

# UNIVERSITÀ DEGLI STUDI DEL MOLISE



Dipartimento di Bioscienze e Territorio

---

DOTTORATO DI RICERCA IN BIOSCIENZE E TERRITORIO

XXXIII Ciclo

CURRICULUM INFORMATICO-MATEMATICO:

Settore disciplinare di riferimento: ING-INF/05 (Sistemi di elaborazione delle informazioni)

TESI DI DOTTORATO

## **UN FRAMEWORK DI SIMULAZIONE DI PROCESSI INDUSTRIALI RAPPRESENTABILI COME SISTEMI INTERCONNESSI**

Coordinatore del Corso di Dottorato: Prof. Ing. Giovanni Fabbrocino

Tutor: Prof. Stefano Ricciardi

Dottorando: Ing. Filippo Catalano  
162457

ANNO ACCADEMICO 2019/2020

# INDICE

<b>ABSTRACT</b>	1
<b>1. INTRODUZIONE</b>	2
<b>2. STATO DELL'ARTE</b>	5
2.1 Modellazione dei processi industriali	5
2.2. Industry 4.0 e IoT	6
2.3. Automazione, Ottimizzazione e Controllo	8
2.4 Rilevanza del controllo di processo: l'esempio della gestione delle filiere alimentari (Food Supply Chain)	12
2.5. Demand Side Management e Software applicativi	15
<b>3. MATERIALI E METODI</b>	19
3.1. Sistemi Interconnessi	20
3.2. Individual System Modelling	21
3.3. Un quadro generale per i modelli di sottosistemi indipendenti	22
3.4. Un nuovo approccio per gli ISM	26
3.5. Il nuovo approccio applicato ai sistemi Linear Time Invariant: rappresentazione nello spazio degli stati e condizioni di stabilità	29
3.6. Il Framework di simulazione	33
<b>4. RISULTATI E DISCUSSIONE</b>	37
4.1. Node Editor	37
4.1. Runtime VR	40
4.2. Test numerici su Sistemi Lineari Autonomi	42
4.2. Analisi di performance	44
4.5. Caso studio reale	45
<b>5. CONCLUSIONI</b>	57
<b>BIBLIOGRAFIA</b>	59

## ABSTRACT

Il problema della simulazione di sistemi complessi quali impianti industriali ed intere linee di produzione, processi di trasformazione, ecc., rappresenta oggi una opportunità che porta con sé il grande vantaggio di limitare i costi di progettazione. Tuttavia, pur disponendo di diversi strumenti software, il progettista è deve innanzitutto definire ed implementare i modelli matematici che descrivono il comportamento delle diverse macchine, effettuare le opportune verifiche di debug degli stessi prima di poterli utilizzare senza incorrere in possibili errori di progettazione legati ad errori di modellazione, deve riprogrammare il tutto laddove ha necessità di modificare il modello anche di un solo sotto sistema.

È per tale motivo che in questo progetto di dottorato è stato studiato e realizzato un nuovo Framework per l'utilizzo di Individual Subsystem Models (ISM) per la modellazione e la simulazione di sistemi interconnessi. Inoltre, dallo studio dello stato dell'arte è emersa la mancanza di algoritmi numerici efficienti e sufficientemente generali, ma nello stesso tempo semplici da utilizzare, per risolvere le equazioni algebrico-differenziali derivanti dalla simulazione di ISM. Il nuovo approccio proposto segue il paradigma dei metodi della co-simulazione, inclusi anche i metodi della teoria dei grafi per risolvere in modo semplice ed efficiente l'ISM generale. In questo approccio si richiede che ogni sottosistema abbia la propria rappresentazione indipendentemente dagli altri sottosistemi. In questo modo è sempre possibile sostituire qualsiasi sottosistema ogni volta che si renda disponibile una rappresentazione aggiornata, rendendo flessibile la manutenzione e l'evoluzione dell'intero ISM. Tale flessibilità si basa sul fatto che l'utente non ha bisogno di conoscere necessariamente il contenuto di ogni sottosistema, ad es. quando un impianto, una macchina, ecc. vengono rappresentati da una black box in quanto la proprietà del suo contenuto è di un'azienda esterna che non vuole condividerne i dettagli. In tal caso gli input e output sono le uniche informazioni ovviamente note. Pertanto, si è deciso di strutturare il framework in modo da utilizzare algoritmi che risolvono separatamente ogni sottosistema (o un sottogruppo di essi) in un ordine ottimale (o subottimale) basato sulla struttura del grafo dell'intero sistema. In particolare, ogni uscita di un sottosistema (o gruppo di essi) collega l'ingresso di un altro sottosistema. La procedura generale è stata validata nell'ambito degli ISM Lineari e Tempo-Invarianti: in tali ipotesi sono state calcolate le condizioni di stabilità ed effettuati test numerici che mostrano l'efficacia dell'approccio proposto.

# 1. INTRODUZIONE

Un impianto industriale è oggi da considerarsi un sistema complesso in quanto spesso è composto da numerose tipologie di macchine ed è caratterizzato da una forte variazione temporale delle attività. Tale complessità, pertanto, si riflette in modo notevole sulla progettazione degli impianti e aumenta ancora di più se la progettazione deve tenere conto del comportamento degli impianti in fase di esercizio a causa della elevata variabilità dei parametri influenzanti la dinamica del sistema. D'altro canto, oggi il progettista non ha molti mezzi di progettazione che gli consentano la simulazione di una molteplicità di configurazioni impiantistiche tale da poter scegliere di volta in volta quella ottimale. Tutte le soluzioni attualmente disponibili consentono al più di costruire modelli meccanici, o modelli CFD, o ancora modelli multifisici, ma una visione di sistema viene ancora oggi implementata in modo molto parziale se non in applicazioni prevalentemente a scopo didattico o al più molto specifiche e, pertanto, per loro natura molto limitate. Inoltre, le attuali metodologie di progettazione degli impianti non considerano soluzioni dinamiche e flessibili, tranne in alcune applicazioni per l'edilizia civile. Tale aspetto è dovuto sostanzialmente alla mancanza di framework o software sufficientemente efficienti per la simulazione, l'ottimizzazione e il controllo intelligente di un insieme complesso di "oggetti" che comunicando tra loro possano far funzionare l'impianto stesso sempre in modo ottimale.

La simulazione è uno strumento molto utilizzato in campo industriale in quanto permette di valutare e seguire nel tempo il comportamento dei sistemi produttivi. Tuttavia, la modellazione di sistemi complessi come i sistemi di produzione può risultare alquanto difficile con conseguenti semplificazioni nella modellazione che possono avere ripercussioni sull'affidabilità dei risultati. In particolare, negli ultimi anni le diverse tecniche di simulazione hanno consentito ai progettisti di immaginare nuovi sistemi e a coloro chiamati a gestire gli impianti di quantificare e osservare il loro comportamento. Inoltre, indipendentemente dal tipo di sistema simulato si possono valutare e confrontare configurazioni diverse da quella effettivamente implementata al fine di trovare possibili soluzioni a problemi altrimenti difficili da risolvere. Con i modelli alla base di tutte le tecniche di simulazione è possibile studiare il funzionamento di un impianto o di un processo esistente e valutare la sensibilità degli stessi a variazioni nei parametri di funzionamento senza comunque effettuare costosi interventi sull'impianto reale.

Inoltre, proprio a causa della sempre maggiore complessità di tutti quei processi finalizzati ad una produzione manifatturiera sempre più flessibile ed in grado di adattarsi alla crescente variabilità della domanda, le problematiche di modellazione ed analisi di tali processi ne risentono inevitabilmente con conseguenti complicazioni sia nella modellazione stessa che nella successiva

fase di calcolo e simulazione. In tal senso viene in aiuto proprio il paradigma dell'Industria 4.0 che, per sua natura, tende a scorporare in blocchi funzionali i diversi processi, in quanto è possibile farli comunicare sia tra loro per un immediato (real time) intervento, ad esempio correttivo a seguito di anomalie, sia con il centro di controllo aziendale per un ottimale coordinamento di tutte le attività in essere. La suddivisione in blocchi funzionalmente indipendenti e intercomunicanti ha il grande vantaggio, al fine di avere una simulazione più efficiente, di consentire la modellazione separata di ciascun blocco (che a questo punto possiamo a pieno titolo chiamarlo sottosistema dell'intero sistema impianto e/o processo), agevolando, così, le opportune semplificazioni per quei processi che non hanno particolari esigenze di accuratezza e mantenendo, invece, modelli più complessi quando effettivamente necessario.

Per i motivi anzidetti oggi si rende necessario separare alcune tipologie di modelli più semplici da realizzare rispetto ad altre. Infatti, in linea generale, possiamo distinguere modelli statici e quasi-statici da modelli dinamici, nei quali l'evoluzione temporale dei processi è di estrema importanza, modelli causali da modelli statistici, modelli organizzativi, controllo e gestione, da modelli di processo vero e proprio, ecc.

Di particolare rilevanza per il successivo calcolo è la distinzione tra modelli statici e dinamici in quanto per questi ultimi, per i quali la simulazione evolve nel tempo, è necessario passare da modelli continui, per i quali deve essere possibile una soluzione cosiddetta analitica, a modelli discreti. Nella simulazione discreta (nel tempo, ma questo vale anche per la discretizzazione spaziale, sia nello spazio fisico che in spazi multidimensionali) i cambiamenti si verificano in istanti temporali distinti e può essere guidata da incrementi discreti del tempo (Time Step) o da eventi specifici (Event Driven).

Pertanto, la simulazione, intesa come sopra descritto, offre vantaggi sostanziali alle aziende anche se possibili vantaggi diretti di un simulatore sono difficili da determinare. Ad esempio, la simulazione aiuta a ridurre i rischi e i tempi di avvio di un impianto o di ripartenza di processi industriali in seguito ad aggiornamento nelle macchine. Altro possibile vantaggio è nel test di software aziendali di controllo e gestione prima della loro implementazione in azienda.

Lo sviluppo, il test e la formazione di un modello di simulazione consentono, tra l'altro, un approccio incrementale, strettamente integrato con il ciclo di vita del progetto di impianto, che consente di correggere eventuali errori di progettazione.

Tutto quanto sopra ha portato a immaginare un percorso di dottorato focalizzando in modo particolare il lavoro di tesi sulla realizzazione di un Framework, completo di front-end di visualizzazione e progettazione user-friendly, che consenta di simulare, progettare e controllare in modo ottimale (o come vedremo sub-ottimale) un qualsiasi impianto industriale configurato come

un sistema IoT i cui componenti possono essere descritti attraverso modelli fisico-matematici opportuni e che possano trasferire tra loro informazioni in ottica IoT.

Poiché l'obiettivo generale del progetto di ricerca è stato quello di sviluppare un Framework di simulazione di un generico sistema IoT in ambito industriale, la sfida è stata quella di sviluppare un pattern di algoritmi innovativi che consentisse di applicare il software ad un qualsiasi sistema visto come un insieme di ISM.

In seguito allo sviluppo del Framework si è prima effettuata la validazione attraverso alcuni casi studio numerici e successivamente è stato scelto un caso studio reale nell'ambito del settore industriale degli impianti di produzione dell'energia termica e frigorifera a servizio di produzioni industriali.

Durante il primo anno è stato effettuato uno studio approfondito dello stato dell'arte sull'applicazione sperimentale in ambito industriale di diversi algoritmi di simulazione per l'ottimizzazione e la gestione intelligente di sistemi e soluzioni impiantistiche, alla luce del paradigma IoT (Internet of Things).

In seguito a tale studio, durante il secondo anno di Dottorato si è voluto orientare l'attività di ricerca verso la realizzazione del framework oggetto della tesi di dottorato per la simulazione, l'ottimizzazione ed il controllo di un generico impianto industriale in ottica IoT. A tal fine è stato svolto un periodo di internship della durata di 6 mesi all'interno dello spin off del Politecnico di Bari "Wideverse srl", durante il quale sono state approfondite le tecniche di sviluppo grafico per orientare al meglio il Framework verso l'usabilità dell'utilizzatore finale.

Infine, durante il terzo anno è stata effettuata la validazione analitica del framework e sono stati simulati diversi case study reali. In particolare, durante i 3 mesi di collaborazione con la "Universidad del Atlantico Medio" (Las Palmas, Spain), è stata effettuata la validazione del Framework con il case study reale illustrato nella presente tesi che è strutturata come segue: nel capitolo successivo si descrive lo stato dell'arte che ha portato alla presente ricerca, seguito dalla descrizione del Framework sia in termini di aspetti teorici alla base della tecnica numerica sviluppata sia implementativi del software stesso. Vengono poi illustrati alcuni esempi puramente numerici per dimostrare l'efficienza ed efficacia delle tecniche studiate ed il tutto viene applicato ad un caso reale di impianto a servizio di processi industriali. Le tesi, quindi, termina con alcune considerazioni conclusive sui risultati ottenuti e sui possibili sviluppi futuri.

## **2. STATO DELL'ARTE**

Viene ora presentata una analisi dello stato dell'arte riguardante la modellazione dei processi, partendo dai sistemi industriali per approdare, attraverso alcuni approfondimenti intermedi, alla base concettuale del progetto di ricerca: la gestione di sistemi complessi e la modellazione ISM.

### **2.1. Modellazione dei processi industriali**

Un processo industriale è costituito da una serie di unità integrate tra loro in modo razionale con l'obiettivo generale di convertire alcune materie prime in un prodotto finito in modo sostenibile: infatti, durante il suo funzionamento l'impianto deve soddisfare una serie di condizioni specificate al momento del progetto ed imposte dalla situazione generale (economica, sociale, ambientale) in cui opera l'impianto. Particolarmente complesse sono le situazioni in cui un impianto deve funzionare in condizioni dinamiche (Naghshbandi, S. N. et al. 2020) a causa di input che tendono a spostarlo dalle condizioni stazionarie. Questi sono:

- Perturbazioni, indesiderate ma inevitabili in un ambiente industriale (Yang, D. et al., 2018); ad esempio: cambio di materie prime, invecchiamento o imbrattamento delle superfici, fluttuazioni di pressione e temperatura della rete di raffreddamento o riscaldamento, variazioni di temperatura tra notte e giorno, ecc.
- Variazioni delle condizioni operative, volute e stabilite dall'operatore o dal sistema di supervisione dell'impianto (Mellit, A. et al., 2018); ad esempio: cambio tipo di prodotto, ottimizzazione del funzionamento (minimizzazione del consumo energetico, aumento o diminuzione della produzione, ecc.).

Le diverse condizioni operative sono spesso individuate in termini di variabili di processo, anche se, per arrivare ad una maggiore generalizzazione della trattazione, nel presente studio si farà riferimento a generiche variabili di stato del sistema. Uno schema delle variabili di processo di un impianto soggette a variazioni nel corso del suo funzionamento, desiderate o meno come detto sopra, è mostrato in Figura 2.1. Il processo può essere visto come un sistema che, sotto l'azione di determinate variabili di input (cause), genera determinate variabili di output (effetti).

Tra le variabili di input è possibile distinguere variabili manipolabili (i veri input mostrati in figura e che sono a disposizione dell'operatore per apportare correzioni nel processo) e da disturbi (che dipendono dall'ambiente esterno e sono al di fuori del controllo dell'operatore). Mentre le variabili manipolabili sono spesso anche facilmente misurabili, solo alcuni disturbi possono essere misurati o in qualche modo stimati; la maggioranza è invece da considerarsi sconosciuta e non misurabile.

Due categorie possono essere essenzialmente distinte tra le variabili di output del processo: variabili di prestazione e variabili intermedie. Le prime sono le variabili controllate in senso lato (le variabili interne mostrate in figura), generalmente non tutte misurabili direttamente, legate agli obiettivi di

ottimizzazione dell'impianto; le seconde sono le variabili fisiche, misurabili in qualche modo (i veri outputs mostrati in figura, indicanti il funzionamento del sistema e quindi le prestazioni attuali.

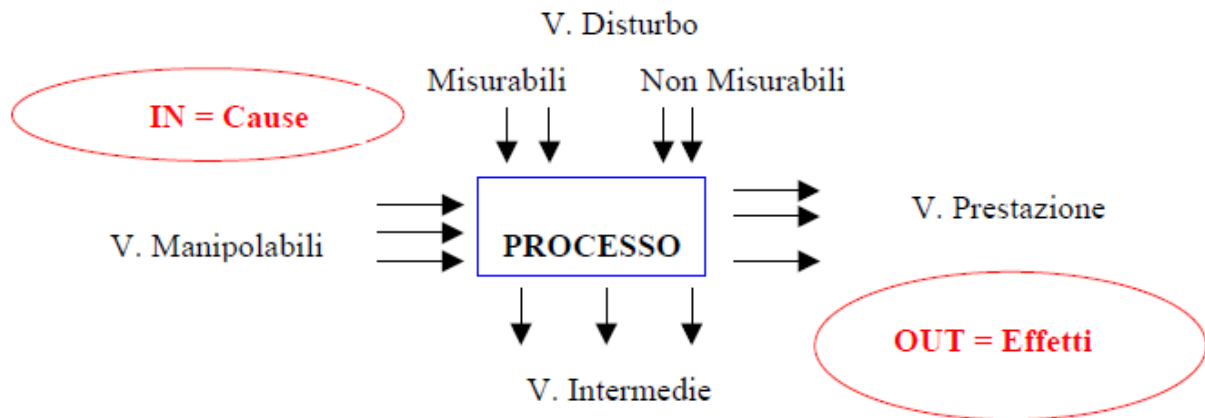


Figura 2.1. schema delle variabili di processo

## 2.2. Industry 4.0 e IoT

Le attività della produzione industriale richiedono, per essere veramente competitive, una elevata qualità non solo delle produzioni ma soprattutto della organizzazione aziendale. In tali situazioni l'IoT consente di organizzare l'attività produttiva attraverso una progettazione integrata, forme di lavoro meglio organizzate, controllo di qualità delle materie prime, del processo di trasformazione nonché del prodotto finito, la logistica ed il mercato nonché il rapporto con i clienti e con i fornitori (Papetti, A. et al., 2018).

La gestione di questi aspetti può essere effettuata sia off-line che in real-time utilizzando ad esempio metodologie ed applicazioni tipiche della Building Automation o del Facility Management (Verma, A. et al., 2019).

Il paradigma dell'Industria 4.0, all'interno del quale si inserisce in modo efficace il mondo dell'IoT, porta a modificare il modo di definire e pianificare il ciclo di vita dei prodotti, la loro gestione durante l'intera catena del valore, anche in contesti diversi, con attori differenti e soprattutto in aree geografiche anche molto distanti.

L'Industria 4.0, evoluzione della Smart Manufacturing (Thoben, K. et al., 2017), si basa sulla connessione e integrazione di differenti attività e tecnologie in modo da consentire il controllo di tutte le fasi produttive, del ciclo di vita del prodotto, della sua distribuzione e commercializzazione. D'altra parte, l'Industria 4.0 applicata al mondo delle imprese deve basarsi su alcuni punti essenziali che facilitano lo sviluppo tecnologico e organizzativo: la verticalizzazione delle attività caratterizzate da una struttura gerarchica importante (integrazione verticale), la parallelizzazione delle attività che possono essere considerate indipendenti tra loro (integrazione orizzontale), la



capacità di gestire in modo intelligente la produzione, gli strumenti utilizzati, non solo le macchine e gli impianti ma anche gli edifici e gli ambienti di lavoro.

Lo Smart Manufacturing è alla base del paradigma IoT, anzi in esso trova la sua più efficace realizzazione. Infatti, ben prima che l'espressione Internet of Things si affacciasse nel mondo delle tecnologie avanzate, si potevano già trovare applicazioni che nulla avevano di meno dell'IoT odierno. Per questo si può, in realtà, considerare, sotto questo punto di vista, l'IoT e l'industria 4.0 un sistema di tecnologie ed applicazioni industriali che iniziano ad essere consolidate e mature.

Lo Smart Manufacturing si sovrappone e anzi completa il sistema Industria 4.0. Tutto ciò è stato reso possibile, partendo dalla sua definizione nel mondo dell'industria tedesca, attraverso politiche di ricerca e sviluppo che hanno consentito l'estensione e l'integrazione delle tecnologie avanzate nel campo dello Smart Manufacturing e dell'IoT.

Internet of Things (IoT) è un paradigma che nasce in Germania e raccoglie tutte quelle tecnologie che consentono non solo di collegare ad Internet qualunque tipo di *oggetto*, ma soprattutto consentire la comunicazione tra i diversi apparati che così potranno dialogare con un server centrale e ancor meglio tra loro anche in modo completamente autonomo (Phuyal, S. et al., 2020; Arnold, C. et al., 2016). Ciò consente di alleggerire il compito delle unità centrali deputate al controllo di ogni singolo apparato, come avviene nei tradizionali sistemi automatici, trasferendo tale capacità alle singole macchine e a gruppi di esse. Tutto ciò porta con sé un'altra esigenza, quella dell'identificazione di ogni oggetto collegato fornendo così una *identità elettronica* ad ogni apparato che può essere riconosciuto dagli altri in un continuo scambio di informazioni.

Inoltre, l'elevato numero di nuovi oggetti che ogni giorno viene connesso ad internet tende a trasferire una enorme mole di dati che vengono raccolti ed immagazzinati (Big Data) in grandi data base, per poi essere rielaborati (ma ciò può avvenire anche in real time) per fornire nuove informazioni ed essere di ausilio nelle fasi di Decision Making (Hajjaji, Y. Et al., 2021).

In questo modo, ad esempio, una macchina si collega, attraverso internet, a diversi sensori che, per diverse ragioni, non possono essere connessi direttamente e riceve informazioni da questi sensori: ad es. condizioni microclimatiche (da sensori che potrebbero anche essere parte integrante della macchina stessa) e di area geografica (con dati da sensori posti a notevole distanza e che forniscono dati macroscopici: forniscono previsioni anche in tempo reale) che consentono di valutare specifici interventi che dipendono dalle condizioni stesse e dalle previsioni metereologiche.

Tutte queste interconnessioni formano reti di oggetti interconnessi che costituiscono il cuore del problema che si sta analizzando: i sistemi interconnessi di grandi dimensioni.

Le principali applicazioni dell'Internet of Things spaziano nei vari campi della tecnologia e delle attività sociali che, in generale, hanno grandi quantità di informazioni da scambiare:

- Dalla casa intelligente alle smart city

- Dai sistemi di generazione e distribuzione dell'energia su larga scala alle installazioni locali (Micro-grid, Smart Grid, ecc.) che si stanno diffondendo a macchia d'olio e che vanno in qualche modo coordinate e controllate.
- Controllo delle produzioni industriali, non limitandosi alla sola automazione dei processi, ma soprattutto al coordinamento con tutte le attività della filiera, dalla produzione delle materie prime fino alla commercializzazione dei prodotti finiti
- La “robotica collaborativa” che interviene anch'essa pesantemente sia nel mondo industriale ma, oggi sempre più anche in agricoltura con la trasformazione dell'agricoltura di precisione in agricoltura 4.0.
- Ospedali intelligenti nei quali i risultati di ogni attività (cartelle cliniche, ordini di medicinali, ecc.) sono immediatamente disponibili per tutte le altre al fine di consentire una gestione intelligente non solo delle risorse, ma soprattutto delle problematiche connesse ad ogni singolo paziente con possibili inclusioni di dati esterni all'a stessa organizzazione ospedaliera (centri medici e di ricerca di tutto il mondo, condivisione dei dati: l'epidemia da Covid 19 lo ha imposto).
- E molte altre...

### **2.3. Automazione, Ottimizzazione e Controllo**

Un processo industriale è, come già detto, costituito da una serie di unità interconnesse a loro volta costituite da macchine elementari, il tutto finalizzato alla realizzazione di prodotti partendo da materie prime. Un impianto oltre a dover far fronte a una serie di eventi e/o perturbazioni che ne possono alterare il corretto funzionamento, è soggetto ad una serie di vincoli operativi di varia natura (Ibarra-Junquera, V. et al., 2021) tra i quali, in ordine di priorità decrescente, si possono elencare:

- Sicurezza
- Regolamentazioni ambientali
- Specifiche di produzione
- Economia dell'impianto

L'obiettivo del sistema di controllo, in senso stretto, è quello di mantenere le variabili di processo sui valori desiderati. In senso più generale, un sistema anche tradizionale di automazione di un impianto include le funzioni di: sicurezza, controllo e ottimizzazione, secondo lo schema di Figura 2.2.

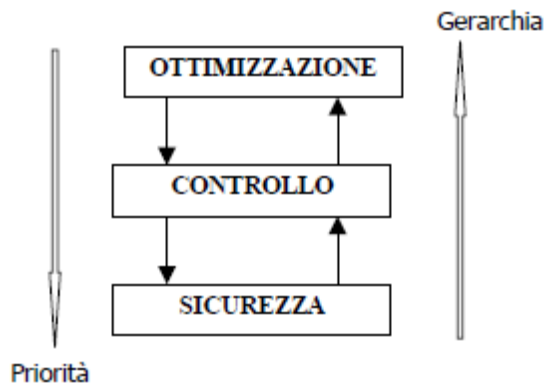


Figura 2.2. Funzioni del sistema di automazione di un impianto

La funzione sicurezza gestisce le condizioni di emergenza; si basa su dispositivi autoazionanti (valvole di sicurezza, di blocco, dischi a rottura, ecc.); ha la priorità più elevata. La funzione controllo gestisce le condizioni ordinarie; si basa su regolatori, in genere di tipo standard (PID).

La funzione ottimizzazione ha lo scopo di mantenere il processo nelle condizioni operative ottimali; determina i valori di riferimento dei regolatori; si basa su sistemi di calcolo più avanzati; non è prioritaria, ma ha il livello tecnologicamente più elevato.

La schematizzazione del problema in questi termini è essenziale ai fini della definizione degli obiettivi, per la scelta delle variabili misurate e manipolate (cfr. Figura 2.1), per la scelta della struttura del sistema di controllo.

Nel caso più semplice, ai fini del controllo, un processo può essere schematizzato con una sola variabile controllata in uscita e una sola variabile manipolata in ingresso, oltre ai disturbi (sistema SISO: Single Input Single Output). Nella maggior parte dei casi i processi hanno molte variabili in ingresso e in uscita (sistemi MIMO: Multi Input Multi Output) e sono caratterizzati da interazione dei diversi ingressi sulle uscite. In casi particolari è ancora possibile identificare variabili di ingresso aventi effetto predominante su certe uscite e ricondurre il sistema multivariabile ad una serie di sistemi di tipo SISO tra loro non interagenti. Negli esempi precedenti possono essere individuati gli elementi essenziali costituenti il processo controllato, schematizzati in termini di diagramma a blocchi nella Figura 2.3.

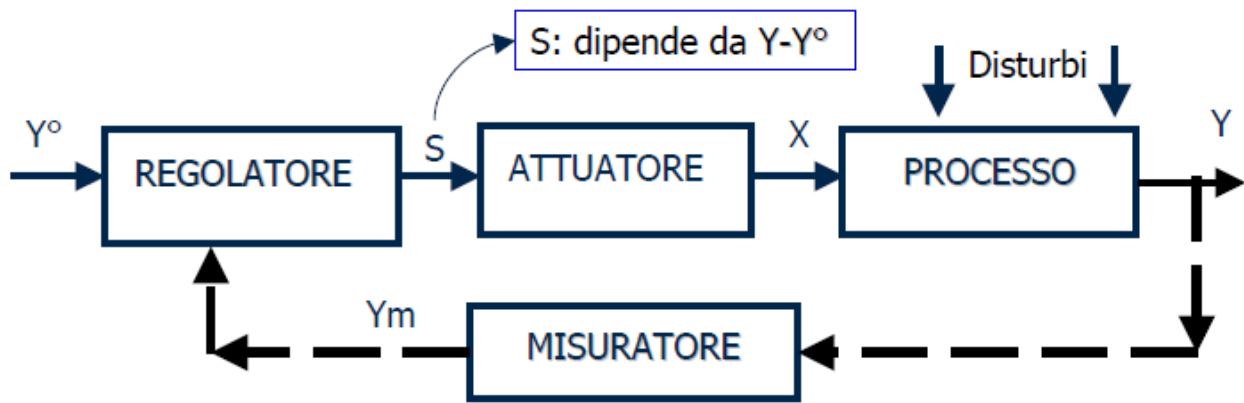


Figura 2.3. Elementi e variabili in un processo controllato in retroazione.

Ciascun blocco rappresenta in modo pittorico il rapporto causa - effetto tra variabili di ingresso e variabili di uscita. Introducendo nel blocco il modello matematico dell'elemento è possibile una valutazione quantitativa dell'evoluzione del sistema nel tempo. Di seguito è riportata una descrizione qualitativa dei diversi elementi e del loro funzionamento nell'anello di regolazione in retroazione (Dodds, S., J., 2015).

- Il Processo: rappresenta l'operazione (le variabili) che si vuole controllare, ovvero l'effetto delle variabili manipolate  $X$  e dei disturbi sull'uscita; normalmente questi due effetti sono diversi tra loro.
- L'Attuatore: rappresenta il dispositivo che trasforma le correzioni  $S$ , decise dal regolatore, in variazioni sulle variabili manipolate  $X$  (ad esempio una valvola di regolazione che agisce su una portata di fluido).
- Il Regolatore: stabilisce l'azione di controllo da effettuare affinché la variabile di uscita  $Y$ , che è nota attraverso la sua misura  $Y_m$ , sia uguale al valore desiderato  $Y^\circ$  (set-point); la variabile  $E = Y_m - Y^\circ$  è definita errore. L'azione di controllo dipende dall'errore (retroazione); l'entità dell'azione correttiva  $[S=f(E)]$ , dipende dalla legge di controllo (algoritmo del regolatore).
- Il Misuratore: trasferisce al sistema di controllo il valore  $Y_m$  della variabile controllata  $Y$ . Il blocco  $M$  ingloba l'elemento sensibile, il trasduttore, la linea di trasmissione. In parallelo la variabile misurata è generalmente registrata per avere una traccia del suo andamento nel tempo.

La struttura del sistema di controllo rappresenta il modo con cui le informazioni sul processo, ottenute attraverso le misure, vengono usate per effettuare le correzioni nelle variabili manipolate in ingresso al processo.

La struttura più comune, come già illustrato in precedenza, è quella in retroazione. Sono utilizzate direttamente le misure delle variabili controllate in uscita per intervenire sulle variabili manipolate; l'entità dell'azione correttiva dipende proprio dallo scostamento della variabile di uscita rispetto ai valori desiderati. Tali sistemi sono anche detti ad anello chiuso in quanto si formano circuiti (anelli)

orientati chiusi (si fa riferimento a strutture a grafo orientato) e che rappresentano proprio il punto di partenza di questo progetto di ricerca.

Un'altra struttura di controllo base è quella in avanti (Feed-Forward), o in Anello Aperto. In questo caso sono utilizzate le misure di alcune variabili di ingresso per effettuare le correzioni sulle variabili manipolate. Al vantaggio di una maggiore velocità di intervento rispetto allo schema in retroazione, corrisponde lo svantaggio della mancanza di una verifica sugli effetti ottenuti sulle variabili in uscita dal processo oltre a possibili situazioni di instabilità che vengono normalmente corrette con sistemi retroazionati. Per questa ragione lo schema in avanti si usa raramente da solo; più comunemente, è accoppiato allo schema in retroazione, al fine di coniugare i vantaggi delle due configurazioni.

Nel caso in cui siano disponibili altre misure, cioè oltre alle variabili di uscita anche di alcune variabili intermedie che possono risentire degli effetti dei disturbi più prontamente delle uscite, queste possono essere usate per stimare alcuni disturbi e migliorare il controllo attraverso schemi predittivi in cascata. In questo caso si possono avere, ad esempio, più regolatori in serie con obiettivi di controllo specifici e stabiliti, dal regolatore più esterno (primario), ad ogni regolatore interno (secondario).

Molto spesso si fa affidamento sulla condizione che tutte le variabili controllate siano anche considerate direttamente misurabili. Tuttavia, in molti casi di rilevante interesse industriale le misure, pur essendo teoricamente attuabili, nella realtà applicativa non sono sempre possibili, oppure i sistemi di misura sono costosi o scarsamente affidabili o hanno proprietà dinamiche sfavorevoli (ritardi). Esempi possono essere le misure di proprietà finali, richieste nel controllo di qualità (grammatura della carta, resistenza meccanica di una fibra), ma anche misure di variabili intermedie (pesi molecolari o loro distribuzione, viscosità, composizione di una miscela a molti componenti).

Anche in questi casi, come per alcuni disturbi, è necessario ricorrere a schemi più complessi nei quali la grandezza non misurata direttamente viene ricostruita per mezzo di uno stimatore, il quale si basa sui valori delle variabili intermedie disponibili e/o su un modello del processo. Queste stime sono usate in uno schema di controllo inferenziale, per effettuare le correzioni sulle variabili di ingresso al processo.

Il grande sviluppo tecnologico degli ultimi anni ha reso disponibili calcolatori di potenzialità crescente a prezzi sempre più bassi, con un impatto profondo nel controllo dei processi. Le possibili applicazioni sono molto diversificate, ma i due modi base di impiego sono:

**Controllo Digitale Diretto:** in questo caso il calcolatore sostituisce il regolatore, nel senso che riceve direttamente le misure dal processo e calcola l'azione di controllo.

Controllo con Calcolatore in Supervisione: in questo caso il calcolatore acquisisce i dati dall'impianto, li elabora in modo da perseguire una strategia di ottimizzazione e stabilisce i valori di set-point per i regolatori che agiscono sulle singole unità del processo. Questi possono essere controllati ancora da calcolatori più piccoli, ma nei casi più comuni i regolatori che agiscono sul processo sono ancora di tipo convenzionale. Le modifiche da effettuare sulle variabili di processo, possono essere attuate automaticamente, oppure dopo consenso da parte dell'operatore.

#### **2.4. Rilevanza del controllo di processo: l'esempio della gestione delle filiere alimentari (Food Supply Chain)**

L'incidenza della parte "Strumentazione e Controllo", che include strumenti di misura, trasmettitori, sistemi di controllo e attuatori, sul costo totale di un impianto può essere stimata intorno al 30%, ma può essere anche molto superiore, dipendendo fortemente dal grado di automazione dello stesso.

Nel caso di progettazione di impianti nuovi, una esigenza che si è affermata è quella della interazione tra le fasi di progettazione dell'impianto e del sistema di controllo, per far sì che la possibilità di garantire certe specifiche (controllabilità dell'impianto) sia affrontata direttamente fin dalle prime fasi e non lasciate alla fase finale. In tal modo, si cerca di limitare i rischi dovuti alla impossibilità (o estrema difficoltà) di controllare un impianto la cui progettazione, ottimale dal punto di vista di processo, non ha tenuto conto delle caratteristiche del sistema di controllo.

Fino a circa 25 anni fa, la modellazione e la simulazione della lavorazione degli alimenti era principalmente dedicata alla conservazione dei prodotti agricoli con adeguate condizioni microclimatiche e secondo i loro requisiti di sicurezza organolettica e igienica, nonché dell'impatto ambientale (Perrot et al., 2011; Bianchi et al., 2015).

A causa della maggiore comprensione dei processi, sono emerse applicazioni che considerano altri attributi di qualità come il contenuto di umidità, il colore, la viscosità, a volte la composizione degli alimenti. Più recentemente, anche a causa della maggiore potenza di calcolo, si è tenuto conto della struttura e della reologia degli alimenti (Theys T.E., 2009). Parallelamente, i progressi nelle tecniche di imaging a vari livelli di dettaglio e metodi analitici migliorati hanno permesso di studiare diverse scale strutturali e interazioni tra specie chimiche, principalmente tra macromolecole e piccole molecole: il cibo è un sistema molto complesso, con varie possibili interazioni alle diverse scale (Baudrit et al., 2010).

Inoltre, negli ultimi anni l'importanza della modellazione dinamica è stata sempre più riconosciuta in vari argomenti scientifici e tecnici come strumento necessario per comprendere, progettare, ottimizzare e controllare prodotti, processi e macchine. (Catalano et al., 2013). Nelle produzioni alimentari le attuali applicazioni si basano principalmente su modelli stazionari, che non possono produrre una valutazione realistica degli effetti transitori. Questi ultimi, invece, sono di gran lunga i

più preponderanti nella trasformazione degli alimenti, con evidenti conseguenze negative in termini di efficienza degli impianti e ambientale (Perone C. et al., 2017; Catalano F. et al., 2020).

Lo studio di sistemi complessi, inclusi aspetti puramente matematici e intelligenza artificiale, fornisce metodi e strumenti utili a descrivere ambienti e processi altamente variabili nel tempo, che spesso non sono prevedibili con tecniche di simulazione convenzionali.

In particolare, le produzioni alimentari, dal campo al post-raccolta, sono caratterizzate da diversi attori. Sono, infatti, organizzati in FSC che produce e distribuisce prodotti ai consumatori.

Gli approcci basati su modelli nella trasformazione degli alimenti sono generalmente soggetti a uno o più limiti (Soysal et al., 2012; Bimbenet et al., 2007; Baudrit et al., 2009; Fito et al., 2007; Ioannou et al., 2006; Perrot et al., 2006; Van Impe, 1996), molti dei quali si verificano simultaneamente nella biotecnologia alimentare e nella trasformazione vera e propria (Van Impe, 1996).

In molti campi, i risultati di studi specifici nell'industria alimentare hanno portato a modelli validi in piccoli domini sia in termini di composizione che di ambiente chimico-fisico. Inoltre, il loro quadro concettuale non consente una facile integrazione dei risultati di altri modelli esistenti (Rodriguez-Fernandez, Balsa-Canto, Egea e Banga, 2007, Catalano et al. 2013). La maggior parte dei modelli è costituita da equazioni differenziali relative al trasferimento di calore e massa, mentre gli aspetti microbiologici o chimici sono descritti per lo più da semplici equazioni cinetiche e raramente da equazioni differenziali: l'accoppiamento è possibile ma non facilmente generalizzabile. In molti campi, alcuni risultati importanti sono ottenuti dalla sintesi di dati complessi e dissimili (Desiere et al., 2001. Ayr et al., 2015) che devono essere elaborati con più modelli fisici.

Come dimostrato da Datta (2008), lo sviluppo della struttura alimentare non è solo funzione di parametri fisici quali temperatura e umidità, ma anche della "storia" del prodotto stesso. Uno dei problemi che è stato poco affrontato è come trattare i requisiti contrastanti di ogni particolare sottosistema, ottimizzato per il proprio dominio di conoscenza. Il punto chiave è la capacità di far fronte a diversi tipi di conoscenza, a diverse scale, espressi in diversi formalismi e di poterli considerare in modo unitario. Tuttavia, questo problema è una chiave per il futuro.

Le interazioni tra i vari campi di studio sono state evidenziate in relazione alle problematiche ambientali e alle risorse naturali (Christakos, 2002), alle questioni biologiche (Olivier et al., 2010) e agli aspetti nutrizionali (McLachlan & Garrett, 2008). Tutto ciò porta alla necessità di definire modelli che facciano riferimento a schemi di calcolo sufficientemente consolidati. Da un punto di vista computazionale, i modelli utilizzati nei processi alimentari sono sempre più complessi (Vitrac & Hayert 2007 Sicard et al., 2009, Banga, J.R et al., 2003).

Oltre ai tradizionali obiettivi della gestione della gestione di una generica Supply Chain (SC), come la riduzione dei costi e il miglioramento della reattività, la gestione della Food Supply Chain (FSC)

richiede un diverso approccio gestionale che consideri anche le caratteristiche intrinseche dei prodotti e dei processi alimentari (Van der Vorst et al., 2011). Un prodotto alimentare deve essere di alta qualità e sostenibile: socialmente equo e rispettoso dell'ambiente, oltre ad essere prodotto in modo efficiente, competitivo e redditizio (Kepler, 2004). La rapida evoluzione degli obiettivi di sviluppo sostenibile cambia in quasi tutte le SC, compresa la FSC, e rende inadeguate le strategie tradizionali. Ciò ha portato allo sviluppo di un nuovo concetto fortemente richiesto nell'industria alimentare: la gestione sostenibile della FSC (Sustainable Food Supply Chain: SFSC) (Seuring and Muller, 2008; Ahumada e Villalobos, 2011) aumentandone la complessità.

Inoltre, Profit, People e Planet sono i motori che guidano l'FSCM verso prestazioni efficienti in senso lato (Van der Vorst et al., 2005). È chiaro che questo cambiamento necessita di un approccio integrato che colleghi le decisioni prese all'interno della FSC ai tre pilastri della sostenibilità economica, ambientale e sociale (Chaabane et al., 2012) e gestendo la qualità del prodotto. Questo rappresenta un approccio noto come gestione della logistica alimentare sostenibile (Sustainable Food Logistic: SFL).

La produzione, il trasporto e lo stoccaggio influenzano in modo significativo le decisioni all'interno di ogni sottoprocesso (Chopra and Meindl, 2010). La maggior parte degli studi cerca di utilizzare un approccio integrato per migliorare l'efficienza di tutti i processi (Eksioglu, SD and Jin, MZ, 2006; Mula et al., 2010).

Nello specifico, alcuni studi propongono di gestire più prodotti con lo stesso modello (You et al., 2012), mentre altri incorporano elementi stocastici nei loro modelli. Alcuni di questi elementi sono la gestione della domanda (ad esempio Ahuja (2007)), i tempi di consegna (Van der Vorst et al., 2000), l'offerta e i costi (Azaron et al., 2008) e il comportamento SC (Dabbene et al., 2008).

Inoltre, la capacità di monitorare la durata di conservazione di qualità dei prodotti alimentari lungo una FSC è incorporata in alcuni modelli (ad esempio Eksioglu and Jin, 2006, Yan et al., 2011; Ahumada e Villalobos, 2011; Rong et al., 2011). Alcuni studi logistici (ad es. Akkerman et al., 2009, Blackburn and Scudder, 2009 Bosona and Gebresenbet, 2011) integrano il controllo della temperatura o dell'entalpia (prodotti congelati), considerando fattori aggiuntivi come i tassi di utilizzo dell'energia termica.

Un altro aspetto di fondamentale importanza nei modelli di trasformazione alimentare all'interno di una FSC è l'utilizzo di modelli deterministici. Questo approccio è accettato in quanto i relativi modelli possono essere risolti in modo relativamente semplice. Tuttavia, la maggior parte dei processi sono soggetti a varie incertezze, come la disponibilità delle informazioni e la tempestività dei dati, l'incertezza nell'offerta, nel processo e nella domanda (Van der Vorst et al., 2000). Pertanto, le ipotesi deterministiche non colgono appieno la complessità dei problemi reali, il che potrebbe persino ostacolare l'applicabilità. Al contrario, la necessaria reattività di una FSC richiede



il rapido adattamento del sistema ai cambiamenti nella domanda dei clienti, del mercato o delle materie prime. Affrontare l'incertezza è un problema chiave nella lavorazione degli alimenti. L'integrazione esplicita dell'incertezza nei problemi di simulazione è cruciale nelle applicazioni industriali e di conseguenza nei processi decisionali (Baudrit et al., 2006; Perrot et al., 2006; Petermeier et al., 2002; Bimbenet et al., 2007; Banga, et al., 2008).

Infine, solo pochi studi tengono conto della sostenibilità nei loro modelli ma, allo stesso tempo, ignorano altri indicatori. L'approccio corretto è gestire più obiettivi, che sono in conflitto tra loro.

In ogni caso, in letteratura non è riscontrabile un vero approccio dinamico che affronti la gestione ottimale dell'offerta e/o della domanda in condizioni reali altamente variabili nel tempo. Le tecniche di gestione lato domanda utilizzate nella gestione della rete intelligente possono essere opportunamente trasferite a SFSC.

## **2.5. Demand Side Management e Software applicativi**

Come detto sopra, vi sono tecniche consolidate in sistemi specifici come l'approvvigionamento energetico ed in particolare nel Demand Side Management (DSM) utilizzato per le smart grid. Queste tecniche possono essere efficacemente applicate nella gestione della FSC per affrontare proprio le difficoltà sopra descritte nella modellazione dei sistemi alimentari. Infatti, l'ottimizzazione dell'avviamento e dell'arresto di un impianto o la modifica improvvisa e non pianificata dei parametri di processo possono incidere notevolmente sui costi e sull'impatto ambientale. Il controllo in linea dei processi continui richiede anche modelli dinamici per la compensazione degli inevitabili disturbi, come le variazioni nella fornitura e nella qualità delle materie prime (Trystram and Courtois, 1996).

Il DSM è una funzione importante nella gestione energetica di una rete intelligente, che fornisce supporto per funzionalità in diverse aree come il controllo e la gestione della domanda di elettricità (Nguyen e Yousefi, 2011; Faria et al., 2011a,b). In particolare, la gestione della domanda orientata al cliente, gioca un ruolo chiave nella pianificazione delle utenze (Schisler et al., 2008) e può essere implementata e orientata alla FSC.

Il DSM include strategie di riduzione della crescita del carico e/o servizi energetici flessibili, nonché pianificazione, monitoraggio e valutazione sistematica dei processi (Yoo et al., 2011; Nguyen, 2010).

Mohagheghi et al. (2010) discutono di una strategia innovativa di gestione della domanda per i piccoli e medi consumatori di elettricità. I programmi DSM sono più efficaci quando includono tutti i concetti di Efficienza Energetica (EE), Risparmio Energetico (ES) e Demand Response (DR) che, in particolare, è l'elemento chiave le cui metodologie possono essere esportate come parte della SFSC. Ad esempio, per garantire che sia disponibile un'adeguata fornitura di energia (quindi anche

cibo) per soddisfare la domanda, è necessario effettuare previsioni di carico corrette (condizioni meteorologiche per i prodotti agricoli) (Bai et al., 2006; Mutanen et al., 2011 ).

L'obiettivo del DSM è ridurre il picco della domanda di elettricità e promuovere dispositivi efficienti dal punto di vista energetico ed eco-sostenibili (He et al., 2008; Pedrasa et al., 2009): questi obiettivi sono simili a quelli richiesti per una SFSC. L'obiettivo è ridurre i picchi di domanda per migliorare il fattore di carico, spostando il carico (prodotti in SFSC) tra le diverse ore del giorno o le diverse stagioni (Ruiz et al., 2009; Amir-Hamed Mohesenian-Rad et al. , 2010; Xiong et al., 2011). Ciò ben si adatta al problema di bilanciare l'uso delle risorse in base alle richieste del mercato.

Gli obiettivi strategici del DSM includono anche il flusso di cassa (Howard et al., 2010; He et al., 2010; Wacks, 2008). Pertanto, il DSM consente di gestire in modo affidabile il sistema di produzione di una SFSC includendo in esso anche l'efficienza energetica e la sostenibilità (Mohsenian-Rad, 2010; Zhou et al., 2008).

Lo studio dei problemi di gestione ottimale del DR, es. la riduzione del carico di punta a lungo termine di una smart grid e quindi il relativo rinvio della necessità di realizzare nuove centrali elettriche, con conseguenti impatti ambientali considerevoli (Darby, 2006; Fahrioglu and Alvarado, 2000), possono essere facilmente esportati in problemi di gestione dei processi all'interno di una SFSC. I programmi lato domanda sono progettati per una corretta pianificazione delle risorse. Il DR viene fornito attraverso la previsione delle risorse dal lato della domanda (Allen e Ilic, 2000). L'applicazione di tali tecniche alla SFSC consentirà a un operatore di mitigare il rischio operativo (ovvero il rischio di prezzo e quantità) relativo alle operazioni quotidiane del sistema.

La DR è anche correlata alla regolazione del consumo di elettricità da parte degli utenti finali, ad es. in risposta a incentivi e/o variazioni di prezzo, quando programmati dalla rete o dall'operatore di mercato (Faria et al., 2015; MacCormack et al., 2012).

L'integrazione delle risorse secondo la DR può essere pienamente affrontata se si considerano anche le risorse disponibili della Distributed Generation (DG). DG (corrispondente alla produzione sul campo in SFSC) e DR (corrispondente alla DR nella SFSC) possono essere messi insieme attraverso l'implementazione di reti intelligenti (virtuali e/o fisiche) (Fang et al., 2012). Poiché DR, DG e unità di stoccaggio in una SFSC possono a loro volta influenzare la domanda stessa, è necessaria un'adeguata pianificazione di queste risorse considerando tutti i possibili feedback. Tan et al., 2014 propone un modello congiunto per l'ottimizzazione della programmazione del sistema di stoccaggio con DR e DG, specificamente focalizzato in quel caso sulla generazione di energia eolica ma facilmente esportabile ad altri contesti, minimizzando i costi totali. In particolare, vengono introdotti i Virtual Power Players (VPP), che mirano ad aggregare piccole risorse energetiche, comprese le piccole smart grid (DG distribuite) e la DR, consentendo di partecipare ai

mercati dei prodotti elettrici. Quindi un VPP può anche gestire parte di una rete di distribuzione SFSC. La gestione delle risorse da parte di un VPP può essere effettuata riducendo al minimo i costi operativi o aumentando i profitti (Morais et al., 2012).

L'ampia integrazione delle risorse DG e DR comporta diverse possibilità di risoluzione dei problemi legati all'intermittenza e all'imprevedibilità della loro disponibilità. In questo quadro, un'adeguata attenzione è richiesta per lo stoccaggio, problema comune nei sistemi di produzione di energia e nei sistemi di produzione alimentare, per mantenere livelli più elevati di qualità operativa e sicurezza (Karangelos and Bouffard, 2012). A tal fine, un VPP è in grado di gestire le risorse disponibili per soddisfare i requisiti di una SFSC (Faria et al., 2012).

L'aggregazione di DG e DR con caratteristiche simili può essere eseguita utilizzando strumenti di clustering (Vale et al., 2011). Possono essere definiti diversi cluster di risorse, che mirano a identificare caratteristiche comuni definendo meglio le diverse risorse in ogni specifica condizione. Una volta ottenuta una serie di scenari operativi, comprendendo la presenza di disturbi sulle principali variabili, è possibile definire gruppi di risorse idonei per la SFSC.

All'interno del DSM, i consumatori possono anche essere inclusi sia individualmente che aggregati. La simulazione richiede la conoscenza dei dati di carico e dei contratti tra i clienti e i loro fornitori. Questi contratti possono includere clausole di flessibilità che consentono al gestore della rete di ridurre il carico per alcuni clienti. Esiste un chiaro parallelismo con i contratti di fornitura della SFSC.

La timeline di funzionamento (e quindi di simulazione) di una smart grid così come di una SFSC è spesso composta da una sequenza di periodi con uno o più eventi che si verificano nel tempo. Ogni singolo intervallo può essere modellato attraverso processi continui; se non si verificano variazioni importanti e si trascurano i transitori, il simulatore si comporta come un sistema con semplici eventi discreti. All'inizio della simulazione, tutti i parametri delle variabili vengono definiti in base allo stato iniziale considerato. Tuttavia, in generale, qualsiasi cambiamento nei parametri del sistema può causare instabilità nella simulazione e quindi deve essere fornito un tempo di simulazione sufficientemente piccolo affinché il sistema sia sempre in uno stato stabile: questo rappresenta uno dei problemi da affrontare.

Nella simulazione di sistemi DG-DR aggregati, lo stato del sistema può essere implementato anche nell'hardware corrispondente, con la possibilità di simulare in tempo reale il comportamento dell'intero sistema. Dopo ogni transizione, lo stato del sistema viene salvato e il primo evento DR viene simulato. Dopo questo evento, il nuovo stato del sistema viene salvato e così via. Questa sequenza viene ripetuta per il numero di periodi di simulazione. Dopo aver salvato i risultati di un evento, questi diventano lo stato iniziale della rete per il periodo successivo: sarà uno dei punti chiave degli algoritmi proposti.

Per quanto riguarda i software di simulazione oggi disponibili sul mercato, qui verrà effettuata solo una breve descrizione di quelli applicabili ad una SFSC al fine di mostrare le peculiarità in un campo specifico senza disperdere la trattazione in inutili digressioni.

DemSi è un simulatore di DR (Faria et al., 2012; Faria e Vale, 2014), mentre SPIDER (Faria et al., 2015) costituisce una piattaforma di simulazione per l'integrazione di DR e DG. Partendo dall'esperienza maturata con l'implementazione e l'utilizzo di DemSi, gli autori di questo studio hanno apportato diversi miglioramenti. In particolare, è stato evidenziato che modelli molto diversi possono essere implementati combinando l'uso di MATLAB (Xiong et al. 2011), una piattaforma molto utile per simulare singoli processi anche complessi, ottimizzando la gestione dei sistemi. Le combinazioni DG-DR sono spesso implementate anche tramite GAMS, un software di ottimizzazione generale (Amir-HamedMohesenian-Rad et al., 2010). PSCAD è uno strumento utile per simulare reti elettriche ma può essere utilizzato per processi che hanno un analogo elettrico (Hindi et al., 2011). Questi modelli, ad esempio, sono stati adattati in (Faria et al., 2015) per essere inclusi in SPIDER. Infine, Simulink viene spesso utilizzato anche come piattaforma di base per la simulazione di reti di processo. I singoli processi possono essere scelti da una serie di modelli già disponibili o, in alternativa, l'utente può progettare il proprio processo da zero e includerlo nella libreria (Xiong et al., 2011).

Quanto appena detto consente di affermare che le tecniche utilizzate nella DSM sono esportabili ad altre applicazioni quale la SFSC. In tutti questi casi ritroviamo un aspetto comune: la presenza di sistemi interconnessi.

### 3. MATERIALI E METODI

Un impianto industriale è oggi da considerarsi un sistema complesso in quanto spesso non solo è composto da numerose tipologie di macchine, ma è anche caratterizzato da una forte variazione temporale delle produzioni. Tale complessità, pertanto, si riflette sulla progettazione degli impianti industriali e aumenta ancora di più se è necessario tener conto del comportamento degli stessi in fase di esercizio a causa della elevata variabilità dei parametri influenzanti la dinamica dei processi coinvolti.

Ad esempio, la progettazione di impianti per la produzione dell'energia viene basata su dati ricavati da diagnosi energetiche effettuate in regime stazionario considerando spesso un intervallo temporale non superiore alla mensilità. Si tratta in sostanza di calcolare il fabbisogno elettrico e termico medio nel periodo considerato: tale procedura, però, ben si adatta qualora si debba solo rispondere a esigenze di tipo normativo. Se invece si deve effettuare una progettazione al di fuori delle prescrizioni legislative, una simulazione energetica condotta in regime dinamico in un arco temporale di almeno tre mesi, può fornire certamente molte informazioni su come il sistema risponde alle sollecitazioni interne ed esterne. Ciò non toglie comunque che, integrando su tutto il periodo di simulazione i risultati ottenuti, sia possibile ricavare il fabbisogno energetico complessivo. Tale approccio è oggi nuovo in ambito industriale, mentre è attualmente in fase di sviluppo normativo in ambito civile, soprattutto nella diagnosi energetica di un edificio, o ancor di più di un sistema di edifici, e viene finalizzata alla progettazione degli impianti di produzione e distribuzione dell'energia per il complesso di edifici analizzato. Inoltre, ad oggi poiché la maggior parte dei sistemi di generazione dell'energia installati nelle aziende definite "energivore" vengono dimensionati utilizzando i dati medi mensili dei consumi elettrici e termici, non sempre i risultati realmente ottenuti durante il loro funzionamento rispecchiano quelli attesi in fase di progettazione.

Di conseguenza gli impianti non solo risultano inadeguati al tipo di carico della singola azienda, ma spesso non sono nemmeno flessibili rispetto all'evoluzione che il processo produttivo può avere negli anni a venire.

Il presente progetto di ricerca, pertanto, vuole seguire la metodologia della simulazione dinamica sia nella progettazione che nella gestione intelligente degli impianti di gestione di energia e più in generale degli impianti industriali di processo.

I tre fattori mancanti per realizzare quanto detto sopra sono:

- Simulazione;
- Controllo intelligente;
- Implementazione di sistemi ad oggetti comunicanti detti anche sistemi IoT (Internet of Things).

Il progettista, infatti, non ha mezzi di progettazione che gli consentano la simulazione di una molteplicità di configurazioni impiantistiche tale da poter scegliere di volta in volta quella ottimale. Inoltre, gli impianti sono progettati senza considerare soluzioni dinamiche e flessibili: tale aspetto è dovuto sostanzialmente alla mancanza di framework adeguati alla simulazione, l'ottimizzazione e il controllo intelligente di un insieme di "oggetti" che comunicando tra loro possano far funzionare l'impianto stesso sempre in modo ottimale.

Per quanto riguarda le applicazioni dell'IoT si riscontra che quelle più diffuse e riuscite sono legate alle soluzioni più semplici e di immediata realizzazione limitando fortemente le potenzialità di apertura e raggiungibilità tipiche dell'Internet delle cose. Tra queste troviamo i contatori intelligenti (smart metering) per misurare i consumi, le soluzioni domotiche, la sicurezza delle persone, i servizi di infomobilità e la registrazione dei parametri di guida. Vi sono però diversi ambiti in cui l'Internet of Things è stato soltanto immaginato. Questo avviene in sperimentazioni di piccola scala e tra queste le più avanzate si trovano in ambito energetico con le Smart Grid.

Pertanto, l'obiettivo del progetto di ricerca è stato quello di realizzare un framework completo di interfaccia user-friendly che consenta di simulare, progettare e controllare in modo ottimale un qualsiasi sistema IoT nell'ottica degli Individual System Model (ISM) e nello specifico sono stati simulati case study in ambito industriale.

### **3.1. Sistemi Interconnessi**

L'interconnessione è intesa come un trasferimento di materia, energia, informazioni, ecc. da un sistema o sottosistema a un altro: questa definizione si applica a diverse tipologie di sistemi: organismi biologici, sistemi sociali, economici, dispositivi elettronici, ecc. o anche concetti astratti come sistemi di equazioni (Martínez-Villegas et al., 2018). In particolare, ogni sottosistema rappresenta la relazione causa-effetto tra le variabili di input e le variabili di output. Un tale sistema (insieme di sottosistemi interconnessi) può essere rappresentato da un grafo i cui nodi sono i sottosistemi del sistema. Vandendorpe e Van Dooren, 2008, hanno fornito una definizione efficace di un sistema interconnesso: può essere visto come un insieme di equazioni algebrico-differenziali lineari/non lineari di tipo Multiple-Input-Multiple-Output. Ciascun sottosistema è a sua volta caratterizzato da una funzione di trasferimento MIMO lineare / non lineare. Pertanto, introducendo il modello matematico degli elementi in ogni sottosistema, è possibile una valutazione quantitativa dell'evoluzione dell'intero sistema nel tempo.

Secondo Goetz e Sheno, 2008, oggi l'industria moderna è fortemente basata sull'uso di diverse risorse naturali e/o generate (ad esempio acqua, elettricità, calore, ecc.) e sulla loro distribuzione con la necessità di metodi che riducano il più possibile gli sprechi: esiste un'ampia letteratura su questo argomento (Feng, B. et al., 2021; Deb, C., & Schlueter, A., 2021). Pertanto, non esiste un

approccio univoco sulla soluzione numerica di tali sistemi (si veda ad esempio Garwood et al.; 2018 Chiacchio et al., 2020; Iannino et al., 2020; Meyer, 2015; Gomes et al., 2018).

### **3.2. Individual System Modelling**

Un interessante metodo di modellazione e simulazione di sistemi interconnessi composti da sottosistemi e interconnessioni complesse si basa sull'approccio noto anche come co-simulazione e sul paradigma dei modelli di sottosistema individuali (Individual System Modelling: ISM).

Concettualmente, gli ISM sono stati introdotti da Di Perna R.A., 1970, dove viene costruito un modello continuo o discreto per ogni sottosistema e le interconnessioni tra questi modelli utilizzate per produrre la simulazione complessiva dell'intero sistema. Nonostante questa vecchia definizione, poco è stato fatto per studiare tale approccio (l'ISM) da un punto di vista abbastanza generale, ma solo alcune sottoclassi di sistemi interconnessi. Una classe di modelli dinamici non lineari abbastanza studiata è quella del Block-Oriented Model (BOM, è una sottoclasse di ISM) in cui si possono trovare modelli non lineari e/o lineari con/senza memoria, ciascuno rappresentato come un singolo sottosistema (Wills et al., 2013). Questi modelli, definiti anche come modelli causali (Kofránek et al., 2017), sono ampiamente applicati nella simulazione numerica di diversi processi (tra molti altri: Shanshiashvili e Kavlashvili, 2020; Juneja et al., 2019; Ira-Ramírez et al., 2014). Un'altra sottoclasse di ISM abbastanza studiata è quella dei modelli acausali (Kofránek et al., 2017) consistenti nella definizione di istanze di ciascun modello di sottosistema in cui sono definite solo le relazioni tra le diverse variabili di ciascun modello. Quindi, l'intero modello viene risolto nel suo insieme o tramite algoritmi utilizzando metodi specifici di riduzione della complessità computazionale.

Il controllo di sistemi interconnessi complessi ha una vasta gamma di applicazioni: Zhang et al., 2020; Martínez-Villegas et al., 2018 (diagnosi dei guasti), Zohdi, T.I., 2018 (veicoli senza pilota), Dolgui et al., 2019 (problemi di pianificazione), Galdiero et al., 2016; Pöchacker et al., 2014 (reti idrauliche), Satin et al., 2019; Tamborrino et al., 2019; McPhee et al., 2004; Shi e McPhee, 2000 (industrie alimentari, sistemi meccanici), Perea-López et al., 2003 (impianti di lavorazione di materiali diversi e gestione della SFSC).

Inoltre, le deviazioni dal corretto funzionamento di questi sistemi interconnessi o di loro parti non sono più solo difficoltà tecniche; rappresentano un pericolo con un impatto sulla sicurezza globale (Galinec e Luić, 2020). Questo è uno degli inconvenienti che vengono studiati di volta in volta con tecniche differenti. Un'altra difficoltà sorge quando si ha a che fare con sistemi interconnessi molto grandi per i quali si utilizzano spesso metodi iterativi globali, necessari per simulare sistemi non lineari, che rallentano il processo di simulazione a causa dell'enorme complessità computazionale (Falade e, Tihamiyu, 2020). Al contrario, uno dei principali vantaggi della modellazione di sistemi

complessi come sistemi interconnessi è la possibilità di sostituire uno dei sottosistemi con una versione aggiornata senza modificare né la struttura del sistema né gli algoritmi di simulazione. Inoltre, come accennato nella introduzione, le caratteristiche di portabilità di tali modelli consentono la sostituzione di un sottosistema con una black box fornita da una azienda. In questo caso, non è possibile modificare gli algoritmi di risoluzione in quanto le equazioni che modellano il nuovo sottosistema non sono note: il modello black box rende difficile l'utilizzo dei benefici derivanti dalla conoscenza delle equazioni di processo di tutti gli altri sottosistemi.

Il nuovo metodo proposto in questa tesi definisce un ordinamento risolutivo a priori di ogni sottosistema basato solo sulla struttura (grafo) del sistema. Esso, ad esempio, consente di effettuare diverse simulazioni al variare del contenuto di alcuni sottosistemi che non alterino la struttura delle interconnessioni. Inoltre, l'affidabilità del metodo proposto non viene alterata quando si utilizza una black box come sottosistema. Non è necessario iterare tutte le equazioni dell'intero sistema, ma le uniche iterazioni richieste sono quelle implementate in ciascun sottosistema. In questo modo è possibile superare molti degli inconvenienti citati in precedenza.

Il metodo proposto, dunque, migliora e generalizza gli algoritmi numerici per la simulazione di ISMs superando i suddetti inconvenienti. La nostra ipotesi non si basa sulla natura di ogni Sistema Individuale che consente di risolvere ogni sottosistema indipendentemente l'uno dall'altro. Viene realizzato un grafo dell'intero ISM e viene determinato automaticamente un ordinamento ottimale (subottimale) della numerazione dei nodi: esso corrisponde all'ordine ottimo (sub-ottimo) di simulazione di ciascun ISM. In particolare, ad ogni passo temporale, risolto l'i-esimo sottosistema, tutte le uscite calcolate vengono trasferite agli ingressi dei successivi sottosistemi ad esso collegati, mentre gli altri ingressi (non ancora trasferiti, per la presenza di retroazioni) rimangono con i valori calcolati durante il passo temporale precedente. Ciò consente di ottimizzare il numero di nuovi input calcolati. Infine, sono state stabilite le condizioni di stabilità per i sistemi Linear Time Invariant (LTI).

### 3.3. Un quadro generale per i modelli di sottosistemi indipendenti

Un ISM in generale dà origine a un sistema di equazioni algebrico-differenziali implicite (Differential-Algebraic Equations: DAE), per ogni sottosistema, della forma:

$$\Phi_i(t, x_i(t), \dot{x}_i(t), y_i(t), v_i(t), u_i(t), p_i) = 0 \quad 3.1$$

dove  $t$  è il tempo,  $x_i$  sono le variabili differenziali,  $y_i$  sono le variabili algebriche in output dal sottosistema,  $v_i(t)$  è il vettore di input del sottosistema (ovvero input i cui valori sono uguali agli output di altri sottosistemi nell'ISM),  $u_i$  è il vettore degli ingressi esterni ISM (ovvero gli eventuali



ingressi provenienti dall'ambiente esterno),  $p_i$  è il vettore dei parametri delle costanti interne del sottosistema. Nelle DAE alcune delle equazioni potrebbero essere puramente algebriche che portano a maggiori difficoltà se combinate con equazioni differenziali (Mattsson, S.E. e Söderlind G., 1993). L'interconnessione dei sottosistemi si ottiene, come accennato, collegando le uscite del sottosistema agli ingressi di altri sottosistemi. Precisamente, l' $i$ -esimo sottosistema è connesso al  $j$ -esimo sottosistema se c'è una connessione diretta da uno dei componenti scalari  $\tilde{y}_{ir}$  del vettore di output  $y_i = (\tilde{y}_{ir})$  dell' $i$ -esimo sottosistema alla componente scalare  $\tilde{u}_{js}$  del vettore di input  $u_j = (\tilde{u}_{js})$  del  $j$ -esimo sottosistema. Senza perdita di generalità, si possono trascurare gli auto-loop (cioè trascuriamo il caso  $j = i$ ) usando la funzione identità come sottosistema ausiliario intermedio.

Concatenando i vettori  $x_i$  otteniamo un vettore di input globale  $X = [x_i]$  e lo stesso può essere fatto per le altre variabili:  $Y = [y_i], V = [v_i], U = [u_i], P = [p_i]$ , e per l'insieme di tutte le DAE:  $\Phi = [\Phi_i]$ . Combinando tutte le equazioni si ottiene un sistema di equazioni di DAE generalmente implicito:

$$\Phi(t, X, \dot{X}, Y, V, U, P) = 0 \quad 3.2$$

che viene risolto con diversi metodi disponibili, inclusa la manipolazione simbolica.

D'altra parte, senza perdita di generalità, si può descrivere il metodo utilizzando la rappresentazione ODE per una migliore leggibilità e utilizzando una rappresentazione grafica adeguata. Le DAE possono essere facilmente introdotte senza alcun cambiamento nel metodo. Per ogni sottosistema abbiamo quindi:

$$f_i[t, x_i, \dot{x}_i, v_i, u_i, p_i] = 0 \quad 3.3a$$

$$y_i(t) = g_i[t, x_i, \dot{x}_i, v_i] \quad 3.3b$$

In questo modo tutte le variabili algebriche scompaiono e le  $x_i$  sono tutte differenziali. A volte le ODE possono essere scritte nella forma esplicita:

$$\dot{x}_i = f_i(t, x_i(t), v_i(t), u_i(t), p_i) \quad 3.4a$$

Un tale ISM interconnesso formato da tutti i sottosistemi può essere rappresentato come mostrato in Figura 3.1. Usando le notazioni precedenti, tutte le interconnessioni sono modellate come:

$$V = MY \quad 3.4b$$

dove ogni elemento  $M_{hk}$  della matrice di connessione  $M$  è  $M_{hk} = 0$  o  $M_{hk} = 1$ .

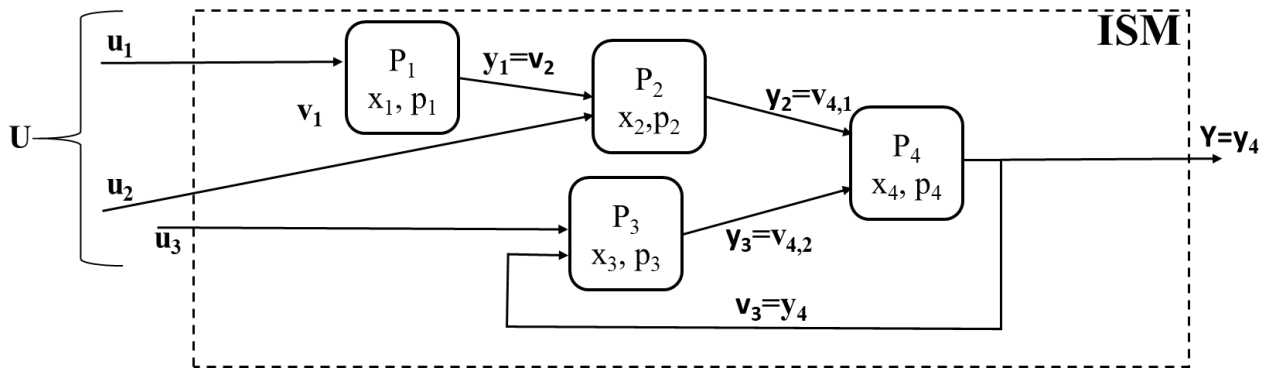


Figura 3.1. Un esempio di schema ISM: il nodo  $P_i$  rappresenta un sottosistema che, ad esempio, modella un industriale

Processo ed è caratterizzato dalle sue variabili di stato interne  $x_i$  e parametri  $p_i$ , ingressi esterni  $u_i$ , ingressi interni  $v_i$ , uscite  $y_i$ .

L'ultima condizione,  $M_{hk} = 1$ , rappresenta una connessione dall'uscita scalare  $Y_k$  all'ingresso scalare interno  $V_k$ . Se non sono consentiti self-loop tutti gli elementi diagonali sono nulli: questa ipotesi può essere accettata in quanto semplifica i calcoli seguenti senza perdere la generalità. Infine, si ipotizza, per semplicità e ancora senza perdita di generalità, che ogni componente scalare di un vettore di output del sottosistema sia connesso a uno e solo un componente scalare di un altro vettore di input interno sottosistema (più connessioni possono essere modellate con più output scalari identici). Con questa condizione, ogni componente di un vettore di output del sottosistema è connesso a un input, il che significa che ogni riga e ogni colonna della matrice  $M$  ha una e una sola componente diversa da zero: la matrice  $M$  è una matrice di permutazione.

Come per qualsiasi altro metodo, la simulazione numerica di sistemi ISM continui richiede innanzitutto l'approssimazione di ciascun sottosistema mediante un modello discreto. L'evoluzione dinamica di un ISM è calcolata in una sequenza di  $n$  punti temporali discreti  $t_n = \sum_{k=0}^n \Delta t_k$ . Il passo temporale  $\Delta t_k$  può essere costante o variabile a seconda del tipo di metodo numerico utilizzato, ad esempio a causa di possibili mal condizionamenti del sistema che richiede la variazione di  $\Delta t_k$  durante la simulazione.

D'altra parte, risolvere sistemi di DAE completamente impliciti richiede un'elevata complessità computazionale (Ascher, U.M. e Petzold, L.R., 1988). I metodi principali utilizzati per ridurre tale costo [Cellier e Kofman, 2006] sono:

- *causalization*;
- *tearing*

generalmente eseguiti mediante elaborazione simbolica. Questi metodi essenzialmente riducono quanto più possibile le variabili algebriche che il risolutore numerico deve calcolare. Le variabili

restanti costituiscono un insieme minimo di variabili: le variabili di stato. Lo stato, dunque, è un insieme minimo di variabili (interne di un ISM) in modo tale che se i loro valori correnti sono noti e tutti i futuri valori di input ISM esterni sono noti, il comportamento futuro dell'ISM è determinato in modo univoco. Ovviamente, la stessa definizione di stato è valida anche per ogni sottosistema preso singolarmente. Se nel sistema di equazioni ci sono anche ODE (vedi 3.4a-b) allora il vettore delle variabili differenziali  $x_i$  che appare in ogni ODE rappresenta naturalmente un insieme adatto di variabili di stato per ciascuna ODE. D'altra parte, per la maggior parte delle DAE un sottoinsieme di variabili sia differenziali che algebriche nella formulazione 3.1 potrebbe ancora essere utilizzato come insieme delle variabili di stato (Ascher, U.M. e Petzold, L.R., 1988).

La prima procedura utilizza una manipolazione simbolica per ricavare l'insieme delle variabili di stato e convertire un modello DAE come quello dell'Eq. (3.1) in un modello numerico iterativo dal quale vengono calcolate tutte le variabili differenziali ed algebriche in un proprio ordine sequenziale. È divisa in due fasi: Matching and Block-Lower Triangular (BLT) ordering. Il metodo più utilizzato è l'algoritmo di Tarjan [Tarjan, 1972], ancora considerato uno dei più efficienti. D'altra parte, uno degli svantaggi più significativi è proprio la manipolazione simbolica: il costo computazionale si sposta dalla risoluzione numerica al calcolo simbolico.

La seconda procedura, tearing (Baharev et al., 2016a; Elmqvist and Otter, 1994; Duff et al., 1986; Kron, 1963), è un metodo per risolvere sistemi sparsi di DAE senza alcuna struttura particolare (ad es. diagonale) ed è meno costoso dal punto di vista computazionale della causalization. Le variabili e le equazioni vengono riordinate per dividere il sistema in due partizioni, la prima altamente strutturata che può essere risolta efficacemente da sola: la soluzione ottenuta può essere utilizzata per risolvere la seconda parte non strutturata.

Mentre l'ordine di variabili ed equazioni ottenuto mediante la BLT garantisce la stabilità numerica, il Tearing no. Questo è un noto inconveniente, difficile da risolvere quando non sono disponibili informazioni precise ed affidabili sulla struttura del sistema o, se note, non possono per qualche ragione essere utilizzate. Tuttavia, quando si ha a che fare con sistemi nei quali ogni sottosistema può essere risolto individualmente calcolando gli output dagli input e dai parametri (implicitamente o esplicitamente non importa), potrebbero essere utilizzati algoritmi numerici che garantiscano la stabilità numerica (Westerberg e Edie, 1971; Gupta et al., 1974).

Dalla discussione di cui sopra deriva che ci sono molti metodi disponibili per modellare sistemi anche di grandi dimensioni, ma nessun approccio globale e completo è stato sviluppato dopo il rapporto di Di Perna. Inoltre, come detto alcuni strumenti risolvono il sistema di DAE nel suo complesso, altri strumenti hanno un approccio diverso per determinare preliminarmente un ordinamento nell'esecuzione dei sottosistemi. Tuttavia, si basano principalmente sulla natura di ciascun sottosistema piuttosto che sulla struttura del grafo che rappresenta l'ISM visto come una

raccolta di sottosistemi interconnessi. Il metodo proposto segue il paradigma dell'approccio della co-simulazione, ma con una nuova prospettiva, inclusi i metodi della teoria dei grafi per risolvere in modo efficiente l'ISM generale.

### 3.4. Un nuovo approccio per gli ISM

Come brevemente dedotto nella sezione precedente, quando si ha a che fare con un ISM, la struttura del problema può essere definita direttamente dal suo grafo rappresentante l'insieme delle connessioni. l' $i$ -esimo sottosistema,  $i = 1 \dots N$ , può essere descritto da un sottoinsieme di DAE come in 3.1, qui ripetuto per semplicità:

$$\Phi_i(t, x_i(t), \dot{x}_i(t), y_i(t), v_i(t), u_i(t), p_i) = 0 \quad 3.1$$

Tutti i sottosistemi possono essere interconnessi tra loro collegando le loro uscite agli ingressi corrispondenti di altri sottosistemi. Questa struttura è ben rappresentata da un grafo  $G(V, E)$ , dove  $V$  è l'insieme degli  $N$  vertici (denominati anche nodi), ed  $E \subseteq V \times V$  è l'insieme degli archi (archi orientati). Ogni nodo  $N_i$  ( $N_i \in V, i = 1 \dots N$ ) corrisponde a un singolo sottosistema. Un arco orientato  $(N_i, N_j) \in E$  esiste se un'uscita scalare del sottosistema  $N_i$  è collegata a un ingresso scalare del sottosistema  $N_j$ . Come detto, in questo studio, senza perdita di generalità, non si considerano i self-loop; cioè si presume che ogni sottosistema  $N_i$  non sia connesso a sé stesso, cioè  $(N_i, N_i) \notin E$ . Si presume che ogni sottosistema  $N_i$  ammetta una rappresentazione DAE come in 3.1. La matrice di incidenza del grafo può essere considerata la matrice di incidenza del sistema di DAE. Il sottoinsieme di DAE di ogni sottosistema può essere risolto utilizzando uno qualsiasi dei metodi descritti in precedenza applicati direttamente nella procedura che definisce il sottosistema. Questo metodo consente di unire metodi diversi (ad esempio, metodi causali o acausali, metodi numerici generali, ecc.) a seconda del contesto.

In questo modo, si può descrivere ogni sottosistema con una sua rappresentazione indipendentemente da ogni altro sottosistema nell'ISM: il grande vantaggio è che lo si può sostituire ogni volta che è necessario con una rappresentazione diversa (cioè, DAE diversi per quel sottosistema e diversi metodi di soluzione), rendendo la manutenzione ed evoluzione dell'intero ISM di facile implementazione. Un altro vantaggio è che non abbiamo bisogno di conoscere il contenuto di ogni sottosistema (è sufficiente avere una black box con input e output noti: questo diventa particolarmente utile quando chiediamo a un'azienda esterna di modellare un particolare componente senza necessariamente avere accesso al suo contenuto completo. Questa rappresentazione appartiene al framework ISM in un modo più generale di quanto fatto in precedenza poiché ogni sottosistema è modellato e risolto individualmente e indipendentemente

dagli altri modelli. La struttura del grafo viene utilizzata per coordinare/ordinare tutti questi solutori per determinare in modo efficiente l'intera evoluzione dinamica dell'ISM. Pertanto, una delle principali novità del mio approccio è che ogni sottosistema (o un sottogruppo di essi come mostrato di seguito) potrebbe essere risolto separatamente in un giusto ordine determinato sulla base della struttura del grafo. Precisamente, ogni output di un sottosistema (o gruppo di essi, Xu et al., 2020) diventa l'input del sottosistema successivo nell'ordine scelto. La matrice  $M$  è una matrice a blocchi in cui ciascun blocco corrisponde alla connessione da un sottosistema verso un altro sottosistema e può essere utilizzata per descrivere matematicamente quanto appena detto: la matrice  $M$  è, infatti, la somma di una matrice strettamente triangolare inferiore a blocchi  $M'$ , e di una matrice strettamente triangolare superiore a blocchi  $M''$ . Come ipotizzato in precedenza, gli elementi diagonali (blocchi) di  $M$  (quindi sia di  $M'$  che di  $M''$ ) sono nulli in quanto, senza perdita di generalità, i self-loop non sono qui considerati.  $M'$  rappresenta tutte le connessioni di tipo feed-forward nell'ordinamento dei nodi considerato, mentre  $M''$  rappresenta tutte le connessioni di tipo feedback. Questo approccio porta a uno schema a metà strada tra quelli puri espliciti e quelli puri impliciti: utilizzando al passaggio temporale  $t_{n+1} \rightarrow t_{n+1}$  solo valori noti delle variabili di uscita dei sottosistemi, cioè  $y_{n+1}$  per archi corrispondenti a connessioni feed-forward (matrice  $M'$ ) e  $y_n$  per gli archi di retroazione (matrice  $M''$ ), la versione discretizzata dell'equazione di matrice 3.1b diventa:

$$v_{n+1} = M'y_{n+1} + M''y_n \quad 3.5$$

Una situazione ottimale, ma ovviamente in generale poco interessante, si verifica quando è possibile suddividere i sottosistemi (ovvero i nodi del grafo che rappresenta l'ISM) in gruppi (Gerbnera et al., 2016), con ciascun gruppo contenente un numero minimo di sottosistemi, in modo che esistano solo archi feed-forward (da ogni gruppo ai gruppi che seguono. Con questo partizionamento, le equazioni di ogni gruppo possono essere facilmente risolte simultaneamente ed il risultato viene utilizzato per risolvere le equazioni dei gruppi successivi nell'ordine selezionato. Sarebbe anche auspicabile avere un numero minimo di sottosistemi in ogni gruppo perché questo consentirebbe di ridurre al minimo il numero di equazioni che devono essere risolte simultaneamente. Per ISM complessi tale partizionamento non è eseguibile se non attraverso algoritmi specifici (Bloem et al., 2006; Xie e Beere, 2000). Non è richiesta alcuna manipolazione. Tuttavia, è molto raro trovarsi in una tale situazione ideale: normalmente sono presenti diversi archi di feedback che possono essere risolti con il partizionamento della matrice di connessione  $M$  come indicato nella Eq. 3.5.

Il processo di partizionamento è dunque solo il primo passo verso la soluzione. Se gli archi di feedback non vengono del tutto cancellati, le variabili corrispondenti sono tali che è possibile eseguire comunque tutti i sottosistemi del gruppo nell'ordine selezionato utilizzando per tali

variabili i valori calcolati con la 3.5, cioè da passi temporali precedenti. Se un gruppo contiene un sottosistema non lineare, una soluzione esplicita non è in generale possibile e si utilizzano metodi iterativi separatamente per ciascun sottosistema non lineare. Risolvere l'ISM senza sfruttare la struttura delle equazioni di tali sistemi interconnessi risulta altamente inefficiente dal punto di vista computazionale. D'altra parte, per ridurre la complessità computazionale, dovrebbe essere scelto il più piccolo insieme possibile di variabili per ogni gruppo partizionato. È stato dimostrato (Charbit et al., 2007) che la scelta di un numero minimo di tali variabili è equivalente a un problema comune nella teoria dei grafi orientati, la determinazione del set minimo di archi di feedback. Le seguenti definizioni (Younger 1963), sono utili per tale identificazione.

- Per un grafo orientato, un set di archi di feedback è un insieme di archi che, se rimossi, portano alla riduzione del numero di circuiti nel grafo risultante.
- Un set di archi di feedback è minimo se nessun altro set di archi di feedback per quel grafo è costituito da un numero inferiore di archi.
- Un ordinamento sequenziale  $R$  di un grafo di  $N$  nodi è una permutazione della sequenza  $1 \dots N$

In generale, una esecuzione puramente sequenziale dei singoli sottosistemi (o gruppi) è possibile solo se il grafo è privo di loop diretti, cioè non vi sono archi di feedback. La procedura di partizionamento ed ordinamento dei nodi del grafo avviene in due fasi. Inizialmente è necessario suddividere i nodi del grafo in nodi *sorgente* (nodi caratterizzati da soli archi uscenti: sono gli input esterno all'intero sistema) e nodi *pozzo* (nodi caratterizzati da soli archi entranti: sono gli output verso l'esterno all'intero sistema e, generalmente, coincidono con alcune delle variabili misurate). Gli archi che lasciano i nodi sorgente (input esterni  $U$ ), non possono mai essere variabili di iterazione in quanto rappresentano grandezze note. Tali nodi sorgente costituiscono il primo gruppo della partizione, mentre i nodi *pozzo* costituiscono l'ultimo gruppo della partizione.

Successivamente, la procedura di partizionamento viene quindi applicata al sottografo, chiamato ISM Ridotto (Reduced Individual Subsystem Model: RISM), ottenuto eliminando i nodi sorgente e pozzo dal grafo originale. Il secondo passo, infatti, consiste nel trovare un set minimo di archi di feedback che determina un insieme di variabili di iterazione, una per ogni arco di feedback: tale procedura di ordinamento dei nodi descritta di seguito viene, pertanto, applicata direttamente al RISM.

Per quanto riguarda l'ordinamento dei nodi, Younger mostra che per ogni ordinamento sequenziale  $R$  dei nodi del grafo, la rimozione di quegli archi  $(i, j)$  per i quali  $R(i) \geq R(j)$  cancella tutti i circuiti orientati. Le variabili corrispondenti a tutti questi archi diventano variabili di iterazione in quanto non sono note al passo temporale  $n + 1$  e quindi devono essere iterate dal passo precedente  $n$  (è la componente  $M^n y_n$  di  $v_{n+1}$ ), mentre tutte le altre sono state calcolate con l'esecuzione di altri

sottosistemi è la componente  $M'y_{n+1}$  di  $v_{n+1}$ ): l'ordine scelto determina l'insieme di variabili di iterazione. Per trovare un numero minimo di tali variabili, si desidera quindi trovare un ordinamento sequenziale  $R$  che minimizzi il numero di archi  $(i, j)$  per i quali  $R(i) \geq R(j)$ . Questo è esattamente il problema di determinare un set minimo di archi di feedback. Dettagli su possibili procedure possono essere trovati in Younger (1963), ma anche altri e più efficienti algoritmi possono essere trovati in letteratura (es Rajasingh et al., 2011) dando un ordinamento a volte ottimale ma spesso subottimale: infatti, il problema generale è NP-hard (Charbit et al., 2007). Molti algoritmi di ordinamento subottimali hanno il grande vantaggio di un'esecuzione rapida, in alcuni casi con complessità di calcolo direttamente proporzionale a  $N$ . Il metodo proposto viene analizzato in dettaglio per i sistemi LTI nel paragrafo successivo.

### 3.5. Il nuovo approccio applicato ai sistemi Linear Time Invariant: rappresentazione nello spazio degli stati e condizioni di stabilità

Il metodo di partizionamento delineato nel paragrafo precedente è ora applicato ai sistemi LTI, le equazioni 3.4a-b possono essere scritte esplicitamente. Per distinguere i vettori dalle matrici, usiamo lettere maiuscole per le seconde e lettere minuscole per le prime (es.  $x$  per il vettore di stato invece del simbolo  $X$  usato nel paragrafo precedente):

$$\dot{x} = Ax + Bv + Pu \quad 3.6a$$

$$y = Cx \quad 3.6b$$

$$v = My \quad 3.6c$$

Le matrici  $A, B, C$  sono diagonali a blocchi dove ogni blocco rappresenta ogni sottosistema nell'ISM: possono dipendere anche dal tempo e in questo caso il sistema non è autonomo, rendendo solo più complicata la discussione. La matrice  $P$  rappresenta la dipendenza del vettore di stato  $x$  dal vettore di input esterni  $u$  (sorgenti). Per quanto riguarda le uscite esterne (pozzi), possono essere scelte, senza perdita di generalità, tra le componenti del vettore di uscite interne  $y$ .

Sostituendo le ultime due equazioni 3.6b e 3.6c nella prima 3.6a si ottiene il seguente sistema di equazioni che può essere risolto calcolando  $x(t)$  come sono note tutte le matrici e gli input esterni  $u$ :

$$\dot{x} = Kx + Pu \quad 3.7$$

dove  $K = A + BMC$  corrisponde alla matrice di transizione di stato (precisamente per i sistemi LTI la matrice di transizione di stato è  $e^{Kt}$ ). Il sistema 3.7 è detto asintoticamente stabile (AS) se per ogni  $\varepsilon > 0$  esiste  $\delta(\varepsilon)$  tale che per  $u = 0$  e una norma adeguata  $\|\cdot\|$  si ha:

$$\|x(0)\| < \delta \Rightarrow \begin{cases} \|x(t)\| < \varepsilon \forall t > 0 \\ \lim_{t \rightarrow \infty} \|x(t)\| = 0 \end{cases} \quad 3.8$$

Il sistema è AS se e solo se le parti reali degli autovalori  $\lambda(K)$  della matrice  $\lambda(K)$  sono tutte negative:  $Re[\lambda(K)] < 0$ .

Se il passo temporale  $\Delta t$  è costante per cui si ha  $t_n = n\Delta t$ , la versione discreta del sistema 3.7 può essere ottenuta approssimando  $\dot{x}$  come segue:

$$\dot{x} \sim \frac{x_{n+1} - x_n}{\Delta t} \quad 3.9$$

dove  $x_n = x(t_n)$ . Senza perdita di generalità, possono essere usate altre approssimazioni per migliorare la convergenza specialmente quando si tratta di problemi mal condizionati: inoltre, in tali casi può essere utilizzato anche un passo temporale variabile.

Esistono diversi modi di trattare il sistema 3.7: esplicitamente o implicitamente. Utilizzando lo schema di Eulero, nel primo caso calcoliamo il lato destro della 3.7 all'istante  $t_n$  ottenendo:

$$x_{n+1} = (I + \Delta t K)x_n + \Delta t P u_n \quad 3.10$$

Questo schema è stabile se il più grande valore assoluto  $\rho(I + \Delta t K)$ , il raggio spettrale, degli autovalori della matrice  $I + \Delta t K$  è minore di 1 cioè, per i sistemi AS, quando:

$$\Delta t < \frac{2}{\rho(K)} \quad 3.11$$

Nel secondo caso (schema completamente implicito) il lato destro del sistema 3.7 viene calcolato all'istante  $t_{n+1}$  ottenendo:

$$x_{n+1} = (I - \Delta t K)^{-1} x_n + (I - \Delta t K)^{-1} \Delta t P u_n \quad 3.12$$

Definendo  $H^{-1} = (I - \Delta t K)^{-1}$  la matrice di transizione temporale discreta a un passo, la condizione di stabilità diventa  $\rho[H^{-1}] < 1$  ed è vera per qualsiasi matrice  $K$  avente  $Re[\lambda(K)] < 0$ , quindi lo schema è incondizionatamente stabile per i sistemi AS.



Gli schemi di cui sopra non considerano direttamente la struttura del sistema e generalmente vengono utilizzate procedure differenti per ridurre lo sforzo computazionale come descritto nei paragrafi precedenti.

Come già detto all'inizio di questo capitolo, la matrice  $M$  è la somma di una matrice triangolare inferiore a blocchi  $M'$  e di una matrice triangolare superiore a blocchi  $M''$ .  $M'$  rappresenta tutte le connessioni feed-forward nell'ordinamento dei nodi considerato, mentre  $M''$  rappresenta tutte le connessioni di feedback: ovviamente siamo nell'ipotesi di aver già effettuato le operazioni di ordinamento ottimale – subottimale – dei nodi ed il risultato è rappresentato dal sistema 3.6a-b-c.

Quando si ottiene l'ordinamento ottimale (o subottimale),  $M''$  è massimamente sparsa (o abbastanza vicina a tale condizione). L'equazione di matrice 3.6c viene suddivisa nelle due parti relative agli archi di feedback e a quelli restanti:

$$v_{n+1} = M'y_{n+1} + M''y_n \quad 3.13$$

L'equazione 3.6b può essere scritta all'istante  $t_n$  usando  $y_n = Cx_n$  e all'istante  $t_{n+1}$  usando  $y_{n+1} = Cx_{n+1}$ . Sostituendo queste due condizioni nella (3.13) diventa:

$$v_{n+1} = M'Cx_{n+1} + M''Cx_n \quad 3.14$$

Infine, si utilizza la versione discreta dell'equazione 3.6a (Schema di Eulero implicito per quanto riguarda i termini a destra):

$$\frac{x_{n+1} - x_n}{\Delta t} = Ax_{n+1} + Bv_{n+1} + Pu_{n+1} \quad 3.15$$

che diventa:

$$\frac{x_{n+1} - x_n}{\Delta t} = Ax_{n+1} + BMM'Cx_{n+1} + BM''Cx_n + Pu_{n+1} \quad 3.16$$

L'equazione 3.6a-b-c può essere risolta esplicitamente per ottenere:

$$x_{n+1} = H_1^{-1}x_n + (I - \Delta tK')^{-1}\Delta tPu_{n+1} \quad 3.17$$

con  $H_1 = (I + \Delta tBM''C)^{-1}(I - \Delta tK')$  e  $K' = A + BM'C$ .

Lo schema è stabile se sono soddisfatte due condizioni, una per ogni termine in 3.17:  $H_1^{-1}$ , la matrice di transizione di stato ad un passo e su  $(I - \Delta tK')^{-1}\Delta tP$ , la matrice di transizione ingresso - stato):

- $[H_1^{-1}] < 1$ .
- $K'$  deve avere autovalori con parte reale negativa: questo significa che ogni sottosistema deve essere stabile, essendo  $K'$  la matrice di transizione di stato dell'ISM con archi di feedback cancellati: se uno qualsiasi dei sottosistemi è instabile e viene stabilizzato riportando in retroazione un'uscita (metodo di stabilizzazione mediante feedback), eliminando il feedback corrispondente il sistema risultante torna ad essere nuovamente instabile.

Per quanto riguarda la prima condizione, la matrice  $BM''C$  è il prodotto di matrici diagonali blocchi con  $M''$ , una matrice a blocchi strettamente triangolare superiore, dando come risultato ancora matrice a blocchi strettamente triangolare superiore: tale matrice è nilpotente, cioè  $(BM''C)^r = 0$  con  $r$  il suo rango. Questa proprietà permette di calcolare facilmente la matrice inversa di  $I + \Delta tBM''C$ :

$$(I + \Delta tBM''C)^{-1} = I + \sum_{i=1}^{r-1} (-1)^i (\Delta tBM''C)^i \quad 3.18$$

Sostituendo la 3.18 nella definizione di  $H_1$  si ottiene:

$$\begin{aligned} H_1 &= (I + \Delta tBM''C)^{-1}(I - \Delta tK') = \\ &= I - \Delta tK' - \Delta tBM''C - \Delta t^2BM''CK' + \left[ \sum_{i=2}^{n-1} (-1)^i (\Delta tBM''C)^i \right] (I - \Delta tK') = \\ &= I - \Delta t(A + BM'C + BM''C) - \Delta t^2BM''CK' + \left[ \sum_{i=2}^{n-1} (-1)^i (\Delta tBM''C)^i \right] (I - \Delta tK') = \\ &= I - \Delta tK + O(\Delta t^2) \end{aligned} \quad 3.19$$

ed infine:

$$H_1 = H + O(\Delta t^2) \quad 3.20$$

Pertanto, gli autovalori della matrice perturbata  $H_1$  sono una versione perturbata di quelli di  $H$  (Eulero implicito, eq. 3.10) che restano maggiori di uno in modulo (quindi minore di uno il modulo degli autovalori di  $H_1^{-1}$ ) se la perturbazione del secondo ordine è sufficientemente piccola. In ogni caso, il metodo qui proposto è solo del primo ordine, ma possono essere utilizzate ovviamente

approssimazioni di ordine superiore della 3.9 per la derivata temporale  $\dot{x}$  dello stato. Inoltre, la dimensione del passo  $\Delta t$  dovrebbe essere comunque abbastanza piccola mantenere l'errore globale al di sotto di una soglia prefissata indipendentemente dalla condizione di stabilità: ciò può diventare necessario per i sistemi mal condizionati (Nwachukwu e Mokwunyei, 2018).

### **3.6. Il Framework di simulazione**

Sulla base di quanto precedentemente illustrato si è proceduto alla realizzazione del Framework di simulazione, completo di interfaccia grafica e procedure di esecuzione della simulazione, tale da consentire l'ottimizzazione e il controllo di un generico impianto industriale in una prospettiva IoT. Come detto un impianto industriale è oggi un sistema complesso in quanto spesso non solo è composto da numerose tipologie di macchine ma è anche caratterizzato da una forte variazione temporale della produzione. A tal fine sono state progettate e sviluppate le seguenti sezioni del Layer Grafico del framework di simulazione:

- La sezione “libreria”, che contiene gli elementi (blocchi) da inserire nello schema di processo, completo di call a functions matlab che ne descrivano il comportamento fisico-matematico, che vengono utilizzate e richiamate dall'algoritmo di simulazione.
- La sezione “creazione e gestione dello schema di processo”, che costituisce, dunque, il fulcro del framework realizzato. L'interfaccia grafica consente di creare nuove configurazioni o gestire configurazioni già progettate e validate: in particolare le configurazioni saranno composte dagli elementi presenti in libreria o da nuovi elementi da creare e conservare in libreria. Gli elementi presenti in libreria sono/saranno prevalentemente blocchi rappresentativi di componenti fisici (macchinari) e connettori logici che, opportunamente utilizzati, permettono di collegare i diversi blocchi generando così un grafo orientato. Ciascun lato del grafo corrisponderà ad un flusso di informazioni/materia/energia/ecc. da un blocco all'altro e conterrà le variabili di output della function del blocco di partenza i cui valori saranno utilizzati come input per il blocco di arrivo. Sono stati implementati gli opportuni controlli di integrità delle informazioni.
- La sezione di “simulazione”, che permette di simulare il funzionamento della configurazione così progettata. L'algoritmo di simulazione richiama le function dei singoli blocchi costituenti il sistema in un ordine che può essere casuale, definito dal progettista in base alla sua esperienza o gestito dal sistema attraverso l'eventuale ricerca dell'ordine “ottimo” di richiamo delle singole funzioni (blocchi o nodi del grafo che rappresenta l'impianto), che consenta di ridurre, cioè, il numero di operazioni implicite nel metodo di risoluzione (semi-implicito). Poiché, tuttavia, la ricerca dell'ordine ottimo assoluto è un problema di elevata complessità (NP), pertanto si è optato per sistemi di tipo euristico (che consente di trovare

una soluzione caratterizzata da un adeguato grado di approssimazione) che portano a soluzioni cosiddette sub-ottime.

Ogni sezione può essere parzialmente modificata dall'utente del software al fine di consentire la personalizzazione dei progetti da creare e gestire, nonché la ricerca di una configurazione "ottimale" in base a criteri che il progettista definirà di volta in volta.

Nel software verrà definito “Nuovo progetto” una nuova configurazione di elementi che insieme compongono una IoT. Gli elementi da utilizzare per comporre la nuova IoT sono presenti nella libreria del software. Ogni elemento costituirà un singolo blocco della IoT che pertanto sarà composta da un insieme di blocchi collegati tra loro in modo opportuno da connettori di tipo logico anch'essi presenti in libreria.

In seguito a tali osservazioni vengono mostrati gli strumenti sviluppati e implementati nella sezione di simulazione utilizzato per modellare un processo industriale fisico mediante un'interfaccia grafica intuitiva che consente di definire in modo interattivo diverse configurazioni dell'impianto, attraverso la selezione di elementi grafici di base e la loro connessione appropriata.

Tali elementi del Framework sono stati pensati tenendo conto che negli impianti industriali gli elementi essenziali che costituiscono i diversi processi possono essere individuati e schematizzati in termini di schemi a blocchi. Ogni blocco rappresenta in modo pittorico la relazione causa-effetto tra le variabili di input e di output. Introducendo nel blocco il modello matematico dell'elemento si ottiene ciò che abbiamo precedentemente definito un sottosistema (ISM) consentendo così la valutazione quantitativa dell'evoluzione del sistema nel tempo la cui analisi teorica è stata sviluppata nel paragrafo precedente.

Questo percorso ha portato alla definizione di due classi di elementi base:

- **COMPONENTE PROCESSO**
  - ID (attributo)
  - Nome (attributo)
  - Porta (classe):
    - Numero (es. porta numero 1)
    - $N_{MAX}$  linee di flusso elementari trattabili
    - Verso
      - Input
      - Output
      - I/O
    - Dati (stessa struttura dei Dati delle linee di flusso che arrivano)
  - Data processor:
    - Set di equazioni

- Set di vincoli
- Set di parametri
- LINEA DI FLUSSO ELEMENTARE:
  - ID (attributo)
  - Nome (attributo)
  - Nodo Input (attributo)
  - Nodo Output (attributo)
  - Verso (attributo)
  - Direzione (attributo)
  - Dati (attributo strutturato con struttura modificabile)

Nella sottoclasse “Data processor” del “Componente Processo”, gli strumenti implementati consentono di modellare un processo, una macchina, un intero impianto industriale, fino ad arrivare a diverse configurazioni di impianto. Gli strumenti implementati sono:

- modelli matematici per la simulazione di singoli componenti coinvolti in diversi processi tecnologici, ad esempio di tipo energetico.
- modelli matematici di sistema che consentano la simulazione di diverse soluzioni impiantistiche, secondo criteri di gestione intelligente, utilizzando i modelli sviluppati al punto precedente; in particolare sono stati affrontati e ripresi alcuni aspetti relativi alla modellazione di impianti utilizzati per la trasformazione e la conservazione di alimenti e per la conversione energetica.
- algoritmi per la simulazione di una ampia casistica di sistemi e soluzioni impiantistiche, anche in ottica IoT, permettendo, quindi, un utilizzo più trasversale e generale del simulatore stesso; in particolare è stato utilizzato un metodo di tipo semi-implicito per la risoluzione delle equazioni differenziali, sia miste algebriche che ordinarie, coinvolte nei modelli matematici delle singole macchine.
- algoritmi di ottimizzazione che consentano di selezionare, mediante analisi multi-obiettivo, le soluzioni impiantistiche ottimali secondo criteri e parametri di tipo energetico, economico ed ambientale. Ciò consente da un lato di determinare quale tecnica di ottimizzazione meglio si presta allo scopo della ricerca, dall’altro di determinare una o più soluzioni impiantistiche “ottime” secondo i criteri previsti. Sono stati scelti ed implementati opportuni algoritmi di ottimizzazione che consentano di selezionare, mediante analisi multi-obiettivo, le soluzioni impiantistiche ottimali secondo criteri e parametri predefiniti dal progettista. In particolare si è rivolta l’attenzione principalmente al metodo della frontiera di Pareto

utilizzando coppie di obiettivi contrastanti e intersecando tra loro gli insiemi di soluzioni ottenute per ricercare la/le soluzioni ottimali o sub-ottimali. Ciò, infatti, consente da un lato di determinare quale tecnica di ottimizzazione meglio si adatta alle diverse attività sia di ricerca che di trasferimento tecnologico, dall'altro di determinare una o più soluzioni impiantistiche "ottime" secondo criteri anche tra loro contrastanti.

## 4. RISULTATI E DISCUSSIONE

Verrà ora presentato e commentato l'aspetto visuale del Framework di simulazione ed in particolare sia il Node Editor che il modulo di realtà virtuale (Run Time VR). Di seguito vengono presentati e discussi alcuni casi studio numerici che rappresentano modelli di sistemi lineari sia stabili che instabili ma opportunamente stabilizzati. Sulla base dei primi risultati viene effettuata una analisi di performance del metodo proposto. Infine, viene presentato un caso studio reale nel quale viene rappresentato e modellato un impianto di refrigerazione/riscaldamento a servizio di processi industriali modellati a loro volta come sistemi a fluido intermedio.

### 4.1. Node Editor

Lo sviluppo del node editor è stato effettuato in Unity, un motore grafico multiplatforma utilizzato per creare realtà tridimensionale, bidimensionale, virtuale e aumentata, nonché simulazioni e si è avvalso di un framework già esistente: [https://github.com/Seneral/Node\\_Editor\\_Framework](https://github.com/Seneral/Node_Editor_Framework) a cui sono state apportate modifiche per renderlo compatibile con le specifiche sopra indicate.

In particolare, è stato introdotto il nodo CustomFunctionNode, tramite le opportune modifiche della classe Node già creata nel framework di partenza riguardante il parametro AutoLayout impostato a True in modo da avere un nodo che si adatti verticalmente al contenuto. La classe Node consente di ereditare due funzioni come segue.

- NodeGUI: contiene tutta la parte visiva e di rendering del nodo con tutte le istruzioni che permettono di visualizzare i parametri del singolo nodo.
- Calcola: che contiene la parte logica del calcolo della funzione nodo (che in questo caso non verrà utilizzata in quanto delegata ad appositi script Matlab).

Successivamente è stata creata una nuova classe denominata Parameter che permette di avere attributi personalizzati per ogni parametro. Infine, sempre all'interno della stessa classe, sono state aggiunte le seguenti due nuove variabili.

- Parametri: un elenco di parametri con attributi correlati
- FilePath: una stringa che contiene il percorso dello script Matlab contenente le relazioni Input-Output (modello) relative al nodo.

Al termine si otterrà un'immagine come quella mostrata nelle Figure 3.2 e 3.3.

Per quanto riguarda la struttura di un nuovo progetto, questo si trova all'interno di un'apposita cartella in cui troveremo una cartella chiamata "Templates" al cui interno ci saranno due file (se non presenti verranno generati in automatico) chiamati Node.tmp e Graph.tmp che se aperti con un editor di testo ci permetteranno di modificarli:

Il template per il singolo nodo contiene i placeholder di ogni elemento sono contrassegnati dal simbolo | prima e dopo il nome del placeholder stesso:

- |outputs| sono i parametri di output del nodo
- |FunctionName| corrisponde al nome del nodo
- |inputs| corrisponde ai parametri di input del nodo
- |VarAssignment| corrisponde alla lista di assegnazioni per quanto riguarda gli input e i workparameters

Nel template dello script generale come nel template di nodo sono presenti i seguenti placeholder:

- |GraphName| corrisponde al nome del progetto
- |Functions| è la lista delle funzioni scritte nei vari nodi

Lo script Matlab viene preparato automaticamente con tutte le informazioni relative a input, output, parametri di lavoro prelevati dal Template di nodo, lasciando all'utente la possibilità di scrivere il codice del proprio modello subito sotto l'ultimo commento. Nella prima parte del codice viene ricavato (o generato) il percorso per lo script MatLab generale. Nella seconda sezione di codice vengono mostrati i parametri e le variabili aggiunte nel nodo. La prima metà crea i punti di connessione tra un nodo e l'altro con le relative variabili, mentre la seconda metà si occupa di mostrare i nomi dei parametri.

La terza parte si occupa della parte di creazione di un nuovo parametro o variabile con selezione del tipo di parametro e aree di input per introdurre i valori richiesti e relativa unità di misura partendo da un valore di default. Infine, è presente un comando che apre o crea (se non esiste) il file relativo allo script Matlab per ciascun nodo. Il risultato è mostrato nelle Figure 4.1 e 4.2.

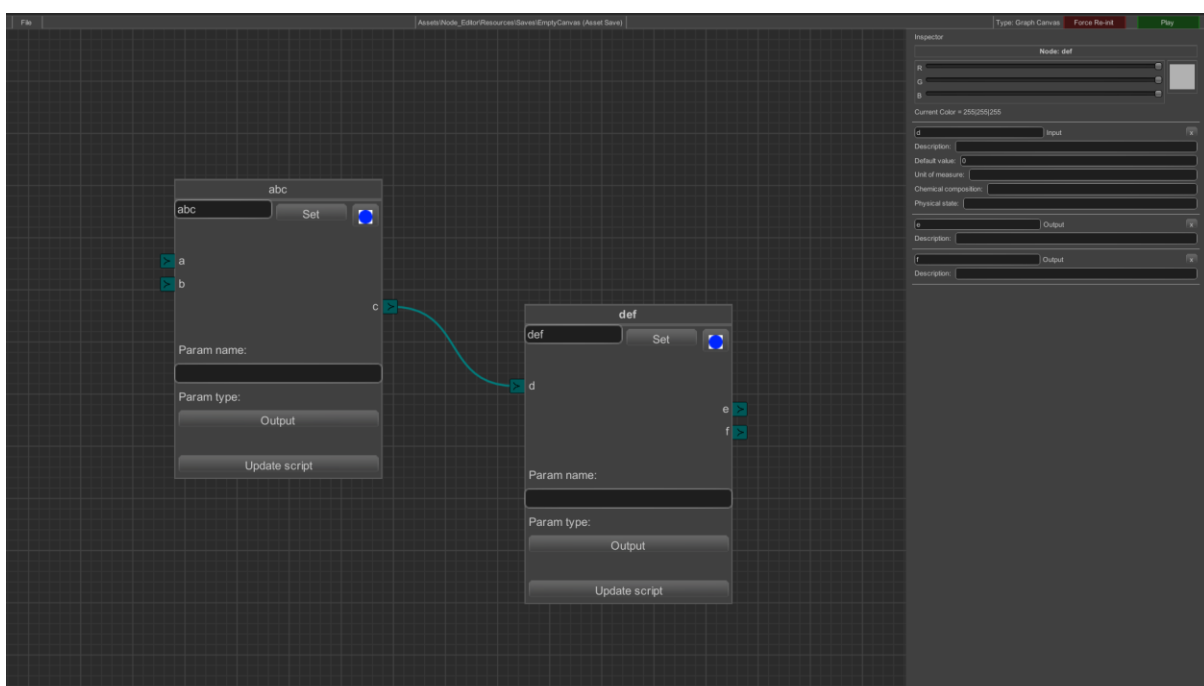




Figura 4.1: Connessione tra due nodi

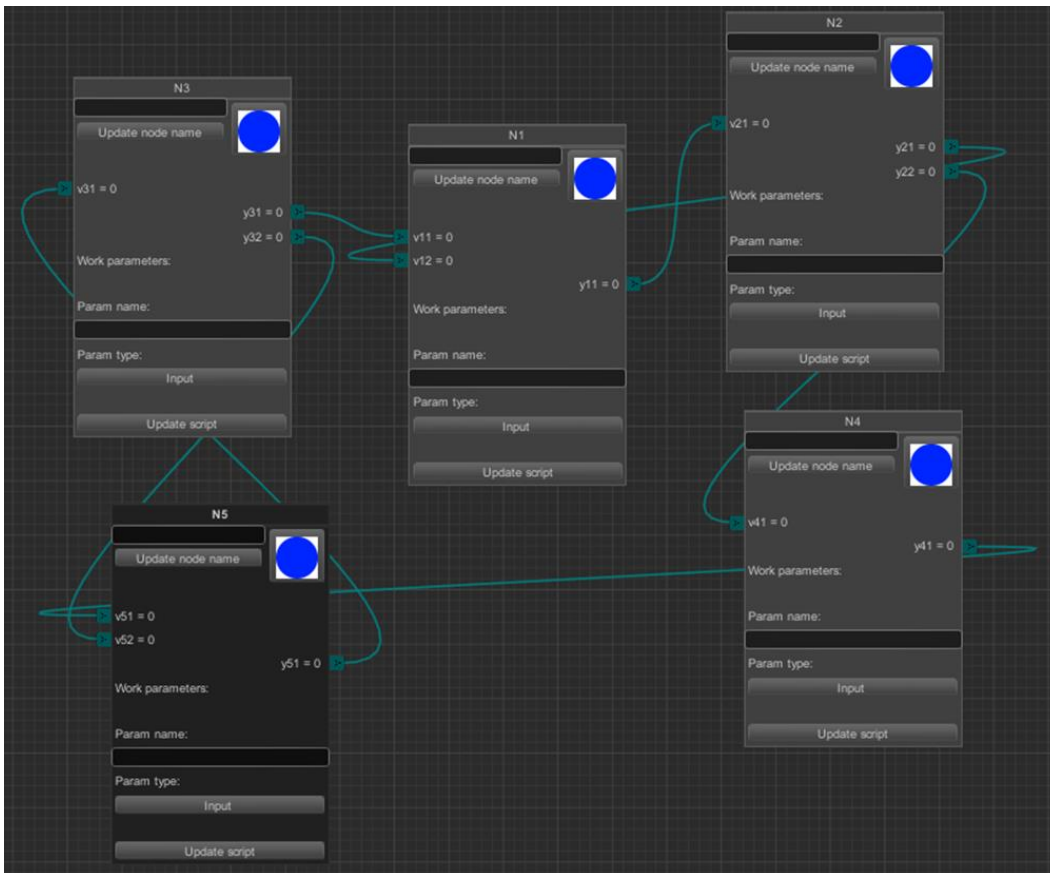


Figura 4.2: un grafico a 5 nodi con connessioni tra variabili di output e input di diversi nodi

Una volta generato lo script MatLab, viene generato il file dei parametri per il quale sono state create due nuove classi denominate SerializedCanvas e JsonNode per strutturare al meglio il file \*.json che conterrà tutti i dati scambiati tra il codice Unity e lo script Matlab.

Durante il Runtime, lo script Matlab legge i dati da questi due nuovi file comprese le costanti  $t_0$  (Istante iniziale),  $\Delta t$  (Passo temporale)  $n_t$  (Numero di passi temporali).

In particolare, la comunicazione la comunicazione Unity  $\leftrightarrow$  MatLab. I passaggi che vengono effettuati sono i seguenti:

- Generazione del file dei parametri utilizzando due classi chiamate SerializedCanvas e JsonNode per strutturare in modo corretto il file \*.json corrispondente
- Il codice legge i dati dai vari nodi presenti all'interno dell'editor
- Esegue gli script dei vari nodi nell'ordine selezionato
- Sostituisce all'interno del template i valori delle variabili calcolati durante l'esecuzione
- Mappa per ogni altro placeholder il proprio valore

Tutte le operazioni di scrittura su file vengono eseguite utilizzando la classe StringBuilder.

Una volta raccolti tutti i dati relativi ai nodi (da *var canvasParameters* a *var data*) viene richiamata la funzione *GenerateParamJsonFile* che non fa altro che prendere i dati dei nodi, aggiungere le costanti (temporaneamente scritte a codice)  $Dt$ ,  $n_t$  e  $t_0$  e serializzare il tutto per generare il file:

## 4.2. Runtime VR

Il runtime VR è un progetto separato dal principale.

La scena si presenta nel modo visualizzato in Figura 4.3

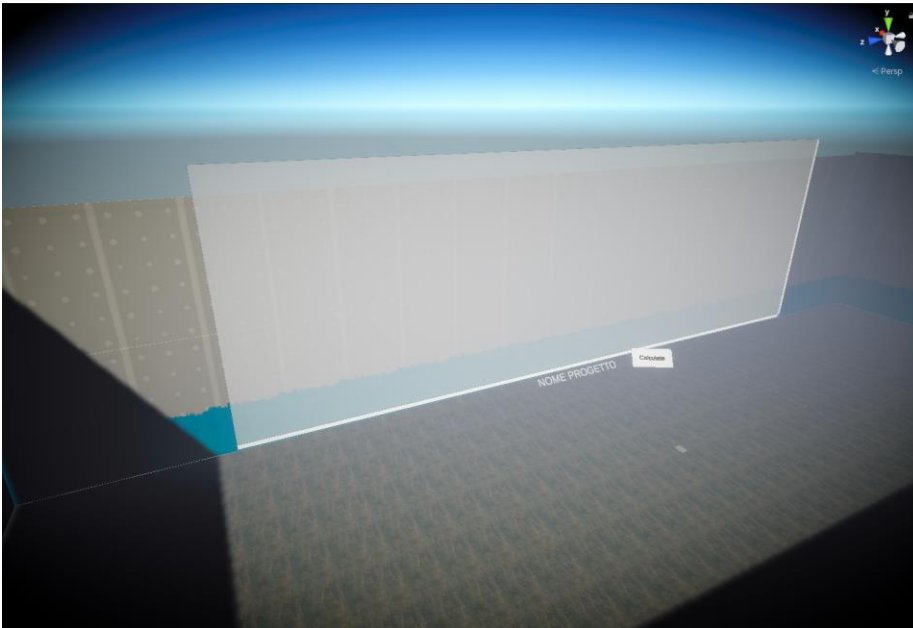


Figura 4.3: Visualizzazione in realtà virtuale della scena iniziale vuota

In pratica nel visore si vede un grande schermo in cui vengono posizionati i nodi creati nell'editor desktop, in basso il nome del progetto attualmente in esecuzione e a sinistra il tasto Calculate che crea un file Json dei parametri dei nodi ed esegue lo script generale di Matlab. Per far funzionare la modalità VR si è fatto uso del nuovo XR System di Unity. Anche in questo caso (come nei precedenti) si esegue un check di integrità sui file json dei parametri. Come detto in precedenza sullo schermo vengono disegnati i nodi con relativi collegamenti (Figure 4.4 e 4.5).



Figura 4.4: Visualizzazione in realtà virtuale con due nodi esempio

Per disegnare la linea di collegamento si è fatto uso di una libreria esterna: *Unity UI Extension* presa dall'asset store. Sul controller sinistro si vede la lista dei parametri del nodo

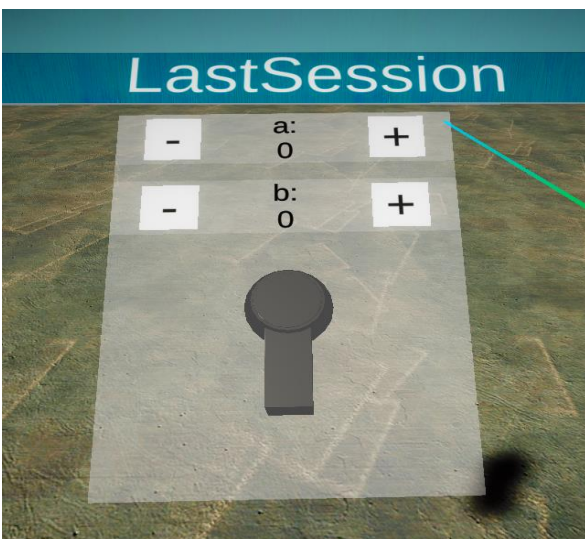


Figura 4.5: Visualizzazione in realtà virtuale di un nodo editabile

### 4.3. Test numerici su Sistemi Lineari Autonomi

Come primo test, consideriamo un ISM lineare stabile composto da cinque sottosistemi lineari collegati come in figura 4.6 dove i sottosistemi (nodi) sono numerati casualmente.

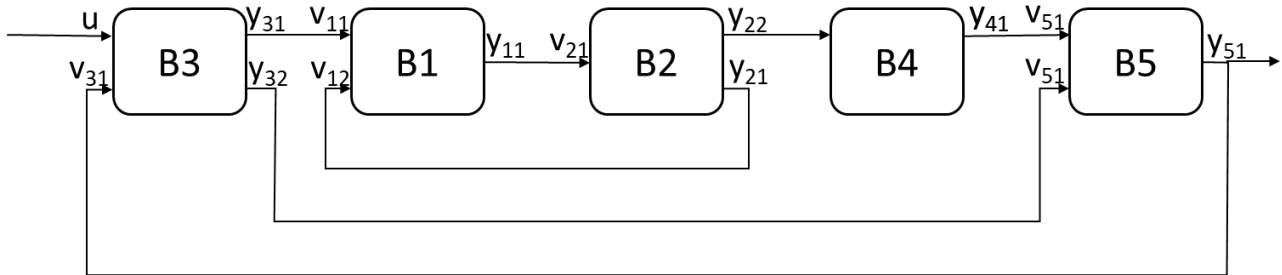


Figura 4.6. Schema grafico del primo LTI ISM testato: qui i sottosistemi sono rappresentati come blocchi (da B1 a B5) in quanto corrispondono ai blocchi delle sottomatrici nel sistema di equazioni.

C'è un solo ingresso esterno ( $u$ ) al sottosistema B1 e una sola uscita esterna coincidente con l'uscita  $y_{51}$  dal sottosistema B5. Ci sono due ordinamenti ottimali dei nodi, corrispondenti ad un insieme minimo di archi di feedback: (2, 4, 3, 5, 1) e (3, 1, 2, 4, 5). Il secondo ordinamento dei nodi verrà analizzato nel successivo esempio. In ogni caso, a causa della stabilità di tutti i sottosistemi, i due ordinamenti sono equivalenti. Viene ora analizzato il primo: di seguito sono mostrate le corrispondenti matrici modificate dopo il riordinamento dei nodi, dove ciascuna matrice è stata partizionata in blocchi, ciascuno rappresentante un sottosistema:

$$A = \begin{matrix} B2 \\ B4 \\ B3 \\ B5 \\ B1 \end{matrix} \left( \begin{array}{ccc|ccc|c} -6 & -2.75 & -1.5 & 0 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & -3 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -3 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -7 \end{array} \right); \quad P = \begin{matrix} B2 \\ B4 \\ B3 \\ B5 \\ B1 \end{matrix} \left( \begin{array}{c} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} \right)$$

$$B = \begin{matrix} B2 \\ B4 \\ B3 \\ B5 \\ B1 \end{matrix} \left( \begin{array}{cccc|cc} 4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{array} \right); \quad C = \begin{matrix} B2 \\ B4 \\ B3 \\ B5 \\ B1 \end{matrix} \left( \begin{array}{ccc|ccc|c} 0 & 0 & 2.6875 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -1 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array} \right)$$

$$\begin{aligned}
M &= \begin{array}{l} B2 \\ B4 \\ B3 \\ B5 \\ B1 \end{array} \left( \begin{array}{cccc|cccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right) \\
&= \begin{array}{l} \\ \\ \\ \\ \\ \end{array} \left( \begin{array}{cccc|cccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right) + \begin{array}{l} \\ \\ \\ \\ \\ \end{array} \left( \begin{array}{cccc|cccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right) = \\
&= M' + M''
\end{aligned}$$

$$K = A + BMC = \begin{array}{l} B2 \\ B4 \\ B3 \\ B5 \\ B1 \end{array} \left( \begin{array}{ccc|cccc} -6 & -2.75 & -1.5 & 0 & 0 & 0 & 0 & -4 \\ 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -3 & 2 & -1 & 0 \\ 0 & 0 & 0 & 0 & 2 & -3 & -1 & 0 \\ 0 & 0 & 0 & -1 & 2 & -1 & -2 & 0 \\ 0 & 0 & 2.6875 & 0 & -1 & 1 & 0 & -7 \end{array} \right)$$

Come detto, sia i sottosistemi che l'intero ISM sono asintoticamente stabili in quanto le parti reali degli autovalori della matrice  $A$  (coincidenti con gli autovalori dei blocchi diagonali), e della matrice  $K$  sono negative. Considerando lo schema di soluzione 4.12, la matrice  $K'$  (non mostrata) ha autovalori con parte reale negativa e l'ampiezza di tutti gli autovalori di  $H_1^{-1}$  (non mostrati) è senza dubbio inferiore a uno per un tempo di campionamento inferiore rispetto alla più piccola costante di tempo del sistema originale.

Un secondo esempio è costituito da un sistema molto simile al precedente ma che sostituisce il sottosistema B2 con uno instabile:

$$A = \begin{array}{l} B2 \\ B4 \\ B3 \\ B5 \\ B1 \end{array} \left( \begin{array}{ccc|cccc} -4 & -1 & -3 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -3 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -7 \end{array} \right)$$

In questo caso, mentre il modulo di tutti gli autovalori di  $H^{-1}$  è indubbiamente minore di uno (lo stesso accade per  $H_1^{-1}$  per il tempo di campionamento  $\Delta t < 0.151$ , cioè lo schema è stabile per ogni condizione iniziale e  $\Delta t < 0.151$ ), la matrice  $K'$  ha un autovalore con parte reale positiva: per ogni input  $u$  lo schema diventa instabile. Ciò, come detto nel paragrafo precedente a proposito della

matrice  $K'$ , è dovuto al fatto che il sistema senza gli archi di feedback diventa instabile: questo accade ovviamente ogni volta che si trattano sottosistemi instabili. Il feedback che va dall'uscita  $y_{21}$  del sottosistema B2 all'ingresso  $v_{12}$  del sottosistema B1 stabilizza il sistema B2: cancellando questo feedback il sistema diventa instabile. Ma questo è semplicemente dovuto a un errore di modellazione: infatti, i sottosistemi B1-B2 devono essere modellati come un unico sottosistema come mostrato in figura 4.7: il sistema B12 rappresenta un sottosistema appropriato mentre i sottosistemi B1 e B2 sono solo ausiliari in quanto utilizzati per la rappresentazione grafica di B12 ma non necessari al fine di rappresentare l'intero ISM.

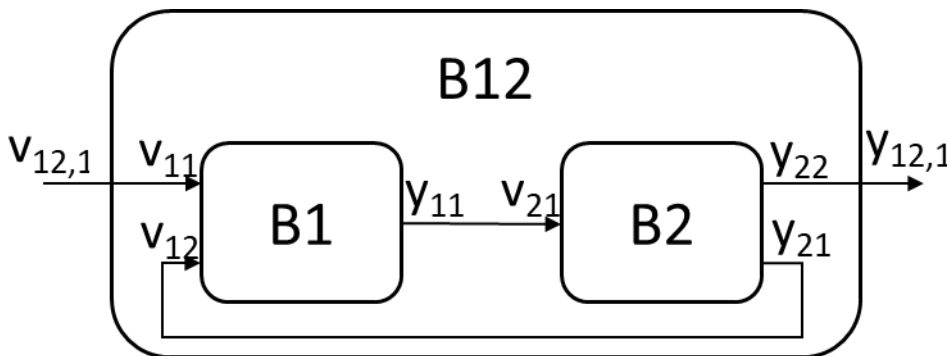


Figura 4.7. Schema grafico del sottosistema B12 ottenuto unendo i sottosistemi ausiliari B1 e B2.

Sostituendo, dunque, il sottosistema B12 nell'ISM di Figura 4.6 al posto della coppia B1-B2, l'ordine ottimale da utilizzare diventa (3, 12, 4, 5). La matrice  $A$  diventa:

$$A = \begin{matrix} B3 \\ B12 \\ B4 \\ B5 \end{matrix} \left( \begin{array}{cc|cccc|cc} -3 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & -3 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & -7 & 0 & 0 & -5.375 & 0 & 0 \\ 0 & 0 & 4 & -4 & -1 & 3 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 \end{array} \right)$$

Per questo sistema le matrici  $K'$  e  $H_1^{-1}$  soddisfano le due condizioni di stabilità.

#### 4.4. Analisi di performance

In tabella 4.1 è mostrato il confronto con gli altri metodi (schemi implicito ed esplicito) per quanto attiene le condizioni di stabilità in presenza di sottosistemi instabili (e stabilizzati da opportuni feedback), la possibilità di una sostituzione di un blocco con una versione aggiornata (o del tipo black box), e la possibilità di scegliere un particolare ordinamento dei nodi per risolvere singolarmente ogni sottosistema in tale ordine.

Metodo	Convergenza	Sostituzione di un blocco	Indipendenza delle equazioni	Necessità di ulteriori manipolazioni
Implicito	SI	NO	NO	SI
Esplicito	Non sempre	SI	SI	SI
Nuovo	SI	SI	SI	NO

*Tabella 4.1. Prestazioni comparative dei metodi standard e di quello nuovo proposto*

In particolare, abbiamo visto che anche se gli schemi Impliciti sono incondizionatamente stabili non consentono la sostituzione di un sottosistema (potremmo voler sostituire il sottosistema instabile B2 del secondo esempio nel paragrafo precedente, ad esempio solo modificando alcuni parametri interni, con il sistema stabile B2 del primo esempio) ma in questo caso è necessario modificare il solutore in quanto si deve tenere conto di tutte le equazioni contemporaneamente che, ovviamente, cambiano ad ogni sostituzione di un blocco. Un algoritmo completamente esplicito consente sicuramente sia l'ordinamento dei nodi (ma senza alcuna preferenza sull'ottimalità) che la sostituzione dei blocchi, poiché i sottosistemi vengono risolti separatamente l'uno dall'altro, ma diventa instabile in presenza di sottosistemi instabili quando gli archi di feedback utilizzati per stabilizzare sottosistemi instabili vengono completamente cancellati. D'altra parte, il nuovo metodo proposto consente sia un ordinamento ottimale (subottimale), velocizzando l'algoritmo di risoluzione rispetto ai metodi Impliciti, sia la sostituzione dei blocchi come fatto nel secondo esempio dove i due sottosistemi B1 e B2 vengono sostituiti con il blocco B12 senza alterare l'algoritmo di soluzione.

#### **4.5. Caso studio reale**

Il processo studiato come case study reale è un impianto di refrigerazione a servizio di un processo di refrigerazione (Cold Process: CP) dove viene refrigerato un primo prodotto fluido (P1). Tale tipo d'impianto è stato scelto in quanto rappresentativo di tutte le principali problematiche tipiche a livello di calcolo che si possono trovare nell'analisi degli impianti industriali, in quanto contiene diverse retroazioni ed una dinamica diversa per ognuno dei sottosistemi che compongono il modello di case study.

Come noto un impianto di refrigerazione (Refrigeration Plant: RP) è costituito da due sezioni principali: una parte fredda e una parte calda del ciclo termodinamico. La sezione fredda ha lo scopo di estrarre calore dal CP sia per espansione diretta che per mezzo di un fluido intermedio, es. glicole in caso di impianti di congelamento o acqua per temperature di refrigerazione superiori a 0 ° C. Quest'ultima opzione è quella scelta in questo caso. La sezione calda ha lo scopo di dissipare sia

il calore proveniente dal CP sia il calore utilizzato per aumentare la temperatura del refrigerante (equivalente termico del lavoro nei cicli di compressione di vapore o cascami termici nei cicli di refrigerazione ad assorbimento). Questo calore può essere dissipato direttamente nell'ambiente esterno tramite un ventilconvettore o recuperato in un fluido di lavoro secondario principalmente acqua da utilizzare nel processo caldo (Hot Process: HP) a valle, riscaldando un altro prodotto fluido (P2). Anche in questo caso, è stata scelta acqua in quanto si suppone che essa sia sempre a temperatura inferiore a 100 ° C ed immagazzinata in un serbatoio di acqua calda isolato (Water Tank: WT). Inoltre, una caldaia ausiliaria (Boiler: B) è stata aggiunta tra WT e HP per aumentare la temperatura dell'acqua alla temperatura HP desiderata. Nella Figura 4.8 è mostrato uno schema dell'impianto scelto con le principali macchine, comprese tubazioni e pompe, e le variabili di input e output.

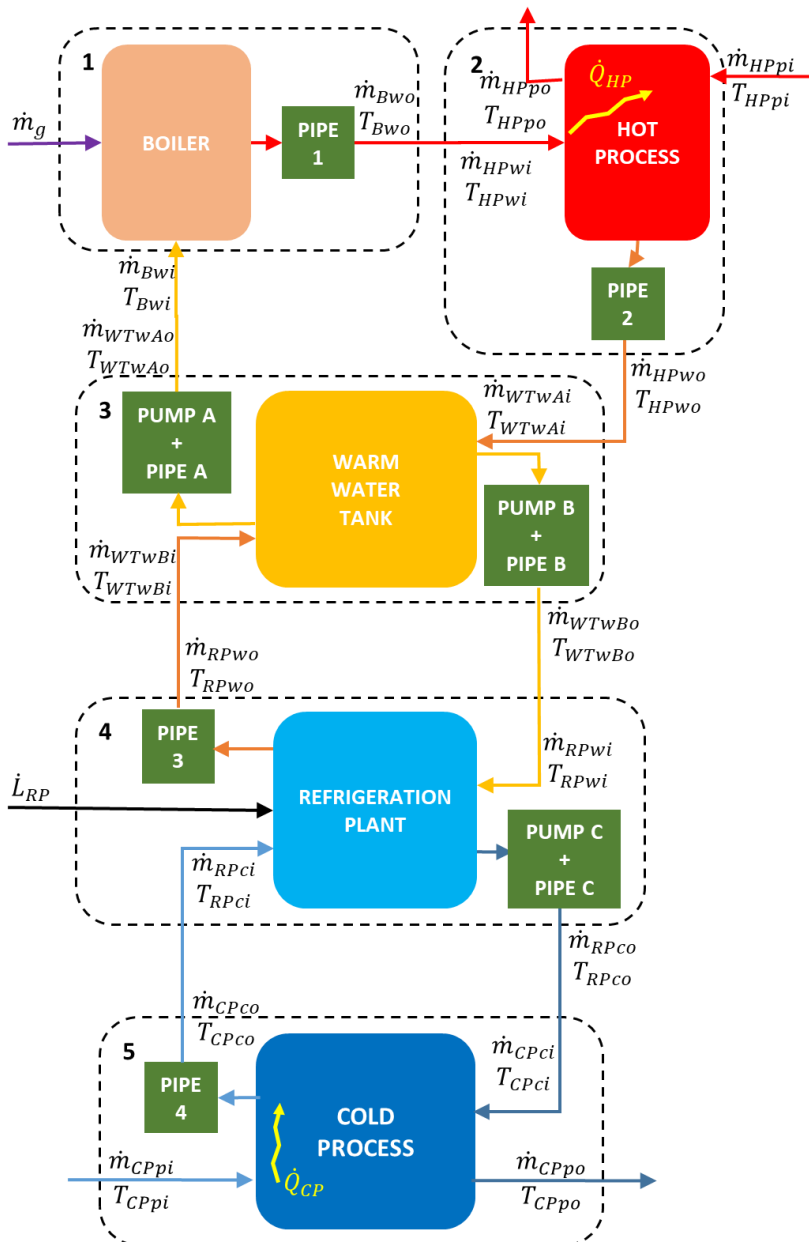


Figura 4.8. Schema dell'impianto simulato



Gli input esterni sono:  $T_E$  la temperatura ambiente esterno (non mostrata in Figura 4.8),  $\dot{m}_{HPpi}$  e  $T_{HPpi}$  la portata e la temperatura del prodotto fluido P1 in ingresso in HP,  $\dot{L}_{RP}$  la potenza fornita all'RP (supponiamo un ciclo di compressione di vapore),  $\dot{m}_{CPpi}$   $T_{CPpi}$  portata e temperatura del prodotto fluido P2 in ingresso al CP,  $\dot{m}_g$  portata del gas (es. metano) che alimenta la caldaia B.

Le uscite esterne sono:  $\dot{m}_{HPpo}$  e  $T_{HPpo}$  portata e temperatura del prodotto fluido P1 in uscita dall'HP,  $\dot{m}_{CPpo}$   $T_{CPpo}$  portata e temperatura del prodotto fluido P2 in uscita dal CP.

Tutti i processi sono interconnessi tra loro tramite archi orientati che rappresentano flussi di massa e di energia, in modo tale che l'intero impianto possa essere rappresentato come un grafo orientato dove i nodi rappresentano i processi. Di seguito si riportano le equazioni che governano ogni processo:

#### *Caldaia (B) e tubazione 1*

Nella caldaia l'acqua proveniente dal WT (Lato A) viene riscaldata utilizzando metano come combustibile:

$$\dot{m}_{Bwo} = \dot{m}_{Bwi} \quad 4.1a$$

$$T_{Bwo} = (1 - \eta_{P1}) \left( T_{Bwi} + \frac{\eta_B \dot{m}_g h_{fg}}{\dot{m}_{Bwi} c_w} \right) + \eta_{P1} T_E \quad 4.1b$$

dove  $\dot{m}_{Bwi}$ ,  $T_{Bwi}$  e  $\dot{m}_{Bwo}$ ,  $T_{Bwi}$  sono rispettivamente le portate massiche, le temperature dell'acqua in entrata in Caldaia e in uscita dal Tubazione 3,  $\eta_{P1}$  l'efficienza di scambio termico del tubazione (il più piccolo possibile se il tubazione è ben isolato: questo vale per tutti tubi),  $\eta_B$  l'efficienza della caldaia,  $h_{fg}$  il potere calorifico inferiore (Lower Heating Value: LHV) del metano,  $c_w$  il calore specifico dell'acqua,  $T_E$  la temperatura esterna.

#### *Processo caldo (HP) e tubazione 2*

Nel HP, il prodotto fluido entra nel processo (lo si può pensare come uno scambiatore di calore anche se qui non è considerato il meccanismo di trasferimento del calore) e viene riscaldata (calore trasferito:  $\dot{Q}_{HP}$ ) utilizzando l'acqua calda proveniente dalla caldaia. Quest'acqua viene quindi raffreddata (durante il processo staziona in un serbatoio di massa totale  $M_{HP}$ ) e lascia HP diretta alla tubazione 2. Le equazioni di riferimento sono:

$$\dot{m}_{HPwo} = \dot{m}_{HPwi} \quad 4.2a$$

$$\dot{m}_{HPpo} = \dot{m}_{HPpi} \quad 4.2b$$

$$\frac{dT_{HP}}{dt} = -\frac{\dot{m}_{HPwi}}{M_{HP}}T_{HP} + \frac{\dot{m}_{HPwi}}{M_{HP}}T_{HPwi} + \frac{\dot{Q}_{HP}}{M_{HP}c_w} \quad 4.2c$$

$$\dot{m}_{HPwi}c_{P1}(T_{HPpo} - T_{HPpi}) = -\dot{Q}_{HP} \quad 4.2d$$

$$T_{HPwo} = (1 - \eta_{P2})T_{HP} + \eta_{P2}T_E \quad 4.2e$$

dove  $\dot{m}_{HPwi}, T_{HPwi}$  e  $\dot{m}_{HPwo}, T_{HPwo}$  sono rispettivamente le portate e le temperature dell'acqua in entrata nel HP e in uscita dalla tubazione 2,  $T_{HP}$  la temperatura dell'acqua in uscita dal HP ed in entrata nella tubazione 2,  $\dot{m}_{HPpi}, T_{HPpi}$  e  $\dot{m}_{HPpo}, T_{HPpo}$  sono rispettivamente le portate e le temperature del prodotto fluido P1 in entrata e in uscita dall'HP,  $\eta_{P2}$  l'efficienza di scambio termico della tubazione (comunque tutte le tubazioni disperdono calore nell'ambiente esterno anche se ben coibentate),  $c_{P1}$  il calore specifico del prodotto fluido P1.

#### *Serbatoio dell'acqua calda (WT) e tubi delle pompe A e B*

Nel WT (di massa totale  $M_{WT}$  e temperatura  $T_{WT}$ ) ci sono due flussi d'acqua, rispettivamente sui lati A e B. L'acqua proveniente dal HP (Lato A) è più fredda dell'acqua nel WT, quindi è un fluido di raffreddamento per il WT, mentre l'acqua proveniente dal RP è più calda dell'acqua nel WT ed è un fluido riscaldante per il WT. Le equazioni di riferimento sono:

$$\dot{m}_{WTwAo} = \dot{m}_{WTwAi} \quad 4.3a$$

$$\dot{m}_{WTwBo} = \dot{m}_{WTwBi} \quad 4.3b$$

$$\frac{d(M_{WT})}{dt} = \dot{m}_{WTwAi} + \dot{m}_{WTwBi} - \dot{m}_{WTwAo} - \dot{m}_{WTwBo} \quad 4.3c$$

$$\frac{d(M_{WT}T_{WT})}{dt} = \dot{m}_{WTwAi}T_{WTwAi} + \dot{m}_{WTwBi}T_{WTwBi} - \dot{m}_{WTwAo}T_{WT} - \dot{m}_{WTwBo}T_{WT} \quad 4.3d$$

$$+ \frac{U_{WT}A_{WT}}{c_w}(T_E - T_{WT})$$

$$T_{WTwAo} = (1 - \eta_{PA})T_{WT} + \eta_{PA}T_E \quad 4.3e$$

$$T_{WTwBo} = (1 - \eta_{PB})T_{WT} + \eta_{PB}T_E \quad 4.3f$$

dove  $\dot{m}_{WTwAi}, T_{WTwAi}$  e  $\dot{m}_{WTwAo}, T_{WTwAo}$  sono rispettivamente le portate e le temperature dell'acqua in entrata nel WT e in uscita dalla tubazione A (lato HP),  $\dot{m}_{WTwBi}, T_{WTwBi}$  e  $\dot{m}_{WTwBo}, T_{WTwBo}$  le portate e le temperature dell'acqua in entrata nel WT e in uscita dalla tubazione B (lato RP),  $\eta_{PA}$  e  $\eta_{PB}$  le efficienze di scambio termico dei tubi A e B,  $U_{WT}$  e  $A_{WT}$  rispettivamente il coefficiente di scambio termico globale e la superficie esterna del WT.

*Impianto di refrigerazione (RP), Tubazione 3 e Tubazione della pompa C*

Nel RP viene operato un ciclo di compressione di vapore in modo da refrigerare l'acqua proveniente dal CP. Il calore ceduto dall'acqua al refrigerante RP (qui non specificato: non viene eseguita una analisi dettagliata del RP, ma ciascun processo può essere sostituito in qualunque momento con uno schema dettagliato senza per questo influire sul solutore) e il lavoro fornito al ciclo di compressione riscaldano l'acqua proveniente dal WT. Le equazioni di riferimento sono:

$$\dot{m}_{RPwo} = \dot{m}_{RPwi} \quad 4.4a$$

$$\dot{m}_{RPco} = \dot{m}_{RPci} \quad 4.4b$$

$$T_{RPwo} = (1 - \eta_{P3}) \left( T_{RPwi} + \frac{(1 + COP)}{\dot{m}_{RPwi} c_w} \dot{L}_{RP} \right) + \eta_{P3} T_E \quad 4.4c$$

$$T_{RPco} = (1 - \eta_{PC}) \left( T_{RPci} - \frac{COP}{\dot{m}_{RPci} c_w} \dot{L}_{RP} \right) + \eta_{PC} T_E \quad 4.4d$$

dove  $\dot{m}_{RPwi}, T_{RPwi}$  e  $\dot{m}_{RPwo}, T_{RPwo}$  sono rispettivamente le portate e le temperature dell'acqua in entrata e uscita dalla tubazione 2,  $\dot{m}_{RPci}, T_{RPci}$  e  $\dot{m}_{RPco}, T_{RPco}$  sono rispettivamente le portate e le temperature del prodotto fluido che lascia HP,  $\eta_{P3}$  e  $\eta_{PC}$  le efficienze di scambio termico dei tubi 3 e C,  $COP$  il Coefficiente di Prestazione del RP.

*Processo freddo (CP) e tubazione 4*

Nel CP l'acqua fredda refrigera il prodotto fluido P2 (calore trasferito:  $\dot{Q}_{CP}$ ). Quest'acqua viene quindi riscaldata (durante il processo staziona in un serbatoio di massa totale  $M_{CP}$  e temperatura  $T_{CP}$ ) e lascia il CP andando nella tubazione 4. Le equazioni di riferimento sono:

$$\dot{m}_{CPco} = \dot{m}_{CPci} \quad 4.5a$$

$$\frac{dT_{CP}}{dt} = -\frac{\dot{m}_{CPci}}{M_{CP}} T_{CP} + \frac{\dot{m}_{CPci}}{M_{CP}} T_{CPci} + \frac{\dot{Q}_{CP}}{M_{CP} c_w} \quad 4.5b$$

$$\dot{m}_{CPwi} c_{P2} (T_{CPpo} - T_{CPpi}) = -\dot{Q}_{CP} \quad 4.5c$$

$$T_{CPco} = (1 - \eta_{P4}) T_{CP} + \eta_{P4} T_E \quad 4.5d$$

dove  $\dot{m}_{CPci}, T_{CPci}$  e  $\dot{m}_{CPco}, T_{CPco}$  sono rispettivamente le portate e le temperature dell'acqua in entrata nel CP e in uscita dalla tubazione 4,  $\eta_{P4}$  l'efficienza di scambio termico della tubazione 4.

*Solutore*

Le equazioni precedenti sono in generale equazioni algebriche differenziali non lineari (DAE): il vettore di stato è  $x = (T_{HP}, T_{WT}, T_{CP})$  e il vettore degli ingressi controllati è  $(\dot{m}_g, \dot{L}_{RP})$ .

Un'altra condizione generalmente assunta per semplificare la trattazione è la condizione di non ritardo: si trascura il tempo di risposta delle pompe in quanto abbastanza piccolo rispetto alla variazione dei parametri controllati. Anche il lavoro per azionarle viene in questo caso trascurato. Inoltre, si può supporre che la portata non vari all'interno di ogni circuito:

$$\dot{m}_{WTWBo} = \dot{m}_{Bwi} = \dot{m}_{Bwo} = \dot{m}_{HPwi} = \dot{m}_{HPwo} = \dot{m}_{WTBwi} = \dot{m}_{PPA} \quad 4.6a$$

$$\dot{m}_{WTWAO} = \dot{m}_{RPwi} = \dot{m}_{RPwo} = \dot{m}_{WTAwi} = \dot{m}_{PPB} \quad 4.6b$$

$$\dot{m}_{RPCo} = \dot{m}_{CPci} = \dot{m}_{CPco} = \dot{m}_{RPCi} = \dot{m}_{PPC} \quad 4.6c$$

Utilizzando regole di interconnessione e lo schema di algoritmo sviluppato, è possibile ricavare relazioni simili per le altre variabili interne ottenendo un sistema di Equazioni Differenziali Ordinarie (ODE):

$$\dot{x} = Kx + Pu \quad 4.7$$

dove

$$K = \begin{bmatrix} -\frac{\dot{m}_{PPA}}{M_{HP}} & \frac{\dot{m}_{PPA}(1-\eta_{P1})(1-\eta_{PA})}{M_{HP}} & 0 \\ \frac{\dot{m}_{PPA}(1-\eta_{P2})}{M_{WT}} & -\frac{\dot{m}_{PPB}[1-(1-\eta_{P3})(1-\eta_{PB})] + \dot{m}_{PPA} + \frac{U_{WT}A_{WT}}{c_w}}{M_{WT}} & 0 \\ 0 & 0 & -\frac{\dot{m}_{PPC}[1-(1-\eta_{PC})(1-\eta_{P4})]}{M_{CP}} \end{bmatrix}$$

$$P = \begin{bmatrix} \frac{\dot{m}_{PPA}(1-\eta_{P1})\eta_{PA} + \dot{m}_{PPA}\eta_{P1}}{M_{HP}} & \frac{1}{M_{HP}c_w} & 0 & 0 & \frac{(1-\eta_{P1})\eta_B h_{fg}}{M_{HP}c_w} \\ \frac{\dot{m}_{PPB}(1-\eta_{P3}\eta_{PB} + \eta_{P3}) + \dot{m}_{PPA}\eta_{P2} + \frac{U_{WT}A_{WT}}{c_w}}{M_{WT}} & 0 & \frac{(1-\eta_{P3})(1+COP)}{M_{WT}c_w} & 0 & 0 \\ \frac{\dot{m}_{PPC}(1-\eta_{PC}\eta_{P4} + \eta_{PC})}{M_{CP}} & 0 & -\frac{(1-\eta_{PC})COP}{M_{CP}c_w} & \frac{1}{M_{CP}c_w} & 0 \end{bmatrix}$$

Ulteriori manipolazioni (non mostrate qui) dimostrano che la matrice K ha autovalori reali negativi. La soluzione analitica stabile del sistema di cui sopra è:

$$x = e^{Kt}(x_0 + K^{-1}Pu) - K^{-1}Pu \quad 4.8$$

Prima di procedere alla soluzione di questo caso studio sono stati preventivamente implementati nel Framework di simulazione sviluppato i diversi componenti dell'impianto ed inseriti nella libreria del Framework stesso.

Lo schema finale dell'impianto è stato generato utilizzando tutti i modelli di libreria ed è mostrato nelle Figure 4.9 e 4.10: la Figura 4.9 mostra l'intero impianto e la Figura 4.10 mostra un dettaglio. Inoltre, sono stati aggiunti altri due blocchi: il controller HP e il controller CS da utilizzare per ulteriori miglioramenti di questo modello: nel presente studio forniscono solo valori costanti per i parametri controllati ( $\dot{m}_g, \dot{L}_{RP}$ )

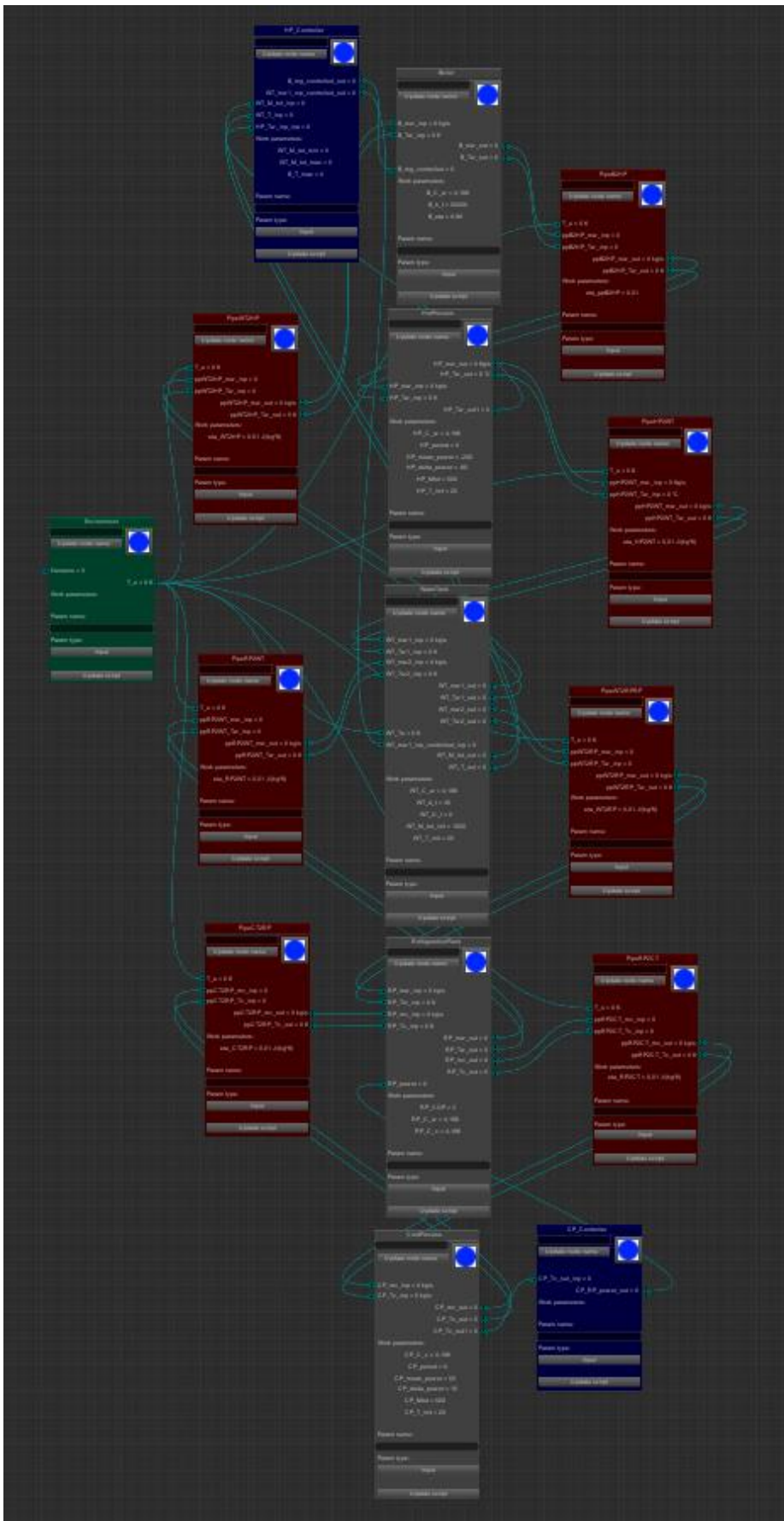


Figura 4.9. Lo schema dell'intero impianto: EE in verde, PP in rosso, controller in blu, processi in grigio.

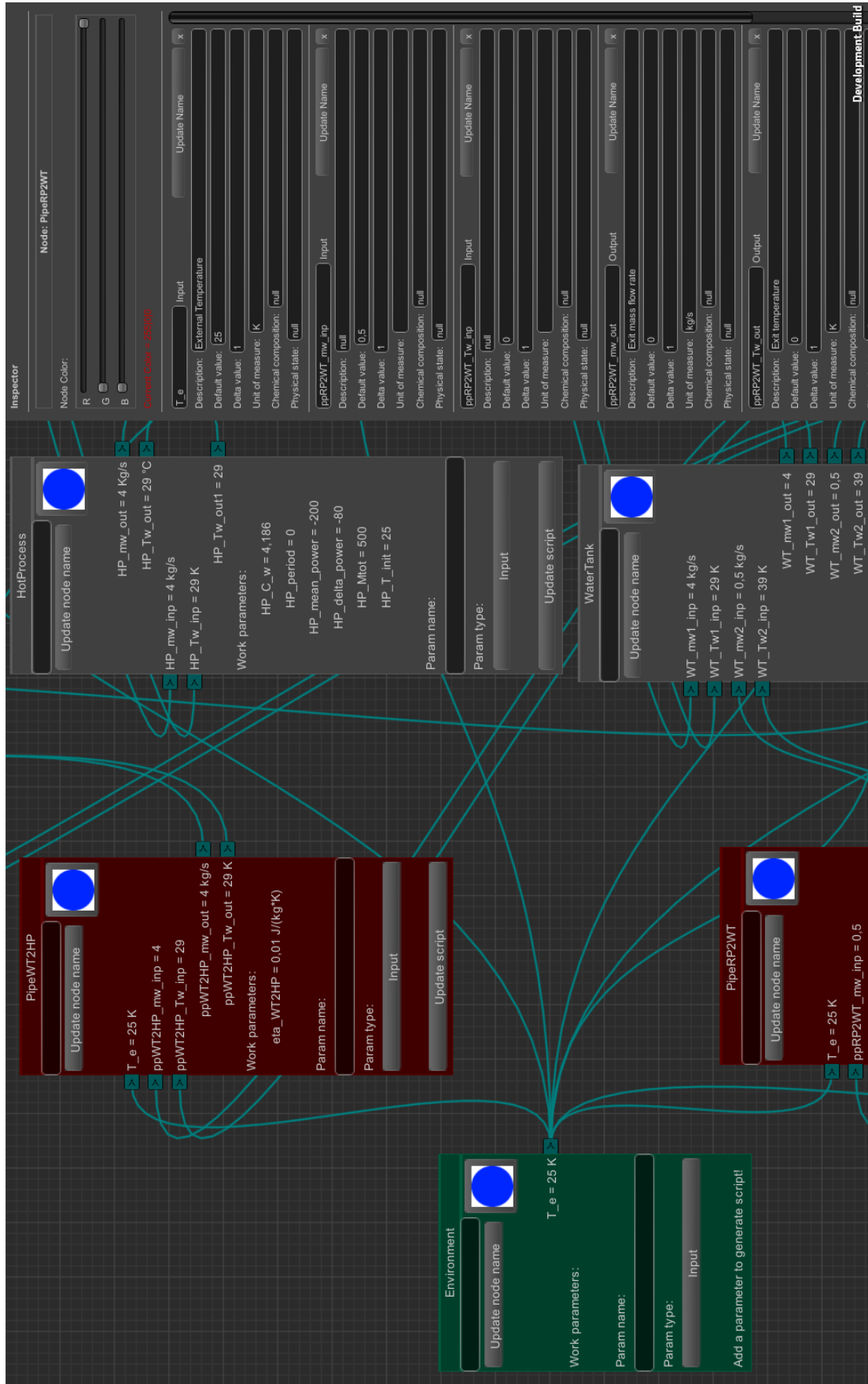


Figura 4.10. Particolare dello schema dell'impianto che mostra l'ambiente esterno, i processi HP e WT e due tubazioni

Nella Tabella 4.2 sono riportati i valori di tutte le costanti utilizzate nella simulazione, mentre le variabili di stato hanno lo stesso valore iniziale:  $T_{HP} = T_{WT} = T_{CP} = 25^{\circ}C$ .

Parametro	Valori	Parametro	Valori
$\dot{m}_{PPA}$	4 kg/s	$c_w$	4.187 kJ/kgK
$\dot{m}_{PPB}$	0.5 kg/s	$h_{fg}$	50,000 kJ/kg
$\dot{m}_{PPC}$	20 kg/s	$\eta_B$	0.98
$M_{HP}$	500 kg	$COP$	3
$M_{WT}$	1000 kg	$T_E$	25 °C
$M_{CP}$	500 kg	$\dot{Q}_{HP}$	-280 kW
$\eta$	0.01	$\dot{L}_{RP}$	28.5 kW
$U_{WT}$	0 kW/m <sup>2</sup> K	$\dot{Q}_{CP}$	60 kW
$A_{WT}$	30 m <sup>2</sup>	$\dot{m}_g$	0.0025 kg/s

Tabella 4.2. Costanti utilizzate nella simulazione

In questo caso la forma esplicita della soluzione analitica per il vettore di stato  $x$  è:

$$\begin{cases} T_{HP} = 14.7930 \exp(-0.000796 * t) + 10.207 \\ T_{WT} = 6.4373 \exp(-0.011924 * t) - 10.640 \exp(-0.000086386 * t) + 29.200 \\ T_{CP} = -3.2210 \exp(-0.011924 * t) - 10.739 \exp(-0.000086386 * t) + 38.960 \end{cases} \quad 4.8$$

il cui valore finale (stazionario) è:

$$x_{\infty} = -K^{-1}Pu = \begin{cases} T_{HP} = 10.207 \\ T_{WT} = 29.200 \\ T_{CP} = 38.960 \end{cases} \quad 4.9$$

La simulazione è stata effettuata utilizzando un passo temporale  $\Delta t = 10$  s, mentre il numero di passi temporali è  $n_t = 8640$  per una durata totale  $t_{fin} = 86400$  s = 24 h. Le figure 4.11, 4.12, 4.13 mostrano l'andamento delle soluzioni rispettivamente per  $T_{HP}$ ,  $T_{WT}$ ,  $T_{CP}$ . In ogni figura vengono mostrate la soluzione analitica, la soluzione ottenuta utilizzando un metodo completamente implicito e la soluzione ottenuta con il metodo proposto e implementato nel framework.



Va sottolineato che per la soluzione ottenuta con la tecnica numerica proposta l'errore assoluto è inferiore a  $0.15\text{ }^{\circ}\text{C}$  (0,65% del valore assoluto ottenuto con la soluzione analitica) per  $T_{HP}$ , inferiore a  $0.14\text{ }^{\circ}\text{C}$  (0.45% della soluzione analitica) per  $T_{WT}$ , nullo per  $T_{CP}$ .

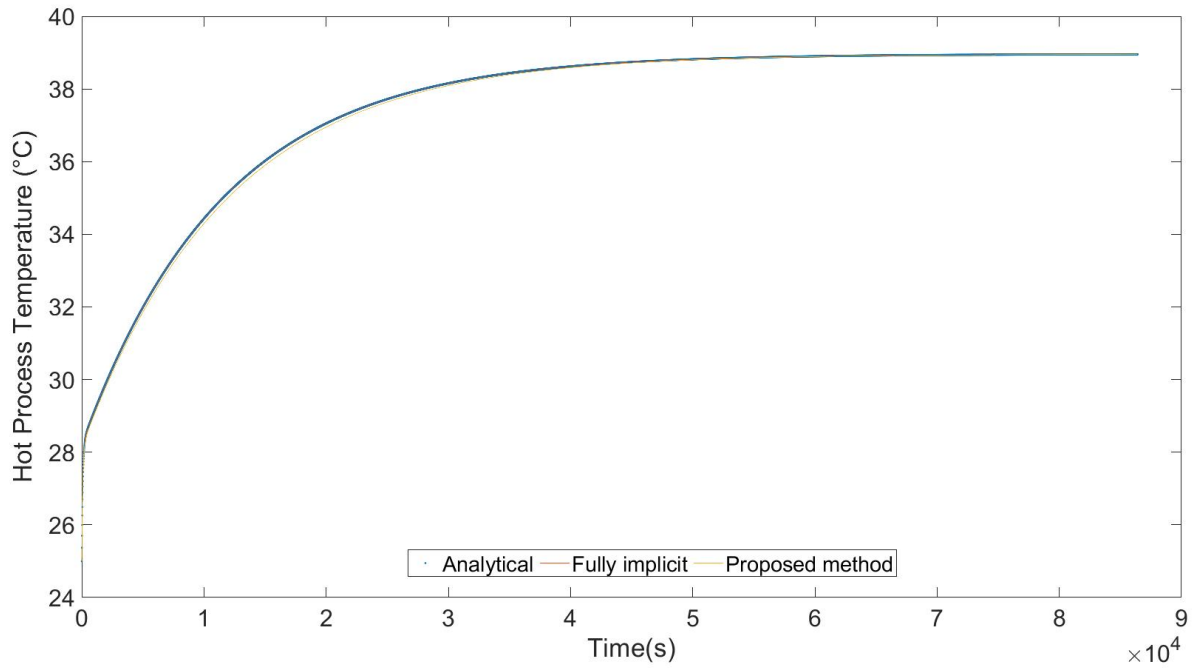


Figura 4.11. Plot delle soluzioni per  $T_{HP}$

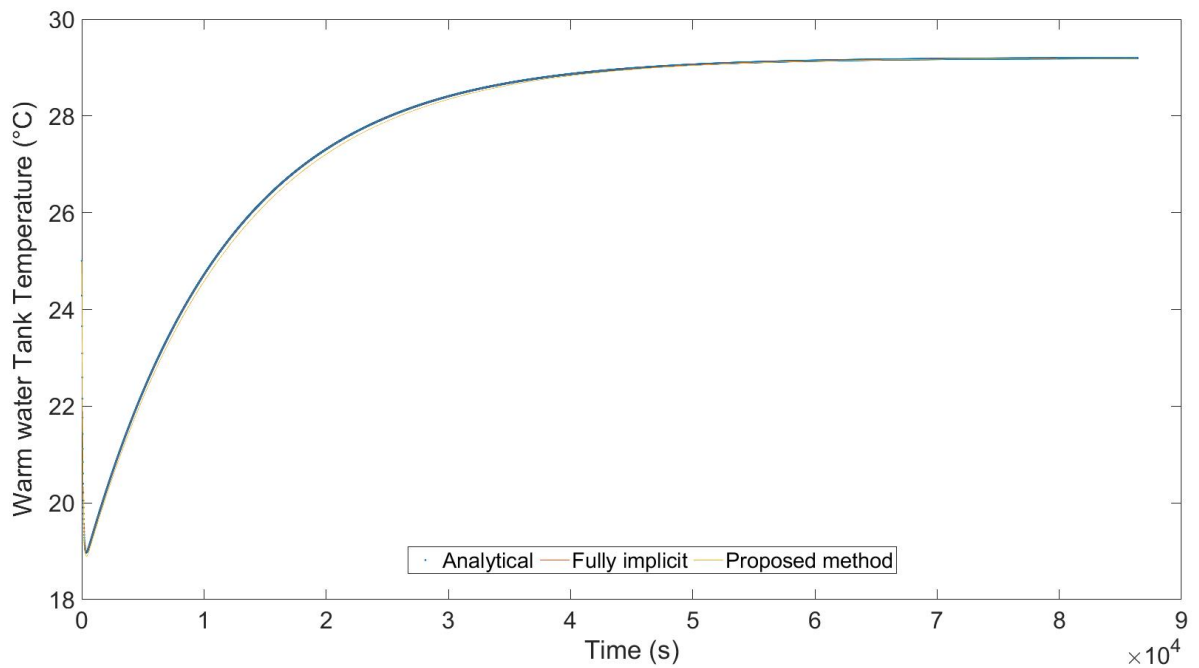


Figura 4.12. Plot delle soluzioni per  $T_{WT}$

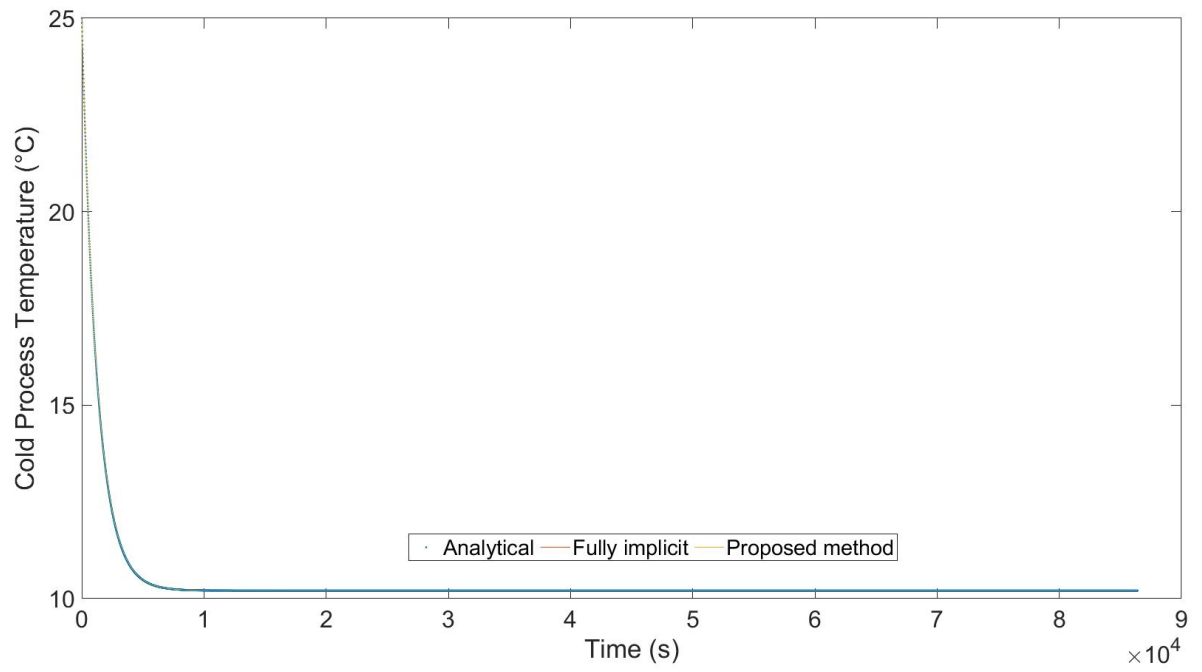


Figura 4.13. Plot delle soluzioni per  $T_{CP}$

## 5. CONCLUSIONI

In questa tesi viene presentato un nuovo metodo per trattare lo studio e la simulazione di modelli di sottosistemi indipendenti (ISM).

Oggi ci sono pochissimi esempi di applicazione del ISM nell'ambito delle procedure di progettazione assistita. D'altra parte, si è potuto osservare che tali modelli possono essere resi ancora più flessibili ed efficaci sfruttando anche alcuni concetti e metodiche tipiche dei sistemi gestionali delle filiere di produzione manifatturiera. Ad esempio, il Programma di lavoro della Commissione Europea su "Sicurezza alimentare, agricoltura e silvicoltura sostenibili, ricerca marina e marittima e sulle acque interne e bioeconomia" sin dal 2016 ha richiamato, per le proposte di progetti di ricerca e sviluppo, anche i modelli del tipo *Demand Side Management* (DSM) quali modelli base per la progettazione di sistemi avanzati di produzione nell'ambito del paradigma Industria 4.0 con l'implementazione di sistemi IoT. Tuttavia, la maggior parte di queste applicazioni riguarda la gestione dell'efficienza energetica all'interno delle filiere produttive, non solo alimentari, basata su DSM. Tali modelli comportano una ovvia evoluzione verso la logica dei sistemi interconnessi e quindi ben adatti alle tecniche tipiche degli ISM.

Tutto ciò ci porta a concludere che mentre in alcune applicazioni sono disponibili tecniche specifiche riguardanti la modellazione e l'ottimizzazione dei processi durante la produzione industriale, tali tecniche, come si è potuto vedere dalla letteratura analizzata, riguardano principalmente metodi stazionari o, al più lentamente variabili nel tempo, mentre pochi sono gli esempi di modellazione dinamica e simulazione di processi rapidamente e imprevedibilmente variabili nel tempo. Per tali motivazioni il framework studiato, progettato e realizzato nel presente progetto di dottorato è stato basato su diversi metodi che, una volta uniti in un framework completo, consentono di affrontare una classe ISM più ampia rispetto a quanto trovato nello stato dell'arte. In particolare, il metodo proposto fa riferimento all'approccio di co-simulazione, ma con la novità di integrare metodi di teoria dei grafi come il set di feedback minimo di un grafo e il corrispondente ordinamento dei nodi.

Questo approccio porta ad ottenere una risoluzione sequenziale dei sottosistemi, indipendentemente dal metodo specifico utilizzato per ogni sottosistema, riducendo la complessità del risolutore dell'intero sistema e velocizzando la procedura di risoluzione grazie all'ordinamento ottimale (subottimale) della soluzione dei sottosistemi basata solo sulle interconnessioni struttura e non sulle equazioni stesse. È stata delineata la procedura generale del nuovo metodo proposto, quindi è stata ottenuta una condizione di stabilità per i sistemi Linear Time Invariant e vengono mostrati alcuni esempi.

La ricerca futura dovrebbe indirizzare due diversi obiettivi: prima di tutto si dovrebbe cercare una condizione di stabilità più generale per sistemi non lineari o almeno quasi non lineari, quindi potrebbe essere sviluppato un nuovo quadro di simulazione in cui il metodo proposto può essere incorporato. Come già detto, i vantaggi che ne derivano sono evidenti soprattutto per quanto riguarda la flessibilità di utilizzo, modelli implementati nei singoli sottosistemi facilmente aggiornabili nonché ridotta complessità di calcolo dell'intero sistema. Pertanto, potranno essere esplorati diverse tipologie di sistemi anche alquanto complesse al fine di simulare impianti non solo macchine o parti di impianti ma intere linee di produzione. Altra importante opportunità è rappresentata dalla possibilità di cercare la configurazione ottima tra le diverse configurazioni disponibili. A tal fine sarà necessario ampliare le funzioni del Framework includendo la possibilità di creare sottosistemi a loro volta composti da blocchi interni, implementare modalità di parametrizzazione controllabile da altri blocchi da includere in libreria (questa funzione consentirà la ricerca di configurazioni ottime o sub ottime attraverso metodi di ottimizzazione parametrica e non), blocchi di libreria che eseguano le operazioni di monitoraggio e controllo degli impianti.

## BIBLIOGRAFIA

- Ahuja, R. K. (2007). A heuristic approach to the multi-period single-sourcing problem with production and inventory capacities and perishability constraints. *Inform Journal on Computing*. 19 (1), 14-26
- Ahumada, O., Villalobos, J. R. (2011). Operational model for planning the harvest and distribution of perishable agricultural products. *International Journal of Production Economics*. 133 (2). 677-687
- Akkerman, R., Yang, W., and Grunow, M. (2009). MILP approaches to sustainable production and distribution of meal elements, Paper presented at: International Conference on Computers & Industrial Engineering (Troyes, 6-9 July 2009). 973-978
- Allen, E. H. and Ilic, M. D. (2000). Reserve markets for power systems reliability. *IEEE Trans. Power Syst.* 15 (1). 228–233
- Amir-HamedMohesenian-Rad, Wong, V.W. S., Jatskevich, J., Schober, R. and Leon-Garcia A. (2010). A. Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid. *IEEE Trans. on Smart Grid*. 1 (3). 320-331
- Ascher, U.M., and Petzold, L.R. (1988). *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. SIAM Bookstore. <https://my.siam.org/Store/Product/viewproduct/?ProductId=961>
- Arnold, C., Kiel, D., & Voigt, K. -. (2016). How the industrial internet of things changes business models in different manufacturing industries. *International Journal of Innovation Management*, 20(8).
- Ayr U., Tamborrino A., Catalano P., Bianchi B., Leone A. 2015. 3D computational fluid dynamics simulation and experimental validation for prediction of heat transfer in a new malaxer machine. *Journal of Food Engineering*, vol. 154, p. 30-38, ISSN: 0260-8774, doi: 10.1117/12.2030414.
- Azaron, A., Brown, K. N., Tarim, S. A., and Modarres, M. (2008). A multi-objective stochastic programming approach for supply chain design considering risk. *International Journal of Production Economics*. 116 (1). 129-138
- Bai, J., Gooi, H. B., Xia, L. M. Strbac, G., and Venkatesh, B. (2006). A probabilistic reserve market incorporating interruptible load. *IEEE Trans. Power Syst.* 21 (3). 1079-1087
- Banga, J.R., Balsa-Canto, E., Moles, C.G., and Alonso, A. A. (2003). Improving food processing using modern optimization methods. *Trends in Food Science & Technology*. 14. 131-144

- Banga, J.R., Balsa-Canto, E., and Alonso, A. A. (2008). Quality and safety models and optimization as part of computer-integrated manufacturing. *Comprehensive reviews in food science and food safety*. 7, 168-174
- Baudrit, C., Dubois, D., & Guyonnet, D. (2006). Joint Propagation and Exploitation of Probabilistic and Possibilistic Information in Risk Assessment Models. *IEEE Trans. on Fuzzy Syst.* 14(5). 593-608
- Baudrit, C., Hélias, A., & Perrot, N. (2009). A Joint treatment of imprecision and variability in food engineering: Application to cheese mass loss during ripening. *Journal of Food Engineering*. 93. 284-292
- Baudrit, C., Sicard, M., Wuillemin, P. H. & Perrot N. (2010). Towards a global modelling of the Camembert-type cheese ripening process by coupling heterogeneous knowledge with dynamic Bayesian networks. *Journal of Food Engineering*. 98. 283-293
- Bianchi B., Cavone G., Cice G., Tamborrino A., Amodio M., Capotorto I., Catalano P. (2015). CO2 Employment as Refrigerant Fluid with a Low Environmental Impact. *Experimental Tests on Arugula and Design Criteria for a Test Bench. SUSTAINABILITY*, vol. 7, p. 3734-3752, ISSN: 2071-1050, doi: 10.3390/su7043734
- Bimbenet, J.J., Schubert, H., & Trystram, G. (2007). Advances in research in food process engineering as presented at ICEF9. *Journal of Food Engineering*. 78. 390–404.
- Blackburn, J., Scudder, G. (2009). Supply chain strategies for perishable products: The case of fresh produce. *Production and Operations Management*. 18 (2). 129-137
- Bloem R., Gabow H.N. And Somenzi F. (2006). An Algorithm for Strongly Connected Component Analysis in  $n \log n$  Symbolic Steps. *Formal Methods in System Design*, 28, 37–56
- Bosona, T. G., Gebresenbet, G. (2011). Cluster building and logistics network integration of local food supply chain. *Biosystems Engineering*. 108 (4). 293-302
- Catalano F., Perone C., Iannacci V., Leone A., Tamborrino A., Bianchi B. (2020). Energetic analysis and optimal design of a CHP plant in a frozen food processing factory through a dynamical simulation model. *Energy Conversion and Management*, 225 (2020), 113444. doi.org/10.1016/j.enconman.2020.113444.
- Catalano P., Fucci F., Giametta F., La Fianza G. Bianchi B., 2013. Vibration analysis using contactless acquisition system. *Sensing Technologies for Biomaterial, Food and Agriculture. Proc. of SPIE Vol. 8881, 888108* © 2013 SPIE CCC code: 0277-786X/13/\$18, doi: 10.1117/12.2030414, pp. 1-7.
- Cellier, F.E., Kofman, E., (2006), Springer: Berlin, Germany, 2006.

- Chaabane, A., Ramudhin, A., and Paquet, M. (2012). Design of sustainable supply chains under the emission trading scheme. *International Journal of Production Economics*. 135 (1). 37-49
- Charbit, P., Thomassé, S., Yeo, A., 2007, The Minimum Feedback Arc Set Problem is NP-Hard for Tournaments, *Combinatorics Probability and Computing* 16(1):1-4
- Chiacchio, F., Iacono, A., Compagno, L., D'Urso, D., 2020, A general framework for dependability modelling coupling discrete-event and time-driven simulation, *Reliability Engineering and System Safety*, 199, 106904.
- Chopra, S., Meindl, P. (2010). *Supply Chain Management: Strategy, Planning and Operation* (ed. Pearson Education, New Jersey)
- Christakos, G. (2002). On the assimilation of uncertain physical knowledge bases: Bayesian and non Bayesian techniques. *Advances in Water Resources*. 25. 1257-1274
- Dabbene, F., Gay, P., and Sacco, N. (2008). Optimisation of fresh-food supply chains in uncertain environments, Part I: Background and methodology. *Biosystems Engineering*. 99 (3). 348-359
- Darby, S. (2006). The effectiveness of feedback on energy consumption. A review for Defra of the literature on metering, billing and direct displays. In *Technical Report of Environmental Change Inst. (Univ. Oxford, Oxford, U.K.)*
- Datta, A. K. (2008). Status of physics-based models in the design of food products, processes, and equipment. *Comprehensive Reviews in Food Science and Food Safety*. 7(1). 121-129
- Deb, C., & Schlueter, A. (2021). Review of data-driven energy modelling techniques for building retrofit. *Renewable and Sustainable Energy Reviews*, 144
- Desiere, F., German, B., Watzke, H., Pfeifer, A., & Saguy, S. (2001). Bioinformatics and data knowledge: the new frontiers for nutrition and foods. *Trends in Food Science & Technology*. 12 (7). 215-229
- Di Perna R.A. (1970). Computational methods for digital simulation of continuous systems. Technical Report No. 123. Columbia University New York, N. Y.
- Dodds, S.,J. (2015). *Feedback Control Linear, Nonlinear and Robust Techniques and Design with Industrial Applications*. Springer series on Advanced Textbooks in Control and Signal Processing. ISBN 978-1-4471-6675-7 (eB
- Dolgui, A., Ivanov, D., Sethi, S.P., Sokolov, B. (2019) Scheduling in production, supply chain and Industry 4.0 systems by optimal control: fundamentals, state-of-the-art and applications, *International Journal of Production Research*, 57:2, 411-432
- Eksioglu, S. D., Jin, M. Z. (2006). Cross-facility production and transportation planning problem with perishable inventory. *Computational Science and Its Applications*. 3982. 708-717

- Fahrioglu, M. and Alvarado, F. L. (2000). Designing incentive compatible contracts for effective demand management,” *IEEE Trans. Power Syst.* 15 (4). 1255–1260
- Falade, K., Tihamiyu, A.T. (2020). Numerical Solution of Partial Differential Equations with Fractional Variable Coefficients Using New Iterative Method (NIM), *I.J. Mathematical Sciences and Computing*, 2020, 3, 12-21
- Fang, X., Misra, S., Guoliang, X., Yang, D. (2012). Smart Grid—The New and Improved Power Grid: A Survey. *IEEE Commun. Surv. Tutor.* 14. 944–980
- Faria, P.; Vale, Z. (2011a). Demand response in electrical energy supply: An optimal real time pricing approach. *Energy.* 36. 5374–5384
- Faria, P., Vale, Z., Soares, J., Ferreira, J. (2011b). Demand response management in power systems using a particle swarm optimization approach. *IEEE Intelligent Systems.* 99. 1-9
- Faria, P.; Vale, Z.; Soares, T.; Morais, H. (2012). Energy and reserve provision dispatch considering distributed generation and demand response. Paper presented at: 3rd IEEE PES International Conference and Exhibition on Innovative Smart Grid Technologies (ISGT Europe). (Berlin, Germany)
- Faria, P.; Vale, Z. (2014). Decision Support Concerning Demand Response Programs Design and Use—A Conceptual Framework and Simulation Tool. *Appl. Math. Inf. Sci.* 8. 161–169
- Faria, P., Vale, Z. and Baptista, J. (2015). Demand Response Programs Design and Use Considering Intensive Penetration of Distributed Generation. *Energies.* 8. 6230-6246
- Feng, B., Zhuo, L., Xie, D., Mao, Y., Gao, J., Xie, P., & Wu, P. (2021). A quantitative review of water footprint accounting and simulation for crop production based on publications during 2002–2018. *Ecological Indicators*, 120
- Fito, P., LeMaguer, M., Betoret, N. & Fito, P.J. (2007). Advanced food process engineering to model real foods and processes: The “SAFES” methodology. *Journal of Food Engineering*, 83. 390–404
- Galdiero, E., De Paola, F., Fontana, N., Giugni, M. and Savic, D. (2016). Decision support system for the optimal design of district metered areas. *Journal of Hydroinformatics.* 18(1)
- Galinec, D., Luić, L., 2020, Design of Conceptual Model for Raising Awareness of Digital Threats, *WSEAS TRANSACTIONS on ENVIRONMENT and DEVELOPMENT*, Volume 16, 493-504
- Garwood, T.L., Hughes, B.R., Oates, M.R., O’Connor, D., Hughes, R., 2018, A review of energy simulation tools for the manufacturing sector, *Renewable and Sustainable Energy Reviews* 81, 895–911.
- Gerbnera D., Keszegha B., Palmer C., Pálvölgyi C. (2016). Topological orderings of weighted directed acyclic graphs. *Information Processing Letters* 116. 564–568.



- Goetz, E. and Shenoi, S. (Eds.) (2008). *Critical Infrastructure Protection*. Springer, Boston, Massachusetts (US)
- Gomes, C., Thule, C., Broman, D., Gorm Larsen, P., Vangheluwe, H., 2018. CoSimulation: A Survey. *ACM Comput. Surv.* 51, 3, Article 49, 33 pages
- Hajjaji, Y., Boulila, W., Farah, I. R., Romdhani, I., & Hussain, A. (2021). Big data and IoT-based applications in smart environments: A systematic review. *Computer Science Review*, 39.
- He, M. M., Reutzel, E. M., Jiang, X., Katz, R. H., Sanders, S. R., Culler, D. E. and Lutz, K. (2008). An architecture for local energy generation, distribution, and sharing. *Proceedings of the IEEE Energy 2030 Conference*
- Hindi, H., Greene, D. and Laventall, C. (2011). Coordinating regulation and demand response in electric power grids using multirate model predictive control. *Proceedings of IEEE Innovative Smart Grid Technologies (ISGT)*. 1-8
- Howard, J., Ham, H., Maxemchuk, N. F. (2010). Smart Air Conditioners. *Proceedings of the Global Smart Grid Symposium (US-Korea Conference on Science, Technology, and Entrepreneurship)*
- Iannino, V., Mocci, C., Vannocci, M., Colla, V., Caputo, A., Ferraris, F., 2020, An Event-Driven Agent-Based Simulation Model for Industrial Processes, *Appl. Sci.*, 10, 4343
- Ibarra-Junquera, V., Gonzalez-Potes, A., Paredes, C. M., Martinez-Castro, D., & Nunez-Vizcaino, R. A. (2021). Component-based microservices for flexible and scalable automation of industrial bioprocesses. *IEEE Access*, 9, 58192-58207.
- Ioannou, I., Mauris, G., Trystram, G., & Perrot, N. (2006). Back-propagation of imprecision in a cheese ripening fuzzy model based on human sensory evaluations. *Fuzzy Sets and Systems*. 157. 1179-1187
- Ira-Ramírez, H., GAO, Z., Cuevas- Ramírez, L., 2014, Tracking in Interconnected Gantry Crane Systems: A Decentralized Active Disturbance Rejection Control, *Proceedings of the 33rd Chinese Control Conference*, July 28-30, 2014, Nanjing, China, 4342-4347.
- Juneja, M., Mohanty, S.R., Naga, S.K., 2019, Robust optimisation-based order reduction and stability analysis of autonomous DC microgrid with consideration of non-linearity, *International Transactions on Electrical Energy Systems* 30(3), e12228
- Karangelos, E. and Bouffard, F. (2012). Towards Full Integration of Demand-Side Resources in Joint forward Energy/Reserve Electricity Markets. *IEEE Trans. Power Syst.* 27. 280–289
- Kepler, E. F. (2004). Supply chain approach to sustainable beef production from a Brazilian perspective". *Livestock Production Science*. 90 (1). 53-61

- Kofránek, J., Kulhánek, T., Matejak, M., Ježek, F., Silar, J., 2017, Integrative physiology in Modelica, Proceedings of the 12th International Modelica Conference, May 15-17, Prague, Czech Republic, 589-603
- MacCormack, J., Zareipour, H., Rosehart, W.D. (2012). Long-Term Market Equilibrium Model with Strategic, Competitive, and Inflexible Generation. *IEEE Trans. Power Syst.* 27. 2291–2292
- Martínez-Villegas, C.T., Theilliol, D., Torres, L. (2018). Review of two noncentralized observer-based diagnosis schemes for interconnected systems. 10th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes, SAFEPROCESS 2018, Aug 2018, Warsaw, Poland.
- Mattsson, S.E., and Söderlind G. (1993). Index Reduction in Differential-Algebraic Equations Using Dummy Derivatives. *SIAM J. Sci. Comput.*, 14(3), 677–692
- McLachlan, M. and Garrett, J., (2008). Nutrition change strategies: the new frontier. *Public Health Nutrition.* 11 (10). 1063-1075
- McPhee, J., Schmitke, C. and Redmond, S. (2004). Dynamic Modelling of Mechatronic Multibody Systems With Symbolic Computing and Linear Graph Theory. *Mathematical and Computer Modelling of Dynamical Systems.* 10 (1), 1–23
- Mellit, A., Tina, G. M., & Kalogirou, S. A. (2018). Fault detection and diagnosis methods for photovoltaic systems: A review. *Renewable and Sustainable Energy Reviews*, 91, 1-17.
- Meyer R., (2015), Event-driven multi-agent simulation. In: Grimaldo F, Norling E, editors. Multi-agent-based simul. XV, Int Work MABS 2014, 9002. Paris, France: Springer International Publishing, 3–16.
- Mohagheghi, S., Stoupis, J., Wang, Z., Li, Z. and Kazemzadeh, H. (2010). Demand response architecture- Integration into the distribution management system. Proceedings of IEEE First International Conference on Smart Grid Communications (SmartGridComm). 501-506
- Mohsenian-Rad, A-H., Wong, V.W.S., Jatskevich, J. and Schober, R. (2010). Optimal and Autonomous Incentive-based Energy Consumption Scheduling Algorithm for Smart Grid. Proceedings of the IEEE Conference on Innovative Smart Grid Technologies
- Morais, H., Pinto, T., Vale, Z., Praça, (2012). I. Multilevel Negotiation in Smart Grids for VPP Management of Distributed Resources. *IEEE Intell. Syst.* 27. 8–16
- Mula, J., Peidro, D., Diaz-Madronero, M., and Vicens, E. (2010). Mathematical programming models for supply chain production and transport planning. *European Journal of Operational Research.* 204 (3). 377-390
- Mutanen, A., Ruska, M., Repo, S. and Jarventausta, P. (2011). Customer classification and load profiling method for distribution systems. *IEEE Trans.on Power Delivery.* 26 (3). 1755-1763

- Naghshbandi, S. N., Varga, L., Purvis, A., McWilliam, R., Minisci, E., Vasile, M., Troffaes M., Sedighi T., Guo W., Manley E., Jones, D. H. (2020). A review of methods to study resilience of complex engineering and engineered systems. *IEEE Access*, 8.
- Nguyen, D. T. (2010). Demand Response for Domestic and Small Business Consumers: A New Challenge. *Proceedings of IEEE Transmission and Distribution Conference and Exposition*. 1-7
- Nguyen, T. T. and Yousefi, A. (2011). Multi-objective demand response allocation in restructured energy market. *Proceedings of IEEE Innovative Smart Grid Technologies (ISGT)*,. 1-8
- Nwachukwu, G. C., Mokuwunyei N.E. (2018), Generalized Adams-Type Second Derivative Methods for Stiff Systems of ODEs. *International Journal of Applied Mathematics*, 48(4).
- Olivier, N., Luengo-Oroz, M., Duloquin, L., Faure, E., Savy, T., Veilleux, I., Solinas, X., Débarre, D., Bourguine, P., Santos, A., et al. (2010). Cell lineage reconstruction of early Zebrafish embryos using label-free nonlinear microscopy *Science*. 329. 967-971
- Papetti, A., Gregori, F., Pandolfi, M., Peruzzini, M., & Germani, M. (2018). Iot to enable social sustainability in manufacturing systems. Paper presented at the *Advances in Transdisciplinary Engineering*, 7 53-62.
- Pedrasa, M. A. A., Spooner, T. D. and MacGill, I. F. (2009). Scheduling of demand side resources using Binary Particle Swarm Optimization. *IEEE Trans. on Power Syst.* 24 (3). 1173-1181
- Perea-López, E., Ydstie, B. E., Grossmann, I.E. (2003). A model predictive control strategy for supply chain optimization. *Computers and Chemical Engineering* 27: 1201-1218
- Perone C., Catalano F., Tamborrino A., Giametta F., Bianchi B., Ayr U. 2017. Study and Analysis of a Cogeneration System with Microturbines in a Food Farming of Dry Pasta. *Chemical Engineering Transactions*, vol. 58, 2017, 499-504, doi: 10.3303/CET1758084.
- Perrot, N., Ioannou, I., Allais, I., Curt, C., Hossenlopp, J., & Trystram, G. (2006). Fuzzy concepts applied to food product quality control: a review. *Fuzzy Sets and Systems*. 157. 1145-1154.
- Perrot, N., Trelea, J.C., Baudrit,, Trystram, G., Bourguine, P. (2011). Modelling and analysis of complex food systems: State of the art and new trends. *Trends in Food Science and Technology*. 22 (6).304 - 314
- Petermeier, H., Benning, R., Delgado, A;, Kulozik, U., Hinrichs, J., & Becker, T. (2002). Hybrid model of the fouling process in tubular heat exchangers for the dairy industry. *Journal of Food Engineering*. 55. 9-17.
- Phuyal, S., Bista, D., Bista, R. (2020). Challenges, Opportunities and Future Directions of Smart Manufacturing: A State of Art Review. *Sustainable Futures* 2 (2020) 100023

- Pöchacker, Khatib, M.T., Elmenreich, W. (2014), The Microgrid Simulation Tool RAPSIm: Description and Case Study, Proceedings of the IEEE Innovative Smart Grid Technologies Asia (ISGT-ASIA 2014), Kuala Lumpur, Malaysia.
- Rajasingh, I., Rajan, B., Little Joice, S.L., 2011, Feedback Arc Set In Oriented Graphs, *J. Comp. & Math. Sci.* Vol.2 (6), 804-811.
- Rong, A. Y., Akkerman, R., and Grunow, M. (2011). An optimization approach for managing fresh food quality throughout the supply chain. *International Journal of Production Economics.* 131 (1). 421-429
- Ruiz, N., Cobelo, I. and Oyarzabal, J. (2009). A direct load control model for virtual power plant management. *IEEE Trans. on Power Syst.* 24 (2). 959-966
- Satin, A., Savelev, R., Smagin, D., Napreenko, K. and Neveshkina, A. (2019). Application SimInTech Software for Optimization Fuel System Parameters of the Perspective Helicopter. *MATEC Web of Conferences.*
- Schisler, K., Sick, T. and Brief, K. (2008). Role of demand response in ancillary services markets. *Proceedings of IEEE Transmission and Distribution Conference and Exposition.* 1-3
- Seuring, S., Muller, M. (2008). From a literature review to a conceptual framework for sustainable supply chain management. *Journal of Cleaner Production.* 16 (15). 1699-1710
- Shanshiashvili, B., Kavlashvili, N., 2020, Parameter Identification of Nonlinear Dynamic Systems of Industrial Processes, *Biological and Chemical Research*, Vol.7, 21-34
- Shi, P., McPhee, J. (2000). Dynamics of Flexible Multibody Systems Using Virtual Work and Linear Graph Theory. *Multibody System Dynamics* 4, 355–381.
- Sicard M., Perrot N., Baudrit C., Reuillon R., Bourguine P., Alvarez I. & Martin S. (2009). Paper presented at: The viability theory to control complex food processes. *European Conference on Complex Systems-ECCS'09* (University of Warwick, UK)
- Soysal, M., Bloemhof-Ruwaard, J.M., Meuwissen, M.P.M. and Van der Vorst, J.G.A.J. (2012). A Review on Quantitative Models for Sustainable Food Logistics Management. *Int. J. Food System Dynamics.* 3 (2). 136-155
- Tamborrino, A., Perone, C., Catalano, F., Squeo, G., Caponio, P., and Bianchi, B. (2019). Modelling energy consumption and energy-saving in high-quality olive oil decanter centrifuge: Numerical study and experimental validation. *Energies*, 12(13)
- Tan, Z., Li, H., Ju, L., Song, Y. (2014). An Optimization Model for Large-Scale Wind Power Grid Connection Considering Demand Response and Energy Storage Systems. *Energies.* 7. 7282–7304

- Tarjan, R. E. (1972). "Depth-first search and linear graph algorithms". *SIAM Journal on Computing* 1:2, 146–160.
- Theys, T. E., Geeraerd A. H. & J. F. Van Impe (2009). Evaluation of a mathematical model structure describing the effect of (gel) structure on the growth of *Listeria innocua*, *Lactococcus lactis* and *Salmonella Typhimurium*. *Journal of Applied Microbiology*. 107(3). 775-784
- Thoben, K., Wiesner, S. A., & Wuest, T. (2017). "Industrie 4.0" and smart manufacturing-a review of research issues and application examples. *International Journal of Automation Technology*, 11(1), 4-16.
- Trystram, G. & Courtois, F. (1996). Food process modelling and simulation, In *Computerized control systems in the food industry* (Ed G. Mittal, M. Dekker, New York) 55-85
- Vale, Z., Morais, H., Ramos, S., Soares, J., Faria, P. (2011). Using data mining techniques to support DR programs definition in smart grids. Paper presented at: Proceedings of the IEEE PES General Meeting, (Detroit, MI, USA)
- Vandendorpe A., Van Dooren P. (2008) Model Reduction of Interconnected Systems. In: Schilders W.H.A., van der Vorst H.A., Rommes J. (eds) *Model Order Reduction: Theory, Research Aspects and Applications*. Mathematics in Industry (The European Consortium for Mathematics in Industry), vol 13. Springer, Berlin, Heidelberg.
- Van der Vorst, J. G. A. J., Beulens, A. J. M. and Van Beek, P. (2000). Modelling and simulating multi-echelon food systems. *European Journal of Operational Research*. 122 (2). 354-366
- Van der Vorst, J. G. A. J., Beulens, A. J. M., and Van Beek, P. (2005). Innovations in logistics and ICT in food supply chain networks. In *Innovation in Agri-Food Systems* (Wageningen Academic Publishers, Wageningen). 245-292
- Van der Vorst, J. G. A. J., Van Kooten, O., and Luning, P. (2011). Towards a diagnostic instrument to identify improvement opportunities for quality controlled logistics in agrifood supply chain networks. *International journal on food system dynamics*. 2 (1). 94-105
- Van Impe, J.F. (1996). Power and limitations of model based bioprocess optimization. *Mathematics & Computers in simulation*. 42. 159-169
- Verma, A., Prakash, S., Srivastava, V., Kumar, A., & Mukhopadhyay, S. C. (2019). Sensing, controlling, and IoT infrastructure in smart building: A review. *IEEE Sensors Journal*, 19(20), 9036-9046.
- Vitrac, O. & Hayert, M. (2007) Effect of the distribution of sorption sites on transport diffusivities: A contribution to the transport of medium-weight-molecules in polymeric materials. *Chemical Engineering Science* 62 (9). 2503-2521
- Wacks, K. (2008). Home Area Networks for Electricity Demand Management. *iHomes& Buildings*. 15-17

- Wills, A., Schön, T.B., Ljung, L., Ninness, B., 2012, Identification of Hammerstein–Wiener Models, *Automatica*, (49), 1, 70-81
- Xie A. and Beere P. A., (2000). Implicit Enumeration of Strongly Connected Components and an Application to Formal Verification. *IEEE Transactions on computer-aided design of integrated circuits and systems*. 19. 10.
- Xiong, G., Chen, C., Kishore, S. and Yener, A. (2011). Smart (in-home) power scheduling for demand response on the smart grid. *Proceedings of Innovative Smart Grid Technologies (ISGT)*. 1-7
- Xu T., Wang G., Yang J., (2020). Finding strongly connected components of simple digraphs based on granulation strategy. *International Journal of Approximate Reasoning*. 118. 64-78
- Yan, C., Banerjee, A., and Yang, L. (2011). An integrated production-distribution model for a deteriorating inventory item. *International Journal of Production Economics*. 133 (1). 228-232
- Yang, D., Gao, X., Xu, L., & Guo, Q. (2018). Constraint-adaptation challenges and resilience transitions of the industry–environmental system in a resource-dependent city. *Resources, Conservation and Recycling*, 134, 196-205.
- Yoo, T. H., Kwon, H.G., Lee, H.C., Rhee, C.H., Yoon, Y.T. and Park, J.K. (2011). Development of reliability based demand response program in Korea. *Proceedings of IEEE Innovative Smart Grid Technologies (ISGT)*. 1-6
- You, F., Tao, L., Graziano, D. J., and Snyder, S. W. (2012). Optimal design of sustainable cellulosic biofuel supply chains: Multiobjective optimization coupled with life cycle assessment and input–output analysis. *AIChE Journal*. 58 (4). 1157-1180
- Younger, D.H., 1963, Minimum Feedback Arc Sets for a Directed Graph, *IEEE Trans. Circuit Theory* CT-10(2), 229-245.
- Zhang, Y., Li, Y., Tomsovic, K., Djouadi, S., Yue, M. (2020). Review on Set-Theoretic Methods for Safety Verification and Control of Power System. <https://arxiv.org/abs/2001.00080v2>
- Zhou, M., Gao, Y., and Li, G. (2008). Study on improvement of available transfer capability by demand side management,” Paper presented at: 3rd Int. Conf. Electric Utility Deregulation and Restructuring and Power Technologies, DRPT’08. 545–550
- Zohdi, T.I. (2018). Multiple UAVs for Mapping: A Review of Basic Modeling, Simulation, and Applications. *Annu. Rev. Environ. Resour.* 43:523–43