



UNIVERSITA' POLITECNICA DELLE MARCHE

FACOLTA' DI INGEGNERIA

Corso di Laurea triennale in Ingegneria Meccanica

L'importanza del Digital Twin nell'industria 4.0

The importance of Digital Twin in industry 4.0

Relatore: Chiar.mo

Prof. Ciarapica Filippo Emanuele

Tesi di Laurea di:

Cieri Lorenzo

Correlatore: Chiar.mo

Prof. Mazzuto Giovanni

Anno Accademico 2019/2020

Indice

1 Introduzione	1
2 Digital Twin	6
2.1 Simulation	6
2.2 Dalla simulazione al Digital Twin	10
2.3 Di cosa si tratta	11
2.4 Software di simulazione per la realizzazione di un Digital Twin	22
3 Openmodelica	28
3.1 Progettare un impianto idraulico	32
3.1.1 Prime operazioni	34
3.1.2 Costruzione grafica e inserimento parametri Impianto	35
3.1.3 Inserimento sensori	41
4 Digitalizzare un impianto	49
4.1 Descrizione impianto	50
4.2 Componenti idraulici e sensori	54
4.3 Elettronica di controllo	74
4.4 Dal reale al digitale	76
5 Conclusioni	103

Bibliografia	107
Sitografia	109

1 Introduzione

Il termine Industria 4.0 (Industry 4.0) indica una tendenza dell'automazione industriale che integra alcune nuove tecnologie produttive per migliorare le condizioni di lavoro, aumentare la produttività e la qualità produttiva degli impianti. Industria 4.0 prende il nome dal piano industriale del governo tedesco, presentato nel 2011 e concretizzato alla fine del 2013, che prevedeva investimenti su infrastrutture, scuole, sistemi energetici, enti di ricerca e aziende per ammodernare il sistema produttivo tedesco e riportare la manifattura tedesca ai vertici mondiali rendendola competitiva a livello globale.

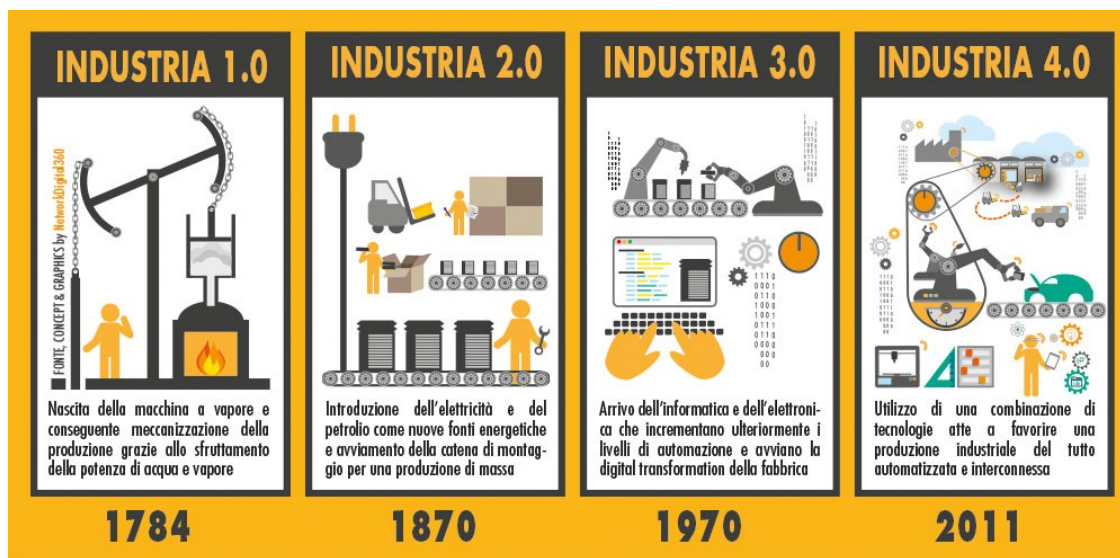


Figura A: Industria 4.0: storia, significato ed evoluzioni tecnologiche (www.digital4.biz).

I risultati ottenuti dalla Germania a livello produttivo hanno portato molti altri paesi a perseguire questa politica tanto da avviare numerosi studi sui vantaggi dell'industria 4.0. Questi studi hanno portato a definire l'impatto che queste nuove politiche avranno sul contesto sociale ed economico, definendo questo passaggio storico "**Quarta rivoluzione industriale**". Quel suffisso 4.0, infatti, corrisponde alle tappe di un'evoluzione estremamente precisa. Dall'introduzione della macchina a vapore all'uso sempre più pervasivo dell'automazione, dall'informatizzazione alla digitalizzazione, il passaggio alla quarta rivoluzione industriale traghetta le imprese verso una nuova dimensione detta bimodale, poiché costituita da un ecosistema di risorse fisiche e virtuali.

Negli anni 70 l'industria si incentrava sulla riduzione dei costi, negli '80 sulla qualità, negli anni '90 sulla velocità di produzione per rispondere in maniera più efficiente alle esigenze del consumatore. Con l'Industria 4.0 si entra in quella che viene definita "economia della conoscenza" il cui focus risiede nella diffusione di conoscenza e nei servizi, i processi diventano smart in un'ottica disruptive (cioè capace di creare un nuovo mercato).

Questa rivoluzione, rispetto alle precedenti, vede come protagonista l'utilizzo all'interno dell'impianto di produzione di "tecnologie abilitanti" (in inglese KET, key enabling technology), soluzioni o miglioramenti tecnologici, che racchiudono al loro interno molta attività di ricerca e sviluppo e sono in grado di "rivitalizzare il sistema produttivo".

Utilizzare tecnologie abilitanti significa dunque che i processi legati all'industria saranno dotati di una interconnessione veloce, chiara e diretta tra tutti gli asset aziendali. La produttività aumenta, gli sprechi diminuiscono.

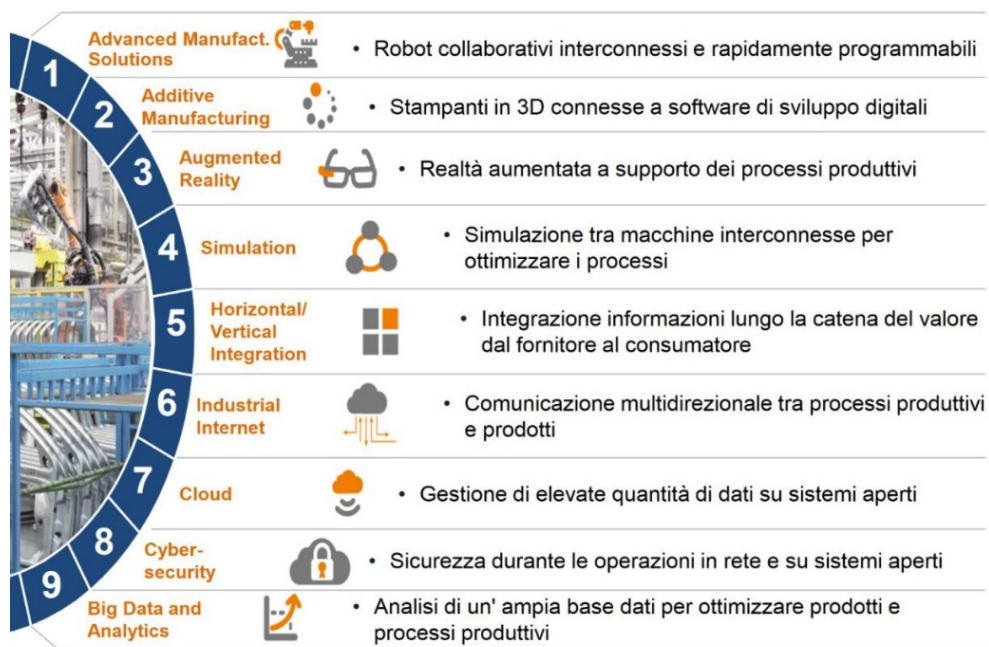


Figura B: Le tecnologie abilitanti del Piano Nazionale Industria 4.0, (www.focusindustria40.com).

L'implementazione di queste nuove tecnologie richiede una fase di transizione: dal vecchio concetto di fabbrica alla nuova fabbrica intelligente (*smart factory*), caratterizzata da una produzione digitalizzata, che funziona in maniera dinamica e “brillante” (*brilliant manufacturing*), composta da processi più fluidi, interconnessi e da sistemi di produzione adeguati alla modernità e ai suoi bisogni, capaci di utilizzare al meglio le risorse disponibili.

In concreto i benefici della rivoluzione che sta travolgendo l'ecosistema produttivo proprio in questo momento sono:

- Maggiore flessibilità, attraverso la produzione di piccoli lotti ai costi della grande scala;
- Maggiore velocità, dal prototipo alla produzione in serie attraverso tecnologie innovative;
- Maggiore produttività, attraverso minori tempi di set-up, riduzione errori e fermi macchina;
- Migliore qualità e minori scarti, mediante sensori che monitorano la produzione in tempo reale;
- Maggiore competitività del prodotto, grazie a maggiori funzionalità derivanti dall'utilizzo di Internet.

Tramite il collegamento di tutti gli asset coinvolti nella filiera logistico-produttiva è possibile disporre di tutte le informazioni pertinenti in tempo reale, che rappresenta un vantaggio ulteriore per le aziende che hanno introdotto le tecnologie abilitanti mentre i sistemi di acquisizione dei dati (come i sistemi MES – Manufacturing Execution Systems) sono messi a disposizione ai livelli aziendali più alti in cui è possibile gestire e monitorare aggiornamenti, interventi di manutenzione e molti altri dati in tempo reale.

Il modello di interconnessione tra sistemi e macchine può essere sfruttato dalla fase di progettazione fino alla fase operativa dell'industria. Ciò richiede uno scambio di dati ininterrotto tale da garantire la continuità digitale fra la fabbrica reale e la sua rappresentazione virtuale, mantenendo la coerenza fra dati di varia origine (es. piani di produzione, monitoraggio, previsioni di domanda, ecc.) che concorrono a definire l'evoluzione nel tempo del sistema produttivo.

2 Digital Twin

2.1 Simulation

“Non è possibile costruire un’auto, un edificio o un ponte senza simulazione”

Younis, ex dirigente Google. Fonte: www.focusindustria40.com

Il punto chiave della quarta rivoluzione industriale è la connessione che vede emergere un vantaggio competitivo basato sui dati come fonte di ricchezza informativa attraverso l'utilizzo pervasivo delle tecnologie digitali. Grazie alle 9 tecnologie abilitanti vengono sviluppate soluzioni e miglioramenti tecnologici attraverso attività di ricerca, una di queste è la **Simulazione**.

Per la pagina web www.eidonlab.org,

“La simulazione numerica è uno strumento sperimentale di analisi molto potente, utilizzato in vari ambiti scientifici e tecnologici, grazie al quale è possibile superare le difficoltà o le impossibilità che si affrontano in un laboratorio reale. Tali difficoltà sono da ricondursi alla riproduzione fisica delle effettive condizioni di un sistema oggetto di studio e analisi.”

Questa tecnologia è, quindi, assimilabile ad una sorta di laboratorio virtuale e copre l'intero ciclo di vita di un processo, dalla ricerca e sviluppo, alla progettazione concettuale, al funzionamento dell'impianto.

L'obiettivo principale della simulazione di prodotti, processi o servizi è quello di utilizzare modelli digitali adatti a simulare il comportamento in base alla variazione di alcune dimensioni oggettivamente misurabili.

La pagina www.eidonlab.org offre anche una classificazione delle tecnologie di simulazione in funzione delle loro finalità:

- Tecnologie di simulazione a supporto delle decisioni;
- Tecnologie di simulazione a supporto della progettazione e ingegnerizzazione dei prodotti;
- Tecnologie di simulazione a supporto dell'analisi dei processi dell'industria manifatturiera e di processo.

La simulazione dei processi è largamente diffusa e comincia a farsi strada anche nelle piccole aziende. L'offerta commerciale dei software di simulazione è ormai ampia, e le software house propongono spesso pacchetti integrati di virtual manufacturing: lo stampaggio, la saldatura, l'imbutitura, la piegatura, la calandratura, la fonderia, l'estrusione e persino processi piuttosto difficili da simulare come la fresatura o la sinterizzazione sono ormai simulabili e certamente molto adatti allo scopo di ottimizzare i processi.

Anche la simulazione a eventi discreti, molto adatta a simulare come si comporta non solo un singolo processo o macchina ma come funziona una fabbrica o un reparto dal punto di vista dei flussi di materiali e utensili, è una realtà solida.

Una dimostrazione sull'utilità della simulazione è fornita dalla Hensley Industries, azienda specializzata in accessori e parti usurabili per apparecchiature di movimento terra impiegate nei campi dell'estrazione, dell'edilizia, dello scavo e della bonifica. Per la progettazione di una benna (organo per il sollevamento e trasporto) di un escavatore gli aspetti da considerare nella fase di progettazione sono molteplici, questo perché dopo aver scavato una certa quantità di rocce, i denti della benna si usurano fino a diventare inservibili e devono essere sostituiti.

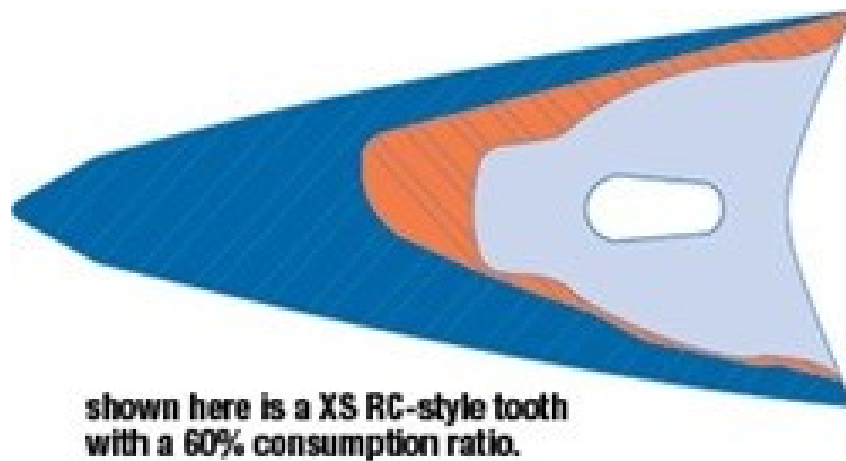


Figura C: dente della benna sottoposto ad usura. (www.ptc.com)

Dunque, le principali domande da porsi sono: il dente può essere progettato in modo da poter essere sostituito rapidamente sul campo dall'operatore o dal tecnico senza speciali competenze, utensili e forza? Come si usurerà la parte nel tempo?

I progettisti della Hensley ottengono la risposta a queste domande eseguendo simulazioni con “Creo Simulate”, il progettista infatti può applicare carichi statici e dinamici a un modello CAD direttamente in Creo e osservare come si comporterà nel mondo reale molto prima di creare prototipi reali.

La domanda per uno strumento software di simulazione che consente ai produttori di testare linee di produzione nuove o riprogettate prima di avviare la produzione è in forte aumento. Entro i prossimi cinque anni, secondo la società ABI Research che si occupa di ricerche di mercato, si stima che 110.000 aziende in tutto il mondo utilizzeranno un software di simulazione, rispetto alle 60.000 del 2018. Secondo l'indagine, i produttori sono pronti a spendere più di 2 miliardi di euro all'anno per i software di simulazione entro il 2025, mentre uno studio dell'Aberdeen Group nel 2017 ha rilevato che le aziende, provenienti dai settori automobilistico ed aerospaziale-difesa, che hanno introdotto la simulazione nei loro processi industriali, hanno raggiunto gli obiettivi prefissati in termini di time-to-market (tempo che intercorre dall'ideazione di un prodotto alla sua effettiva commercializzazione), riduzione dei costi e qualità. I risultati mostrano un aumento di oltre il 20% rispetto ai loro competitors che non la usano.

2.2 Dalla simulazione al Digital Twin

Oltre ai benefici già indicati, la simulazione offre i vantaggi maggiori insieme a l'Internet of Things, il Cloud e la Big Data Analytics (tutte tecnologie abilitanti) permettendo la creazione del cosiddetto gemello digitale o virtuale di processi, prodotti o servizi (**Digital Twin**).

“Rispetto al paradigma dell’Industria 4.0, il concetto di Digital Twin rappresenta lo stato dell’arte.”

Luca Zanotti, “Digital twin: cos'è e come funziona il modello del gemello digitale”. Fonte: www.digital4.biz

Grazie a cicli di sviluppo più snelli e una maggiore collaborazione, sia interna che nei confronti di partner e fornitori esterni, le aziende riducono i tempi di sviluppo di prodotti complessi del 25%, con risparmi nell'ordine del 10-15% (Fonte: “Digital Twins: Identical, But Different” Oliver Wyman).

2.3 Di cosa si tratta

Il concetto di Digital Twin fu usato per la prima volta nel 2001 da Michael Grieves, oggi Chief Scientist for Advanced Manufacturing presso il Florida Institute of Technology, che durante un corso di Product Lifecycle Management (PLM) presso l'Università del Michigan descriveva il gemello digitale come l'equivalente virtuale e digitale di un prodotto fisico.

Conceptual Ideal for PLM

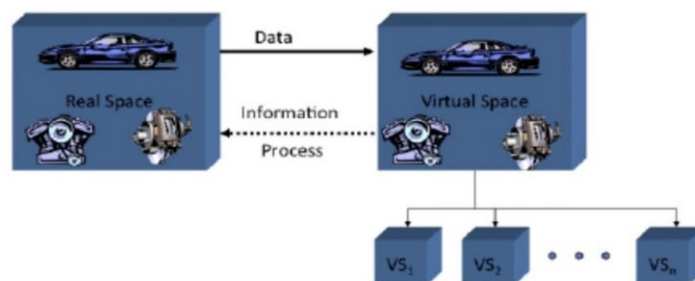


Figura D: l'idea del Digital twin di Michael Grieves. (www.digital4.biz)

Si tratta dunque di una rappresentazione virtuale che varia al variare di determinati parametri controllabili dagli utilizzatori. Nel suo approccio al PLM, Grieves evidenziava un Mirrored Spaces Model riferito a una rappresentazione estremamente dinamica: la dimensione reale e la dimensione virtuale, infatti, rimanevano collegate durante l'intero ciclo di vita del sistema

Condizione necessaria alla realizzazione di un gemello digitale è l'esistenza di:

- prodotti fisici nello spazio reale;
- prodotti virtuali nello spazio virtuale;
- sistemi di collegamento del flusso di dati e informazioni che uniscono lo spazio fisico a quello virtuale e ai sottospazi virtuali.

“Il Digital Twin è una sorta di modello in vivo di un impianto o di una linea di produzione, realizzato dall'incrocio tra i dati reali di funzionamento e i disegni CAD e digitali serviti per la progettazione. Il Digital Twin consente di costruire una copia virtuale dell'impianto reale (ma anche di una singola macchina) in grado di replicare in tutto e per tutto il funzionamento reale o di verificarne tutte le possibili alternative”

Robert Plana, Innovation e ecosystem director di GE Digital, società del gruppo General Electric.

Fonte: www.ilsole24ore.com

Con l'utilizzo del gemello digitale sarà possibile per le aziende:

- *Prevedere e anticipare problemi di produzione:* grazie ai Digital Twin è possibile testare e capire come si comporteranno i sistemi e i prodotti che si vogliono realizzare in un'ampia varietà di ambienti;

- *Migliorare lo sviluppo di prodotto e diminuire costi di realizzazione:*
l'associazione tra realtà fisica e realtà virtuale consente di attivare un'analisi dei dati e un monitoraggio dei sistemi tale per cui è possibile ragionare in modalità predittiva, affrontando i problemi prima ancora che questi si verifichino oltre a prevenire anomalie, tempi di inattività e inefficienze;
- *Gestire il know-how aziendale:* gestire l'insieme delle informazioni e dei processi che la rendono unica e che le permettono di distinguersi dalle altre aziende usando la simulazione per fare esperienza e gestendone i risultati in modo che diventino un patrimonio riutilizzabile;
- *Mantenere un vantaggio competitivo sostenibile nel tempo:* utilizzando simulazioni appropriate è possibile sviluppare nuove opportunità, pianificando i business futuri, innovando costantemente e riducendo al contempo i rischi relativi al fare innovazione.

L'eliminazione dei tempi di inattività non pianificati e la riduzione dei costi di manutenzione sono i principali fattori considerati dalle aziende di vari settori per migliorare la produttività e l'efficienza.

Secondo gli analisti, oltre un'azienda su dieci ha subito almeno un'interruzione di inattività non pianificata in un arco di tre anni (Fonte: Vanson Bourne, 2017). Questi tempi di fermo nelle aziende influiscono su tutto il processo di produzione, impattando negativamente sulla produttività, sull'efficienza e sul servizio clienti. La riduzione dei tempi di fermo può essere dunque raggiunta con l'implementazione di un gemello digitale che aiuta le aziende a raccogliere dati e utilizzarli per generare risultati aziendali specifici. I dati recuperati vengono analizzati rispetto a vari parametri e vengono adottate, in modo proattivo, misure preventive e predittive per evitare danni a prodotto, processo e sistema che a loro volta aiuterebbero a ridurre i tempi di fermo e di conseguenza aumentare l'efficienza complessiva.

Risulta fondamentale anche la consapevolezza delle aziende in merito ai consumi energetici e la possibilità di ridurli, valutando le inefficienze e quali sono le aree produttive meno sviluppate e con più necessità di intervento. I gemelli digitali possono anche utilizzare sistemi di machine learning e di AI per elaborare i dati e produrre nuova conoscenza.

Oltre a portare vantaggi al processo produttivo, il modello Digital Twin aiuta a regolamentare i rapporti fra i diversi reparti in un'ottica di condivisione e di collaborazione che consente di immettere sul mercato prodotti altamente funzionali.

Non va dimenticato che un prodotto spesso è un aggregato di varie componenti, normalmente realizzate da aziende diverse. In un'ottica di azienda estesa i Digital Twin permettono massima trasparenza informativa in fase di progettazione e di realizzazione. Un'azienda sviluppa il suo modello e lo consegna ad una seconda che lo integrerà nel proprio modello (o lo farà interagire) fino ad arrivare all'azienda che dovrà assemblare tutti i componenti che simulerà questa operazione sui diversi modelli ricevuti.

La Boeing (industria di aeronautica statunitense), da diversi anni progetta i suoi aerei in questo modo (a partire dal Boeing 777): i suoi fornitori ricevono le specifiche in “bit” delle parti con cui il loro componente dovrà interfacciarsi, sia questo un rivetto o un flight management system, e a loro volta forniranno alla Boeing una rappresentazione in bit del futuro componente. La Boeing effettua l'integrazione a livello dei bit, esegue test e simulazioni per poi attivare la produzione, sua e dei fornitori. Pertanto, consentendo una continuità digitale dall'ideazione alla produzione fino a coprire anche i servizi post-vendita, la gestione dei gemelli digitali sta prendendo piede a livello globale in diversi settori a partire dallo sport, i Digital twin sono utilizzati per perfezionare le corse automobilistiche di Formula 1, alla medicina, i Digital Twin possono rivoluzionare sia le operazioni sanitarie che l'assistenza ai pazienti.

Avere un gemello digitale di un paziente, con una rappresentazione tridimensionale di organi, ossa, sistema venoso, linfatico e nervoso consente ai chirurghi e agli operatori sanitari di sperimentare le procedure in un ambiente simulato piuttosto che su un paziente reale, mentre i sensori delle dimensioni delle bende possono monitorare i pazienti e produrre modelli digitali che possono essere monitorati dall'Intelligenza Artificiale e utilizzati per migliorare l'assistenza.

Esiste persino un gemello digitale di Singapore: considerando tutte le variabili relative alla gestione di una città, il gemello digitale aiuta gli urbanisti a comprendere come migliorare l'efficienza dei consumi energetici, a gestire il traffico, i servizi pubblici e a identificare nuove applicazioni che possono migliorare la vita dei cittadini.



Figura E: Digital twin di Singapore (www.smartcitylab.com)

L'importanza del Digital Twin è sostenuta anche dalle aziende stesse che forniscono dati sull'utilizzo del gemello digitale e i vantaggi che ne derivano.

Dallara, un'azienda italiana costruttrice di automobili da competizione, anziché sviluppare prototipi di scocche e testarli in gallerie del vento e altri laboratori, costruisce dei modelli digitali ed è attraverso questi che vengono effettuate tutte le prove e le simulazioni necessarie fino a portare a termine ogni progetto.

La General Electric da alcuni anni ha esteso il loro utilizzo alle sue turbine (sia quelle che sono utilizzate nei sistemi eolici sia quelle che operano nei motori dei jet più moderni come il Boeing 787) oltre che alla produzione, anche alla loro manutenzione e controllo. Quando la turbina viene venduta è associata ad un modello in bit, un gemello digitale, che da quel momento sarà specifico di quella turbina.

La turbina contiene una varietà di sensori che in tempo quasi reale comunicano con la turbina virtuale, informandola dell'utilizzo in corso. Ad esempio, nel caso delle pale eoliche, verrà segnalata l'accensione, la velocità di rotazione delle pale, la potenza elettrica istantanea e così via. Stesso discorso per la turbina di un motore di aereo (qui i parametri segnalati saranno molti di più). Venti anni fa un aereo generava circa 1kB di informazioni per ogni volo, registrando lo stato dell'aereo 3 volte per ogni volo, stato basato su una trentina di parametri; oggi un aereo di ultima generazione genera per ogni volo 500GB di dati, con lo stato aggiornato ogni secondo e basato su circa cinquemila parametri (dati forniti da General Electric).

Questi stati vengono inviati in tempo quasi reale (ogni minuto o ogni tre minuti a seconda dell'area in cui sta volando l'aereo) e consentono da un lato di simulare la situazione operativa da parte del gemello digitale e dall'altro di rilevare difformità di funzionamento tra il Digital Twin e la turbina fatta di atomi.



Figura F: General Electric, turbina con sensori per il Digital Twin (www.ge.com)

In maniera del tutto simile la Tesla riceve informazioni dalle sue auto ogni giorno: centinaia di migliaia di auto comunicano dove stanno viaggiando, gli ostacoli identificati lungo il percorso, il funzionamento del motore. Ogni giorno Tesla riceve l'equivalente di oltre due milioni di Km percorsi, sommando i contributi di tutte le auto. Anche nel caso della Tesla ogni auto ha un suo “Digital Twin” costantemente aggiornato.

L'azienda che crea dei gemelli in bit (la General Electric ne ha operativi oltre 550.000 a fine 2017) si apre uno spazio di business basato sull'offerta di servizi, fornisce difatti un monitoraggio dell'operatività del prodotto e può spesso intervenire per porre rimedio a problemi o comunque suggerire interventi tempestivi di manutenzione.

Dal punto di vista dell'azienda monitorare i prodotti durante il loro utilizzo offre anche la possibilità di verificarne le funzionalità e come queste sono utilizzate. In pratica permette di continuare il testing in situazione di uso reale e di continuare ad affinare il prodotto. Considerando che oggi esiste una componente molto rilevante di software in ogni prodotto l'azienda può variare il software ed aggiornare il prodotto (o offrire un aggiornamento a pagamento quando vengono forniti miglioramenti) in tempo reale.

Queste aziende non solo hanno utilizzato i gemelli digitali per migliorare la propria offerta ma hanno anche cambiato l'idea stessa di prodotto: non si è più in presenza di un oggetto che viene acquistato tramite un rivenditore che disaccoppia il produttore dall'utilizzatore, al contrario produttore ed utilizzatore rimangono in contatto, addirittura costantemente in alcuni casi. La filiera si modifica e questo è un elemento che caratterizza l'Industria 4.0. Non cambia infatti soltanto la produzione all'interno dell'azienda ma anche la relazione tra i diversi attori lungo la filiera con il coinvolgimento dell'utente finale.

La presenza di Digital Twin apre anche, potenzialmente, uno spazio per terze parti, spesso PMI (piccole e medie imprese) e start up, che possono sviluppare servizi a partire dai modelli digitali.

Essendo questi basati su bit i costi transazionali sono molto limitati, mettendo lo sviluppo di applicazioni alla portata di un grande numero di attori e a loro volta questi creano nel tempo un ecosistema che arricchisce il prodotto stesso.

Tuttavia, l'utilizzo del Digital Twin, insieme alla quarta rivoluzione industriale, porta con sé un problema legato alla sicurezza. Se infatti l'interconnessione porta benefici a livello di efficienza e gestione, dall'altro tutto ciò che è connesso in rete rappresenta un potenziale rischio per tutto il sistema produttivo, persone comprese. Potenzialmente il sistema diventa più vulnerabile e sono possibili attacchi da parte di hacker, spie industriali, sabotatori e inoltre è da tenere presente che gli attacchi non si limitano al furto di documenti e file ma riguardano possibili danni agli impianti e alla reputazione. Questo problema può essere risolto investendo maggiormente sulla cybersecurity, aspetto divenuto di vitale importanza per tutte le imprese moderne. Il mercato della sicurezza informatica e il mercato del Digital Twin sono forte crescita; da qui a cinque anni gli analisti stimano una crescita esplosiva. (Investor Relation Presentation, Annual Reports, www.digital4.biz).

“Inizialmente l’idea del Digital Twin sembrava un concetto teorico.

Successivamente, procedendo con il percorso digitale che abbiamo avviato in Emerson, abbiamo riconosciuto l’importanza della Model-Based-Enterprise (MBE) e abbiamo condotto numerose iniziative per coglierne i vantaggi. [...] Abbiamo investito sulle persone e sul software perché crediamo in questa strategia e desideriamo che il digital thread attraversi l’organizzazione per eliminare la ridondanza e fornire informazioni di qualità su cui poter contare.”

Douglas Hill, Manager, Teamcenter Manufacturing, Emerson Electric. Fonte:

www.eng.it

Visto il maggior utilizzo sono molte oggi le piattaforme di analisi di dati che consentono la creazione di un modello digitale e tra i fornitori principali troviamo MathWorks, Siemens, Cisco, Intel, ma anche molti altri.

Vista l’ampia scelta è dunque fondamentale scegliere il software giusto in base alle proprie esigenze, considerando sia lo scopo finale da raggiungere, alcuni possono essere utilizzati solo in alcuni settori (Circuit Simulator può essere impiegato solo per circuiti elettrici); sia quanto investire in un software di questo tipo (MatLab e Simulink hanno un costo annuo totale di 2000 EUR, Openmodelica è open-source).

2.4 Software di simulazione per la realizzazione di un Digital Twin

Sono qui riportati alcuni esempi di software, tra i più conosciuti e utilizzati, di simulazione e analisi dati che consentono la creazione di un modello digitale.

Matlab e Simulink



Matlab (MATrix LABoratory) è un ambiente di programmazione, sviluppato dalla compagnia statunitense MathWorks, per applicazioni scientifiche, di analisi numerica e per la simulazione di sistemi dinamici.

Matlab contiene un vasto set di funzioni di base e la possibilità di definire nuove funzioni e insieme a Simulink permette la realizzazione di un Digital Twin.

Simulink è un'interfaccia grafica interattiva per la modellazione, la simulazione e l'analisi di sistemi dinamici. Estende le potenzialità di Matlab, aggiungendo molte funzioni specifiche e mantenendo le caratteristiche generali. Viene utilizzato sia nella fase di definizione del modello da simulare che in quella di analisi del sistema stesso. Spesso questi due passi vengono eseguiti sequenzialmente modificando i parametri del sistema al fine di ottenere il comportamento desiderato.

Le ragioni che portano a scegliere Simulink sono prevalentemente la presenza di blocchi predefiniti da connettere, l'elevata flessibilità nella variazione del progetto e la condivisione con Matlab di tutte le variabili definite nel workspace.

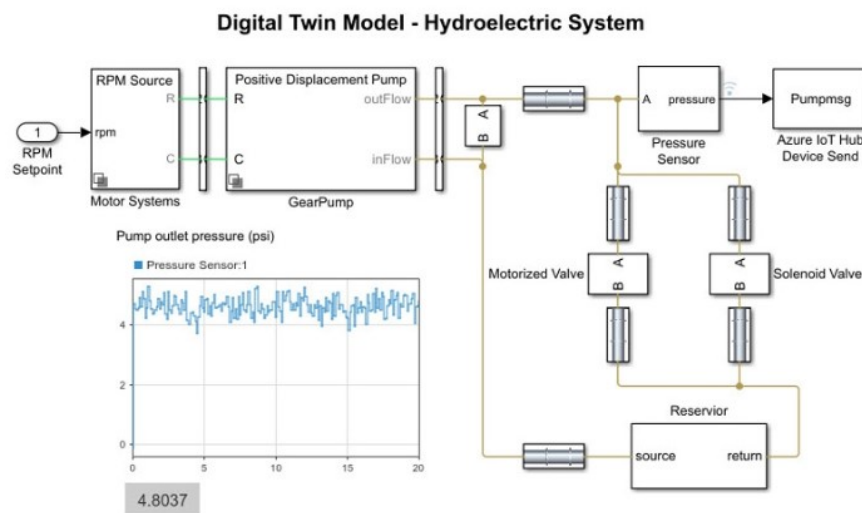


Figura G: Esempio di un Digital twin realizzato con Simulink, (www.mathworks.com).

Società come Huawei e Fujitsu utilizzano questo software per valutare prestazioni e identificare potenziali problemi.

Simulink ha un costo annuo di 1200 EUR e necessita la presenza di MATLAB che ha un costo annuo di 800 EUR.

Siemens NX



Il software Siemens NX è una soluzione integrata, potente e flessibile, che consente di realizzare i prodotti in modo più rapido ed efficiente. NX offre una nuova generazione di soluzioni di progettazione, simulazione e produzione, permettendo alle aziende di realizzare un Digital Twin nel miglior modo possibile.

Supportando ogni singolo aspetto dello sviluppo dei prodotti, dal design del concetto alla progettazione e alla fabbricazione, NX offre un set di strumenti integrati, preserva l'integrità dei dati e la funzionalità di progettazione e semplifica l'operazione.

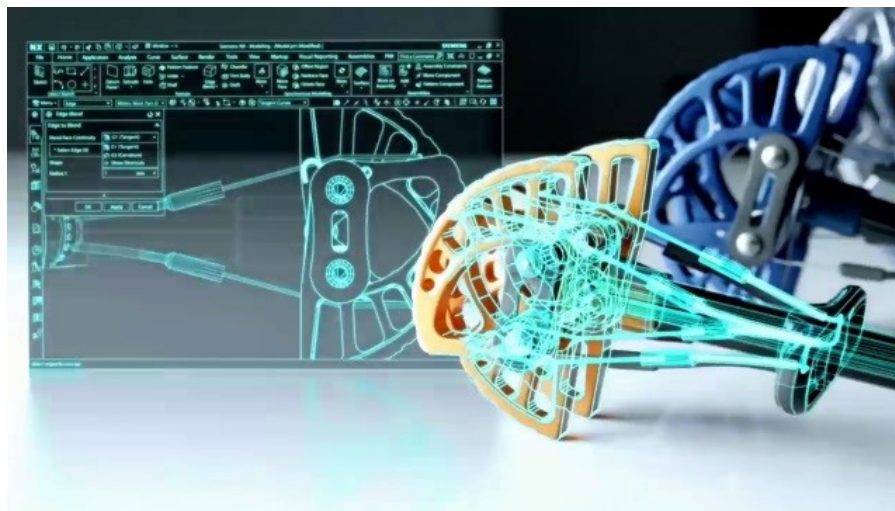


Figura H: Dal modello virtuale al modello reale, (www.plm.automation.siemens.com).

NX consente di creare progetti migliori più velocemente e con maggiore sicurezza, fornisce una comprensione più approfondita del design e una comunicazione più efficace, valuta istantaneamente l'impatto delle modifiche di progettazione e integra gli ultimi aggiornamenti nel flusso di lavoro riducendo al minimo la curva di apprendimento.

La Yamaha Motor per la fase di progettazione e per un miglior utilizzo e condivisione dei dati utilizza Siemens NX.

I vari strumenti che caratterizzano NX hanno un costo annuo compreso tra i 150 e i 700 EUR.

Scilab e Xcos



SciLab è un pacchetto software di computazione numerica open source, il cui sviluppo è portato avanti da Scilab Enterprises che include, all'interno di librerie, centinaia di funzioni utili ai più svariati utilizzi: dal calcolo matematico alla simulazione di sistemi di controllo. Offre inoltre la possibilità di progettare delle personali funzioni in linguaggio C, C++ o Scilab.

Scilab include anche Xcos, editor grafico per la progettazione di modelli di sistemi dinamici ibridi. I modelli possono essere progettati, caricati, salvati, compilati e simulati.

Soluzione ergonomica ed efficiente per esigenze industriali e accademiche, Scilab e Xcos erano inizialmente pensati come strumenti di progettazione e analisi del sistema di controllo, è stato tuttavia sfruttato anche in molti altri campi come la modellazione di sistemi meccanici (automobilistico, aeronautico) e di circuiti idraulici (diga, modellazione di tubi ...) grazie anche al supporto di Modelica, ma rimane comunque uno strumento adattato alle esigenze di controllo.

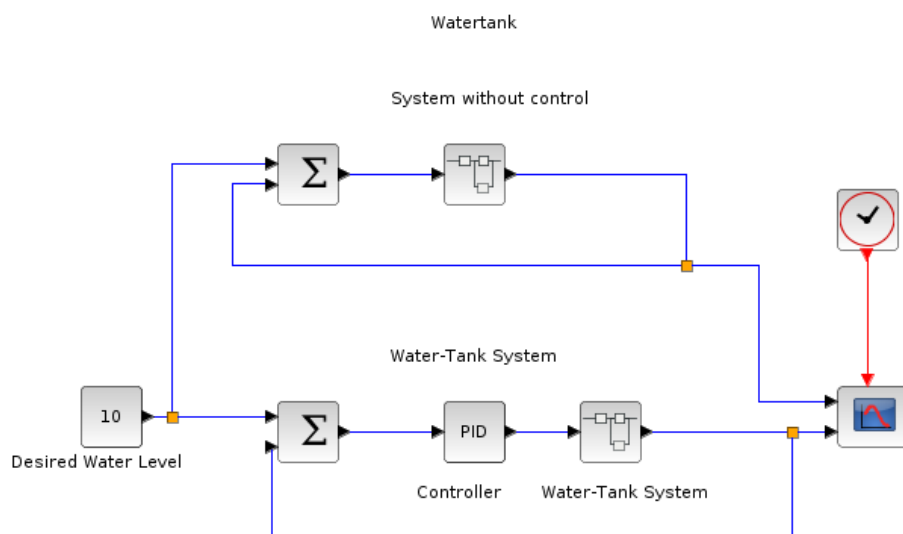


Figura I: Digital twin di un sistema di controllo realizzato con Xcos, (www.scilab.org).

Grazie alla sua capacità di interconnessione con tecnologie e applicazioni di terze parti, Scilab può anche fungere da piattaforma unica per riunire codici scritti in diversi linguaggi di programmazione in un unico linguaggio unificato, facilitando così la loro distribuzione, il loro backup e utilizzo.

Essendo un software open-source alcuni problemi non sempre trovano una soluzione immediata mentre alcuni comandi simili hanno sintassi completamente diverse fra loro.

Xcos è disponibile gratuitamente, ma necessita di SciLab anch'esso distribuito gratuitamente.

3 Openmodelica

OpenModelica

Un approfondimento maggiore è dedicato a Openmodelica, software basato sul linguaggio di programmazione Modelica, utilizzato per modellare, simulare, ottimizzare e analizzare sistemi dinamici complessi. È sviluppato attivamente dal Consorzio Open Source Modelica, un'organizzazione non governativa senza scopo di lucro. Fra i vari utilizzi, può essere impiegato per la costruzione di circuiti analogici e digitali, macchine per la modellazione di motori, generatori elettrici, per modellare un flusso monodimensionale in una rete di tubi o per risolvere equazioni. I vantaggi principali di Openmodelica risiedono nell'essere un software open-source facile da utilizzare e nell'impiegare parametri che possono essere facilmente ricavati, il che rende il software particolarmente adatto all'ambito accademico ma anche in quello industriale.

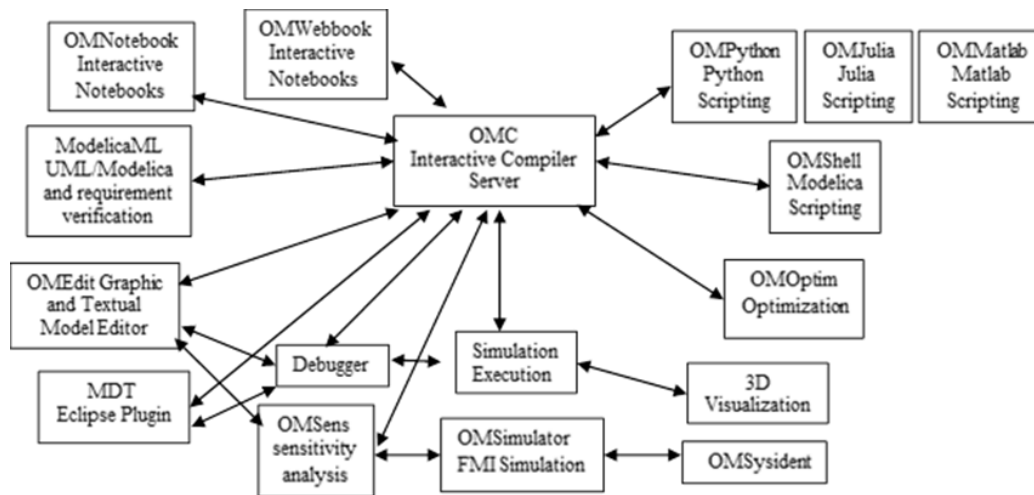


Figura J: The architecture of the OpenModelica environment, (www.openmodelica.org).

Gli strumenti che costituiscono questo software sono diversi, alcuni sono essenziali e alla base di ogni operazione:

- (OMC) Advanced Interactive OpenModelica Compiler. OMC è un compilatore Modelica, che traduce il linguaggio Modelica in codice C, con una tabella di simboli contenente definizioni di classi, funzioni e variabili. Tali definizioni possono essere predefinite, definite dall'utente o ottenute da librerie;
- (OMShell) Interactive OpenModelica Shell, il suo scopo è quello di fornire un ambiente interattivo per lavorare con OMC;

- (OMEdit), OpenModelica Connection Editor. OMEdit è un'interfaccia grafica utente intuitiva e avanzata che offre agli utenti la creazione di modelli di facile utilizzo, la modifica delle connessioni, la simulazione di modelli e la stampa di risultati.

Altri strumenti sono usati per applicazioni specifiche:

- OMOptim, sottosistema della piattaforma OpenModelica facilita l'ottimizzazione in ingegneria. Può essere utilizzato per aumentare l'efficienza energetica di un processo industriale o per diverse attività come l'ottimizzazione della configurazione (ad es. scelta ottimale dei componenti), ottimizzazione dei parametri, ottimizzazione del controllo dinamico.
- (OMPython) OpenModelica Python Interface, sottosistema della piattaforma OpenModelica, che consente agli utenti di sperimentare le capacità di modellazione e simulazione di all'interno dell'ambiente Python.

Grazie alla presenza di diverse librerie, Openmodelica offre un ambiente nel quale è possibile costruire sistemi diversi come un circuito elettrico e un impianto idraulico ma offre anche la possibilità di far interagire questi due sistemi fra loro, ed è proprio con i fluidi che questo software mostra maggiormente come sia immediata la costruzione di un impianto e simulare il suo funzionamento.

3.1 Progettare un impianto idraulico

Come detto precedentemente, Openmodelica risulta particolarmente adatto alla definizione di un impianto idraulico, è possibile ad esempio valutare come cambia il volume di un liquido contenuto in tre serbatoi collegati fra loro e dopo quanto tempo si raggiunge una condizione di equilibrio.

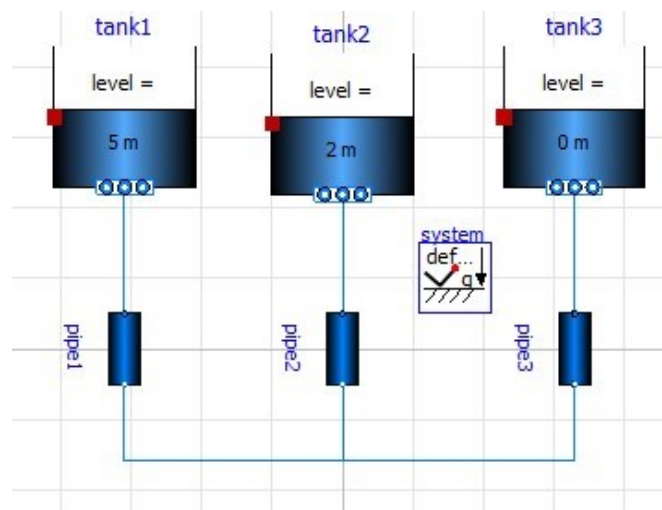
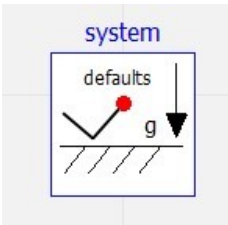


Figura K: modello Tanks realizzato tramite OMEdit

SIMBOLO	INDICE	DESCRIZIONE
	System	<p>Nome: System Library: Fluid</p> <p>È necessario un componente di sistema in ciascun modello di fluido per fornire impostazioni a livello di sistema, come condizioni ambientali e ipotesi generali di modellazione. Le impostazioni del sistema vengono propagate ai modelli fluidi mediante il meccanismo interno / esterno.</p>

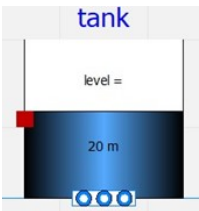

SIMBOLO	INDICE	DESCRIZIONE
	Tank	<p>Nome: Opentank Package: Vessels Library: Fluid</p> <p>Modello di un serbatoio aperto all'ambiente a pressione fissa p_{ambient}.</p>
	Pipe	<p>Nome: StaticPipe Package: Pipes Library: Fluid</p> <p>Modello di un tubo diritto con sezione trasversale costante e con bilanci di massa, momento ed energia stazionari, ovvero il modello non immagazzina massa o energia.</p>

Tabella 1: legenda modello Tanks

3.1.1 Prime operazioni

La prima cosa da fare, dopo aver aperto il software, è creare una nuova classe modelica, indicando il nome che vogliamo dare al nostro impianto, ad esempio “tanks” (operazione comune per qualsiasi tipologia di impianto si voglia realizzare). A questo punto si aprirà OMEdit, che all’interno del software prende il nome di “Vista diagramma” nel quale tramite le librerie a disposizione è possibile costruire graficamente l’impianto. Da questa finestra è possibile inoltre passare a OMShell, tramite l’icona “Text View”.

Prima di passare alla modellazione dell’impianto è importante definire nell’OMShell quale fluido usare. All’interno di Openmodelica i fluidi prendono il nome di media e possono essere costituiti da una o più fasi. Fra questi è possibile usare l’aria sotto forma di aria secca (SimpleAir), aria secca come gas ideale (DryAirNasa) o aria umida (MostAir); liquidi comprimibili, liquidi incomprimibili, fluido refrigerante R-134 e acqua sotto forma di acqua liquido semplice (Constant Property Liquid Water), acqua come gas ideale (IdealSteam), e acqua ad alta precisione (Modelli derivati da WaterIF97).

È possibile inoltre aggiungere nuovi modelli di fluido tramite librerie aggiuntive oppure tramite un’implementazione autonoma, in questo caso bisogna definire tutte le caratteristiche del fluido ed avere un giusto livello di dimestichezza con OMShell.

Definiamo in questo caso acqua con proprietà costanti nel tempo (Constant Property Liquid Water), tramite la funzione “replaceable package”

```
model tanks
  replaceable package Medium =
    Modelica.Media.Water.ConstantPropertyLiquidWater
```

Figura L: definizione media, “Text View”

Il tipo di fluido utilizzato andrà poi dichiarato per ogni elemento con cui viene a contatto tramite la funzione “**redeclare package** Medium =Medium”.

A questo punto è possibile tornare nell’ OMEdit e inserire i componenti che costituiranno il modello “tanks”.

3.1.2 Costruzione grafica e inserimento parametri impianto

Tramite OMEdit, dalle librerie a disposizione, consultabili tramite il navigatore di librerie, solitamente presente a sinistra, viene costruito l’impianto utilizzando modelli predefiniti. Essi vengono inseriti trascinando i modelli con il mouse all’interno del diagramma. Il primo componente da inserire quando si lavora con i fluidi è il componente di sistema (System), per fornire impostazioni a livello di sistema, come condizioni ambientali e ipotesi generali di modellazione.

Gli unici altri componenti di cui abbiamo bisogno in questo caso sono i serbatoi aperti (opentank) e i condotti (pipe) i quali troviamo entrambi nella libreria Fluid rispettivamente nei package “Vessels” e “Pipes”.

Per ciascun componente inserito devono essere dichiarati i parametri che li caratterizzano e ciò può essere fatto premendo il pulsante destro del mouse con il cursore posizionato su di un componente per poi aprire la finestra parametri.

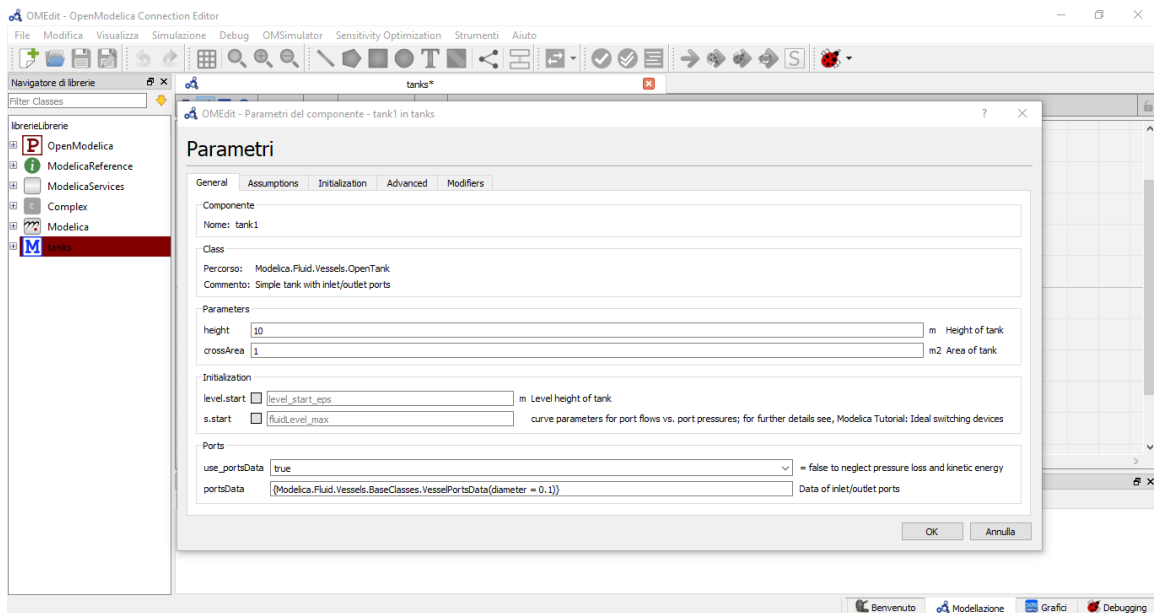


Figura M: dichiarazione parametri per il componente Tank1 del modello “tanks”.

Oltre ad essi è importante definire per ogni componente anche le condizioni iniziali e porre attenzione su quali vengono inserite, devono infatti essere coerenti fra loro e permettere una corretta simulazione.

Tutto ciò che viene inserito nell’ OMEdit, viene automaticamente riportato nell’OMShell, fino ad avere definito tutti i valori che permettono il funzionamento.

È possibile anche inserire parametri direttamente nell'OMShell, in questo caso però bisogna rispettare le regole del linguaggio di programmazione

```
Modelica.Fluid.Vessels.OpenTank tank1(  
  redeclare package Medium =Medium,  
  crossArea = 1, height = 10, level_start = 5,nPorts= 1, portsData =  
{Modelica.Fluid.Vessels.BaseClasses.VesselPortsData(diameter = 0.1)}, use_portsData = true) annotation(  
  Placement(visible = true, transformation(origin = {-62, 66}, extent = {{-20, -20}, {20, 20}}, rotation =  
0)));
```

Figura N: caratteristiche tank1, Text View

Altro elemento importante è il collegamento tra i vari componenti, ognuno di essi presenta uno o più ingressi per comunicare con l'esterno, alcuni per specifici utilizzi come un segnale in ingresso che permetta ad una valvola il passaggio del fluido oppure il trasferimento di calore. Gli ingressi sono indicati come “port”, port_a, port_b e così via e i collegamenti vengono realizzati con il cursore del mouse.

Bisogna inoltre porre attenzione nel momento in cui il modello viene modificato, se ad esempio al serbatoio 1, venisse aggiunto un secondo ingresso, nell' OMShell il parametro del tank1 “nPorts” andrebbe corretto inserendo 2 al posto di 1.

L'ultimo passaggio prevede la verifica del modello, verificare cioè che non siano presenti più incognite delle equazioni, non siano presenti condizioni iniziali in contrasto fra loro e che tutti i componenti siano definiti nel modo giusto, per poi avviare la simulazione e ottenere grafici sull'andamento dei parametri.

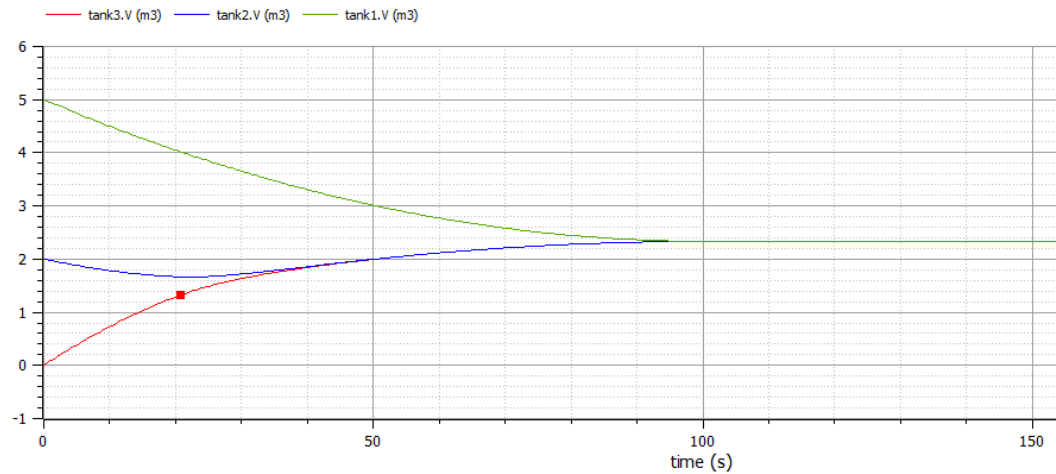


Figura O: andamento del volume presente all'interno dei serbatoi Tank1, Tank2, Tank3 del modello "tanks".

L'impianto fin'ora preso in esame , può essere leggermente modificato inserendo una valvola tra i serbatoi 2 e 3 che impedisce il passaggio di acqua per i primi 50 secondi.

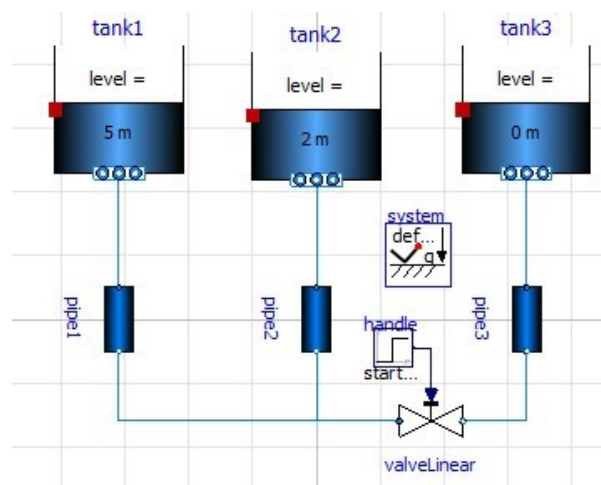
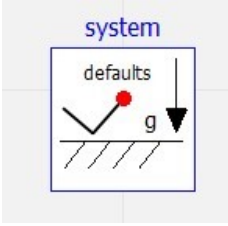
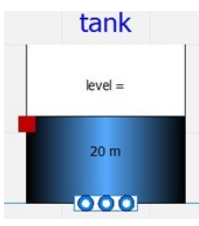
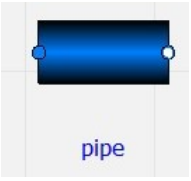
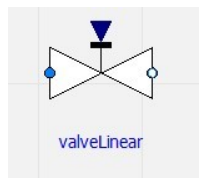


Figura P: modello "tanks" con aggiunta di una valvola

SIMBOLO	INDICE	DESCRIZIONE
	System	<p>Nome: System Library: Fluid</p> <p>È necessario un componente di sistema in ciascun modello di fluido per fornire impostazioni a livello di sistema, come condizioni ambientali e ipotesi generali di modellazione. Le impostazioni del sistema vengono propagate ai modelli fluidi mediante il meccanismo interno / esterno.</p>
	Tank	<p>Nome: Opentank Package: Vessels Library: Fluid</p> <p>Modello di un serbatoio aperto all'ambiente a pressione fissa p_{ambient}.</p>
	Pipe	<p>Nome: StaticPipe Package: Pipes Library: Fluid</p> <p>Modello di un tubo dritto con sezione trasversale costante e con bilanci di massa, momento ed energia stazionari, ovvero il modello non immagazzina massa o energia.</p>
	ValveLinear	<p>Nome: ValveLinear Package: Valves Library: Fluid</p> <p>Questo modello molto semplice di valvola fornisce una caduta di pressione proporzionale alla portata in ingresso, senza calcolare alcuna proprietà del fluido. Può essere utilizzato a scopo di test, quando è necessario un semplice modello di perdita di pressione variabile.</p>

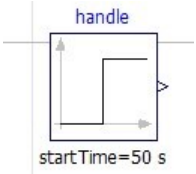
SIMBOLO	INDICE	DESCRIZIONE
	Handle	Nome: Step Package: Sources Library: Blocks Funzione Step.

Tabella 2: legenda modello Tanks con valvola.

Per l’inserimento della valvola, bisogna scegliere quella più opportuna tra quelle presenti nel package “Valves” e inserire un segnale di ingresso che permetta l’apertura della valvola dopo un determinato tempo, può essere ad esempio usata la funzione “Step” indicando come valore al parametro Startime 50 secondi. La funzione “Step” è presente nella libreria Blocks nel package “Sources” e interagisce con la valvola grazie all’ingresso “RealInput” presente sulla valvola stessa.

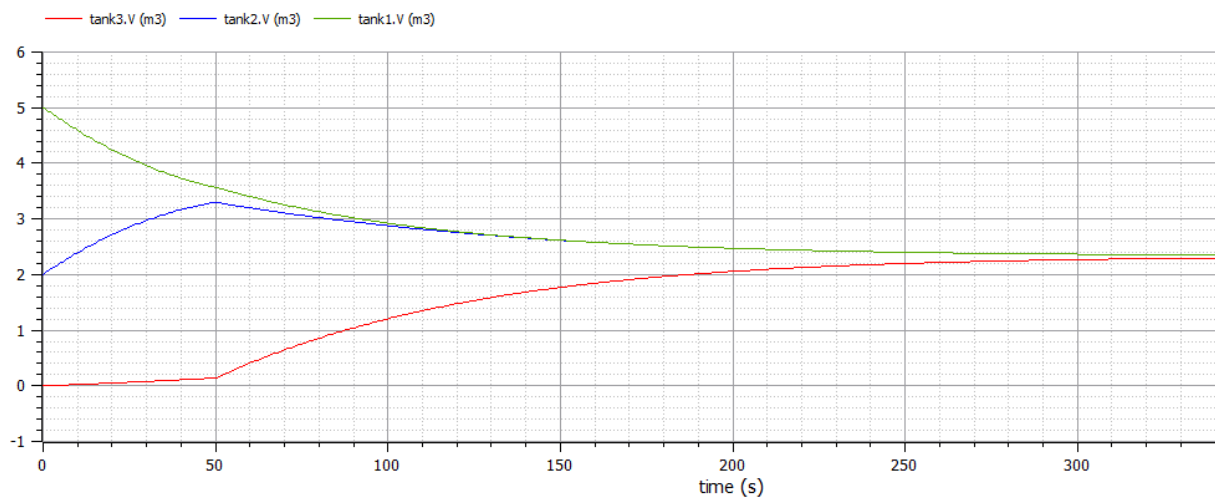


Figura Q: andamento del volume presente all'interno dei serbatoi Tank1, Tank2, Tank3 del modello "tanks" con l'aggiunta di una valvola.

Il risultato finale è lo stesso ma la condizione di equilibrio viene raggiunta con più tempo.

3.1.3 Inserimento sensori

È possibile complicare ulteriormente l'impianto inserendo altri componenti e sensori che permettono di mantenere sotto controllo parametri come la pressione, la temperatura e la portata massica e ottenendo grafici che non solo forniscono informazioni sull'andamento nel tempo, ma permettono di confrontare i diversi valori ottenuti con sensori posizionati i punti diversi dell'impianto.

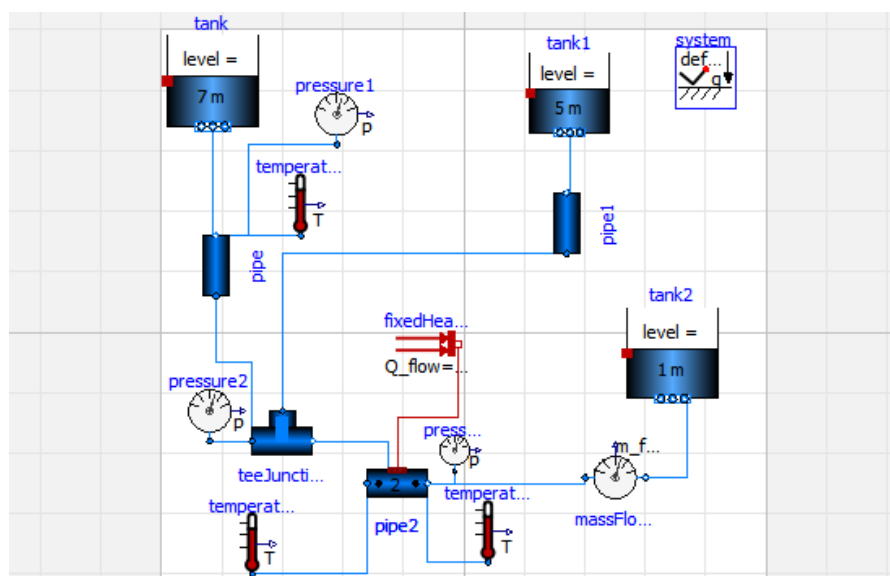
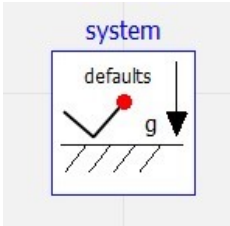
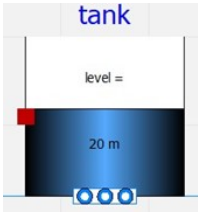

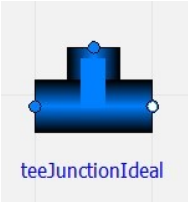
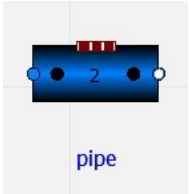


Figura R: modello “Tanks2” realizzato con OMEdit.

SIMBOLO	INDICE	DESCRIZIONE
	System	<p>Nome: System Library: Fluid</p> <p>È necessario un componente di sistema in ciascun modello di fluido per fornire impostazioni a livello di sistema, come condizioni ambientali e ipotesi generali di modellazione. Le impostazioni del sistema vengono propagate ai modelli fluidi mediante il meccanismo interno / esterno.</p>
	Tank	<p>Nome: Opentank Package: Vessels Library: Fluid</p> <p>Modello di un serbatoio aperto all'ambiente a pressione fissa p_{ambient}.</p>

SIMBOLO	INDICE	DESCRIZIONE
 <p>pipe</p>	Pipe	<p>Nome: StaticPipe Package: Pipes Library: Fluid</p> <p>Modello di un tubo dritto con sezione trasversale costante e con bilanci di massa, momento ed energia stazionari, ovvero il modello non immagazzina massa o energia.</p>
 <p>teeJunctionIdeal</p>	TeeJunctionIdeal	<p>Nome: TeeJunctionIdeal Package: Fittings Library: Fluid</p> <p>Questo modello è l'implementazione più semplice per un componente di divisione / unione per tre flussi. Non è richiesto il suo utilizzo. Formula semplicemente le equazioni di equilibrio nello stesso modo in cui la semantica di connessione le formulerebbe comunque. Il vantaggio principale dell'utilizzo di questo componente è che l'utente non viene confuso quando osserva l'entalpia specifica in ciascuna porta, il che potrebbe creare confusione quando non utilizza un componente di divisione / unione.</p>
 <p>pipe</p>	Pipe	<p>Nome: DynamicPipe Package: Pipes Library: Fluid</p> <p>Modello di un tubo dritto con bilanci distribuiti di massa, energia e quantità di moto. Fornisce le equazioni di bilanciamento complete per il flusso del fluido unidimensionale. Questo modello generico offre un gran numero di combinazioni di possibili impostazioni dei parametri.</p>

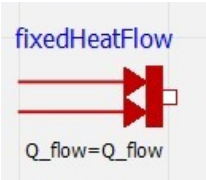
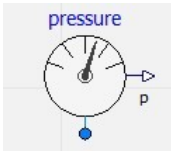
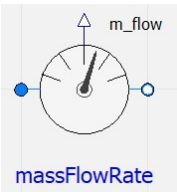
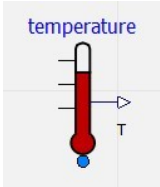
SIMBOLO	INDICE	DESCRIZIONE
	FixedHeatFlow	<p>Nome: FixedHeatFlow Package: HeatTransfer.Sources Library: Thermal</p> <p>Questo modello consente di "iniettare" una determinata quantità di portata di calore in un sistema termico in una determinata porta. La quantità costante di portata di calore Q_flow è indicata come parametro. Il calore fluisce nel componente a cui è collegato il componente FixedHeatFlow, se il parametro Q_flow è positivo.</p>
	Pressure	<p>Nome: Pressure Package: Sensors Library: Fluid</p> <p>Questo componente controlla la pressione assoluta nella sua porta del fluido. Il sensore è ideale, ovvero non influenza il fluido.</p>
	MassFlowRate	<p>Nome: MassFlowRate Package: Sensors Library: Fluid</p> <p>Questo componente controlla la portata massica che scorre da port_a a port_b. Il sensore è ideale, ovvero non influenza il fluido.</p>
	Temperatura	<p>Nome: Temperatura Package: Sensors Library: Fluid</p> <p>Questo componente controlla la temperatura del fluido che passa attraverso la sua porta. Il sensore è ideale, ovvero non influenza il fluido.</p>

Tabella 3: legenda modello Tanks2.

In questo modello le difficoltà maggiori sono date, oltre che dall'inserimento della fonte di calore che per essere collegata ad un condotto ha bisogno di un "DynamicPipe", dal giusto inserimento dei sensori.

Anche in questo caso per ogni sensore bisogna definire nell'OMShell con quale fluido vengono a contatto e attribuire una giusta nomenclatura agli output per ottenere grafici chiari e comprensibili. Inoltre grandezze diverse vengono misurate diversamente, essendo la portata massica una grandezza di flusso, il sensore "massFlowRate.m_flow" ha un'ingresso e un'uscita e quindi un output positivo se il flusso entra da "ports_a" ed esce da "ports_b", negativo viceversa.

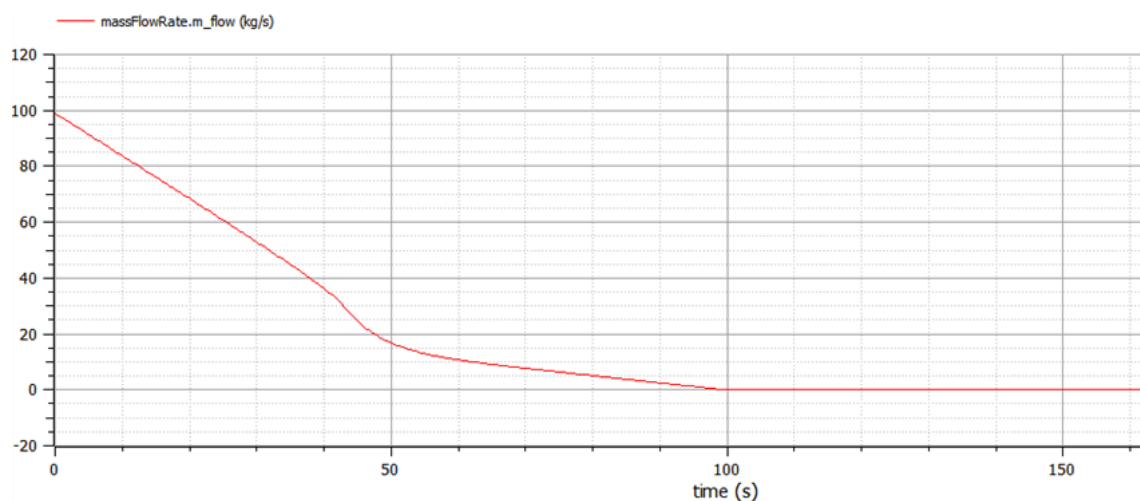


Figura 5: andamento portata massica misurato con il sensore "massFlowRate".

Nel modello “Tanks2” a cento secondi la portata massica scende a zero e rimane costante poiché per ogni componente presente è stata inserita l’opzione “allowFlowReversal=false”, cioè il fluido può muoversi in un’unica direzione, come se fossero presenti una serie di valvole di non ritorno.

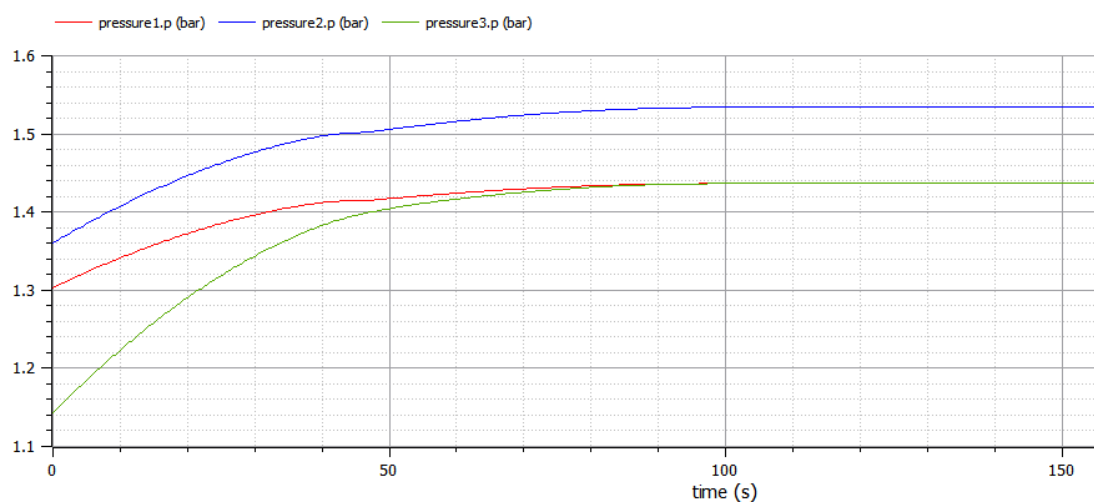


Figura T: andamento pressione misurato con i sensori: pressure1, pressure2, pressure3.

Questi sensori sono però da considerare come ideali, nella realtà la presenza di sensori può portare a perdite di carico, anche se solitamente trascurabili.

Openmodelica permette dunque non solo la progettazione di un impianto idraulico ma anche la possibilità di valutare e conoscere come variano le caratteristiche dei fluidi rispetto alle condizioni iniziali, dato fondamentale per la scelta dei materiali e dei componenti che andranno a costituire l’impianto fisico.

È possibile, ad esempio, valutare come varia l'andamento della temperatura misurato dai tre sensori “temperatura1”, “temperatura2” e “temperatura3” raddoppiando la quantità di calore fornito al condotto 2 (pipe2).

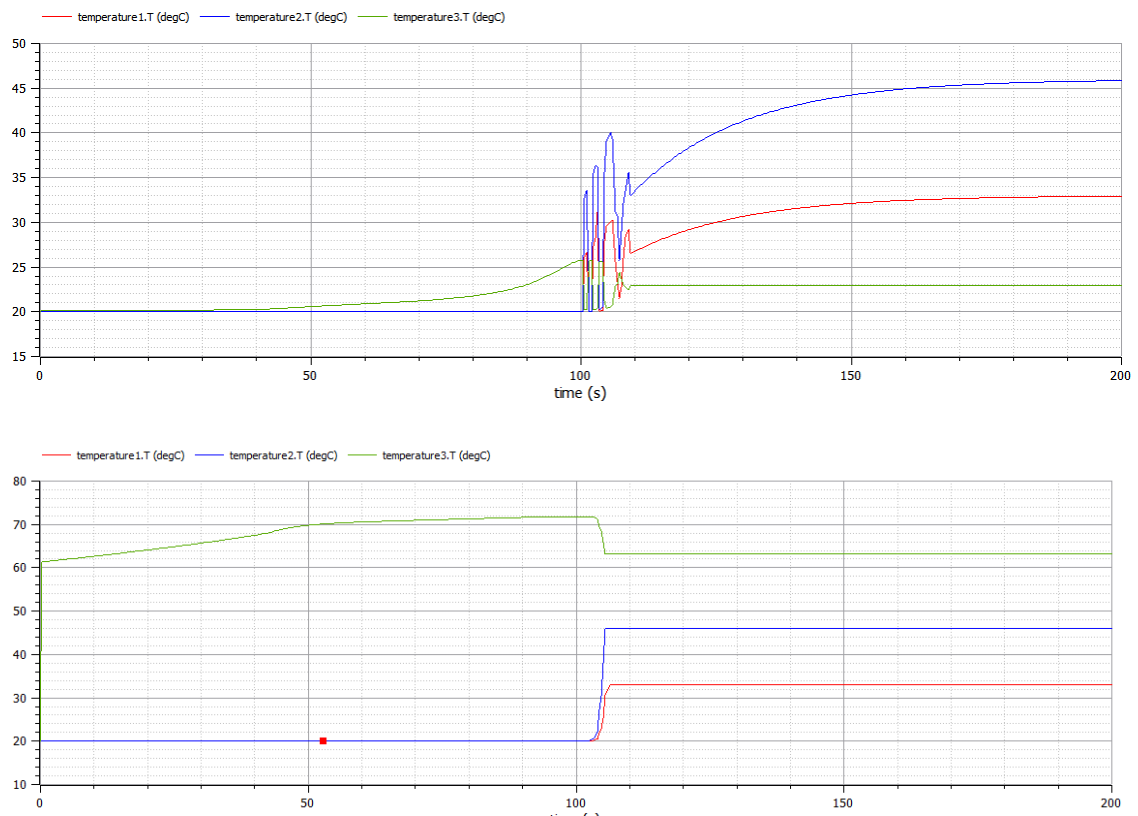


Figura U: variazioni di temperatura misurate dai sensori temperatura1, temperatura2, temperatura3, fornendo prima una quantità di calore pari a 1.6W e poi 3.2 W.

I componenti e i sensori che possono essere inseriti sono molti di più di quelli visti e possono, gran parte di essi, interagire anche con componenti non appartenenti alla libreria Fluid.

Il funzionamento di una pompa può essere ad esempio regolato da una funzione trapezoidale mentre il segnale in uscita di un sensore può regolare, tramite interpolazioni matematiche, il funzionamento di una valvola.

Pertanto, nella realizzazione di modelli come quelli visti, Openmodelica risulta essere un software molto utile e veloce, ma presenta al suo interno delle problematiche che non sempre trovano soluzione immediata: nelle librerie non troviamo componenti come compressori, eiettori o evaporatori, la possibilità di miscelare fluidi diversi o inserire condizioni ambientali che variano nel tempo. Alcuni di questi problemi trovano soluzione tramite librerie aggizionali che possono essere aggiunte alle librerie di partenza, altri per il momento non hanno ancora soluzione.

4 Digitalizzare un impianto

Openmodelica oltre alla fase di progettazione, risulta molto utile anche nella fase di analisi di un impianto. Quello che viene fatto in questo caso è digitalizzare l'impianto già esistente e confrontare i parametri ottenuti nella simulazione con quelli ricavati nella realtà. Essendo, come detto precedentemente, un software che non è in grado di riprodurre tutti i casi reali (senza il supporto di altre estensioni), per realizzare un Digital Twin bisogna: studiare l'impianto fisico, confrontare gli elementi che lo caratterizzano con Openmodelica ed eventualmente sfruttare le proprie conoscenze per risolvere problemi di incongruenza.

Un esempio è la realizzazione di un gemello digitale di un sistema di pompaggio ad eiettore per il trasporto di fluidi bifase gas-liquido situato all'interno dell'Università Politecnica delle Marche.

4.1 Descrizione impianto

L'impianto simula una classica situazione che si presenta nei processi di estrazione, cioè quella di sfruttare la pressione di un giacimento la cui pressione è più alta di quella di trasporto per creare un'aspirazione su un giacimento la cui pressione non è sufficientemente alta per il trasporto sulla linea. Nello specifico il fluido motore, cioè quello a pressione più alta, sarà un liquido (petrolio) mentre il secondo fluido, cioè quello aspirato sarà un gas (metano). Nell'impianto sperimentale saranno utilizzati acqua e aria.

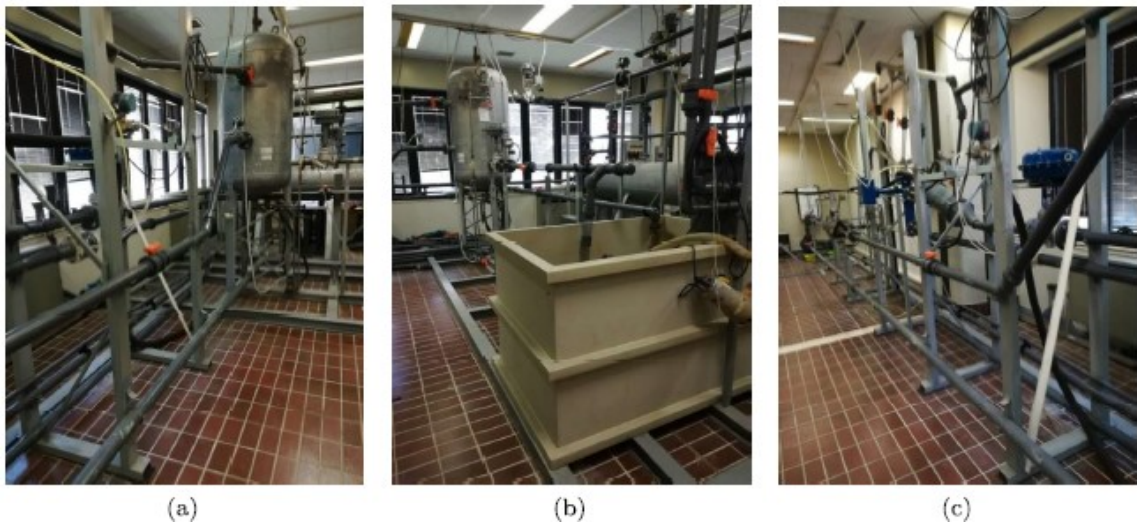


Figura V: foto sistema di pompaggio ad eiettore

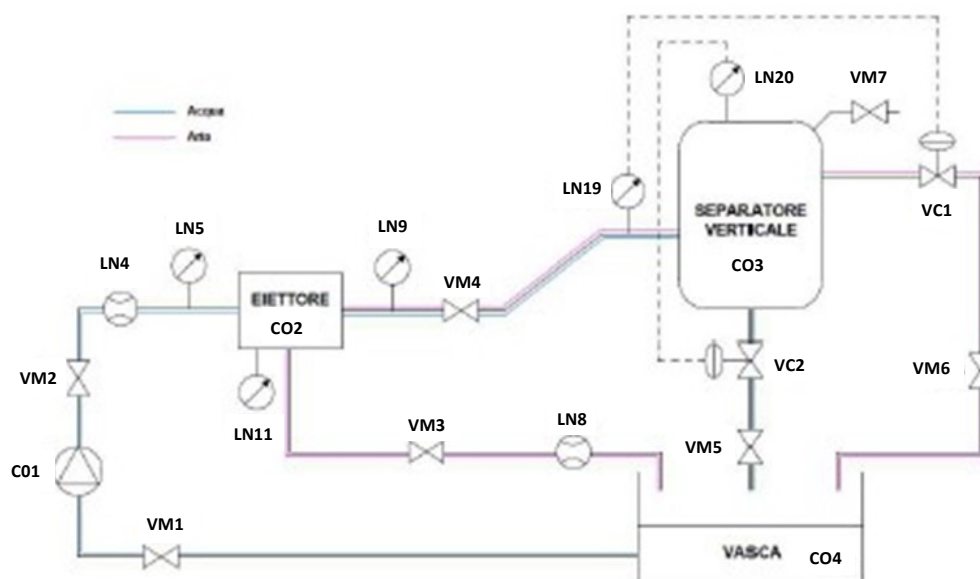


Figura W: schema impianto

Indice	Descrizione
CO1	Pompa centrifuga monoblocco Vergani 32-201
CO2	Eiettore
CO3	Serbatoio/Separatore verticale Elbi 780-I
CO4	Serbatoio Aperto/Vasca
LN4	Sensore portata Foxboro Magnetic Flowtransmitter
LN5/LN11/LN9	Sensori pressione assoluta Setra 280E

Indice	Descrizione
LN19	Sensore pressione relativa Foxboro 841GM-CI1
LN20	Sensore pressione differenziale Foxboro IDP-10
LN8	Sensore portata volumetrica Foxboro Vortex DN 50
VC1/VC2	Elettrovalvole pneumatiche ECKARDT MB6713
VM1/VM2/VM3 VM4/VM5/VM6	Valvole intercettazione 2 vie FIP DN 50 PN 16
VM7	Valvola di sicurezza

Tabella 4: legenda schema impianto

Come è possibile vedere dalla fig.W, gli elementi principali che costituiscono l'impianto sono: un serbatoio aperto, una pompa, un eiettore e un serbatoio chiuso che funge da separatore verticale.

La pompa, collegata al serbatoio aperto, preleva una determinata portata di acqua e la invia, ad una certa pressione, all'eiettore, dove l'energia di pressione del liquido viene trasformata in energia cinetica.

La depressione che ne risulta richiama dall'esterno una portata di aria, alla pressione atmosferica. All'uscita dell'eiettore i due fluidi vengono a contatto e si ha il recupero di pressione della miscela.

La miscela bifase formata all'uscita dell'eiettore confluisce nel serbatoio chiuso che funge da separatore verticale e tramite due elettrovalvole è possibile controllare il flusso di acqua e aria in uscita. L'impianto inoltre è dotato di una serie di sensori che permettono di monitorare il processo. I suddetti sensori sono tutti di tipo analogico e la loro lettura è totalmente gestita dalla piattaforma hardware Arduino Mega 2560 la quale integra un microcontrollore ATmega2560. Oltre alla lettura dei sensori, la piattaforma Arduino ha il compito di controllare le elettrovalvole in modo da mantenere una pressione ed un livello nel serbatoio prestabiliti.

4.2 Componenti idraulici e sensori

Di seguito sarà fatta una descrizione di tutte le componenti dell'impianto e sulla loro compatibilità con il software Openmodelica.

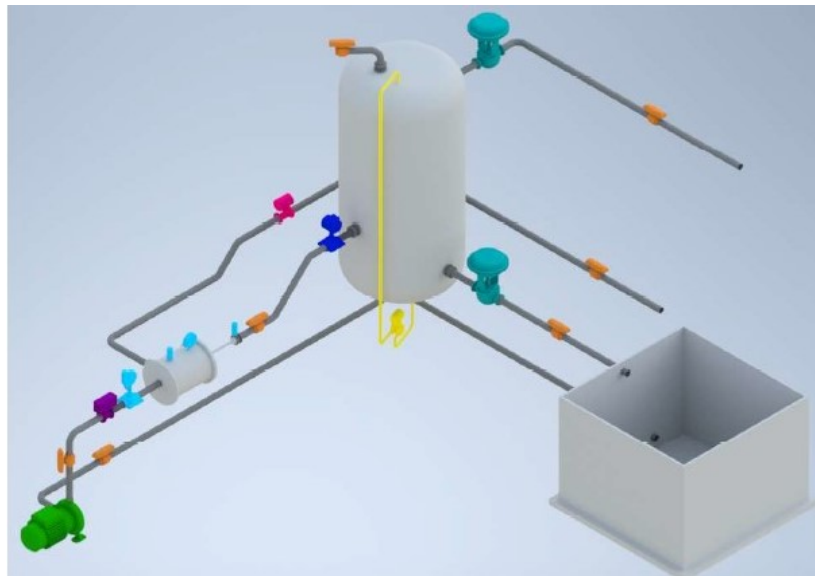


Figura X: impianto 3D

a) Serbatoio aperto

Il serbatoio aperto rappresenta il punto di partenza e di arrivo dell'intero ciclo dell'acqua. Da questo serbatoio, la pompa preleva acqua e la invia all'eiettore per poi confluire all'interno del serbatoio chiuso, dove è presente un punto di uscita per la componente liquida che tornerà nel serbatoio aperto di partenza.



Figura a: serbatoio aperto

All'interno di Openmodelica il serbatoio aperto è ben rappresentato dal componente “OpenTank”, modello di un serbatoio aperto all'ambiente con le seguenti ipotesi:

- Il serbatoio è riempito con un mezzo a sostanza singola o multipla avente una densità superiore alla densità del mezzo ambientale;
- Il fluido ha densità, temperatura e frazioni di massa uniformi;
- Nessun liquido lascia il serbatoio attraverso la parte superiore aperta; la simulazione si interrompe se il livello del liquido aumenta oltre l'altezza.

b) Pompa

La pompa presente nell'impianto è una pompa centrifuga monoblocco flangiata direttamente al motore. Il corpo è diviso verticalmente, con la bocca aspirante assiale e quella premente radiale rivolta verso l'alto; l'albero motore prolungato è supportato da cuscinetti a sfere lubrificati a grasso, che non necessitano di lubrificazione periodica. La girante è calettata direttamente sull'albero motore.

Con Openmodelica questo elemento può essere rappresentato dal componente "ControlledPump", pompa centrifuga con portata massica o pressione idealmente controllate. I valori nominali vengono utilizzati per predefinire le caratteristiche di una pompa esemplare e per definire il funzionamento della pompa.

CARATTERISTICHE POMPA

MODELLO	Vergani 32-201
GIRI AL MINUTO	2900 rpm
CAPACITÀ	233 l/min
PORTATA	14 m ³ /h
POTENZA	7,5 Kw
PREVALENZA	54,25 m



Figura b: pompa Vergani 32-201

c) Eiettore

L'eiettore è un dispositivo molto simile ad una pompa, con due sostanziali differenze:

- Assenza di parti in movimento
- È alimentato dal flusso di un fluido

La mancanza di parti in movimento e la conseguente estrema semplicità dell'apparato, i bassi costi di manutenzione e di installazione e l'assenza di problemi di tenuta, fanno dell'eiettore gas-liquido una macchina che può offrire notevoli possibilità di applicazione nel trasporto sia di miscele di idrocarburi che di gas corrosivi, tossici o radioattivi. Il principio di funzionamento dell'eiettore è basato sul fenomeno idrodinamico detto effetto Venturi, per cui la pressione di una corrente fluida aumenta con il diminuire della velocità. Ciò che succede all'interno dell'eiettore è rappresentato in maniera semplificata nella fig.c2. Il fluido motore (acqua nel caso in esame) muovendosi all'interno dell'ugello, il quale provocherà una riduzione della sezione di passaggio, subirà, per il teorema di Bernoulli un aumento di velocità e pertanto una riduzione di pressione.

$$P + \frac{\rho v^2}{2} + \rho gh = \text{costante}$$

Teorema di Bernoulli

P	Pressione
ρ	Densità
v	Velocità
g	Accelerazione gravitazionale
h	Altezza



Figura c1: eiettore

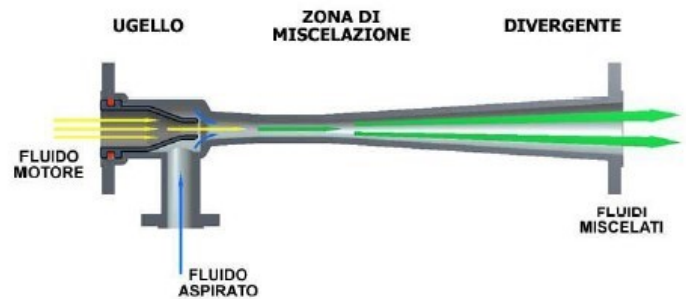


Figura c2: schema semplificato funzionamento eiettore

Il fluido fuoriuscito dall'ugello è caratterizzato da un'alta velocità tale per cui, secondo l'effetto Venturi, si crea una depressione nella camera di miscelazione. Questa depressione promuove l'ingresso del fluido di aspirazione (aria nel caso in esame). Una volta miscelati, l'elevata quantità di moto del fluido motore si trasmette al fluido di alimentazione che ha una bassa quantità di moto. Attraverso il cono divergente, la miscela viene rallentata e l'energia cinetica è riconvertita in energia statica di pressione. Le librerie standard di Openmodelica non presentano l'eiettore come componente.

Per la sua realizzazione, o comunque per simulare il suo funzionamento, è possibile utilizzare una funzione matematica che regola il flusso dell'aria in funzione della portata e della pressione dell'acqua prima dell'eiettore e la pressione dell'acqua nel serbatoio chiuso.

$$-0,60(Pressione\ H_2O[bar]) - 0,87(Pressione\ serbatoio\ chiuso[bar]) + 1,75\left(Portata\ H_2O\left[\frac{m^3}{h}\right]\right) = Portata\ aria\left[\frac{m^3}{h}\right]$$

In particolare, vengono utilizzati gli output di sensori di pressione e portata opportunamente posizionati all'interno del modello, moltiplicati con coefficienti che prendono in considerazione l'equazione vista e le unità di misura, sommati fra loro considerando i segni e utilizzati come input per un componente impiegato come fonte per un fluido "MassFlowSource_T". I passaggi matematici sono realizzati tramite i componenti "Product", funzione che calcola l'uscita y come prodotto dei due ingressi u1 e u2: $y = u1 * u2$; "Constant", funzione con valore costante; "MultiSum", funzione che calcola l'uscita reale scalare "y" come somma degli elementi del vettore di segnale di ingresso reale u, $y = k[1] * u[1] + k[2] * u[2] + \dots k[N] * u[N]$.

L'output di "MultiSum" è in $\left[\frac{m^3}{h}\right]$ e necessita la conversione in $\left[\frac{kg}{s}\right]$, cioè da portata volumetrica in portata massica, per poter essere utilizzato come segnale in ingresso.

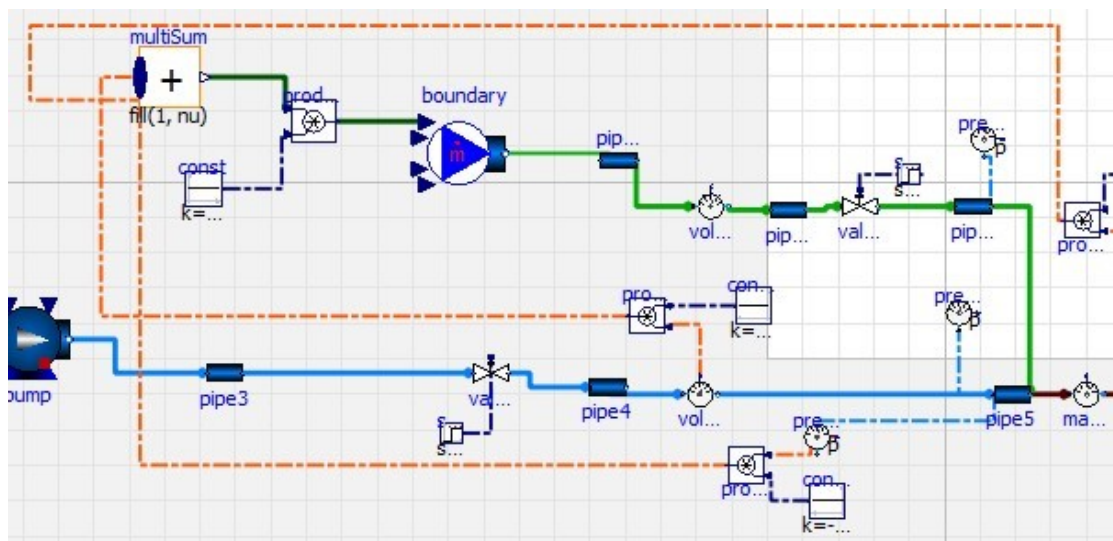


Figura Y: sostituzione eiettore tramite funzione matematiche

d) Serbatoio

Serbatoio chiuso adatto all'accumulo di acqua in pressione che svolge anche il compito di separatore verticale. Il serbatoio è dotato di un punto di accesso per il fluido bifase in uscita dall'eiettore, un punto di uscita della componente liquida, un punto di uscita della componente gassosa e un punto di uscita collegato ad una valvola di intercettazione per simulare una eventuale foratura del serbatoio. Vi sono collegati inoltre un manometro meccanico a tubo di Bourdon per monitorare la pressione e una valvola di sicurezza a molla tarata, secondo la normativa, ad una pressione del 10% minore del valore della pressione massima di prova.

CARATTERISTICHE SERBATOIO

MODELLO	Elbi 780-I
CAPACITÀ	780l
PRESSIONE MASSIMA	9,8 bar
TEMPERATURA ESERCIZIO	-10/+50°C
MATERIALE	Acciaio zincato



Figura d: serbatoio Elbi 780-I

All'interno del software Openmodelica il serbatoio può essere rappresentato dal componente "ClosedVolume", volume di dimensioni fisse e chiuso rispetto l'ambiente esterno, con porte di ingresso / uscita. Si differenzia dal serbatoio reale principalmente per non svolgere la funzione di separatore verticale: con questo componente non possiamo utilizzare un fluido bifase per poi ottenere due uscite diverse.

e) Conduittura

Tutti i tubi sono in PVC-U (polivinilcloruro-non plastificato), del tipo PN16, ed è quindi garantita una pressione di 16 bar alla temperatura di 20°C. Essendo la pressione massima di esercizio dettata dalla prevalenza della pompa, 54,25m, che corrispondono all'incirca a 5 bar, possiamo tranquillamente affermare che la pressione di esercizio massima è abbondantemente al di sotto di quella limite dettata dalle tubature.

Il PVC-U rappresenta una fra le soluzioni economicamente più valide nel campo dei materiali termoplastici e metallici per risolvere i problemi che si incontrano nel trasporto dei fluidi corrosivi industriali e nella distribuzione-trattamento delle acque in genere. I motivi fondamentali di questa preferenza (dovuti prevalentemente alle peculiari caratteristiche della resina) sono:

- Il PVC-U è generalmente inerte alla maggior parte delle soluzioni di acidi, alcoli e sali, ed idrocarburi paraffinici / alifatici.
- Virtuale eliminazione dei problemi di condensazione e contenuta perdita di calore nel trasporto di fluidi caldi grazie ad un ridotto coefficiente di conducibilità termica ($\lambda = 0,15 \text{ W/m } ^\circ\text{C}$ secondo ASTM C177)
- Bassa permeabilità all'ossigeno e ridotto assorbimento d'acqua (0,1% a 23°C secondo ASTM D 570)

CARATTERISTICHE CONDUTTURA

MATERIALE	PVC-U
DIAMETRO ESTERNO	63mm
DIAMETRO INTERNO	53.6mm
SPESSORE	4.7mm



Figura e: condotta in PVC-U

Il trasporto dei fluidi all'interno di Openmodelica in questo caso può essere effettuato tramite il modello "StaticPipe", un tubo dritto con sezione trasversale costante e con bilanci di massa, momento ed energia stazionari, ovvero non immagazzina massa o energia.

f) Valvole di intercettazione e regolazione

Nell'impianto in esame sono presenti 6 valvole di intercettazione manuali, identificate nello schema di fig. W e nella tabella 1 con la dicitura VM, e 2 valvole di regolazione identificate nello schema di fig. W e nella tabella 1 con la dicitura VC.

Le valvole di intercettazione sono posizionate nelle zone più significative dell'impianto: la valvola "VM2" sarà utilizzata per gestire la pressione del fluido motore, chiudendo la suddetta valvola si genererà una perdita di carico sul fluido motore che comporterà un automatico abbassamento della pressione. Le restanti valvole sono invece inserite per simulare potenziali anomalie riscontrabili sull'impianto come per esempio le occlusioni delle tubature. Le valvole prese in esame sono comuni valvole a sfera e il loro funzionamento si basa sulla rotazione di 90° di un otturatore sferico dotato di una cavità cilindrica coassiale al flusso. L'azionamento è molto rapido (si deve fare solo un quarto di giro) e quindi se da una parte consentono una chiusura ed apertura rapida e un facile azionamento, d'altra parte il loro utilizzo provoca brusche aperture e chiusure che causano colpi di ariete, che potrebbero danneggiare apparecchiature limitrofe e non permettono la regolazione del flusso se non in modo molto approssimativo.

CARATTERISTICHE VALVOLE

INTERCETTAZIONE

MODELLO	FIP DN 50 PN 16
MATERIALE	PVC-U
PRESSIONE MASSIMA	16 bar
TEMPERATURA ESERCIZIO	0/+60°C



Figura f1: valvola di intercettazione

Le valvole di regolazione, “VC1” e “VC2”, gestiscono rispettivamente il flusso in uscita del liquido e il flusso di uscita del gas dal serbatoio chiuso. Si tratta di elettrovalvole pneumatiche in cui la chiusura e l’apertura della valvola è realizzata attraverso un attuatore pneumatico, il quale è a sua volta controllato da un posizionatore. Quest’ultimo restituisce un segnale in corrente compreso nell’intervallo 0-20 mA il quale sta a significare rispettivamente valvola completamente aperta e valvola completamente chiusa. Il controllo delle elettrovalvole è completamente affidato al sistema di controllo basato sulla piattaforma hardware Arduino Mega 2560.

Attraverso un controllo PID (sistema in retroazione negativa ampiamente impiegato nei sistemi di controllo) la scheda Arduino controlla:

- La valvola “VC1” affinché la pressione letta dal sensore “LN19” sia la pressione desiderata;
- La valvola “VC2” affinché il livello letto dal sensore “LN20” sia il livello desiderato.

CARATTERISTICHE

VALVOLE DI REGOLAZIONE

MODELLO	ECKARDT MB6713
ATTUATORE	ECKARDT MB6986
CONTROLLO	corrente
INPUT	4-20 mA



Figura f2: valvole di intercettazione

Sia le valvole di intercettazione che di regolazione sono rappresentate in Openmodelica nel package “Valves” all’interno del quale troviamo diverse valvole ognuna delle quali può essere utilizzata per fluidi con caratteristiche diverse ma ognuna con lo stesso obiettivo cioè regolare e controllare il flusso di un fluido.

Il flusso viene regolato tramite un segnale in ingresso nella valvola che varia tra 1 e 0, cioè valvola completamente aperta e valvola completamente chiusa. Il segnale in ingresso può essere definito in diversi modi, per le valvole di intercettazione può essere utilizzata la funzione matematica “Step” ed indicare dopo quanti secondi aprire o chiudere la valvola, per le valvole di regolazione è possibile sfruttare il segnale in uscita da un sensore e con l’utilizzo di funzioni matematiche inviare un segnale che permetta di chiudere o aprire una valvola al variare di quanto misurato dal sensore stesso. Tra le valvole a disposizione per l’acqua e l’aria vengono utilizzate rispettivamente “ValveLinear”, valvola per flussi acqua/vapore con caduta di pressione lineare, e “ValveCompressible”, valvola per fluidi comprimibili.

g) Sensori di pressione

All'interno dell'impianto sono presenti tre diversi sensori di pressione.

- ❖ 3 sensori Setra 280E di pressione assoluta: sensori in acciaio inox con un elettrodo isolato che forma un condensatore variabile; all'aumentare della pressione la capacità diminuisce. La variazione di capacità viene rilevata e convertita in un segnale in uscita DC lineare. Nell'impianto è possibile trovare due tipologie di questo sensore, nello specifico i sensori "LN9" e "LN5" hanno un campo di misura di 0-250 psi (pound per square inch) mentre il sensore "LN11" ha un campo di misura di 0-100 psi.

CARATTERISTICHE SETRA

280E

ALIMENTAZIONE	15-32V DC
OUTPUT	0-5V DC
CAMPO DI MISURA	0-100/0-250 psi



Figura g1: sensore di pressione Setra 280E

- ❖ 1 sensore Foxboro 841GM-CI1 di pressione relativa: trasduttore di pressione che converte la pressione in un segnale elettrico analogico. La conversione della pressione in un segnale elettrico viene ottenuta attraverso la deformazione fisica degli estensimetri connessi alla membrana del trasduttore di pressione e cablati in una configurazione a ponte di Wheatstone. La pressione applicata al trasduttore di pressione produce una flessione del diaframma che a sua volta porta alla deformazione degli estensimetri. La deformazione produrrà un cambiamento della resistenza elettrica proporzionale alla pressione dando un segnale in uscita da 4 a 20 mA in DC.

CARATTERISTICHE FOXBORO

841GM-CI1

ALIMENTAZIONE	12,5-36V DC
OUTPUT	4-20 mA
CAMPO DI MISURA	0-300 psi
TEMPERATURA MAX	120°C
DIAMETRO	50mm



Figura g2: sensore di pressione Foxboro 841GM-CI1

- ❖ 1 sensore Foxboro IDP-10 di pressione differenziale: sensore utilizzato in quest'applicazione per la misura del livello di liquido all'interno del serbatoio. Il principio di funzionamento è basato sulla rilevazione della differenza di pressione tra due lati opposti di un micro-sensore di estensimetri al silicio all'interno del sensore. Questo micro-sensore converte la pressione differenziale in un cambiamento di resistenza che viene quindi convertito in un segnale da 4 a 20 mA proporzionale alla pressione differenziale. La pressione differenziale sarà misurata tra la testa e il fondo del serbatoio, pertanto la pressione differenziale sarà indice del livello di liquido nel serbatoio. In fase di calibrazione il segnale in uscita dal sensore sarà convertito direttamente in una misura di lunghezza al posto di una misura di pressione.

CARATTERISTICHE FOXBORO

IDP-10

ALIMENTAZIONE	12,5-42V DC
OUTPUT	4-20 mA
CAMPO DI MISURA	0-300 psi
PRESSIONE MAX	250 bar

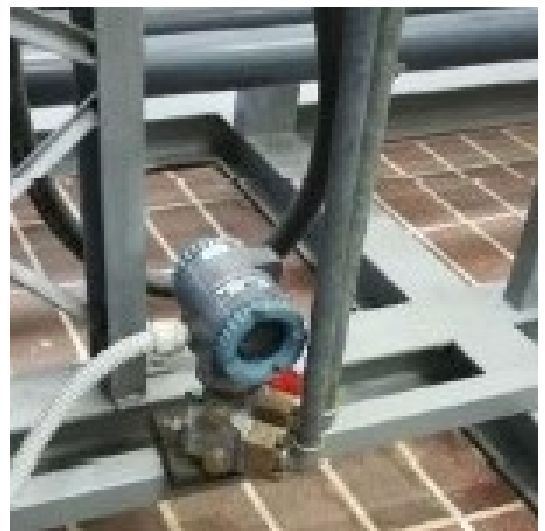


Figura g3: sensore di pressione Foxboro IDP-10

All'interno del software Openmodelica tutti i sensori di pressione sono rappresentati da due sensori ideali, "Pressure" e "RelativePressure". Il primo è un sensore di pressione ideale che controlla la pressione assoluta nella sua porta del fluido, il secondo è un sensore di pressione relativa ideale, determina la pressione relativa "port_a.p - port_b.p" tra le due porte di questo componente e viene fornita come segnale di uscita. Il sensore deve essere collegato in parallelo con altre apparecchiature e non è consentito alcun flusso attraverso il sensore.

h) Sensori di portata

All'interno dell'impianto sono presenti due diversi sensori di portata.

- ❖ 1 sensore Foxboro Magnetic Flow Transmitter 8000 series: rilevatore elettromagnetico di portata di liquido. Il misuratore utilizza le caratteristiche di un liquido conduttivo per generare una tensione indotta direttamente proporzionale alla portata. Può essere usato per fluidi non omogenei, inoltre viscosità e densità del liquido misurato non hanno effetto sull'accuratezza della misura.

CARATTERISTICHE FOXBORO

FLOWTRANSMITTER 8000

SERIES

SENSORE	Foxboro Magnetic Flowtube 801H- WCT
DIAMETRO	50 mm
ALIMENTAZIONE	220V
OUTPUT	4-20 mA
CAMPO DI MISURA	1-100 m ³ /h



Figura h1: sensore di portata Foxboro Magnetic Flow Transmitter 8000 series

- ❖ 1 sensore Foxboro Vortex DN50: si tratta di un sensore che misura la portata del fluido usando il principio del distacco dei vortici. Il suo funzionamento si basa sulla scia di Karman. Su entrambi i lati di un corpo deflettore lambito dal fluido, si formano vortici alternati; questi vortici si distaccano a causa del trascinamento operato dal fluido in moto e si forma la cosiddetta scia di Karman. La frequenza con cui si distaccano i vortici è direttamente proporzionale alla velocità del flusso e inversamente proporzionale alla larghezza del corpo deflettore. Le variazioni di pressione locali derivanti dal distacco dei vortici vengono rilevate da un sensore piezoelettrico e convertite in impulsi elettrici in base alla frequenza di distacco dei vortici. Il sensore produce un segnale analogico da 4 a 20 mA proporzionale alla portata volumetrica.

Questa tipologia di sensori è in grado di rilevare la portata esclusivamente in fluidi allo stato gassoso.

CARATTERISTICHE FOXBORO

VORTEX DN50

DIAMETRO	50mm
ALIMENTAZIONE	24V AC
OUTUPUT	4-20 mA
CAMPO DI MISURA	0-200 m ³ /h
TEMPERATURA MAX	200°C



Figura h2: sensore di portata Foxboro Vortex DN50

Così come per i sensori di pressione, anche i sensori di portata sono rappresentati in digitale da sensori ideali, “MassFlowRate” e “VolumeFlowRate”, il primo è un sensore per la portata massica mentre il secondo è un sensore per la portata volumetrica. Per entrambi i sensori si ha un valore positivo per flussi che scorrono tra port_a e port_b.

4.3 Elettronica di controllo

L'acquisizione dei segnali analogici provenienti dai sensori e la gestione delle elettrovalvole è totalmente affidata alla piattaforma hardware Arduino Mega 2560. Arduino è una piattaforma hardware composta da una serie di schede elettroniche dotate di un microcontrollore. È stata ideata e sviluppata nel 2005 da alcuni membri dell'Interaction Design Institute di Ivrea come strumento per la prototipazione rapida. La piattaforma fisica si basa su un circuito stampato che integra un microcontrollore con dei pin connessi alle porte I/O (input/output), un regolatore di tensione e, quando necessario, un'interfaccia USB che permette la comunicazione con il computer utilizzato per programmare. Negli anni sono stati realizzati dalla casa produttrice diversi modelli con caratteristiche differenti. Per il progetto in esame è stato utilizzato Arduino Mega 2560, in quanto rispetto alla versione standard (Arduino UNO) presenta un maggior numero di porte I/O ed una maggiore memoria interna. Un aspetto molto interessante di Arduino è che in commercio è possibile trovare numerose shield di terze parti, compatibili con la piattaforma, atte a svolgere i compiti più disparati. Per questo progetto è stata utilizzata la shield "Ethernet Shield W5100" la quale aggiunge ad Arduino una porta Ethernet in modo da poterlo controllare via rete locale o via internet e offre la possibilità di utilizzare uno slot microSD.

CARATTERISTICHE TECNICHE ARDUINO MEGA 2560

MICROCONTROLLER	ATmega2560
OPERATING VOLTAGE	5V
INPUT VOLTAGE (RECOMMENDED)	7-12V
INPUT VOLTAGE (LIMIT)	6-20V
DIGITAL I/O PINS	54 (of which 15 provide PWM output)
ANALOG INPUT PINS	16
DC CURRENT FOR I/O PIN	20 mA
DC CURRENT FOR 3.3V PIN	50 mA
FLASH MEMORY	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
CLOCK SPEED	16 MHz
LED_BUILTIN	13
LENGTH	101.52 mm
WIDTH	53.3 mm
WEIGHT	37g

La piattaforma hardware Arduino Mega 2560 è sostituita nel software da funzioni logiche e matematiche che permettono il controllo dell'impianto, inoltre i sensori forniscono automaticamente un risultato numerico.

4.4 Dal reale al digitale

Come visto per i vari componenti, non sempre durante il processo di digitalizzazione si riescono a mantenere tutte le caratteristiche dell'impianto reale; in particolare per quanto riguarda l'impianto in esame le differenze più significative riguardano l'eiettore, elemento non presente nelle librerie standard, e il serbatoio che non svolge la funzione di separatore verticale. A queste problematiche se ne aggiunge un'altra specifica di questo impianto, la miscelazione all'interno dell'eiettore di due fluidi diversi, in questo caso aria e acqua. Openmodelica infatti permette il funzionamento di impianti con fluidi bi-fase ma non la miscelazione e la successiva separazione di essi, un fluido bi-fase infatti deve essere definito come tale in partenza e viene considerato come un fluido che presenta un'unica fase ma con caratteristiche che derivano dalla miscelazione. Bisogna comunque sottolineare come ciò sia valido per l'impianto in esame; nel caso in cui il mio intento finale sia lo studio di trasformazioni chimiche, le miscelazioni sono consentite con gli elementi presenti nella libreria "chemical".

È evidente dunque come non sia possibile riprodurre un Digital Twin che rispetti tutte le caratteristiche dell'impianto presente in laboratorio con Openmodelica (senza il supporto di altri software e componenti aggiuntivi) ma è comunque possibile realizzare un impianto che ne simuli il funzionamento operando esclusivamente con un fluido, l'acqua.

Un impianto che lavora esclusivamente ad acqua risolve gran parte delle problematiche viste finora ad eccezione dell'eiettore che dovrà essere comunque sostituito come visto precedentemente.

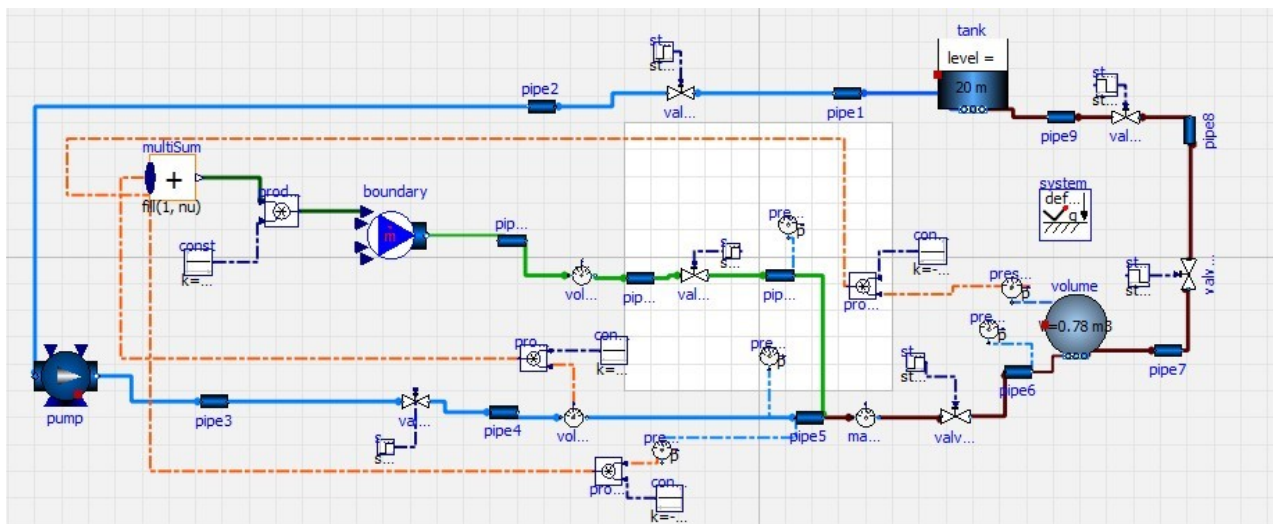


Figura Z: Digital Twin

Ma analizziamo con ordine i vari passaggi che permettono di costruire un modello di questo tipo.

Nel caso in esame, è stato utilizzato come media acqua con le caratteristiche di un gas ideale (IdealSteam).

```

1 model DigitalTwin
2 replaceable package Medium =
3     Modelica.Media.Water.IdealSteam
4     constrainedby
5     Modelica.Media.Interfaces.PartialMedium "Medium in the component"
6     annotation (choicesAllMatching = true);

```

Figura A1: definizione media, "Text View"

È stata aggiunta anche l'annotazione “choicesAllMatching=true” che permette di visualizzare automaticamente un elenco di scelte corrispondenti in un elenco di parametri grafico. Anche in questo caso il tipo di fluido utilizzato andrà poi dichiarato per ogni elemento con cui viene a contatto tramite la funzione “redeclare package Medium =Medium”.

A questo punto è possibile passare alla modellazione dell'impianto che possiamo suddividere in tre passaggi:

- 1 inserimento componenti principali e definizione parametri;
- 2 inserimento elementi di supporto e sensori;
- 3 connettere i vari componenti e verifica.

Queste operazioni non sono però da considerarsi definitive e vincolanti, cioè per avere un maggior controllo sulla presenza di eventuali errori è bene partire da modelli semplici, con pochi elementi, effettuare delle simulazioni e progressivamente aumentare la complessità dell'impianto. Verifiche e simulazioni del modello forniscono spesso informazioni sugli errori commessi o su quali elementi bisogna effettuare modifiche.

Ovviamente l'utilizzo esclusivo dell'acqua porterà altre modifiche all'interno dell'impianto come l'utilizzo di un serbatoio aperto di maggiori dimensioni, essendo la quantità di acqua maggiore, e diversi valori di portata, pressione e temperatura nel tratto precedentemente interessato al fluido bi-fase.

Per inserire i componenti che andranno a costituire il modello “DigitalTwin” basta cercare ognuno di essi tramite il navigatore di librerie e trascinarli con il mouse all’interno del diagramma.

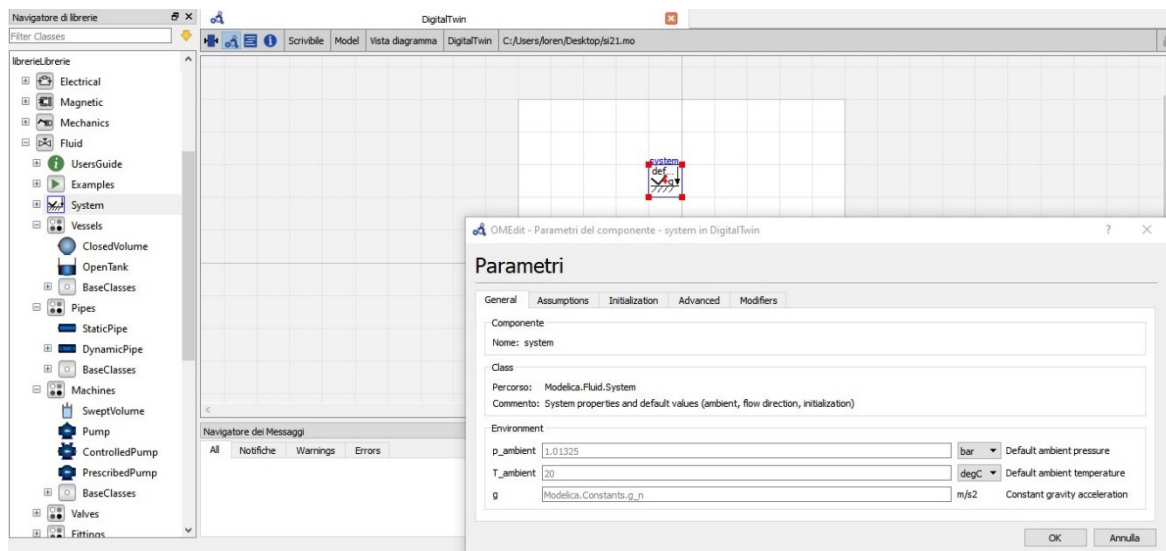


Figura AA1: parametri “System”

Come è possibile notare nella fig. B1, la libreria “Fluid” presenta diversi package all’interno dei quali è possibile trovare vari elementi da inserire. Il primo, necessario per ogni modello che utilizza i fluidi, è il componente “System”, che fornisce impostazioni a livello di sistema, come condizioni ambientali e ipotesi generali di modellazione. Nella fig. B1 è possibile notare anche alcuni dei parametri che caratterizzano “System” i quali sono già definiti, ma eventualmente anche modificare, come la pressione o la temperatura ambiente, rispettivamente 1.01325 bar e 20°C.

A questo punto, è possibile passare alla digitalizzazione della parte dell'impianto reale coinvolta nel flusso di acqua per poi aggiungere la parte che coinvolge il flusso di aria, che nel caso digitale sarà sempre acqua.

Inseriamo dunque, facendo sempre riferimento all'impianto reale, gli elementi: **“OpenTank”** (Libreria: Fluid, Package: Vessels) , modello di un serbatoio aperto all'ambiente a pressione fissa p_{ambient} ; **“ControlledPump”**(Libreria: Fluid, Package: Machines), pompa centrifuga (o un gruppo di pompe parallele) con portata massica o pressione idealmente controllate; **“ClosedVolume”** (Libreria: Fluid, Package: Vessels), volume di dimensioni fisse chiuso rispetto l'ambiente esterno, con porte di ingresso / uscita; **“ValveLinear”**(Libreria: Fluid, Package: Valves), modello molto semplice di valvola che fornisce una caduta di pressione proporzionale alla portata in ingresso, senza calcolare alcuna proprietà del fluido; **“StaticPipe”**(Libreria: Fluid, Package: Pipes), modello di un tubo dritto con sezione trasversale costante e con bilanci di massa, momento ed energia stazionari. Per ognuno di essi andranno poi inseriti diversi parametri.

Parametri OpenTank:

General:

- Height: 100 m (altezza serbatoio)
- crossArea: 10 m² (area base)
- use_portsData: False (Con l'impostazione use_portsData=false, la pressione della porta rappresenta la prevalenza statica all'altezza della rispettiva porta)

Initialization:

- level_start: 20 m (Valore iniziale del livello del serbatoio)

I parametri che caratterizzano “OpenTank”, così come per altri componenti, sono molti di più di quelli elencati, ad esempio la pressione e la temperatura iniziale, ma hanno già un valore assegnato e per il nostro caso in esame possono non essere modificati.

Parametri

General	Assumptions	Initialization	Advanced	Modifiers
Parameters				
level_start	20	m Start value of tank level		
use_T_start	true	= true, use T_start, otherwise h_start		
T_start	if use_T_start then system.T_start else Medium.temperature_phX(p_start, h_start, X_start)			Start value of temperature
h_start	if use_T_start then Medium.specificEnthalpy_pTX(p_start, T_start, X_start) else Medium.h_default			Start value of specific enthalpy
X_start	Medium.X_default			Start value of mass fractions m_j/m
C_start	Medium.C_default			Start value of trace substances

Figura C1: parametri “OpenTank”

Parametri ControlledPump:

General:

- `p_a_nominal`: 110000 Pa (Pressione di ingresso nominale per caratteristiche predefinite della pompa [Pa])
- `p_b_nominal`: 200000 Pa (Pressione di uscita nominale, fissa se non `control_m_flow` e `not use_p_set` [Pa])
- `m_flow_nominal`: 2 kg/s (Portata massica nominale, fissa se `control_m_flow`= true e `use_m_flow_set`=false)
- `control_m_flow`: true (false per controllare la pressione di uscita `port_b.p` anziché `m_flow`)
- `use_m_flow_set`: false (true per utilizzare il segnale di input `m_flow_set` anziché `m_flow_nominal`)
- `use_p_set`: false (true per utilizzare il segnale di input `p_set` anziché `p_b_nominal`)
- `N_nominal`: 2900 rev/min (Velocità di rotazione nominale per caratteristica di flusso)

Assumptions:

- `allowFlowReversal`: false (true per consentire l'inversione del flusso)

Parametri ClosedVolume:

General:

- V: 0.78 m³ (volume)
- use_portsData: false (false per trascurare la perdita di pressione e l'energia cinetica)

Initialization:

- p_start: 110000 Pa (Valore iniziale della pressione)

```
Modelica.Fluid.Vessels.ClosedVolume volume(redeclare package Medium = Medium,  
V = 0.78,  
nPorts= 2, p_start = 110000, use_portsData = false) annotation( ... );
```

Figura BB1: parametri "Closed Volume", TextView

Parametri ValveLinear:

General:

- m_flow_nominal: 100 kg/s (portata massica nominale alla massima apertura)
- dp_nominal: 0.1 bar (Perdita di carico nominale alla massima apertura)

Parametri StaticPipe:

General:

- length: da 0.5 a 2 m (Lunghezza)
- diameter: 0.536 (Diametro del tubo circolare)
- allowFlowReversal: false (true per consentire l'inversione del flusso)

Per un corretto funzionamento il numero di “StaticPipe” da inserire è condizionato dal numero di elementi presenti da connettere.

Resta dunque evidente come i parametri da inserire siano tutti facilmente ricavabili dall’impianto reale o tramite l’utilizzo di semplici formule matematiche.

Una volta inseriti nella quantità giusta, tutti questi componenti devono essere connessi correttamente facendo attenzione a diversi particolari:

il collegamento viene effettuato tramite il cursore del mouse formando delle linee che partono dall’ingresso di un componente e finiscono nell’ingresso di un secondo elemento;

il numero di ingressi/uscite, (compresi anche i sensori), di “OpenTank” e “ClosedVolume” deve essere indicato nell’ OMShell tra le caratteristiche dei due serbatoi, tramite la dicitura “nPorts= (numero ingressi/uscite)”;

per gli elementi con il parametro “allowFlowReversal: false” l’ingresso deve essere sempre port_a;

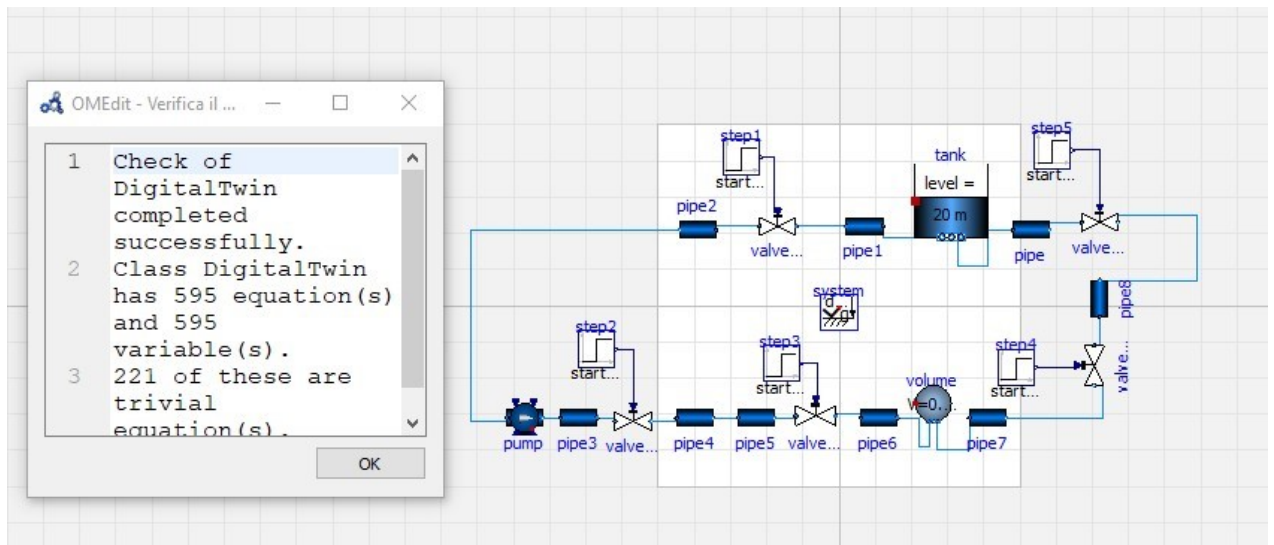


Figura F1: modello DigitalTwin con verifica

Se anche la simulazione ha esito positivo è possibile ottenere dei grafici sull'andamento di alcuni parametri e continuare con la progettazione del modello, in particolare aggiungere i sensori presenti nel modello reale, 4 sensori di pressione, "Pressure" (Libreria: Fluid, Package: Sensors) e 2 sensori di portata, "MassFlowRate" e "VolumeFlowRate" (Libreria: Fluid, Package: Sensors).

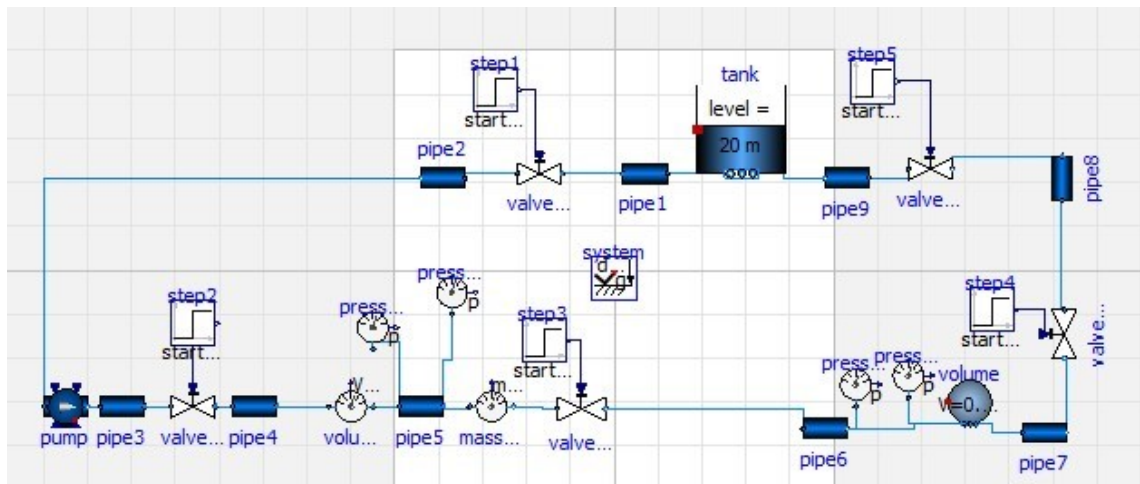


Figura G1: DigitalTwin con aggiunta di sensori

Verificato il modello finora realizzato è possibile passare alla progettazione della restante parte dell'impianto, con particolare attenzione alla digitalizzazione dell'eiettore non essendo presente come componente nelle librerie standard di Openmodelica.

Il primo elemento da inserire è sicuramente “MassFlowSource_T” (Libreria: Fluid, Package: Sources), nel modello prende il nome di “Boundary”, che modella una sorgente di flusso ideale, con valori prescritti di portata, temperatura e composizione e che dunque permette l'aggiunta di un flusso di fluido senza l'utilizzo di un serbatoio. L'unico parametro che andremo a modificare è “use_m_flow_in: true” che permette di ottenere una portata massica da un segnale in ingresso.

Difatti, per simulare la funzione dell'eiettore vengono valutati, tramite i sensori, alcuni parametri del flusso di acqua e tramite interpolazioni matematiche definire la portata massica che sarà utilizzata come input per la fonte "MassFlowSource_T". Bisogna prestare attenzione al collegamento tra segnale di input e il componente poiché "MassFlowSource_T" presenta quattro ingressi diversi dedicati rispettivamente alla portata massica, alla temperatura, alla composizione e infine alla presenza di sostanze in traccia.

La nuova fonte di acqua sarà collegata all'impianto principale attraverso una conduttura per miscelarsi con il flusso principale all'uscita del componente "pipe5". Tenendo sempre in considerazione l'impianto reale, dovranno essere aggiunti 1 "ValveLinear" con la funzione "Step", 1 sensore di portata volumetrica "VolumeFlowRate", 1 sensore di pressione "Pressure" e 3 "StaticPipe". Essendo tutti componenti già presenti nel modello e poiché per tutti valgono le caratteristiche dei precedenti è possibile velocizzare l'inserimento tramite l'opzione "duplica" per ogni singolo elemento. Anche in questo caso, per una maggior sicurezza, è bene verificare e simulare il funzionamento dell'impianto prima tramite un modello più semplice, ad esempio con un valore di portata massica all'ingresso del "MassFlowSource_T" costante, tramite la funzione "Constant" (Libreria: Blocks, Package: Sources) e poi un valore condizionato dal flusso del fluido principale.

Uno dei principali vantaggi di Openmodelica è la facilità di utilizzo, anche dal punto di vista grafico infatti è possibile modificare l'impianto innumerevoli volte e nella forma che preferiamo in modo da renderlo più comprensibile. È possibile ad esempio aumentare la grandezza dei componenti principali oppure legare fra loro i componenti in maniera differente dal punto di vista estetico in base al tipo di collegamento.

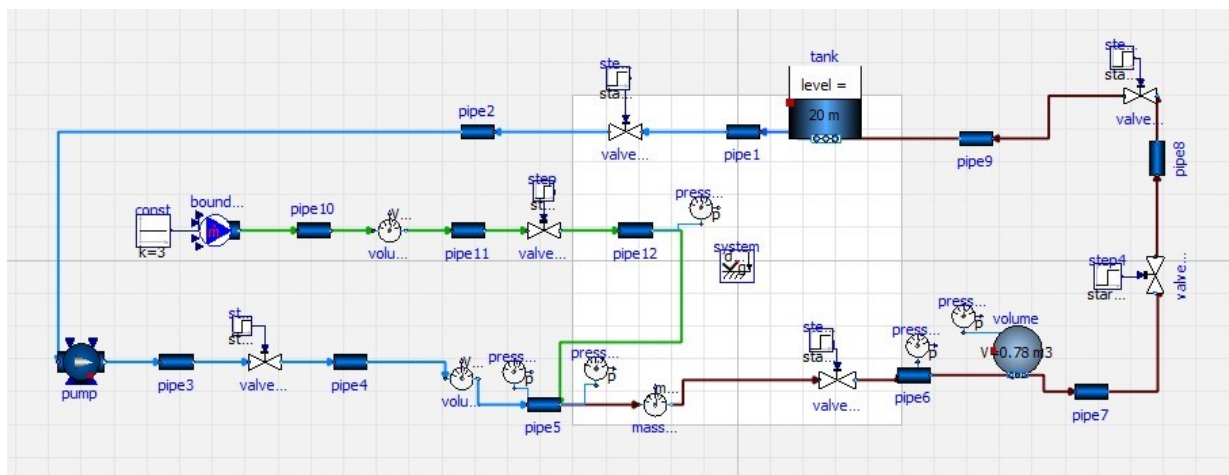


Figura H1: DigitalTwin con ingresso “boundary” costante

Se anche in questo caso la verifica e la simulazione hanno esito positivo, è possibile eliminare la funzione “Constant” e fornire al “MassFlowSource_T” un input influenzato dal flusso principale.

In particolare, è possibile utilizzare l'equazione:

$$-0,60(Pressione\ H_2O[bar]) - 0,87(Pressione\ serbatoio\ chiuso[bar]) + 1,75 \left(Portata\ H_2O \left[\frac{m^3}{h} \right] \right) = Portata\ aria \left[\frac{m^3}{h} \right]$$

Ciò è possibile grazie all'uso di componenti che contengono funzioni matematiche (MultiSum, Constant, Product), e che dunque in base agli input forniscono un determinato output, e grazie ai sensori che non forniscono soltanto informazioni al termine della simulazione ma presentano anche un'uscita che permette di utilizzare il risultato numerico misurato come parametro. Naturalmente in questo caso, come detto precedentemente, il flusso di aria sarà sostituito da un flusso di acqua.

Nel dettaglio, gli elementi da aggiungere sono:

1. 4 “Product” (Libreria: Blocks, Package: Math), funzione che calcola l'uscita y come prodotto dei due ingressi u1 e u2: $y = u1 * u2$.

Tre di questi componenti saranno utilizzati per applicare i coefficienti ai valori della pressione dell'acqua in uscita dalla pompa, portata dell'acqua in uscita dalla pompa e della pressione dell'acqua nel serbatoio chiuso moltiplicando il valore ottenuto dai sensori con valori costanti. Il quarto sarà utilizzato per convertire la portata volumetrica in portata massica.

2. 4 “Constant” (Libreria: Blocks, Package: Sources), funzione valore costante. Queste funzioni costanti sono utilizzate per moltiplicare i valori in uscita dai sensori e dal componente “MultiSum” e valgono rispettivamente:
 1. “Constant1”: +6300, tramite “Product1” viene moltiplicato per la portata volumetrica misurata dal sensore “volumeFlowRate1”.
 2. “Constant2”: -0.000006, tramite “Product2” viene moltiplicato per la pressione misurata dal sensore “Pressure1”.
 3. “Constant3”: -0.0000087, tramite “Product3” viene moltiplicato per la pressione misurata dal sensore “Pressure4”.
 4. “Constant4”: +0.0003, tramite “Product4” viene moltiplicato per il valore in uscita dal componente “MultiSum”, per passare da portata volumetrica a portata massica, in particolare viene moltiplicato per il valore della densità dell’aria ($1,22 \left[\frac{kg}{m^3} \right]$) e diviso per 3600, per trasformare dunque $\left[\frac{m^3}{h} \right]$ in $\left[\frac{kg}{s} \right]$.
3. 1 “MultiSum” (Libreria: Blocks, Package: Math), funzione che calcola l'uscita reale scalare "y" come somma degli elementi del vettore di segnale di ingresso reale u: $y = k[1] * u[1] + k[2] * u[2] + \dots k[N] * u[N]$. Questo elemento è utilizzato per ottenere in uscita un valore di portata volumetrica che una volta trasformato in portata massica potrà essere utilizzato come input per “MassFlowSource_T”.

Tutte le operazioni matematiche vengono effettuate considerando i numeri senza unità di misura, a tal proposito è necessario moltiplicare i suddetti valori per coefficienti che prendono in considerazione anche le grandezze fisiche.

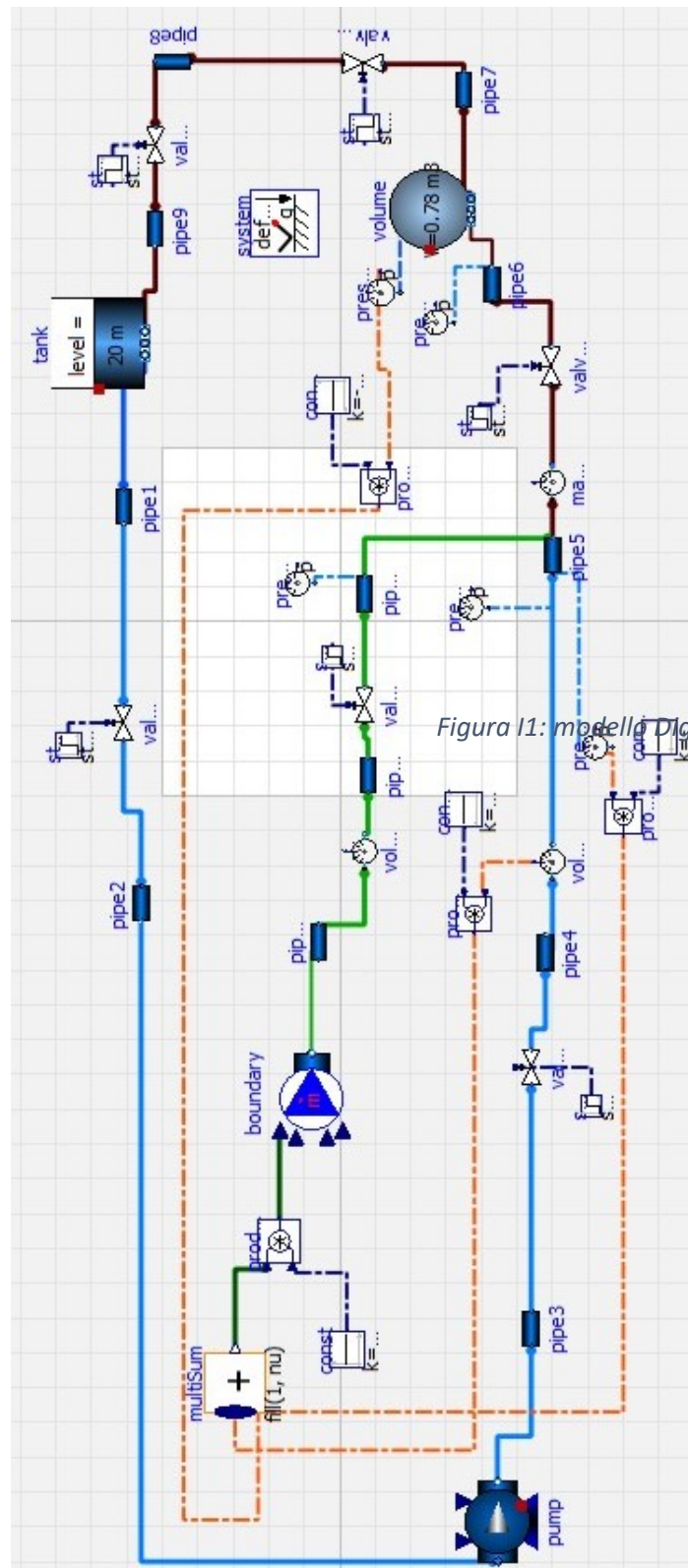
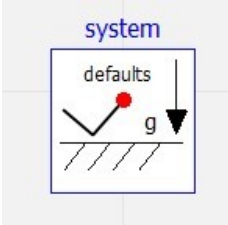
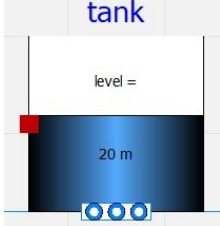

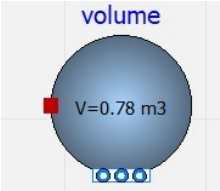

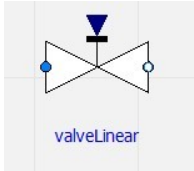
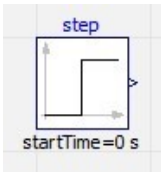
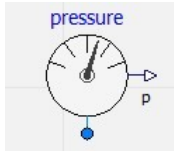
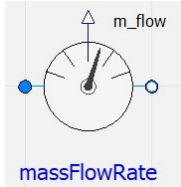
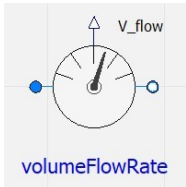

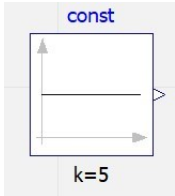


Figura I1: modello DigitalTwin definitivo

SIMBOLO	INDICE	DESCRIZIONE
	System	<p>Nome: System Library: Fluid</p> <p>È necessario un componente di sistema in ciascun modello di fluido per fornire impostazioni a livello di sistema, come condizioni ambientali e ipotesi generali di modellazione. Le impostazioni del sistema vengono propagate ai modelli fluidi mediante il meccanismo interno / esterno.</p>
	Tank	<p>Nome: Opentank Package: Vessels Library: Fluid</p> <p>Modello di un serbatoio aperto all'ambiente a pressione fissa p_{ambient}.</p>
	Pump	<p>Nome: ControlledPump Package: Machines Library: Fluid</p> <p>Pompa centrifuga (o un gruppo di pompe parallele) con portata massica o pressione idealmente controllate.</p>
	Volume	<p>Nome: ClosedVolume Package: Vessels Library: Fluid</p> <p>Volume di dimensioni fisse, chiuso rispetto l'ambiente esterno, con porte di ingresso / uscita.</p>

SIMBOLO	INDICE	DESCRIZIONE
	Pipe	<p>Nome: StaticPipe Package: Pipes Library: Fluid</p> <p>Modello di un tubo diritto con sezione trasversale costante e con bilanci di massa, momento ed energia stazionari, ovvero il modello non immagazzina massa o energia.</p>
	ValveLinear	<p>Nome: ValveLinear Package: Valves Library: Fluid</p> <p>Questo modello molto semplice di valvola fornisce una caduta di pressione proporzionale alla portata in ingresso, senza calcolare alcuna proprietà del fluido. Può essere utilizzato a scopo di test, quando è necessario un semplice modello di perdita di pressione variabile.</p>
	Step	<p>Nome: Step Package: Sources Library: Blocks</p> <p>Funzione Step.</p>
	Pressure	<p>Nome: Pressure Package: Sensors Library: Fluid</p> <p>Questo componente controlla la pressione assoluta nella sua porta del fluido. Il sensore è ideale, ovvero non influenza il fluido.</p>

SIMBOLO	INDICE	DESCRIZIONE
	MassFlowRate	<p>Nome: MassFlowRate Package: Sensors Library: Fluid</p> <p>Questo componente controlla la portata massica che scorre da port_a a port_b. Il sensore è ideale, ovvero non influenza il fluido.</p>
	VolumeFlowRate	<p>Nome: VolumeFlowRate Package: Sensors Library: Fluid</p> <p>Questo componente controlla la portata del volume che scorre da port_a a port_b. Il sensore è ideale, ovvero non influenza il fluido.</p>
	MultiSum	<p>Nome: MultiSum Package: Math Library: Blocks</p> <p>Questo blocco calcola l'uscita reale scalare "y" come somma degli elementi del vettore di segnale di ingresso reale u:</p> $y = k [1] * u [1] + k [2] * u [2] + \dots k [N] * u [N]$
	Constant	<p>Nome: Constant Package: Sources Library: Blocks</p> <p>Funzione valore costante.</p>

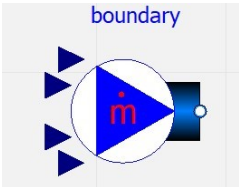
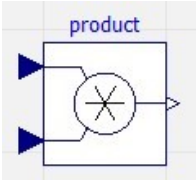
SIMBOLO	INDICE	DESCRIZIONE
	Boundary	<p>Nome: MassFlowSource_T Package: Sources Library: Fluid</p> <p>Modella una sorgente di flusso ideale, con valori prescritti di portata, temperatura, composizione.</p>
	Product	<p>Nome: Product Package: Math Library: Blocks</p> <p>Questo blocco calcola l'uscita y come prodotto dei due ingressi u1 e u2: $y = u1 * u2$.</p>

Tabella 5: legenda Digital Twin

Anche in questo caso, soprattutto grazie all'elevata presenza di sensori, è possibile valutare graficamente come variano determinati parametri nel tempo e in diversi punti dell'impianto.

Effettuiamo una simulazione della durata di cento secondi.

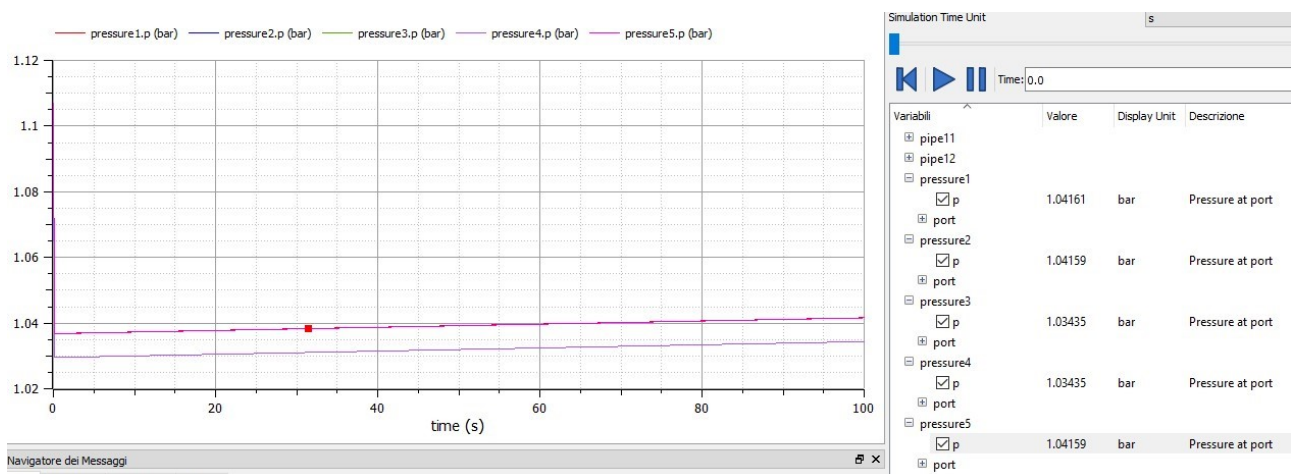


Figura K1: andamento pressioni misurate dai sensori: Pressure1, Pressure2, Pressure3, Pressure4, Pressure5

Dal grafico sull'andamento delle pressioni è possibile notare come in tutti e cinque i sensori la pressione è in aumento, ad eccezione della pressione misurata dal sensore “pressure4”, ovvero la pressione all'interno del serbatoio chiuso, che inizialmente subisce un brusco calo per poi aumentare leggermente.

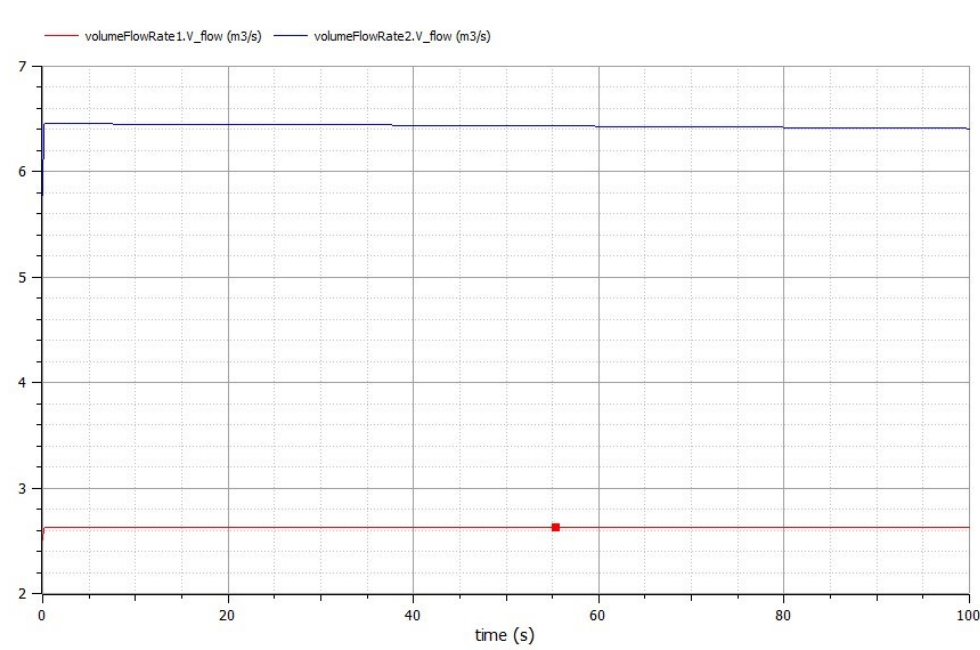


Figura L1: andamento portata volumetrica misurata dai sensori: VolumeFlowRate1 e VolumeFlowRate2

Mentre, analizzando graficamente l'andamento della portata volumetrica misurata dai sensori "VolumeFlowRate1" e "VolumeFlowRate2" è possibile notare come la portata volumetrica in uscita dalla pompa sia sempre costante, mentre la portata erogata dalla fonte "Boundary" varia leggermente, questo perché influenzata dalla pressione dell'acqua e dalla pressione all'interno del serbatoio chiuso entrambe in aumento nel tempo come è possibile notare graficamente.

Altri grafici che è bene osservare riguardano l'andamento del volume di acqua contenuto all'interno dei due serbatoi.

Come è possibile notare dalla fig.M1, il volume di acqua contenuto nel serbatoio chiuso, “ClosedVolume” è costante, dunque poiché varia la pressione nel serbatoio, cambierà anche la temperatura dell’acqua.

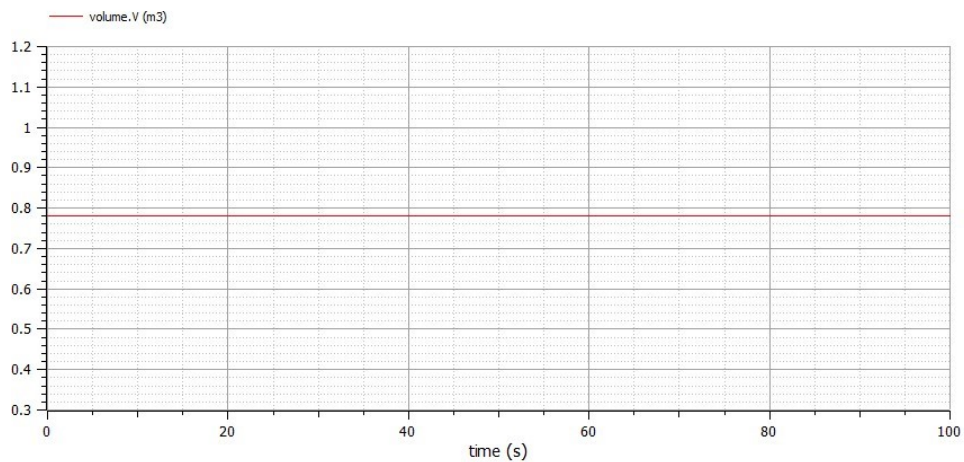


Figura M1: volume di acqua contenuto all’interno del serbatoio chiuso

A differenza di quello che accade nel serbatoio chiuso, nel serbatoio aperto, “Opentank”, il volume di acqua contenuto è in costante aumento.

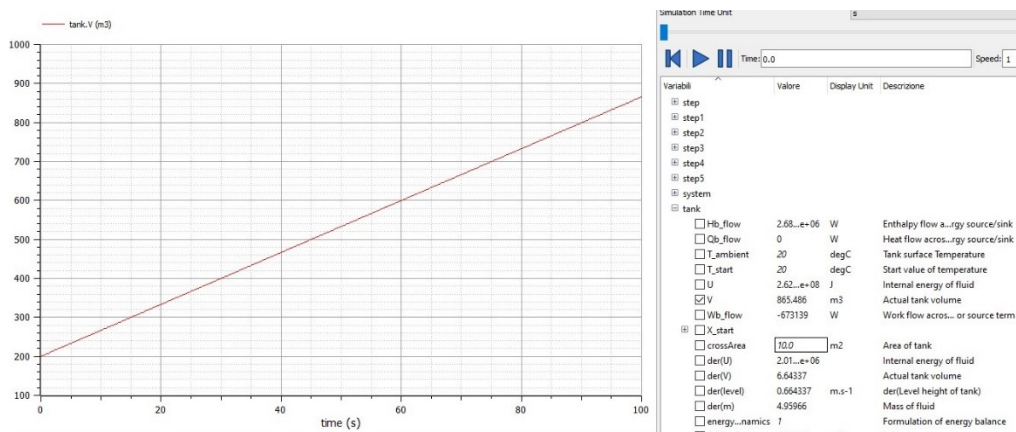


Figura N1: andamento de volume di' acqua contenuto all’interno del serbatoio aperto

Ciò è dovuto alla continua immissione di fluido da parte del componente “MassFlowSource_T” che provoca un aumento della quantità di acqua presente all’interno del serbatoio. Essendo inoltre il livello massimo di acqua contenuto nel serbatoio pari a 1000 litri, basta aumentare di poco la portata massica in uscita dalla pompa oppure aumentare il tempo della simulazione per ottenere un’interruzione della stessa.

Una volta che l’impianto è stato definito, dopo aver verificato che funzioni correttamente e che la simulazione permetta la realizzazione di grafici, ad esso possono essere applicate ulteriori modifiche per simulare diverse situazioni e valutare come variano altri parametri; è possibile ad esempio variare il fluido utilizzandone uno che ha proprietà più simili al petrolio oppure variare le condizioni ambientali e quindi le caratteristiche iniziali del fluido.

Queste modifiche non sempre possono essere realizzate con l’impianto reale e che dunque evidenziano maggiormente l’importanza dell’utilizzo di un gemello digitale.

5 conclusioni

Sia nella fase di progettazione che nella fase di analisi, Openmodelica offre dunque numerosi vantaggi ma altresì diverse problematiche. La sua peculiarità è comunque quella di essere un programma open-source e facilmente accessibile; se infatti dal punto di vista industriale presenta limitazioni dovute prevalentemente al non poter riprodurre in maniera esatta tutti i componenti e le condizioni di lavoro che coinvolgono un impianto, dal punto di vista didattico e accademico risulta particolarmente utile, grazie soprattutto alla facilità di utilizzo che permette di studiare ed elaborare modelli semplici in poco tempo. Inoltre, avendo un funzionamento simile a quello di altri software per lo sviluppo di un Digital Twin, Openmodelica diviene uno strumento con cui fare pratica senza un investimento economico e quindi un punto di partenza per lo sviluppo di simulazioni di analisi, simulazioni di ingegnerizzazione, simulazioni di processo e la creazione di un gemello digitale.

D'altronde la conoscenza e lo studio di strumenti riguardanti le nuove tecnologie abilitanti è uno degli aspetti fondamentali dell'industria 4.0. Dalla ricerca "The Future of the Jobs", presentata al World Economic Forum nel 2016, è emerso che, nei prossimi anni, fattori tecnologici e demografici influenzeranno profondamente l'evoluzione del mercato del lavoro.

L'effetto sarà la creazione di 2 nuovi milioni di posti di lavoro, ma contemporaneamente ne spariranno 7, con un saldo netto negativo di oltre 5 milioni. Per quanto riguarda l'Italia, un rapporto pubblicato dalla Commissione lavoro del Senato nel 2019 presenta una situazione in cui circa il 10% dei lavoratori rischia di essere sostituiti dai robot e dall'automazione, mentre il 44% sarà obbligato a ridefinire e stravolgere le proprie competenze. A livello di gruppi professionali le perdite si concentreranno nelle aree amministrative e della produzione: rispettivamente 4,8 e 1,6 milioni di posti distrutti. Secondo la ricerca compenseranno parzialmente queste perdite l'area finanziaria, il management, l'informatica e l'ingegneria. Un domani, l'introduzione di tecnologie sempre più all'avanguardia creerà nuovi posti di lavoro, legati soprattutto alla lettura ed elaborazione dei dati, installazione e gestione dei componenti elettrici e la loro stessa produzione, inoltre, sarà sempre maggiore il bisogno di esperti di cybersicurezza, sviluppatori e hardware engineer. Cambiano di conseguenza le competenze e abilità ricercate: nel 2020 il problem solving rimarrà la soft skill più ricercata, ma diventeranno più importanti il pensiero critico, la creatività, la capacità di interagire con le macchine e leggere i dati raccolti, comprenderli e sfruttarli per migliorare lo scenario esistente.

Luciano Pero, docente di Organization Theory and Design al MIP-Politecnico di Milano (Fonte: www.economyup.it), sostiene che il lavoro continuerà ad essere centrale, tuttavia esso perde le caratteristiche di pura manualità e di “forza lavoro” da controllare con supervisione diretta. Il lavoro acquista invece sempre di più le caratteristiche di lavoro intelligente, cioè mescola interventi manuali con forti capacità tecniche di analisi e diagnosi e in certi casi anche di ragionamento scientifico, con applicazione di conoscenze complesse.

La sostituzione di alcuni lavori con le macchine va di pari passo con nuovi tipi di lavori, di mansioni, di conoscenze e questo fa parte di uno sviluppo più grande che non riguarda solo i settori ma tutta la società, comprese le persone che ne fanno parte e da cui dipende di fatto il futuro della tecnologia e dei suoi utilizzi.

Come disse Melvin Kranzberg, storico e accademico statunitense,

“Technology is neither good nor bad; nor is it neutral”,

(La tecnologia non è né buona né cattiva; né è neutrale) ovvero la tecnologia non ha effetti positivi e negativi, buoni o cattivi, secondo un’ottica deterministica; tuttavia allo stesso tempo non è nemmeno neutrale perché gli usi che facciamo di una tecnologia hanno degli effetti e gli utilizzi che l’uomo ne fa dipendono dalla cultura di riferimento. La tecnologia è un “attore”, svolge un ruolo.

Con l'epidemia del Coronavirus l'innovazione tecnologica ha mostrato i benefici maggiori sia per i lavoratori che per le industrie: molte di esse hanno infatti evitato il fermo totale della produzione e di alcune servizi come l'assistenza clienti grazie allo Smart Working. Lo sviluppo e controllo di un Digital Twin rientra tra le attività che possono essere svolte a distanza e dunque ha dato la possibilità alle aziende più tecnologicamente avanzate di continuare a raccogliere e analizzare dati.

Per quanto riguarda gli effetti totali della quarta rivoluzione industriale bisognerà comunque aspettare i prossimi anni.

Quel che è certo è che le tecnologie abilitanti, la simulazione e la realizzazione di gemelli digitali hanno e avranno un ruolo sempre più significativo all'interno della società e dunque il loro utilizzo sarà sempre più centrale nel mondo del lavoro e dell'ingegneria.

Bibliografia

- “An Object-Oriented R744 Two-Phase Ejector Reduced-Order Model for Dynamic Simulations”, Michal Haida, Rafal Fingas, Wojciech Sz wajnoch, Jacek Smolka, Michal Palacz, Jakub Bodys and Andrzej J. Nowak.
- “Digital Twins: Identical, But Different”, Oliver Wyman.
- “Funzionamento Impianto”, tesi di laurea di Fabio Di Carlo.
- “Impatto sul mercato del lavoro della quarta rivoluzione industriale”, Senato della Repubblica - 11a Commissione Lavoro, previdenza sociale, Paola Mecchia, (www.senato.it).
- “Impianto di smiscelamento aria-acqua”, tesi di laurea di D’Erme s Vincenzo.
- “Introduzione a Matlab e Simulink”, Simone Ciotti.
- “OpenModelica User's Guide”, Open Source Modelica Consortium, (www.openmodelica.org).
- “The Future of Jobs”, World Economic Forum 2016.

- “Transforming manufacturing: the power of data in action”,
Dassault Systèmes, (www.digital4.biz).
- “Using Modelica as a Design Tool for an Ejector Test Bench”,
C.C. Richter, C. Tischendorf, R. Fiorenzano, P. Cavalcante, W.
Tegethoff, J. Koehler, (www.researchgate.net).

Sitografia

- www.3isrl.it
- www.agendadigitale.eu
- www.alternativeto.net
- www.ansa.it
- www.businesspeople.it
- www.criticalcase.com
- www.digital4.biz
- www.digital-leaders.it
- www.economyup.it
- www.eidonlab.org
- www.enginsoft.com
- www.falstad.com
- www.focusindustria40.com
- www.ilsole24ore.com
- www.industria40.it

- www.ip4fvg.it
- www.it.mathworks.com
- www.maplesoft.com
- www.modelica.it
- www.openmodelica.org
- www.plm.automation.siemens.com
- www.ptc.com
- www.researchgate.net
- www.scilab.org
- www.senato.it
- www.smactory.com