiungere la directory in cui è stato installato l’eseguibile (*waiwera-dkr*) alla *PATH* [S27], accessibile mediante il “*Pannello di controllo*” andando quindi a modificare le variabili d’ambiente.

5.2 Esecuzione

Con l’installazione di *PyWaiwera*, oltre all’eseguibile *waiwera-dkr*, si possono ottenere dei problemi di esempio attraverso il seguente comando:

```
waiwera-dkr --examples
```

Si avranno quindi diversi problemi con i relativi file di input, in formato JSON, che possono essere utilizzati per l’esecuzione di *Waiwera* tramite *Docker*. Posizionandosi nella cartella del file di input desiderato si esegue il seguente comando:

```
waiwera-dkr filename.json
```

Questo comando esegue una simulazione in modalità seriale, adatta solo a piccoli problemi. Per eseguire *Waiwera* con *Docker* in modalità parallela è necessario definire il numero di tali processi attraverso il parametro *-np*:

```
waiwera-dkr -np 16 filename.json
```

Cioè verrà eseguito *Waiwera* utilizzando 16 processi in parallelo.

In questo caso specifico il percorso di *waiwera-dkr.exe* era il seguente:

“C:\Users\Admin\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.9_qbz5n2kfra8p0\LocalCache\local-packages\Python39\Scripts”.

Dove è possibile trovare anche la cartella contenente i diversi problemi scaricati “examples” con i relativi file di input come riportato in *Figura 10*. Si può notare che sono presenti quattro tipologie di problemi, in ordine, MINC, che tratta test con rocce fratturate, *Model Intercomparison Study*, ovvero, una selezione di problemi dal “*Geothermal Model Intercomparison Study*” [14] del 1980, NCG, che riguardano simulazioni di miscele di acqua e gas non-condensabili (NCG) insieme all’energia., e *Source*, un insieme di problemi per il controllo delle sorgenti. Aprendo i problemi NCG, ad esempio “*co2_column*”, al suo interno si trovano quattro diversi file di input in formato JSON, il file di mesh in formato MSH e un file in formato

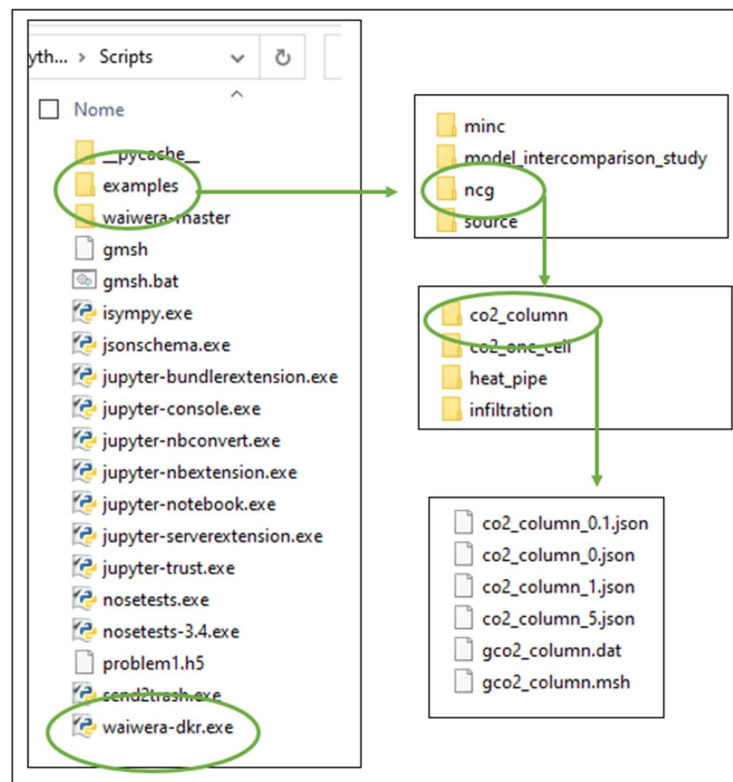


Figura 10 Percorso eseguibile Waiwera - “waiwera-dkr” e Problemi di esempio.

DAT [S28] che descrive le geometrie del problema. I diversi file JSON fanno riferimento alle diverse percentuali in massa di CO₂, presenti nel fluido iniettato, rispettivamente 0%, 0.1%, 1% e 5%. Per l’esecuzione di Waiwera si è quindi eseguito il seguente comando, in riferimento a “*co2_column_1.json*” e posizionandosi nella stessa cartella in cui è presente il file di input:

```
waiwera-dkr co2_column_1.json
```

Ripetendo lo stesso comando anche per “*co2_column_0.1.json*” e “*co2_column_5.json*”, nella medesima cartella si creeranno i file di output in formato HDF₅ e YAML, come riportato in *Figura 11*.

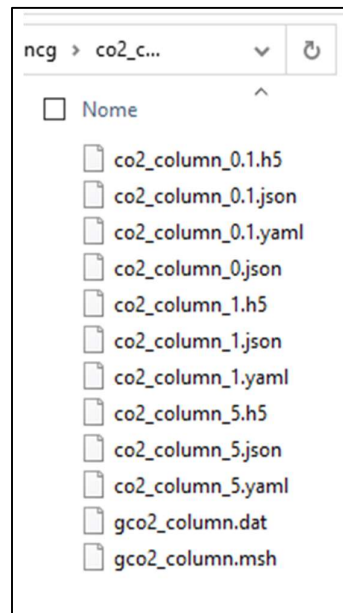


Figura 11 File di output generati con l'esecuzione di Waiwera per i 3 problemi di input *co2_column.x.json*.

I file di input JSON sono in formato ASCII e pertanto è possibile aprirli e leggerli con editor di testo come il Blocco Note o Notepad++. Lo stesso vale per il file in formato YAML e DAT. Per quanto riguarda il file di output in formato HDF₅ questo risulta essere in codice binario e per l'apertura è necessaria una diversa procedura che verrà descritta all'interno del successivo capitolo.

6 Caso studio-flusso 1D con presenza di CO₂

In un manoscritto pubblicato sui proceedings di un congresso [15] vengono descritti i vari problemi forniti a corredo di Waiwera, tra cui quello in esame. Nel presente lavoro, per testare il corretto funzionamento di Waiwera è stato utilizzato il modello di colonna verticale in stato stazionario descritto da O 'Sullivan [16] che ha lo scopo di studiare gli effetti della CO₂ sullo stato naturale del giacimento e sul suo sfruttamento. La colonna ha una profondità di 1 km con una zona di *caprock*, caratterizzata da una permeabilità di 0.5 mD e spessore 300 m, e la zona del reservoir, caratterizzata da permeabilità di 20 mD e spessore 700 m. Sulla superficie vengono applicate pressione pari a 1 bar e temperatura 10°C mentre dal basso vengono iniettati "calore e massa" (2 kg/s/km² con entalpia 1300kJ/kg). Entrambi gli strati sono caratterizzati da una conducibilità termica di 2 W/(m*°C). L'esecuzione del modello avviene da condizioni iniziali idrostatiche e isoterme (senza CO₂) a uno stato stazionario che viene raggiunto dopo 10¹⁵ secondi di simulazione. Nei risultati forniti sono stati considerati tre problemi distinti con diverse percentuali in massa di CO₂ iniettata, ovvero 0,1%, 1% e 5%.

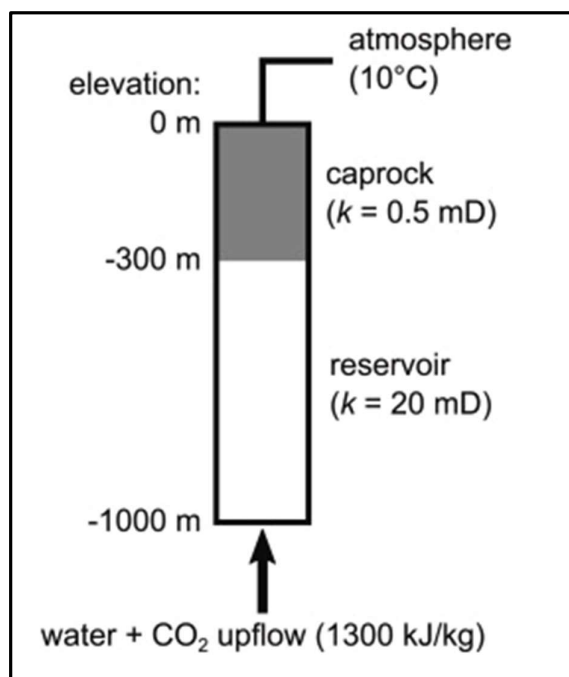


Figura 12 Modello concettuale "cos2_column_1.json". Immagine riportata da una pubblicazione scientifica [9].

6.1 Input “co2_column_1.json”

Tutti i dati riportati nel presente elaborato sono stati dedotti dal file di input “co2_column_1.json”. In *Tabella 15* è riportata la struttura schematica dell’input che è stato interamente riportato nelle seguenti pagine e descritto nei seguenti paragrafi.

Tabella 15 Struttura file di input “co2_column_1.json”

| Name | Type | Value |
|------------------|-----------------------|---------------------------|
| "title" | string | simulation title |
| "mesh" | string object | simulation mesh |
| "rock" | object | rock properties |
| "boundaries" | array | boundary conditions |
| "source" | array | source terms |
| "inizial" | object | initial conditions |
| "gravity" | number array null | gravity |
| "thermodynamics" | string | thermodynamic formulation |
| "eos" | object | equation of state |
| "time" | object | time stepping |
| "output" | boolean object | output results file |

D:\CHIARA\Scripts\examples\ncs

File Modifica Cerca Visualizza

co2_column_1.json

```

1  {
2  "boundaries": [
3  {
4    "faces": {
5      "cells": [
6        0
7      ],
8      "normal": [
9        0,
10       1
11      ]
12     },
13     "primary": [
14       100000,
15       10,
16       0
17     ],
18     "region": 1
19   }
20 ],
21 "eos": {
22   "name": "wce"
23 },
24 "gravity": 9.8,
25 "initial": {
26   "primary": [
27     [
28       246706,
29       10,
30       0
31     ],
32     [
33       540118,
34       10,
35       0
36     ],
37     [
38       833530,
39       10,
40       0
41     ],
42     [
43       1126942,
44       10,
45       0
46     ],
47     [
48       1420354,
49       10,
50       0
51     ],
52     [
53       1713766,
54       10,
55       0
56     ],
57     [
58       2007178,
59       10,
60       0
61     ],
62     [
63       2300590,
64       10,
65       0
66     ],
67     [
68       2594002,
69       10,
70       0
71     ],
72     [
73       2887414,
74       10,
75       0
76     ],
77     [
78       3205277,
79       10,
80       0
81     ],
82     [
83       3547591,
84       10,
85       0
86     ],
87     [
88       3889905,
89       10,
90       0
91     ],
92     [
93       4232219,
94       10,
95       0
96     ],
97     [
98       4574533,
99       10,
100      0
101     ],
102     [
103       4916847,
104       10,
105       0
106     ],
107     [
108       5259161,
109       10,
110       0
111     ],
112     [
113       5601475,
114       10,
115       0
116     ],
117     [
118       5943789,

```

Figura 13 Apertura file di input "cos_column_1.json"(a).

```

co2_column_1.json
119 10,
120 0
121 ],
122 [
123 6286103,
124 10,
125 0
126 ],
127 [
128 6628417,
129 10,
130 0
131 ],
132 [
133 6970731,
134 10,
135 0
136 ],
137 [
138 7313045,
139 10,
140 0
141 ],
142 [
143 7655359,
144 10,
145 0
146 ],
147 [
148 7997673,
149 10,
150 0
151 ],
152 [
153 8339987,
154 10,
155 0
156 ],
157 [
158 8682301,
159 10,
160 0
161 ],
162 [
163 9024615,
164 10,
165 0
166 ],
167 [
168 9366929,
169 10,
170 0
171 ],
172 [
173 9709243,
174 10,
175 0
176 ]
177 ],

co2_column_1.json
178 "region": 1
179 },
180 "mesh": {
181 "filename": "gco2_column.msh",
182 "thickness": 100
183 },
184 "output": {
185 "fields": {
186 "fluid": [
187 "liquid_saturation",
188 "liquid_density",
189 "liquid_CO2_mass_fraction",
190 "vapour_saturation",
191 "vapour_density",
192 "vapour_CO2_mass_fraction"
193 ]
194 },
195 "filename": "co2_column_1.h5",
196 "final": true,
197 "frequency": 1,
198 "initial": true
199 },
200 "rock": {
201 "capillary_pressure": null,
202 "relative_permeability": {
203 "liquid": [
204 0.35,
205 1
206 ],
207 "type": "linear",
208 "vapour": [
209 0,
210 0.7
211 ]
212 },
213 "types": [
214 {
215 "cells": [
216 10,
217 11,
218 12,
219 13,
220 14,
221 15,
222 16,
223 17,
224 18,
225 19,
226 20,
227 21,
228 22,
229 23,
230 24,
231 25,
232 26,
233 27,
234 28,
235 29
236 ]
237 }
238 ]
239 }

```

Figura 14 Apertura file di input "cos_column_1.json"(b).

```

237     "density": 2600,
238     "dry_conductivity": 2,
239     "name": "dfalt",
240     "permeability": [
241       2e-14,
242       2e-14
243     ],
244     "porosity": 0.1,
245     "specific_heat": 900,
246     "wet_conductivity": 2
247   },
248   {
249     "cells": [
250       0,
251       1,
252       2,
253       3,
254       4,
255       5,
256       6,
257       7,
258       8,
259       9
260     ],
261     "density": 2600,
262     "dry_conductivity": 2,
263     "name": "capr",
264     "permeability": [
265       5e-16,
266       5e-16
267     ],
268     "porosity": 0.1,
269     "specific_heat": 900,
270     "wet_conductivity": 2
271   }
272 ],
273 },
274 "source": [
275   {
276     "cell": 29,
277     "component": 1,
278     "enthalpy": 1300000,
279     "name": "inj 1",
280     "rate": 0.0198
281   },
282   {
283     "cell": 29,
284     "component": 2,
285     "enthalpy": 1300000,
286     "name": "inj 2",
287     "rate": 0.0002
288   }
289 ],
290 "thermodynamics": "ifc67",
291 "time": {
292   "start": 0,
293   "step": {
294     "adapt": {
295       "amplification": 2,
296       "maximum": 8,
297       "method": "iteration",
298       "minimum": 5,
299       "on": true,
300       "reduction": 0.2
301     },
302     "maximum": {
303       "number": 500,
304       "size": null
305     },
306     "method": "beuler",
307     "size": 100000,
308     "solver": {
309       "linear": {
310         "preconditioner": {
311           "type": "bjacobi"
312         },
313         "type": "bcgs"
314       },
315       "nonlinear": {
316         "maximum": {
317           "iterations": 8
318         },
319         "minimum": {
320           "iterations": 1
321         },
322         "tolerance": {
323           "function": {
324             "absolute": null,
325             "relative": null
326           }
327         }
328       }
329     },
330     "stop": 1e+15
331   },
332 },
333 "title": "CO2 column"
334 }

```

Figura 15 Apertura file di input "cos_column_1.json"(c).

6.1.1 Title

La sezione "title" è riportata per ultima (l'ordine non ha alcuna importanza) e viene utilizzata per assegnare un titolo alla simulazione; risulta comunque una voce facoltativa.

6.1.2 Mesh

In “*mesh*” sono contenuti tutti i dati relativi alla mesh (o griglia, relativa alla discretizzazione spaziale del dominio); è necessario indicare il file di mesh, in questo caso “*gco2_column.msh*”, contenuto nella stessa cartella in cui è presente il file JSON di input. In questo caso è stato specificato anche il “*thickness*” (spessore/larghezza nel caso di mesh 2D) pari a 100 m. In *Figura 16* è stato riportato uno schema rappresentativo della mesh e il contenuto del file “*gco2_column.dat*”, anch’esso presente nella cartella del problema e che riporta le informazioni della mesh.

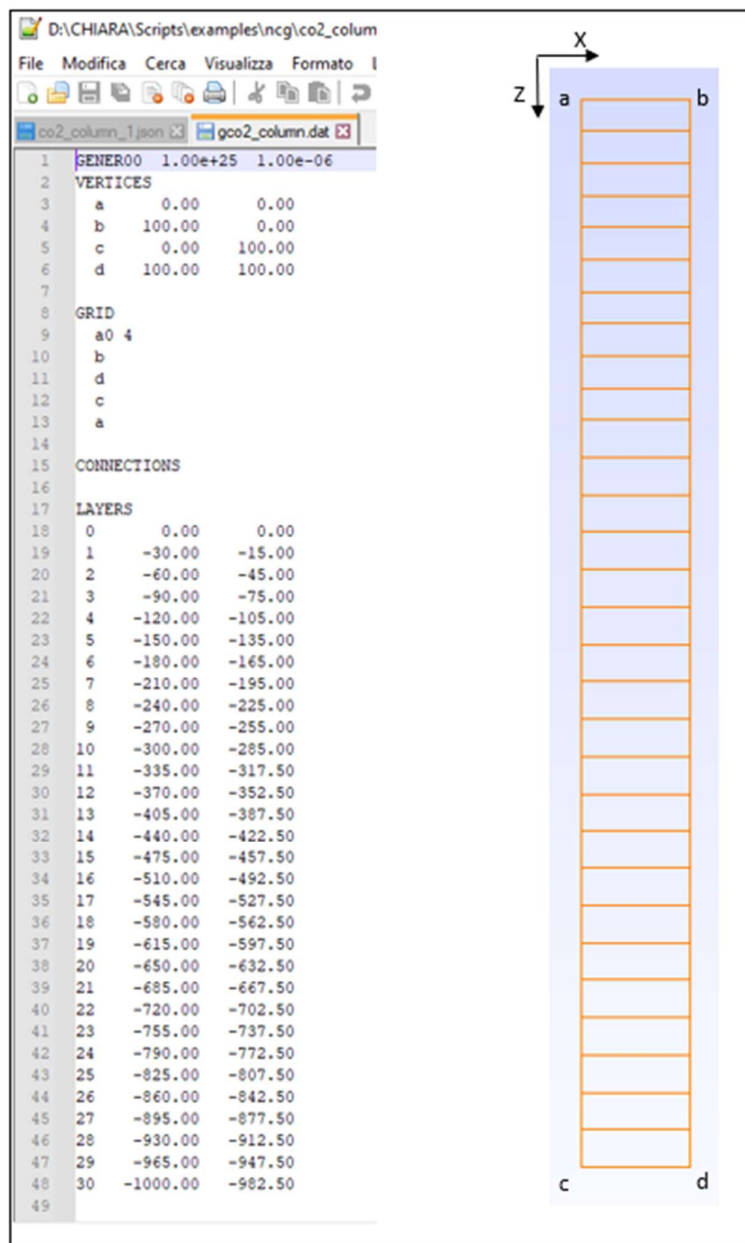


Figura 16 Descrizione mesh “*gco2_column.msh*”.

Le zone di caprock e di reservoir sono state suddivise in celle di dimensioni diverse, in particolare, le prime 10 celle risultano avere uno spessore di 30 m, dalla cella n.11 alla n.30 lo spessore è invece di 35 m.

6.1.3 Rock

La sezione “rock” contiene tutte le informazioni relative al mezzo poroso attraverso “types”, “relative_permeability” e “capillary_pressure”. In questo caso troviamo prima le informazioni sulla pressione capillare che è indicata come nulla, ovvero pari a zero indipendentemente dalla saturazione. La sezione “relative_permeability”, che contiene al suo interno le informazioni delle curve di permeabilità relativa, specifica una tipologia di curva lineare, ovvero, le permeabilità relative sono funzioni lineari di saturazione. Sia per la fase liquida che per la fase vapore, le curve variano quindi in modo lineare, rispettivamente tra i limiti di saturazione [0,35, 1] e [0, 0,7] come mostrato in *Figura 17*. Nella sezione “types” vengono infine elencate le proprietà della roccia (riportate in *Tabella 16*) dividendo le celle nelle due tipologie, caprock e reservoir.

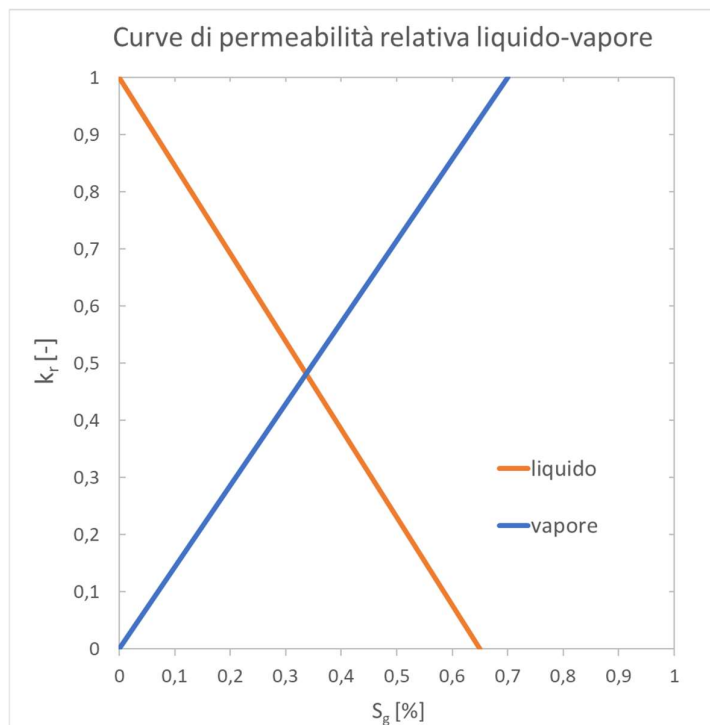


Figura 17 Curve di permeabilità relativa. Andamento lineare "co2_column_1.json".

Tabella 16 Problema "co2_column_1.json"- "types" input

| Types | | |
|--------------------------------|---------|------------|
| | caprock | reservoir |
| wet_conductivity [W/m°C] | 2 | 2 |
| dry_conductivity [W/m°C] | 2 | 2 |
| porosity | 0.1 | 0.1 |
| density [kg/m ³] | 2600 | 2600 |
| specific_heat [J/kg°C] | 900 | 900 |
| permeability [m ²] | 5E-16 | 2E-14 |
| cells | 0-9 | da 10 a 29 |
| name | capr | dfalt |

6.1.4 Eos

In questo caso le equazioni di stato sono descritte dal modulo "wce" (*water, carbon dioxide and energy*) specifico per simulazioni con miscele di acqua e gas incondensabili NCG e in *Figura 18* è riportata la descrizione presente nel manuale Waiwera.

| Water, carbon dioxide and energy ("wce") | |
|--|---|
| abbreviated name: | "wce" |
| component names: | ["water", "CO2", "energy"] |
| phase names: | ["liquid", "vapour"] |
| primary variable names: | single-phase: ["pressure", "temperature", "CO2_partial_pressure"] two-phase: ["pressure", "vapour_saturation", "CO2_partial_pressure"] |
| default primary variables: | [10 ⁵ Pa, 20 °C, 0 Pa] |
| default region: | 1 (liquid) |
| default eos.primary.scale: | {"pressure": 1e6, "temperature": 100, "partial_pressure": "pressure"} |
| default output fluid fields: | ["pressure", "temperature", "region", "CO2_partial_pressure", "vapour_saturation"] |

Figura 18 Eos "wce", water, carbon dioxide and energy.

6.1.5 Boundaries

Le condizioni al contorno "boundaries" sono definite da "faces", "region" e "primary". Con "faces" sono individuate quali sono le facce delle celle a cui applicare le condizioni al contorno e, in particolare, l'indice delle celle di confine "cells" è impostato su 0, quindi si tratta della

cella in superficie, e il vettore normale, definito da “*normal*”, è rivolto verso l’alto, quindi, si fa riferimento alla superficie superiore della cella in superficie. Con “*primary*” vengono invece definite le variabili primarie della cella che, in ordine di scrittura, risultano essere 10^5 Pa, 10°C e 0 Pa (pressione parziale di CO_2) e conformi alla struttura del modulo “*wce*” di *Figura 18*. Infine, con “*region*” viene indicata la regione termodinamica 1, quindi liquido. Per tanto, la dicitura specifica una condizione al contorno allo stato liquido di acqua, CO_2 ed energia sull’unica faccia della superficie superiore della mesh con le tre variabili primarie descritte precedentemente.

6.1.6 Sources

Avendo delle sorgenti che iniettano un fluido, i termini di portata sono positivi e si ha una divisione tra le due componenti acqua e CO_2 . Con “*cell*” viene precisato l’indice della cella sorgente; in questo caso 29 quindi l’ultima cella a partire dalla superficie. Con la voce “*component*” viene specificato il componente e, in questo caso, si fa riferimento all’indice che identifica il componente relativo a “*component names*” di *Figura 18*. Pertanto, la dicitura “*component: 1*” indica acqua, mentre “*component: 2*” indica CO_2 . La sezione “*name*” è opzionale e indica i nomi delle due iniezioni, in questo caso “*inj1*” e “*inj2*”. Con “*enthalpy*” viene impostata l’entalpia di iniezione, che in questo caso è 1300000 J/kg, mentre in “*rate*” viene inserito il valore di portata, espresso in kg/s. Quest’ultimo è ricavabile da semplici passaggi algebrici nota la portata totale in ingresso (2 kg/s/km²), la percentuale in massa di CO_2 (1%) e assumendo come dimensione della superficie di iniezione $0,01$ km² (100×100 m). Risulta quindi una portata d’acqua di $0,0198$ kg/s e $0,0002$ kg/s di CO_2 .

6.1.7 Initial

Le condizioni iniziali delle celle “*initial*” vengono specificate sempre in relazione alle equazioni di stato utilizzate, quindi anche qui si hanno pressione, temperatura e pressione parziale di CO_2 . Viene inoltre definita la regione termodinamica, anche in questo caso pari a 1 quindi liquido.

6.1.8 Gravity

Il termine gravitativo “*gravity*” imposta il vettore gravità su $[0, -9.81]$ m/s² come è normale che sia.

6.1.9 Thermodynamics

Con “*thermodynamics*” viene scelta la formulazione termodinamica per l’acqua IFC-67 che divide il diagramma P,T in cinque regioni e che identificano le diverse condizioni di fase.

6.1.10 Time

I parametri relativi all’evoluzione temporale della simulazione vengono definiti tramite “*time*” che contiene al suo interno “*start*”, inizio della simulazione, “*stop*”, fine della simulazione e “*step*”, contenente tutti i parametri passo-temporali. In questo caso è stato preso come tempo di inizio 0 e di fine 1E+15 s (circa 30 milioni di anni). Per la sezione “*step*”, al suo interno si trovano diverse voci: “*method*”, “*size*”, “*adapt*”, “*maximum*” e “*solver*”. Il metodo è impostato su “*beuler*”, ovvero, viene specificato il metodo del passo temporale di Eulero all’indietro (un metodo implicito e accurato solo al primo ordine ma molto stabile). Con “*size*” si imposta l’intervallo iniziale del passo temporale a 100000 s in quanto risulta attivo “*adapt*”. Infatti, quando si utilizza il passo temporale adattivo “*adapt*”, lo stepper temporale regola automaticamente la dimensione del passo con l’avanzare della simulazione grazie all’ algoritmo di adattamento. All’interno di “*adapt*” si trovano tutti i termini necessari all’algoritmo. In “*maximum*” si trova “*size*” impostato su `null` quindi non viene imposta una dimensione massima del passo temporale, ma “*number*” pari a 500 definisce il numero massimo di passi temporali permesse. Infine, con “*solver*”, vengono assegnati i parametri del solutore lineare e non-lineare. Per la soluzione di equazioni non lineari, “*nonlinear*”, viene utilizzato il solutore PETSc con iterazione Raphson-Newton e viene definito il numero massimo e minimo di iterazioni, in questo caso rispettivamente 8 e 1. Inoltre, con “*tolerance*” vengono specificate le tolleranze relative e assolute necessarie a studiare la convergenza della soluzione non-lineare. Per la soluzione di equazioni lineari, “*linear*”, viene usata la suite di risolutori PETSc specificandone il tipo con “*type*” e tramite “*preconditioner*” è stato scelto un preconditionatore “*bjacobi*”.

6.1.11 Outputfile

Il modulo “*output*” consente di gestire gli outputs della simulazione. Con “*filename*” viene assegnato il nome del file di output in formato HDF5 che verrà creato dopo aver eseguito la simulazione, in questo caso “*co2_column_1.h5*”. La voce “*initial*”, impostata su `true`, indica che le condizioni iniziali sono incluse nell’output, e analogamente “*final*” impostato su `true`. Con “*frequency*” si ha la possibilità di salvare in output i dati (o *printout*) meno frequentemente, quindi saltare la stampa di alcuni step temporali in modo da ridurre la dimensione del file di output. In questo caso la frequenza è impostata su 1 quindi vengono riportati output per ogni passo temporale. In “*fields*” si trova la voce “*fluid*” contenente i campi degli outputs.

6.2 Elaborazione dati

Dopo aver eseguito “*co2_column_1.json*” è stato possibile visualizzare i file di output in formato HDF5 e YAML. In particolare, è stato utilizzato HDF5view [S29] per aprire i file in formato HDF5, mentre per il file di registro YAML è stato sufficiente utilizzare applicazioni quali Blocco Note o Notepad++. Per ottenere una visualizzazione grafica del file di output HDF5 ottenuto è stato utilizzato il programma free open-source ParaView [S30]] (dedicato alla visualizzazione scientifica interattiva) che è risultato utile per una visione più intuitiva dei risultati. È stato inoltre possibile costruire dei grafici con entrambi i file di output installando ed utilizzando le librerie h5py [S31] e PyYAML [S32] di Python. Nei successivi sotto-paragrafi sono state riportate le procedure effettuate per l’utilizzo dei programmi HDFview, ParaView e per la costruzione dei grafici da script Python.

6.2.1 Utilizzo di HDF5view

Per la visualizzazione del file di output HDF5, si è scelto di utilizzare *HDFview* disponibile gratuitamente online [S29]. Considerando il problema “*problem1.json*”, all’interno della cartella “*model_intercomparison_study*”, anche in questo caso è possibile eseguire *Waiwera* ottenendo l’output “*problem1.h5*” e quest’ultimo può essere aperto in *HDFview* lanciando dapprima il programma e in seguito trascinando il file nella sezione verticale a sinistra della finestra, come mostrato in *Figura 19*. Gli outputs vengono divisi nei due gruppi “*cell-fields*” e “*source_fiels*” e possono essere aperti con un clic ottenendo valori tabellati dove vengono

visualizzate le varie proprietà delle celle per i diversi step temporali della simulazione.

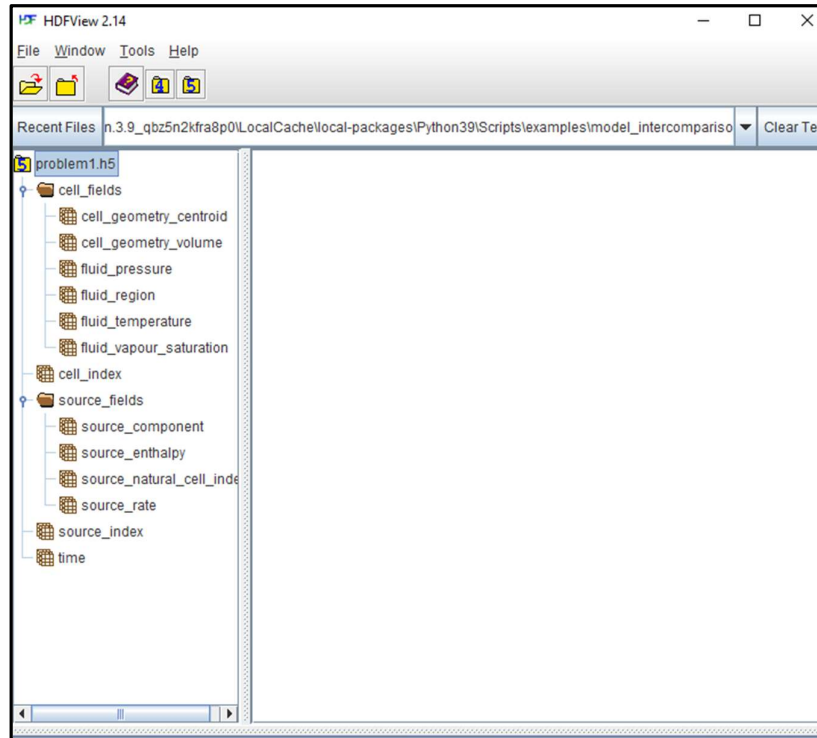


Figura 19 Apertura primo output HDF5view.

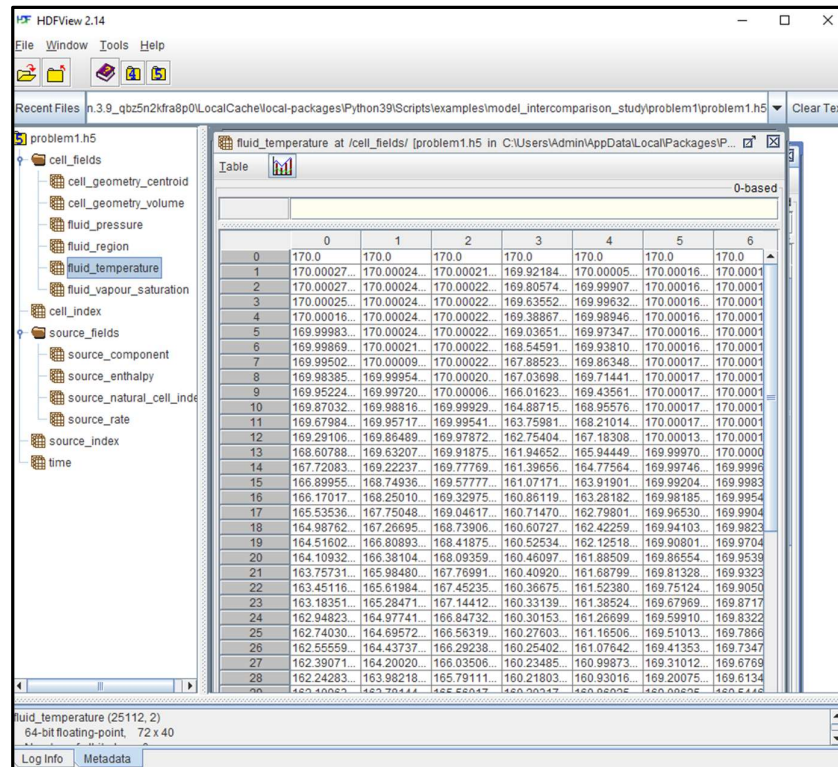


Figura 20 Visualizzazione output "problem1.h5" con apertura tabella "fluid_temperature" tramite HDFview.

Nell'esempio riportato in *Figura 20* è stato scelto in modo arbitrario di aprire l'output relativo alla temperatura del fluido “*fluid_temperature*” del file di output “*problem1.h5*”. In particolare, procedendo da sinistra verso destra, possono essere letti valori relativi alle diverse celle, mentre dall'alto verso il basso vengono letti i valori in funzione dei vari step temporali. Lo stesso ragionamento può essere fatto per tutte le altre schede appartenenti a “*cell-fields*”.

6.2.2 Utilizzo di ParaView

Il file di output in formato HDF5 può essere aperto anche mediante altri software per avere ad esempio una visualizzazione grafica dei risultati. A tale scopo è stato utilizzato il programma gratuito *ParaView* [S30]. Una volta effettuato il download ed eseguito l'apposito installatore, è sufficiente aprire l'eseguibile e trascinare nella sua finestra il file in formato HDF5. In questo caso si è fatto riferimento sempre al problema “*co2_column_1.json*” con output “*co2_column_1.h5*”. Appena importato *Paraview*, verrà visualizzata la schermata di *Figura 21*; è necessario selezionare “*VisitPixieReader*”, altrimenti non verrà visualizzato nulla e verranno segnalati degli errori. A questo punto, spostandosi sulla sinistra, è possibile selezionare diversi elementi per ottenere dei risultati grafici.

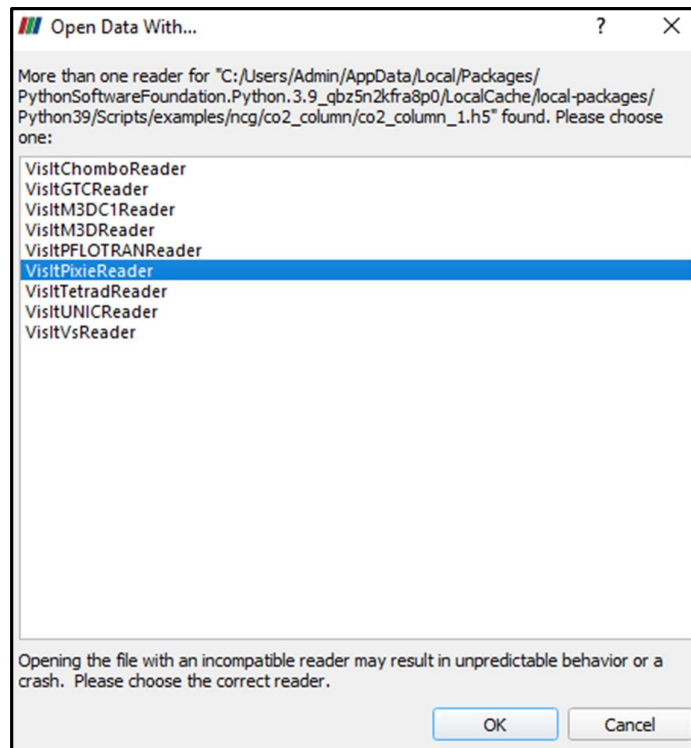


Figura 21 Inizializzazione ParaView.

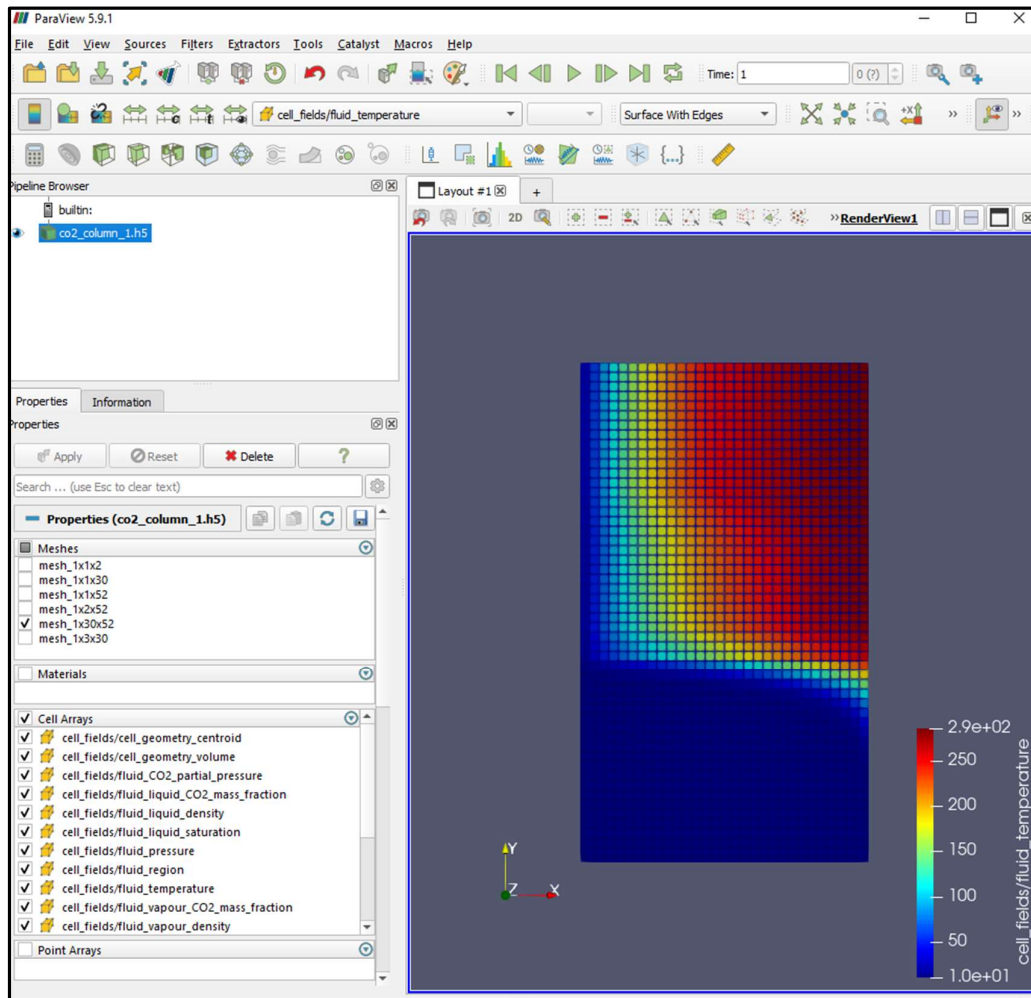


Figura 22 Visualizzazione "co2_column_1.h5" tramite ParaView.

Con riferimento alla Figura 22, il primo è il gruppo "Meshes", ed è stata selezionata la mesh più grande 1x30x52 (che era anche l'unica a fornire dei risultati). Il secondo gruppo è Cell Array e qui sono state spuntate tutte le caselle. Nella sezione "Rappresentation" è stato quindi selezionato "Surface With Edges" e a questo punto nella sezione Coloring è possibile scegliere quale categoria di output visualizzare, in questo caso, in maniera del tutto arbitraria, è stato selezionato "cell_fiels/fluid_temperature". In riferimento alla Figura 22, le 30 celle componenti il lato più corto del rettangolo sono in realtà la mesh del problema, e le 52 celle componenti il lato più lungo fanno invece riferimento al numero di step temporali. Risulta quindi più corretta la visualizzazione di Figura 23 ottenuta con una rotazione di 90° in senso orario attraverso i comandi appositi presenti nella barra degli strumenti. L'asse indicato come X, in questo caso positivo verso il basso, rappresenta quindi la profondità, o meglio, le celle della mesh dalla numero uno in superficie, alla trentesima cella. Sull'asse delle Y troviamo invece gli step temporali, dal primo partendo da sinistra al cinquantaduesimo a destra.

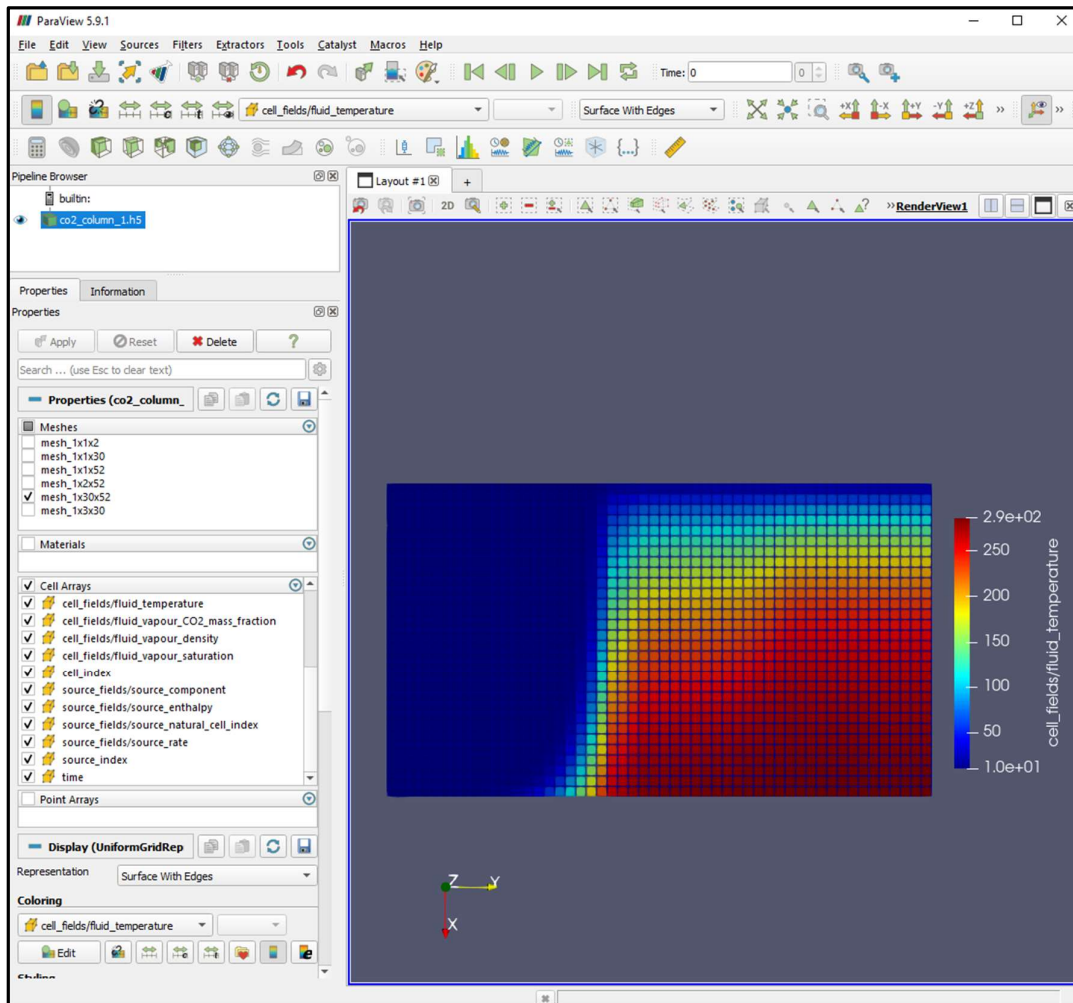


Figura 23 Modifica visualizzazione co2_column_1.h5 ParaView.

Utilizzando i comandi presenti in *ParaView* è possibile estrarre anche i valori di output dell'intero problema e anche di singole celle o gruppi di celle. Ad esempio, aprendo una nuova finestra di dialogo (cliccando sul simbolo “+” a destra di “Layout#1”), è possibile selezionare diverse visualizzazioni (Figura 24). Scegliendo “*SpreadSheet View*” e in seguito sostituendo nella sezione “*Attribute*” “*Point Data*” con “*Cell Data*” vengono forniti i valori di tutte le categorie di output per ogni cella.

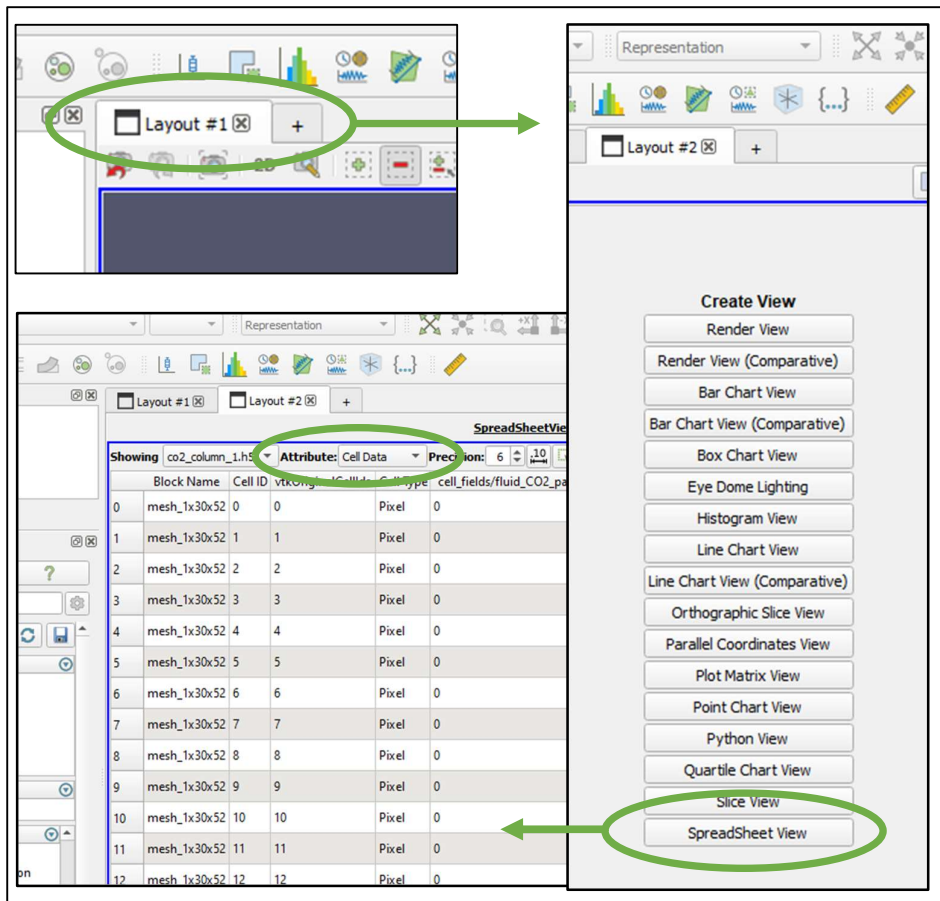


Figura 24 Visualizzazione valori outputs tabellati tramite ParaView.

Nel caso in cui si volessero visualizzare graficamente solo alcune celle o, come in questo caso, un singolo step temporale, è possibile selezionare le celle desiderate attraverso il comando “*Select Cells On*” e creare un nuovo elemento di estrazione attraverso il comando “*Extract Selection*” come rappresentato in Figura 25. Nella finestra “*Pipeline Browser*” verrà creata una nuova voce denominata “*ExtractSelection1*” che può essere attivata cliccando sull’occhio chiuso a sinistra e, spegnendo la voce “*co2_column_1.h5*”, è possibile visualizzare graficamente solo lo stato delle celle all’ultimo step temporale come riportato in Figura 26.

In modo analogo a quello spiegato precedentemente è inoltre possibile ottenere i dati tabellati di ogni cella relativi ad ogni step temporale.

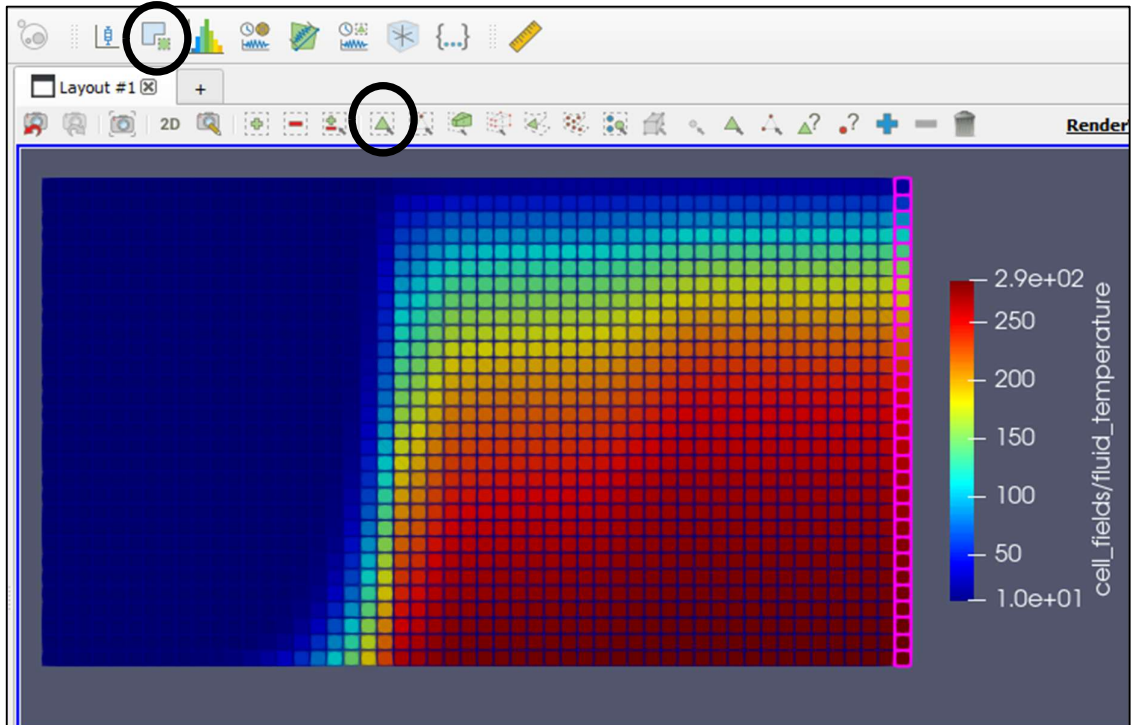


Figura 25 Selezione ultimo step temporale ParaView.

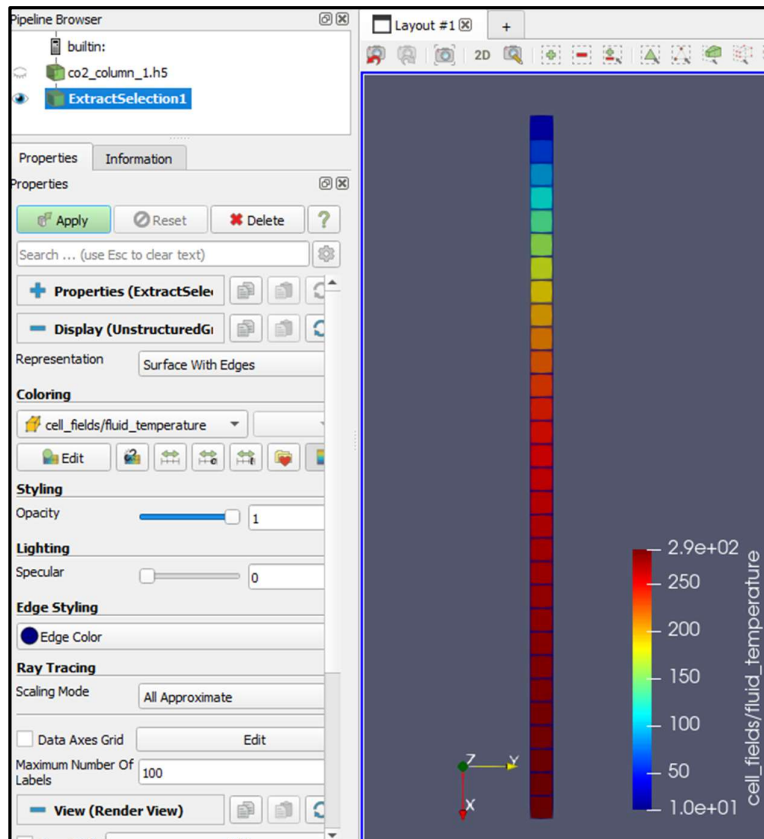


Figura 26 Visualizzazione ultimo step temporale ParaView.

6.2.3 Costruzione grafici dai dati di output

Con lo scopo di verificare la corretta installazione e configurazione di Waiwera, mediante gli output ottenuti utilizzando come file di input “*co2_column_1.json*”, sono stati costruiti dei grafici e ne è stata verificata l’uguaglianza con gli stessi presenti nella guida di Waiwera. La guida Waiwera fornisce inoltre degli appositi script Python [S33] sia per maneggiare i dati dell’output in formato HDF5 sia per l’output in formato YAML. Si è provveduto pertanto a creare copie funzionanti di tali script.

6.2.3.1 Script Python per file HDF5

Per gestire i file in formato HDF5 in post-elaborazione può essere utilizzata la libreria *h5py* [S31] di Python. Di seguito, è riportato lo script di esempio che utilizza *h5py* per l’apertura di un file HDF5 e la creazione di un grafico che descrive il valore della temperatura alle varie profondità all’ultimo step temporale nel caso di modello di colonna verticale descritto in precedenza.

```
import h5py
import matplotlib.pyplot as plt

out = h5py.File('model.h5')

index = out['cell_index'][:,0]
z = out['cell_fields']['cell_geometry_centroid'][index, 1]
T = out['cell_fields']['fluid_temperature'][-1, index]

plt.plot(T, z, '-.')
plt.xlabel('Temperature ( $^{\circ}$ C)')
plt.ylabel('elevation (m)')
plt.show()
```

Figura 27 Script Python per ottenere Grafico temperatura/profondità all’ultimo step temporale riportato dalla guida Waiwera partendo da un file di output HDF5.

Lo script di *Figura 27* è stato riscritto e leggermente modificato (grazie alla consultazione di una guida web [S34] per la costruzione di grafici con la libreria *Matplotlib*) con lo scopo di ottenere un grafico temperatura-profondità all’ultimo step temporale dal file di output “*co2_column_1.h5*”. Lo script compilato è riportato in *Figura 28*.

```
h5_1.txt - Blocco note di Windows
File Modifica Formato Visualizza ?
import h5py
import matplotlib.pyplot as plt

out = h5py.File('co2_column_1.h5')

index = out['cell_index'][:,0]
z = out['cell_fields']['cell_geometry_centroid'][index, 1]
T = out['cell_fields']['fluid_temperature'][-1, index]

plt.plot(T, z, marker = ".", color = 'g', markersize = 9)
plt.title("Andamento della Temperatura con l'elevazione")
plt.xlabel('T ( $^{\circ}$ C)')
plt.ylabel('Elevation (m)')
plt.ylim(-1000, 0)
plt.grid()
plt.show()
```

Figura 28 Script Python modificato per ottenere grafico temperatura-profondità all'ultimo step temporale da file di output "co2_column_1.h5".

Inoltre, per ottenere un maggior numero di verifiche, si è fatto riferimento a una pubblicazione [15] che riporta in un grafico l'andamento della pressione parziale di CO₂ con la profondità per diverse percentuali in massa di CO₂ iniettata (0,1%, 1% e 5%). A tale scopo sono stati eseguiti anche i file di input "co2_column_0.1.json" e "co2_column_5.json". Anche in questo caso si è fatto riferimento allo script di *Figura 27* effettuando delle modifiche e delle integrazioni, necessarie per la sovrapposizione dei tre grafici. In *Figura 29* è riportato lo script creato.

```
*h5_CO2_script.txt - Blocco note di Windows
File Modifica Formato Visualizza ?
import h5py
import matplotlib.pyplot as plt

out = h5py.File('co2_column_0.1.h5')

index = out['cell_index'][:,0]
z = out['cell_fields']['cell_geometry_centroid'][index, 1]
T = out['cell_fields']['fluid_CO2_partial_pressure'][-1, index]

plt.plot(T*10**-5, z, marker = ".", color = 'g', markersize = 8)
plt.text(3.0, -900, "0.1%", fontsize = 10)

out = h5py.File('co2_column_1.h5')

index = out['cell_index'][:,0]
z = out['cell_fields']['cell_geometry_centroid'][index, 1]
T = out['cell_fields']['fluid_CO2_partial_pressure'][-1, index]

plt.plot(T*10**-5, z, marker = ".", color = 'r', markersize = 8)
plt.text(12.0, -900, "1%", fontsize = 10)

out = h5py.File('co2_column_5.h5')

index = out['cell_index'][:,0]
z = out['cell_fields']['cell_geometry_centroid'][index, 1]
T = out['cell_fields']['fluid_CO2_partial_pressure'][-1, index]

plt.plot(T*10**-5, z, marker = ".", color = 'y', markersize = 8)
plt.text(22.0, -900, "5%", fontsize = 10)
plt.title("Andamento Pressione parziale CO2 con l'elevazione\n
per diverse percentuali in massa di CO2")
plt.xlabel('CO2 partial pressure (bar)')
plt.ylabel('Elevation (m)')
plt.ylim(-1000, 0)
plt.grid()
plt.show()
```

Figura 29 Script Python creato per ottenere il grafico dell'andamento della pressione parziale di CO₂ con la profondità. Da file "co2_column_0.1.h5", "co2_column_1.h5", "co2_column_5.h5" (ultimo step temporale).

6.2.3.2 Script Python per file YAML

Per gestire file in formato YAML sono disponibili anche in questo caso delle librerie, tra cui la libreria *Python PyYAML* [S32]. Per l'installazione di quest'ultima è sufficiente eseguire il seguente comando:

```
pip install pyyaml
```

Di seguito è riportato lo script Python che utilizza *PyYAML* per la lettura di un file YAML e fornisce in un grafico le dimensioni del passo temporale per una simulazione di stato stazionario.

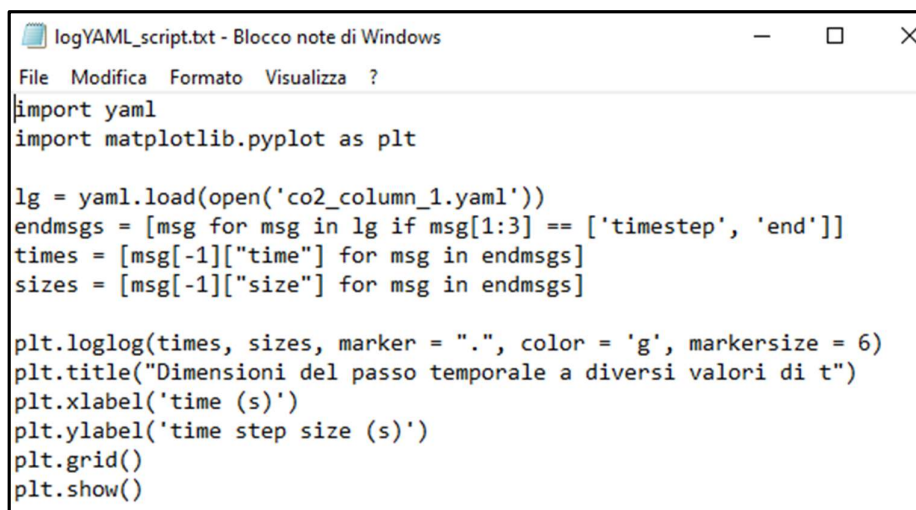
```
import yaml
import matplotlib.pyplot as plt

lg = yaml.load(file('model.yaml'))
endmsgs = [msg for msg in lg if msg[1:3] == ['timestep', 'end']]
times = [msg[-1]['time'] for msg in endmsgs]
sizes = [msg[-1]['size'] for msg in endmsgs]

plt.loglog(times, sizes, 'o-')
plt.xlabel('time (s)')
plt.ylabel('time step size (s)')
plt.show()
```

Figura 30 Script Python per ottenere Grafico dell'andamento della dimensione del passo temporale riportato dalla guida Waiwera partendo da un file di registro YAML .

In questo caso è stato necessario modificare lo script di riferimento riportato in *Figura 30*, questa volta con lo scopo di ottenere l'andamento della dimensione del passo temporale dal file di output “*co2_column_1.yaml*” e lo script così compilato è riportato in *Figura 31*.



```
logYAML_script.txt - Blocco note di Windows
File Modifica Formato Visualizza ?
import yaml
import matplotlib.pyplot as plt

lg = yaml.load(open('co2_column_1.yaml'))
endmsgs = [msg for msg in lg if msg[1:3] == ['timestep', 'end']]
times = [msg[-1]["time"] for msg in endmsgs]
sizes = [msg[-1]["size"] for msg in endmsgs]

plt.loglog(times, sizes, marker = ".", color = 'g', markersize = 6)
plt.title("Dimensioni del passo temporale a diversi valori di t")
plt.xlabel('time (s)')
plt.ylabel('time step size (s)')
plt.grid()
plt.show()
```

Figura 31 Script Python modificato per ottenere grafico dell'andamento della dimensione del passo temporale da file “*co2_column_1.yaml*”.

6.3 Risultati

6.3.1 Risultati numerici

Il manuale di Waiwera fornisce un'immagine di output relativo all'input "co2_column_1.json" visualizzato con *Silx* (un altro programma per l'apertura dei file in formato HDF5) che è riportata in *Figura 32*.

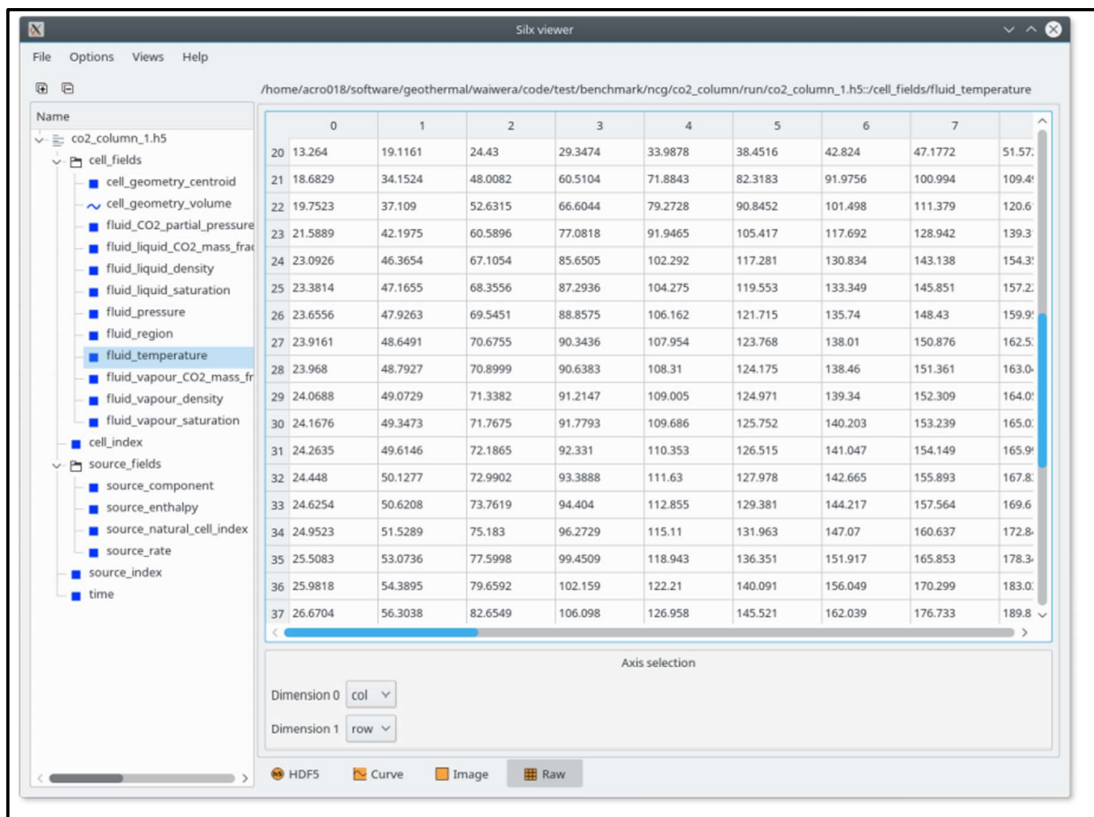


Figura 32 Visualizzazione output tramite Silx da manuale Waiwera.

Quest'ultima serie di dati è stata confrontata con l'output ottenuto e visualizzato tramite HDFview (*Figura 33*). Come si può osservare le soluzioni risultano coincidere, questo presuppone che Waiwera sia stato correttamente installato e che gli outputs siano validi.

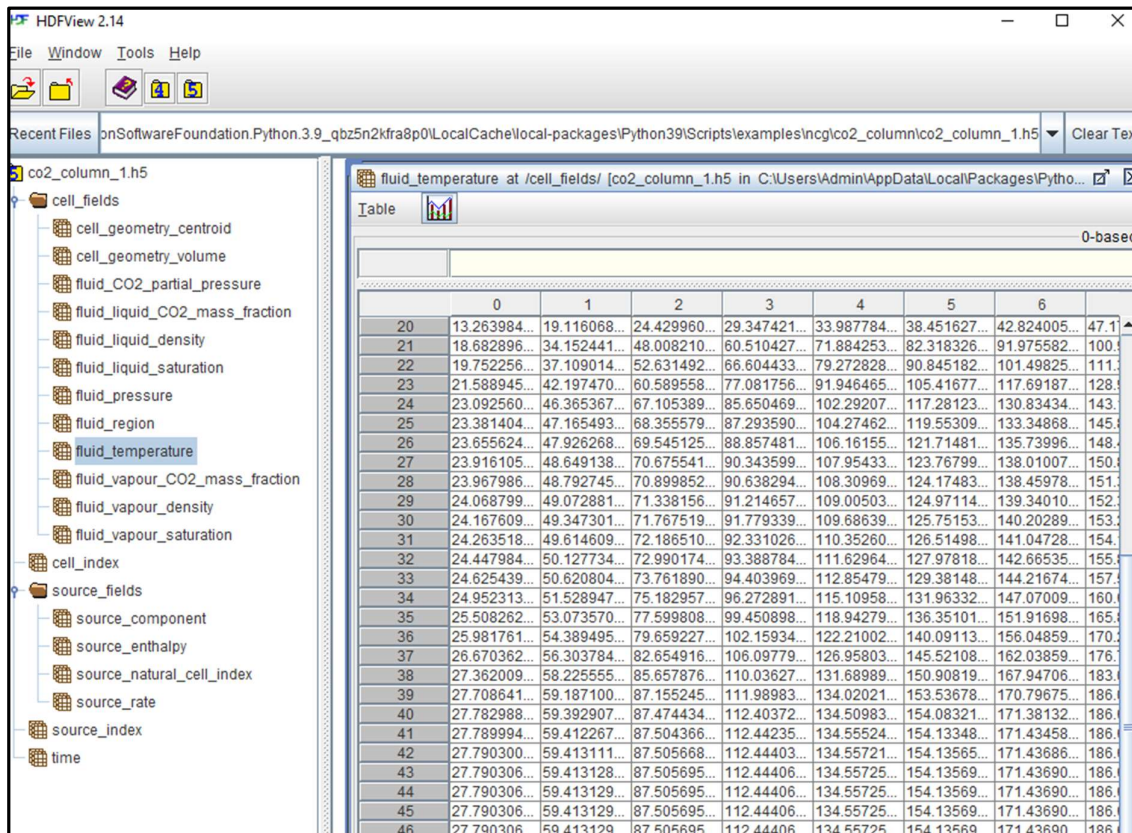


Figura 33 Visualizzazione output "co2_column_1.h5".

6.3.2 Risultati grafici

Per eseguire gli script in linguaggio Python, creati e riportati nel paragrafo 6.2.3, è stato necessario eseguire il seguente comando:

```
python
```

In questo modo viene avviato l'interprete Python in modalità iterativa e per tanto è stato possibile eseguire gli script in linguaggio Python creati. In *Figura 34* è mostrato il grafico che si dovrebbe ottenere dallo script di *Figura 27* come riportato dalla guida Waiwera mentre in *Figura 35* è stato riportato il grafico effettivamente ottenuto dallo script di *Figura 28* creato.

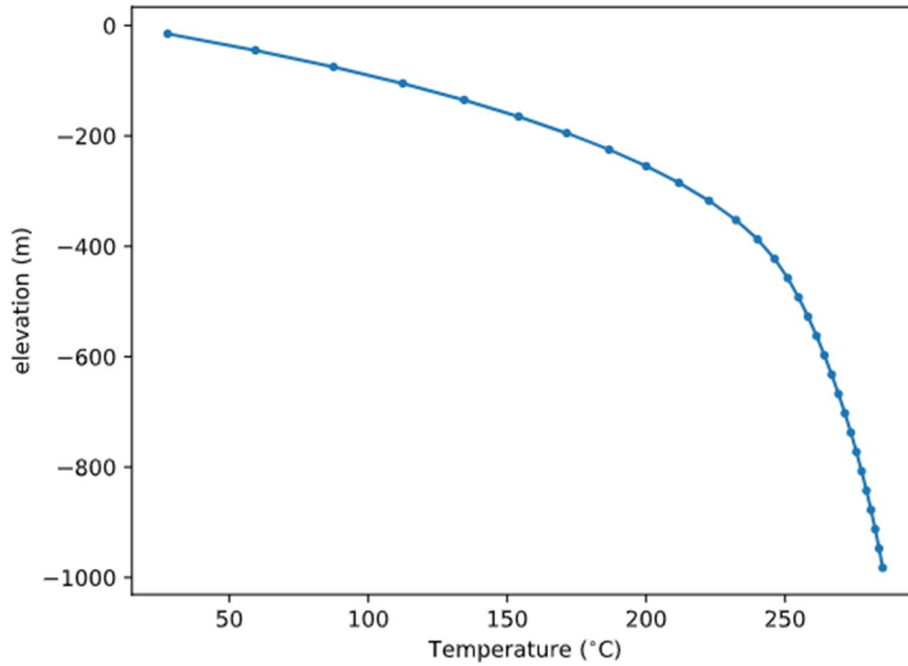


Figura 34 Grafico temperatura/profondità ottenuto da script Python e riportato nella guida Waiwera relativo all'ultimo step temporale del file di output "co2_column_1.h5".

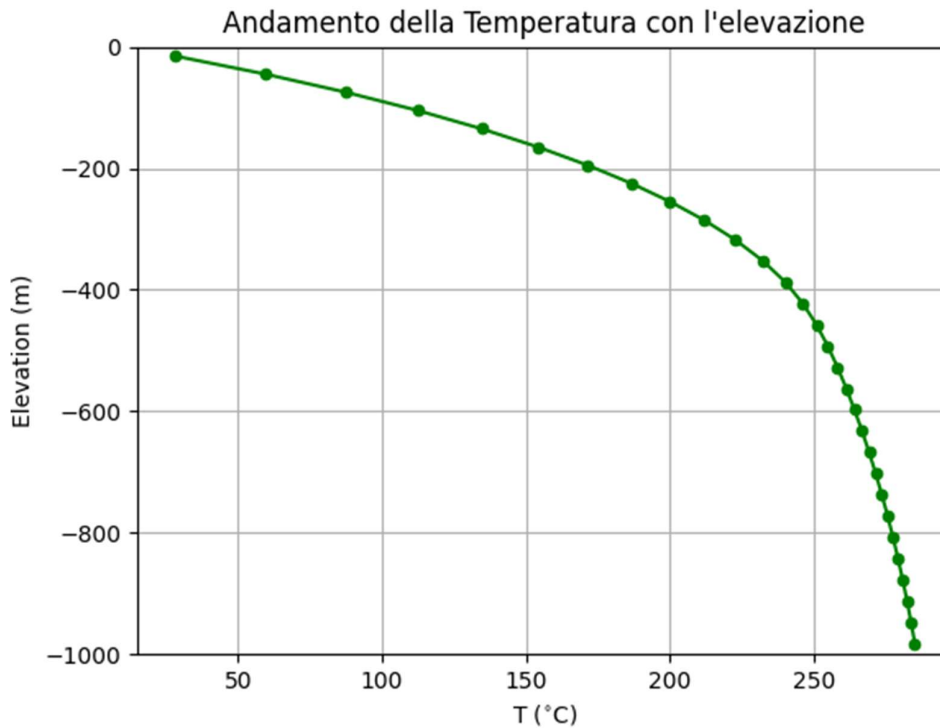


Figura 35 Grafico temperatura/profondità ottenuto da script Python modificato relativo all'ultimo step temporale del file di output "co2_column_1.h5".

Da una semplice comparazione visiva, si evince che il grafico di *Figura 34* e quello di *Figura 35* coincidono.

Come scritto precedentemente, si è fatto riferimento anche ad una pubblicazione contenente risultati inerenti al problema “*co2_column*” e pubblicati nel *proceedings* di un congresso [15]. In particolare, sono presenti in un unico grafico (*Figura 36*) i tre andamenti della pressione parziale di CO₂ con la profondità per tre diverse percentuali in massa di CO₂ iniettata (0,1%, 1% e 5%), in riferimento all’ultimo step temporale. I risultati che vengono riportati riguardano anche altri due software oltre a Waiwera, in particolare, AUTOUGH2 (una versione di TOUGH2 ideata dall’Università di Auckland) e MULKOM, pertanto, bisognerebbe fare riferimento alla linea continua blu per il confronto dei risultati.

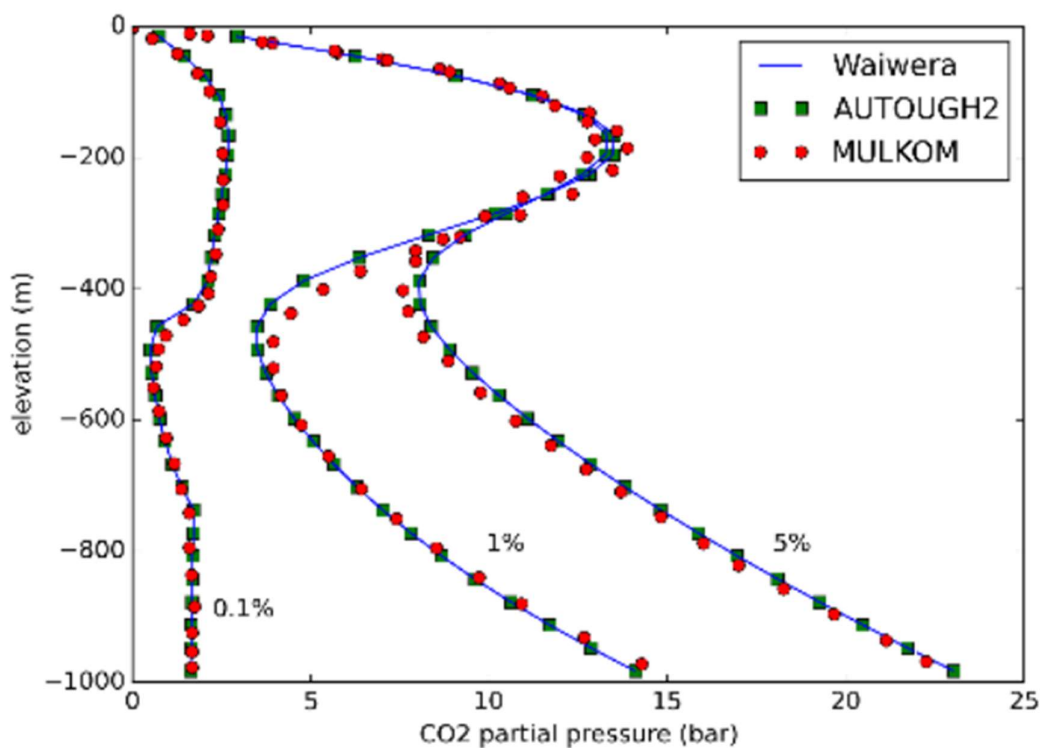


Figura 36 Grafico di riferimento per l'andamento pressione parziale di CO₂ con la profondità. Da file “*co2_column_0.1.h5*”, “*co2_column_1.h5*”, “*co2_column_5.h5*” (ultimo step temporale.)

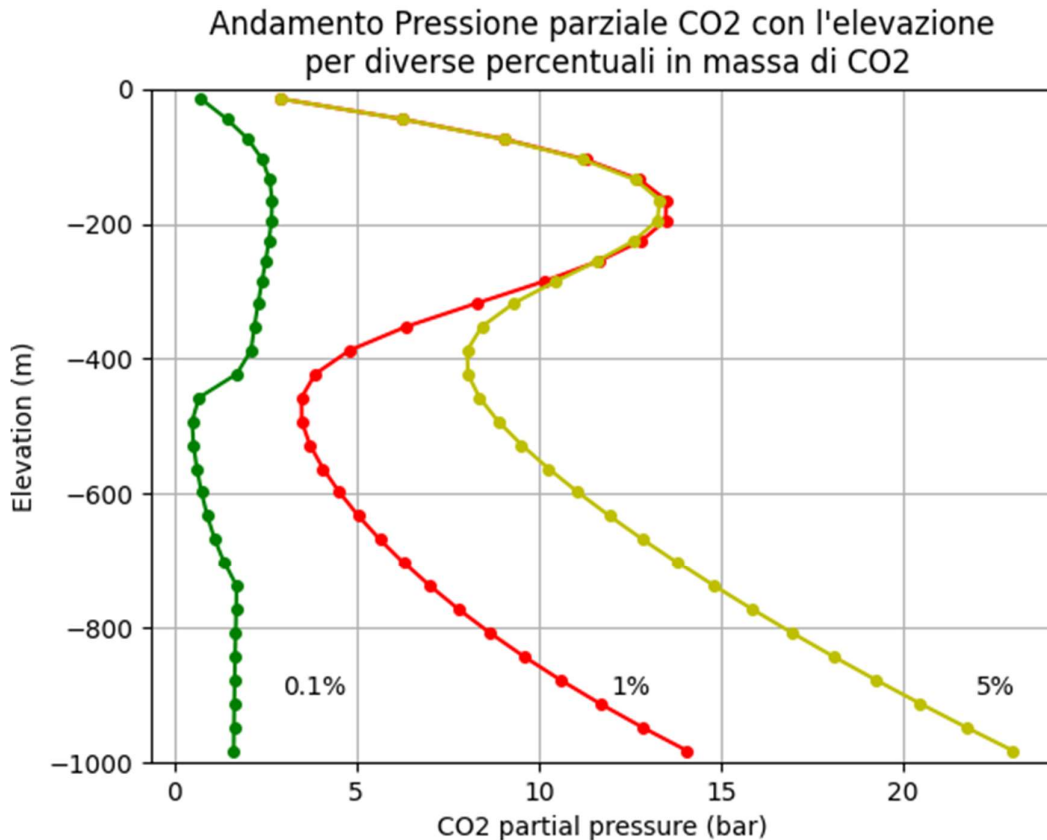


Figura 37 Grafico dell'andamento della pressione parziale di CO₂ con la profondità ottenuto da file "co2_column_0.1.h5", "co2_column_1.h5", "co2_column_5.h5" (ultimo step temporale).

Anche in questo caso, utilizzando lo script creato di *Figura 29*, si è ottenuto un riscontro positivo (*Figura 37*) in quanto i tre andamenti risultano coincidenti con quelli di *Figura 36*.

Per quanto riguarda invece l'elaborazione del file di output in formato YAML, in *Figura 38* è riportato il grafico che si dovrebbe ottenere dallo script mostrato in *Figura 30*, riportato nella guida di Waiwera, mentre in *Figura 39* è riportato il grafico effettivamente ottenuto dallo script creato e mostrato in *Figura 31*. Anche in questo caso si è ottenuto un riscontro positivo.

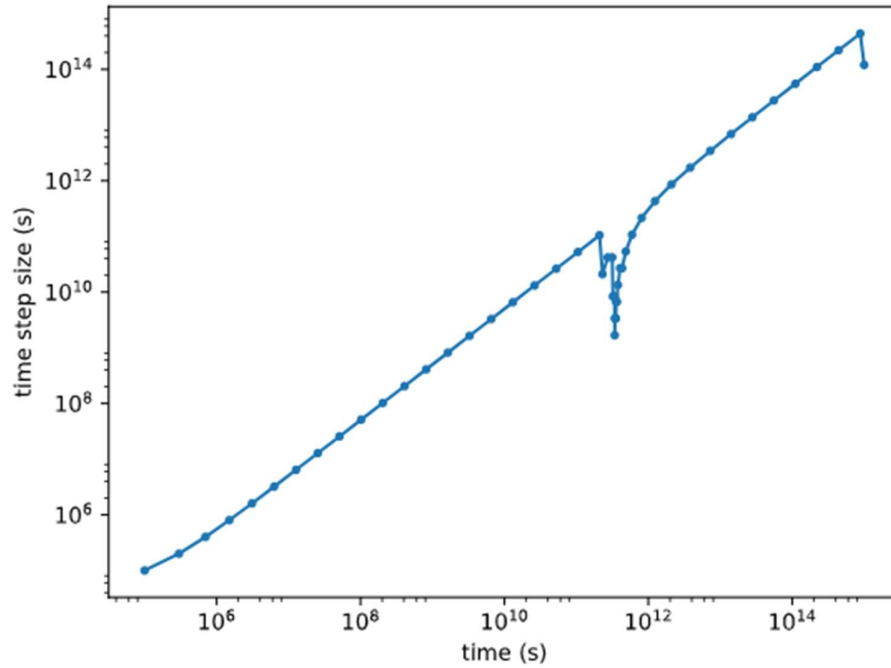


Figura 38 Grafico dimensioni del passo temporale ottenuto da script Python e riportato dalla guida Waiwera. Da file di registro "co2_column_1.yaml".

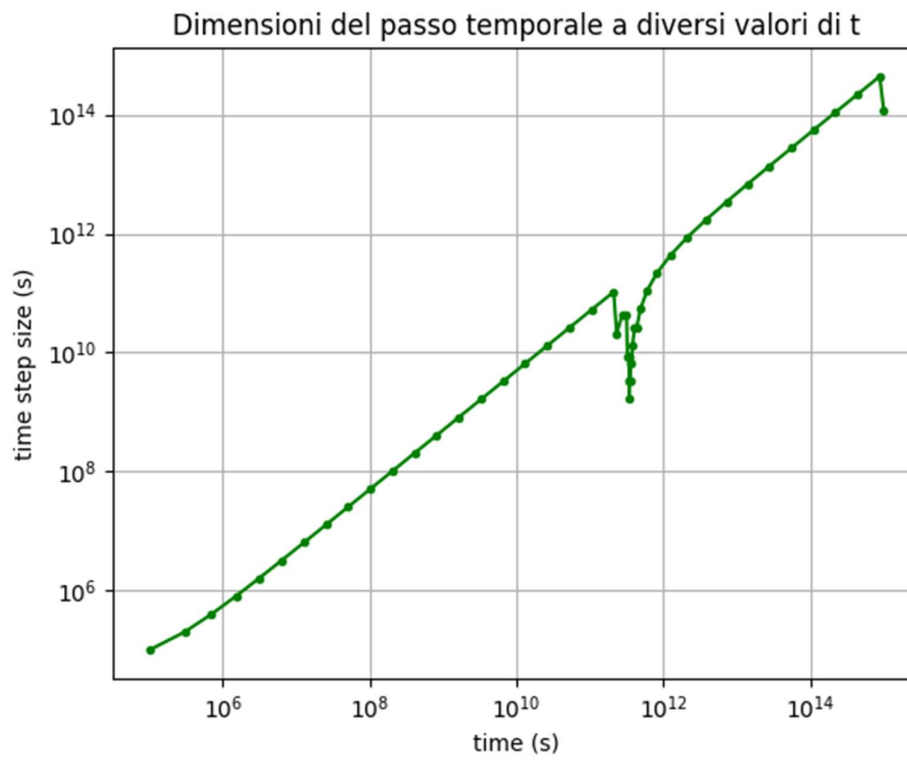


Figura 39 Grafico ottenuto da Script Python con andamento della dimensione del passo temporale da file "co2_column_1.yaml".

Anche se con una procedura solamente qualitativa, i risultati ottenuti, dimostrano che il lavoro di installazione e configurazione di Waiwera è riuscito con successo. Come pure si riconferma l'efficienza di questo nuovo importante simulatore.

6.4 Simulazione del caso studio con TOUGH2-EWASG

Nell'ottica di impostare un lavoro di confronto tra Waiwera e altri simulatori, il modello presentato da O'Sullivan [16], e utilizzato per testare l'installazione di Waiwera, è stato anche implementato in TOUGH2 utilizzando l'EOS 14 (EWASG, Battistelli et al., 1987 [35]) nella versione iTOUGH2 [36] in uso al dipartimento DICAM dell'università di Bologna.

A tale scopo, è stata verificata la capacità di riprodurre con TOUGH2-EWASG il grafico riportato da Croucher et al. (2020) [9] (*Figura 40*) che si riferisce ad una simulazione riguardante lo stesso caso studio trattato nel presente elaborato. Si tratta quindi di una colonna di 1000 metri con un primo materiale roccioso impermeabile (0.5 mD) che si estende da 0 a -300 m di profondità e un secondo, permeabile (20 mD), che si estende da -300 a -1000 m di profondità. Le curve di pressione capillare utilizzate nel lavoro di O'Sullivan sono la Corey, SGB e X-curve e, seguendo le indicazioni di Croucher (2018) [15], è stata utilizzata una funzione di permeabilità relativa (REL_P) lineare con i seguenti parametri: RP(1)=0, RP(2)=0, RP(3)=1, RP(4)=1. Il dominio è stato discretizzato con una griglia di 100 blocchi di spessore 10 m. Le condizioni al contorno sono P=1.013 kPa (pressione atmosferica) alla sommità della colonna mentre le condizioni iniziali sono Saturazione liquido =1, T iniziale 10 °C. Viene simulata l'iniezione dal fondo di 2 kg/s/km² di acqua ad entalpia di 1300 kJ/kg. Per la descrizione puntuale della struttura del file di input e della scelta dei valori associati ai vari parametri ivi presenti, si rimanda al manuale di TOUGH2 [10]. Per verificare la capacità di riprodurre il grafico di *Figura 40* sono stati quindi simulati due scenari di iniezione di acqua contenente CO₂ a concentrazioni pari al 5% e al 20%. In *Figura 41*, creata tramite Excel, sono riportati i due andamenti ottenuti.

Da un punto di vista puramente qualitativo, come si vede dalle due figure, la forma delle curve per la pressione parziale della CO₂ alle varie concentrazioni di iniezione vengono riprodotte abbastanza fedelmente, pur con qualche scostamento.

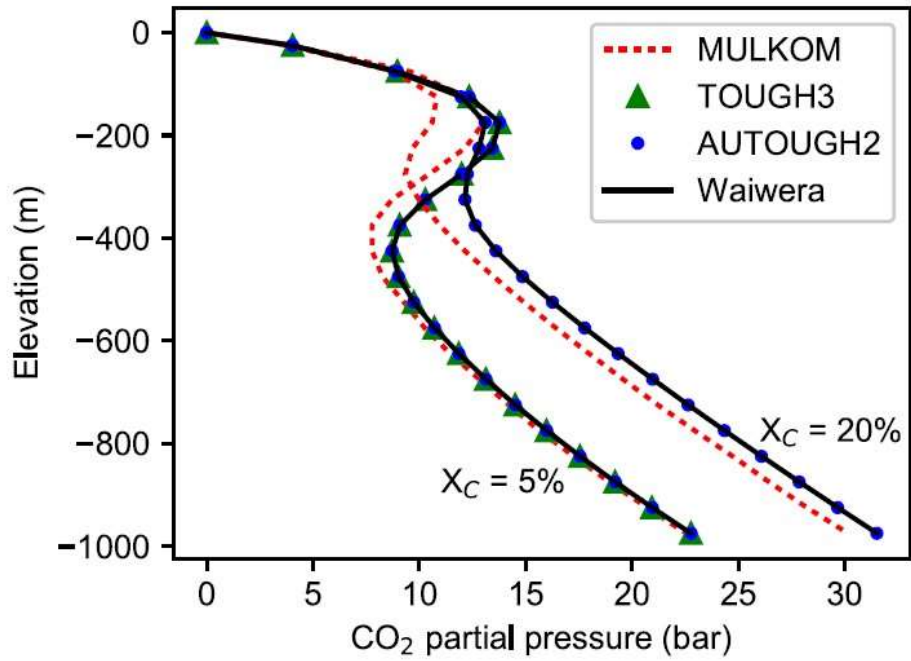


Figura 40 Andamento pressione parziale di CO₂ con la profondità, Croucher 2020 [9].

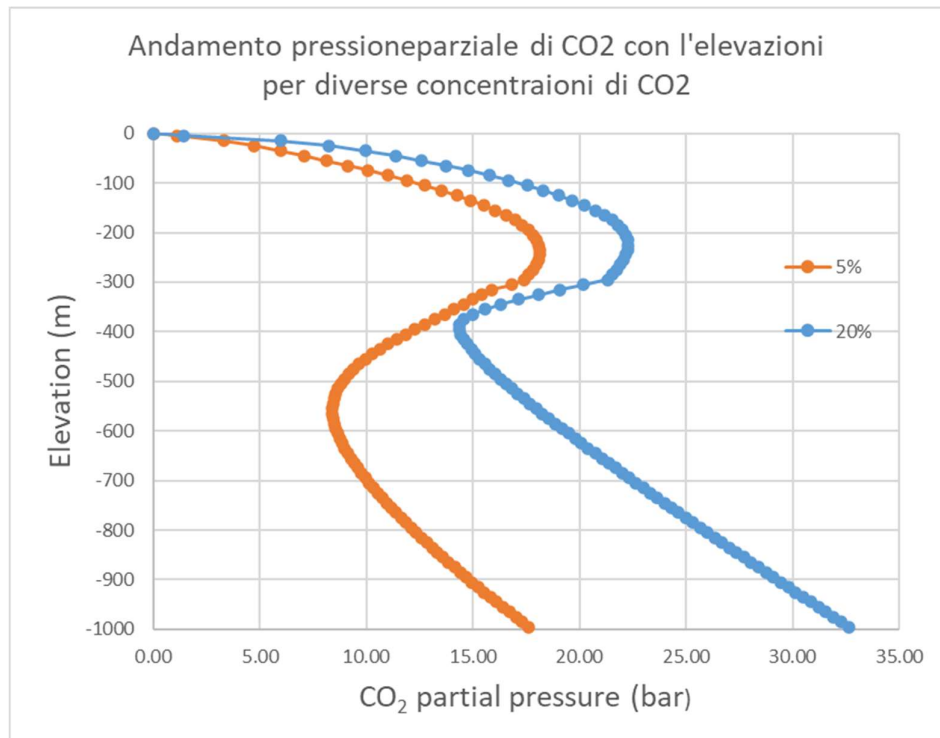


Figura 41 Andamento pressione parziale di CO₂ con la profondità ottenuto con TOUGH2-EWASG.

Una scelta più accurata di alcuni parametri dell'input di TOUGH2-EWASG potrebbe sicuramente migliorare la corrispondenza dei risultati ma, visto il grado complessità e la conseguente quantità di tempo necessaria, tale lavoro rimane confinato nell'ambito di future attività di approfondimento.

Come esempio, nel seguito è riportato l'input integrale di TOUGH2-EWASG utilizzato per la simulazione del caso della miscela acqua-CO₂ al 5%.

```
*1D CO2 injection problem*
ROCKS----1---*---2---*---3---*---4---*---5---*---6---*---7---*---8
ATMO1 3 2600. 0.99 1.0E-19 1.0E-19 1.0E-12 2.30 99920.

1 .00 .00 1. 1.
7 0.44380 8.01E-2 5.792e-07 1.E5 1.
CAPR2 2 2600.e00 0.124.9346E-164.9346E-164.9346E-16 2.51 920.

1 .00 .00 1. 1.
7 0.44380 8.01E-2 5.792e-07 5.E7 1.
SHAL2 2 2600.e00 .121.9738E-141.9738E-141.9738E-14 2.51 920.

1 .00 .00 1. 1.
7 0.44380 8.01E-2 5.792e-07 5.E5 1.

.....IE(4)=1 Andersen et al.;
.....IE(10)=0 VPL off
.....IE(11)=1 tube model for permred
.....IE(15)=3 Phillips et al. brine saturated enthalpy;
.....IE(16)=2 CHOOSE CO2;
SELEC---2---3---4---5---6---7---8---9---10---11---12---13---14---15---16
1 1 1 0 1 0 3 2
3.

.....SPECIFY THREE COMPONENTS (WATER, NAACL, CO2); SOLVE ENERGY EQUATION
MULTI---1---*---2---*---3---*---4---*---5---*---6---*---7---*---8
3 4 3 6
START
PARAM---1---*-123456789012345678901234---*---5---*---6---*---7---*---8
4 49999 9999100030 0020 00300 3
.0 3.154E+13 -1. 9.8065
1.E-6 120.0
1.E-6 1. 1.e-8 1.e+4
10.0 .0000 0.000001 1.0E5
..... Pcap from Yucca M. welded tuff data
MOMOP---1---*---2---*---3---*---4---*---5---*---6---*---7---*---8
2
RPCAP---1---*---2---*---3---*---4---*---5---*---6---*---7---*---8
3 .30 .05 1. 1.
7 0.44380 8.01E-2 5.792e-07 5.E7 1.
TIMES---1---*---2---*---3---*---4---*---5---*---6---*---7---*---8
1 10 3.154E12
3.154E10

GENER5
CU1 1inj 1 COM1 1.95 1300.e3
CU1 1inj 2 COM3 0.05 1300.e3
CU1 1inj 3 COM2 2.E-6 1300.e3

ELEME
A11 2 11.0000E+550.0000E+00 +5.000E+02+5.000E+02-5.000E-02 1.0000E+041.0000E+041.0000E+041.0000E+040.0000E+001.0000E+06
A11 1 21.0000E+070.0000E+00 +5.000E+02+5.000E+02-5.000E+00 1.0000E+041.0000E+041.0000E+041.0000E+040.0000E+001.0000E+06
A21 1 21.0000E+070.0000E+00 +5.000E+02+5.000E+02-1.500E+01 1.0000E+041.0000E+041.0000E+041.0000E+040.0000E+000.0000E+00
```


INCONS - INITIAL CONDITIONS FOR 101 ELEMENTS AT TIME 0.315400E+14

A11 2 0.99000000E+00
0.1013000000000E+06 0.1000000000000E-05 0.1000000000000E-05 0.1000000000000E+02

A11 1 0.12000000E+00
0.1025395549313E+06 0.1020407122034E-05 0.5946442390928E-83 0.1010041778632E+02

A21 1 0.12000000E+00
0.2409207581980E+06 0.1020407122034E-05 0.3072308418081E-83 0.1947542601998E+02

A31 1 0.12000000E+00
0.3714917341111E+06 0.1020407122034E-05 0.1574552767719E-83 0.2861647760244E+02

A41 1 0.12000000E+00
0.4964283798498E+06 0.1020407122034E-05 0.8009844504601E-84 0.3752059838330E+02

A51 1 0.12000000E+00
0.6171073269414E+06 0.1020407122034E-05 0.4046497026372E-84 0.4618668179943E+02

A61 1 0.12000000E+00
0.7344418107371E+06 0.1020407122034E-05 0.2030889565525E-84 0.5461499816027E+02

A71 1 0.12000000E+00
0.8490619017399E+06 0.1020407122034E-05 0.1012915939067E-84 0.6280686526425E+02

A81 1 0.12000000E+00
0.9614172436194E+06 0.1020407122034E-05 0.5021589325190E-85 0.7076441522183E+02

A91 1 0.12000000E+00
0.1071838403901E+07 0.1020407122034E-05 0.2474959033448E-85 0.7849042211051E+02

AA1 1 0.12000000E+00
0.1180574890658E+07 0.1020407122034E-05 0.1212868235664E-85 0.8598817010196E+02

AB1 1 0.12000000E+00
0.1287819475480E+07 0.1020407122034E-05 0.5910473048274E-86 0.9326134975765E+02

AC1 1 0.12000000E+00
0.1393724216706E+07 0.1020407122034E-05 0.2864343664248E-86 0.1003139747298E+03

AD1 1 0.12000000E+00
0.1498411302404E+07 0.1020407122034E-05 0.1380519728060E-86 0.1071503137469E+03

AE1 1 0.12000000E+00
0.1601980569181E+07 0.1020407122034E-05 0.6617376118632E-87 0.1137748343651E+03

AF1 1 0.12000000E+00
0.1704514832092E+07 0.1020407122034E-05 0.3154698674698E-87 0.1201921559817E+03

AG1 1 0.12000000E+00
0.1806083738739E+07 0.1020407122034E-05 0.1495736343670E-87 0.1264070102800E+03

AH1 1 0.12000000E+00
0.1906746606737E+07 0.1020407122034E-05 0.7052895523760E-88 0.1324242077324E+03

AI1 1 0.12000000E+00
0.2006554547024E+07 0.1020407122034E-05 0.3307347212496E-88 0.1382486091103E+03

AJ1 1 0.12000000E+00
0.2105552076359E+07 0.1020407122034E-05 0.1542310561945E-88 0.1438851011817E+03

AK1 1 0.12000000E+00
0.2203778358411E+07 0.1020407122034E-05 0.7151857199781E-89 0.1493385759465E+03

AL1 1 0.12000000E+00
0.2301268170621E+07 0.1020407122034E-05 0.3297553735370E-89 0.1546139128837E+03

AM1 1 0.12000000E+00
0.2398052665744E+07 0.1020407122034E-05 0.1511674073036E-89 0.1597159637845E+03

AN1 1 0.12000000E+00
0.2494159977662E+07 0.1020407122034E-05 0.6889396917283E-90 0.1646495398145E+03

AO1 1 0.12000000E+00
0.2589615707659E+07 0.1020407122034E-05 0.3121196035837E-90 0.1694194005115E+03

AP1 1 0.12000000E+00
0.2684443317945E+07 0.1020407122034E-05 0.1405511462186E-90 0.1740302444652E+03

AQ1 1 0.12000000E+00
0.2778664452479E+07 0.1020407122034E-05 0.6290348840140E-91 0.1784867014686E+03

AR1 1 0.12000000E+00
0.2872299200279E+07 0.1020407122034E-05 0.2797643774911E-91 0.1827933259554E+03

AS1 1 0.12000000E+00
0.2965366312851E+07 0.1020407122034E-05 0.1236330840116E-91 0.1869545915665E+03

AT1 1 0.12000000E+00
0.3057883384722E+07 0.1020407122034E-05 0.5428074081481E-92 0.1909748867075E+03

AU1 1 0.12000000E+00
0.3149867004107E+07 0.1020407122034E-05 0.2367372864583E-92 0.1948585109759E+03

AV1 1 0.12000000E+00
0.3238319179993E+07 0.1020407122034E-05 0.1025499141113E-92 0.1986096815904E+03

AW1 1 0.12000000E+00
0.3323338887043E+07 0.1020407122034E-05 0.4411515430288E-93 0.2022325201260E+03

AX1 1 0.12000000E+00
0.3407947121842E+07 0.1020407122034E-05 0.1884331048258E-93 0.2057310429428E+03
AY1 1 0.12000000E+00
0.3492151476486E+07 0.1020407122034E-05 0.7990462400253E-94 0.2091091701652E+03
AZ1 1 0.12000000E+00
0.3575959592390E+07 0.1020407122034E-05 0.3363233292835E-94 0.2123707251033E+03
B11 1 0.12000000E+00
0.3659379146230E+07 0.1020407122034E-05 0.1404863518747E-94 0.2155194340702E+03
B21 1 0.12000000E+00
0.3742417837167E+07 0.1020407122034E-05 0.5822661228995E-95 0.2185589265437E+03
B31 1 0.12000000E+00
0.3825083375198E+07 0.1020407122034E-05 0.2394052883907E-95 0.2214927356284E+03
B41 1 0.12000000E+00
0.3907383470507E+07 0.1020407122034E-05 0.9762963360566E-96 0.2243242987754E+03
B51 1 0.12000000E+00
0.3989325823699E+07 0.1020407122034E-05 0.3947956122841E-96 0.2270569587260E+03
B61 1 0.12000000E+00
0.4070918116829E+07 0.1020407122034E-05 0.1582736909131E-96 0.2296939646453E+03
B71 1 0.12000000E+00
0.4152168005135E+07 0.1020407122034E-05 0.6289118890927E-97 0.2322384734179E+03
B81 1 0.12000000E+00
0.4233083109402E+07 0.1020407122034E-05 0.2476339003240E-97 0.2346935510803E+03
B91 1 0.12000000E+00
0.4313671008903E+07 0.1020407122034E-05 0.9659611689524E-98 0.2370621743674E+03
BA1 1 0.12000000E+00
0.4393939234853E+07 0.1020407122034E-05 0.3731847271792E-98 0.2393472323516E+03
BB1 1 0.12000000E+00
0.4473895264343E+07 0.1020407122034E-05 0.1427522856574E-98 0.2415515281592E+03
BC1 1 0.12000000E+00
0.4553546514692E+07 0.1020407122034E-05 0.5405209025020E-99 0.2436777807461E+03
BD1 1 0.12000000E+00
0.4632900338212E+07 0.1020407122034E-05 0.2025265524800E-99 0.2457286267192E+03
BE1 1 0.12000000E+00
0.4711964017323E+07 0.1020407122034E-05 0.7506811029549-100 0.2477066221924E+03
BF1 1 0.12000000E+00
0.4790744760024E+07 0.1020407122034E-05 0.2751636843697-100 0.2496142446650E+03
BG1 1 0.12000000E+00
0.4869249695672E+07 0.1020407122034E-05 0.9971041401914-101 0.2514538949132E+03
BH1 1 0.12000000E+00
0.4947485871068E+07 0.1020407122034E-05 0.3570660490986-101 0.2532278988876E+03
BI1 1 0.12000000E+00
0.5025460246820E+07 0.1020407122034E-05 0.1263144747332-101 0.2549385096074E+03
BJ1 1 0.12000000E+00
0.5103179693975E+07 0.1020407122034E-05 0.4412496143506-102 0.2565879090474E+03
BK1 1 0.12000000E+00
0.5180650990914E+07 0.1020407122034E-05 0.1521473811403-102 0.2581782100103E+03
BL1 1 0.12000000E+00
0.5257880820482E+07 0.1020407122034E-05 0.5176153660614-103 0.2597114579810E+03
BM1 1 0.12000000E+00
0.5334875767361E+07 0.1020407122034E-05 0.1736668623931-103 0.2611896329589E+03
BN1 1 0.12000000E+00
0.5411642315666E+07 0.1020407122034E-05 0.5743659476030-104 0.2626146512639E+03
BO1 1 0.12000000E+00
0.5488186846761E+07 0.1020407122034E-05 0.1871573201623-104 0.2639883673148E+03
BP1 1 0.12000000E+00
0.5564515637292E+07 0.1020407122034E-05 0.6005436534919-105 0.2653125753769E+03
BQ1 1 0.12000000E+00
0.5640634857420E+07 0.1020407122034E-05 0.1896546136356-105 0.2665890112769E+03
BR1 1 0.12000000E+00
0.5716550569262E+07 0.1020407122034E-05 0.5891307823311-106 0.2678193540849E+03
BS1 1 0.12000000E+00
0.5792268725520E+07 0.1020407122034E-05 0.1798970814969-106 0.2690052277601E+03
BT1 1 0.12000000E+00
0.5867795168305E+07 0.1020407122034E-05 0.5396600946125-107 0.2701482027627E+03
BU1 1 0.12000000E+00
0.5943135628148E+07 0.1020407122034E-05 0.1589292187449-107 0.2712497976276E+03
BV1 1 0.12000000E+00

0.6018295723183E+07 0.1020407122034E-05 0.4591560871605-108 0.2723114805034E+03
BW1 1 0.12000000E+00
0.6093280958516E+07 0.1020407122034E-05 0.1300339869722-108 0.2733346706534E+03
BX1 1 0.12000000E+00
0.6168096725752E+07 0.1020407122034E-05 0.3606943785987-109 0.2743207399210E+03
BY1 1 0.12000000E+00
0.6242748302700E+07 0.1020407122034E-05 0.9791088530456-110 0.2752710141579E+03
BZ1 1 0.12000000E+00
0.6317240853224E+07 0.1020407122034E-05 0.2598543885247-110 0.2761867746172E+03
C11 1 0.12000000E+00
0.6391579427255E+07 0.1020407122034E-05 0.6736083668114-111 0.2770692593096E+03
C21 1 0.12000000E+00
0.6465768960957E+07 0.1020407122034E-05 0.1703742303513-111 0.2779196643257E+03
C31 1 0.12000000E+00
0.6539814277020E+07 0.1020407122034E-05 0.4199780925546-112 0.2787391451225E+03
C41 1 0.12000000E+00
0.6613720085108E+07 0.1020407122034E-05 0.1007733720866-112 0.2795288177770E+03
C51 1 0.12000000E+00
0.6687490982420E+07 0.1020407122034E-05 0.2350661837276-113 0.2802897602052E+03
C61 1 0.12000000E+00
0.6761131454393E+07 0.1020407122034E-05 0.5322827574305-114 0.2810230133493E+03
C71 1 0.12000000E+00
0.6834645875511E+07 0.1020407122034E-05 0.1168242750569-114 0.2817295823323E+03
C81 1 0.12000000E+00
0.6908038510228E+07 0.1020407122034E-05 0.2481042979101-115 0.2824104375818E+03
C91 1 0.12000000E+00
0.6981313514004E+07 0.1020407122034E-05 0.5089223424647-116 0.2830665159221E+03
CA1 1 0.12000000E+00
0.7054474934429E+07 0.1020407122034E-05 0.1006269712998-116 0.2836987216378E+03
CB1 1 0.12000000E+00
0.7127526712452E+07 0.1020407122034E-05 0.1913663562130-117 0.2843079275069E+03
CC1 1 0.12000000E+00
0.7200472683687E+07 0.1020407122034E-05 0.3491782855650-118 0.2848949758058E+03
CD1 1 0.12000000E+00
0.7273316579800E+07 0.1020407122034E-05 0.6096554738820-119 0.2854606792865E+03
CE1 1 0.12000000E+00
0.7346062029977E+07 0.1020407122034E-05 0.1015466340439-119 0.2860058221265E+03
CF1 1 0.12000000E+00
0.7418712562451E+07 0.1020407122034E-05 0.1608112800349-120 0.2865311608519E+03
CG1 1 0.12000000E+00
0.7491271606095E+07 0.1020407122034E-05 0.2411957521612-121 0.2870374252352E+03
CH1 1 0.12000000E+00
0.7563742492066E+07 0.1020407122034E-05 0.3411328109915-122 0.2875253191668E+03
CI1 1 0.12000000E+00
0.7636128455503E+07 0.1020407122034E-05 0.4526789639003-123 0.2879955215026E+03
CJ1 1 0.12000000E+00
0.7708432637267E+07 0.1020407122034E-05 0.5603085548147-124 0.2884486868875E+03
CK1 1 0.12000000E+00
0.7780658085715E+07 0.1020407122034E-05 0.6424562330068-125 0.2888854465545E+03
CL1 1 0.12000000E+00
0.7852807758512E+07 0.1020407122034E-05 0.6768169934242-126 0.2893064091017E+03
CM1 1 0.12000000E+00
0.7924884524466E+07 0.1020407122034E-05 0.6486087797648-127 0.2897121612468E+03
CN1 1 0.12000000E+00
0.7996891165389E+07 0.1020407122034E-05 0.5584983050842-128 0.2901032685590E+03
CO1 1 0.12000000E+00
0.8068830377972E+07 0.1020407122034E-05 0.4253907096965-129 0.2904802761699E+03
CP1 1 0.12000000E+00
0.8140704775672E+07 0.1020407122034E-05 0.2807801185254-130 0.2908437094642E+03
CQ1 1 0.12000000E+00
0.8212516890617E+07 0.1020407122034E-05 0.1561578924452-131 0.2911940747484E+03
CR1 1 0.12000000E+00
0.8284269175500E+07 0.1020407122034E-05 0.7025783076981-133 0.2915318599014E+03
CS1 1 0.12000000E+00
0.8355964005486E+07 0.1020407122034E-05 0.2397578138067-134 0.2918575350040E+03
CT1 1 0.12000000E+00
0.8427603680114E+07 0.1020407122034E-05 0.5516846272108-136 0.2921715529509E+03

CU1 1 0.1200000E+00
0.8499190425188E+07 0.1020407122034E-05 0.6420293090597-138 0.2924743500428E+03

ENDCY---1---*---2---*---3---*---4---*---5---*---6---*---7---*---8

7 Conclusioni

Lo scopo finale di questa tesi mirava ad una corretta installazione e configurazione su un server equipaggiato con Windows 10 del recente simulatore di flusso geotermico ad alta entalpia, free e open source, chiamato Waiwera. Nella parte finale del presente elaborato, ossia quando si descrive l'utilizzo di Waiwera per simulare un caso studio, si mostra come questo scopo sia stato raggiunto con successo. Dalla fase di verifica, caratterizzata da comparazioni di tipo qualitativo dei risultati, sia grafiche che numeriche, si può affermare di essere riusciti ad ottenere un server per simulazioni geotermiche ad alta entalpia perfettamente funzionante e performante.

La versatilità di Waiwera di operare su diversi sistemi operativi, ha comportato un'iniziale complessità nella sua messa in funzione, caratterizzata da una serie di complesse procedure da attuare per il raggiungimento di tale scopo. La fase di *post-processing* ha richiesto lo studio di specifici programmi, HDFview e ParaView, e, seppur non in modo approfondito, del linguaggio di programmazione Python, per il quale si sono dimostrati essenziali le nozioni assimilate durante il percorso universitario.

La documentazione disponibile, da poter utilizzare per ricavare esempi per effettuare verifiche e confronti, è risultata composta da un esiguo numero di pubblicazioni; questo appare ovvio se si pensa alla giovane età di questo simulatore. Purtroppo, le pubblicazioni reperite hanno comunque reso possibile la comprensione generale di un caso studio, anche se per i dettagli della simulazione in questione è stato comunque di grande importanza lo studio analitico delle strutture dei suoi file di input e output.

La simulazione dello stesso caso studio con TOUGH2-EWASG ha permesso di effettuare un primo passo verso futuri studi più approfonditi per valutare il grado di affidabilità di Waiwera rispetto ad altri simulatori geotermici più comuni e più studiati. In questa ottica, anche valutare l'applicabilità di Waiwera su problemi di iniezione di CO₂ o idrogeno in reservoir è sicuramente un'interessante linea di ricerca.

Bibliografia

- [1] IRENA, *Renewable capacity statistics 2020*. [Online]: https://www.irena.org/-/media/Files/IRENA/Agency/Publication/2020/Mar/IRENA_RE_Capacity_Statistics_2020.pdf
- [2] Doglioni C., *Interno della Terra*, 2007.
- [3] Unione Geotermica Italiana, *La Geotermia Ieri, Oggi, Domani. Numero speciale del Notiziario UGI*, Novembre 2007.
- [4] Bortolotti V., *Simulazione numerica di bacini serbatoio, dispense del corso MGFS, Unibo, marzo 2019*.
- [5] Monegato, G. *Calcolo Numerico*. (Levrotto&Bella: Torino,).
- [6] Chierici, G.L. *Principi di ingegneria dei giacimenti petroliferi*. (ENI: 2004).
- [7] Reverberi A., *Accoppiamento esterno del simulatore per sistemi geotermici ad alta entalpia iTOUGH2 con un simulatore di pozzo. Tesi magistrale, Alma Mater Studiorum-University of Bologna, 2010*.
- [8] Bedient, P. *Ground water contamination: transport and remediation*. (Prentice Hall PTR: Upper Saddle River NJ, 1999).
- [9] Croucher, A.E., O'Sullivan, M.J., O'Sullivan, J.P., J., Yeh, A., Burnell, J. and Kissling, W.: *Waiwera: A parallel open-source geothermal flow simulator. New Zeland (2020)*. [Online]: <https://www.sciencedirect.com/science/article/pii/S0098300419310088?via%3Dihub>
- [10] Pruess K., Oldenburg C., Moridis G., *Earth Sciences Division, TOUGH2 User's Guide Version 2.0, Lawrence Berkeley National Laboratory University of California, Berkeley, 1999*.
- [11] Pan L., Oldenburg C., Wu Y. and Pruess K., *Earth Sciences Division, Twell/ECO2N Version 1.0: Multiphase and Non-Isothermal Model for Coupled Wellbore-Reservoir Flow of Carbon Dioxide and Variable Salinity Water, Lawrence Berkeley National Laboratory, University of California, Berkeley, California 94720, 2011*.
- [12] Croucher, A.E.: *Waiwera user guide – Release 1.2.1 (2020)*.
- [13] Satish Balay. *PETSc users manual. Technical Report ANL-95/11 - Revision 3.11, Argonne National Laboratory, 2019*. [Online]: <https://publications.anl.gov/anlpubs/2019/12/155920.pdf>
- [14] Molloy, M.W.: *Geothermal reservoir engineering code comparison project. Proc. Special Panel on Geothermal Model Intercomparison Study, December, 1980, Stanford University, Stanford, California (1981)*.
- [15] Croucher, A. E., O'Sullivan, J. P., Yeh, A., & O'Sullivan, M. J.: *Benchmarking and experiments with Waiwera, a new geothermal simulator. Proc. 43rd Workshop on Geothermal Reservoir Engineering. Stanford University, Stanford, California, USA (2018)*. [Online]: <https://pangea.stanford.edu/ERE/pdf/IGAstandard/SGW/2018/Croucher.pdf>.

- [16] O'Sullivan, M.J., Bodvarsson, G.S., Pruess, K. and Blakeley, M.R.: *Fluid and heat flow in gas-rich geothermal reservoirs*. *Soc. Pet. Eng. J.*, (1985). [Online]: <https://watermark.silverchair.com>
- [17] Ambrogetti L., *Perfezionamento delle routine del simulatore geotermico T2well-ECO2M dedicate al calcolo delle pressioni capillari*. Tesi magistrale, Alma Mater Studiorum-University of Bologna, 2020.
- [18] O'Sullivan, M., Pruess, K., Lippmann, M., 2001. *State of the art of geothermal reservoir simulation*. *Geothermics* 30, 395–429.
- [19] Vinsome, P., Shook, G., 1993. *Multi-purpose simulation*. *J. Pet. Sci. Eng.* 9, 29–38.
- [20] Pritchett, J., 1995. *STAR: a geothermal reservoir simulation system*. In: *Proceedings of the World Geothermal Congress 1995*. Florence, Italy, pp. 2959–2963.
- [21] Clauser, C., Bartels, J., 2003. *Numerical Simulation of Reactive Flow in Hot Aquifers*. Springer, Berlin, pp. 1–332.
- [22] Blöcher, M., Zimmermann, G., Moeck, I., Brandt, A., Hassanzadegan, A., Magri, F., 2010. *3D numerical modeling of hydrothermal processes during the lifetime of a deep geothermal reservoir*. *Geofluids* 10, 406–421.
- [23] Kolditz, O., Bauer, S., Bilke, L., Böttcher, N., Delfs, J.O., Fischer, T., Görke, U.J., Kalbacher, T., Kosakowski, G., McDermott, C.I., Park, C.H., Radu, F., Rink, K., Shao, H., Shao, H.B., Sun, F., Sun, Y., Singh, A.K., Taron, J., Walther, M., Wang, W., Watanabe, N., Wu, Y., Xie, M., Xu, W., Zehner, B., 2012. *OpenGeoSys: an opensource initiative for numerical simulation of thermo-hydro-mechanical/chemical (THM/C) processes in porous media*. *Environ. Earth Sci.* 67, 589–599.
- [24] Saeid, S., Al-Khoudry, R., Barends, F., 2013. *An efficient computational model for deep low-enthalpy geothermal systems*. *Comput. Geosci.* 51, 400–409.
- [25] Hammond, G., Lichtner, P., Mills, R., 2014. *Evaluating the performance of parallel subsurface simulators: an illustrative example with PFLOTTRAN*. *Water Resour. Res.* 50, 208–228.
- [26] Zyvoloski, G., 2007. *FEHM: A Control Volume Finite Element Code for Simulating Subsurface Multi-Phase Multi-Fluid Heat and Mass Transfer*. Technical Report LAUR-07-3359, Los Alamos National Laboratory, pp. 1–44.
- [27] Akin, S., Kok, M., Uraz, I., 2010. *Optimization of well placement geothermal reservoirs using artificial intelligence*. *Comput. Geosci.* 36, 776–785.
- [28] Stacey, R., Williams, M., 2017. *Validation of ECLIPSE reservoir simulator for geothermal problems*. *Geotherm. Resour. Coun. Trans.* 41, 2095–2109.
- [29] O'Sullivan, M., O'Sullivan, J., 2016. *Reservoir modeling and simulation for geothermal resource characterization and evaluation*. In: DiPippo, R. (Ed.), *Geothermal Power Generation: Developments and Innovation*. Woodhead Publishing, pp. 165–200.

- [30] Yeh, A., Croucher, A., O'Sullivan, M., 2012. *Recent developments in the AUTOUGH2 simulator*. In: *Proceedings of the TOUGH Symposium 2012*. Berkeley, California, pp. 1–8.
- [31] Zhang, K., Wu, Y.-S., Ding, C., Pruess, K., 2003. *TOUGH2_MP: a parallel version of TOUGH2*. In: *Proceedings of the TOUGH Symposium 2003*. Berkeley, California, pp. 1–8.
- [32] Jung, Y., Pau, G.S.H., Finsterle, S., Pollyea, R.M., 2017. *TOUGH3: a new efficient version of the TOUGH suite of multiphase flow and transport simulators*. *Comput. Geosci.* 108, 2–7.
- [33] O'Sullivan, J., Croucher, A., Yeh, A., O'Sullivan, M., 2013. *Improved convergence for air-water and CO2-water TOUGH2 simulations*. In: *Proceedings of the 35th New Zealand Geothermal Workshop*. Rotorua, New Zealand, pp. 1–9.
- [34] Pruess, K., 1988. *SHAFT, MULKOM, TOUGH: A Set of Numerical Simulators for Multiphase Fluid and Heat Flow*. Technical Report LBL-24430, Lawrence Berkeley National Laboratory, University of California, Berkeley, California, pp. 1–25.
- [35] Battistelli A., C. Calore, and K. Pruess K., *The simulator TOUGH2/EWASG for modelling geothermal reservoirs with brines and a noncondensable gas*. *Geothermics*, 1997.
- [36] Finsterle, S., M. Commer, J. Edmiston, Y. Jung, MB Kowalsky, GSH Pau, H. Wainwright e Y. Zhang, *iTOUGH2: un framework di ottimizzazione della simulazione per l'analisi di sistemi multifisici del sottosuolo*, *Computer e geoscienze* , 108 , 8 20, 2017.
[Online]: doi.org/10.1016/j.cageo.2016.09.0052017 .

Sitografia

[S1] Figura della Terra - Wikipedia: [Figura della Terra - Wikipedia](#) . Data di consultazione 09/2021.

[S2] Struttura interna della Terra - Wikipedia:
https://it.wikipedia.org/wiki/Struttura_interna_della_Terra . Data di consultazione 09/2021.

[S3] Teoria dei sistemi - Wikipedia: https://it.wikipedia.org/wiki/Teoria_dei_sistemi . Data di consultazione 09/2021.

[S4] Waiwera- Website: <https://waiwera.github.io/> Data di consultazione 08/2021.

[S5] GNS Science – Nuova Zelanda: <https://www.gns.cri.nz/>. Data di consultazione 07/2021.

[S6] PETSc Wikipedia:
https://en.wikipedia.org/wiki/Portable_Extensible_Toolkit_for_Scientific_Computation . Data di consultazione 07/2020.

[S7] Fortran-Wikipedia: [Fortran - Wikipedia](#) . Data di consultazione 07/2020.

[S8] OOP (Object Oriented Programming):
https://it.wikiversity.org/wiki/Programmazione_orientata_agli_oggetti . Data di consultazione 08/2021.

[S9] JSON (JavaScript Object Notation) - Wikipedia:
https://it.wikipedia.org/wiki/JavaScript_Object_Notation . Data di consultazione 07/2021.

[S10] HDF5 (Hierarchical Data Format) - Wikipedia:
https://en.wikipedia.org/wiki/Hierarchical_Data_Format . Data di consultazione 07/2021.

[S11] YAML - Wikipedia: <https://it.wikipedia.org/wiki/YAML> . Data di consultazione 07/2021.

[S12] ExodusII - Manual: <https://publications.anl.gov/anlpubs/2014/12/79486.pdf> . Data di consultazione 08/2021.

[S13] MSH - Wikipedia: <https://it.wikipedia.org/wiki/Gmsh> . Data di consultazione 08/2021.

[S14] MED: <https://it.scriptcult.com/1345.html> . Data di consultazione 08/2021

[S15] IFC-67: <https://www.steam-water-properties.com/steam%20water%20properties%20international%20formulation.html> Data di consultazione 08/2021.

[S16] IAPWS-97: <http://www.iapws.org/newform.html> Data di consultazione 08/2021.

[S17] Docker Desktop – Website: <https://docs.docker.com/> .Data di consultazione 07/2021.

[S18] Docker Desktop – Installazione per Windows: <https://docs.docker.com/> .Data di consultazione 07/2021.

- [S19] WSL2- Informazioni: <https://docs.microsoft.com/it-it/windows/wsl/about> .Data di consultazione 08/2021.
- [S20] SLAT- Informazioni: [https://en.wikipedia.org/wiki/Second_Level_Address_Translation#:~:text=Second%20Level%20Address%20Translation%20\(SLAT,software%2Dmanaged%20shadow%20page%20tables](https://en.wikipedia.org/wiki/Second_Level_Address_Translation#:~:text=Second%20Level%20Address%20Translation%20(SLAT,software%2Dmanaged%20shadow%20page%20tables) . Data di consultazione 07/2021.
- [S21] WSL2 - Download: https://wslstorestorage.blob.core.windows.net/wslblob/wsl_update_x64.msi Data di consultazione 08/2021.
- [S22] WSL2- Installazione: <https://docs.microsoft.com/en-us/windows/wsl/install-win10#step-1---enable-the-windows-subsystem-for-linux> Data di consultazione 08/2021.
- [S23] BIOS (Basic-Input-Output System): <https://it.wikipedia.org/wiki/BIOS> . Data di consultazione 07/2021.
- [S24] UEFI (Unified Extensible Firmware Interface): <https://it.wikipedia.org/wiki/UEFI> Data di consultazione 07/2021.
- [S25] Coreinfo – Download: <https://docs.microsoft.com/en-us/sysinternals/downloads/coreinfo> . Data di consultazione 07/2021.
- [S26] Python – Download: <https://www.python.org/downloads/> Data di consultazione 08/2021.
- [S27] Variabili d'ambiente: [https://it.wikipedia.org/wiki/Variabile_d%27ambiente_\(Windows\)](https://it.wikipedia.org/wiki/Variabile_d%27ambiente_(Windows)) .Data di consultazione 07/2021.
- [S28] DAT: <https://it.wikipedia.org/wiki/.dat> Data di consultazione 07/2021.
- [S29] HDFview – Download: <https://portal.hdfgroup.org/display/support/HDFView+3.1.2#files> Data di consultazione 08/2021.
- [S30] ParaView – Download e tutorial: <https://www.paraview.org/download/> Data di consultazione 08/2021.
- [S31] H5YP: <https://www.h5py.org/> . Data di consultazione 07/2021.
- [S32] PyYAML – Installazione e informazioni: <https://pypi.org/project/PyYAML/> Data di consultazione 08/2021.
- [S33] Python – Script di visualizzazione output: <https://waiwera.readthedocs.io/en/latest/output.html> Data di consultazione 08/2021.
- [S34] Matplotlib- Guida: <https://matplotlib.org/> . Data di consultazione 08/2021.
- [S35] Possibili schemi di come possa essere rappresentato l'interno della Terra: <http://www.pi.ingv.it/divulgazione/download/esperienze-di-sismologia/interno-della-terra/> . Data di consultazione 09/2021.
- [S36] Flusso termico del globo terrestre: <https://docplayer.it/52009763-La-struttura-della-terra.html> . Data di consultazione 09/2021.

[S37] *Tipologie di impianti per produzione di energia geotermica:*

<https://www.enelgreenpower.com/it/learning-hub/energie-rinnovabili/energia-geotermica/centrale-geotermica> . *Data di consultazione 09/2021.*