

## Article

# Interconnected Systems Modelling in Food Industry: General Solution Scheme and Stability Conditions for Linear Time-Invariant Systems

Filippo Catalano <sup>1</sup>, Moises Diaz <sup>2</sup> , Roberto Romaniello <sup>3,\*</sup> , Gianfranco Semeraro <sup>4</sup>  and Giuseppe Pirlo <sup>5</sup>

<sup>1</sup> CTS s.r.l.—Spin-Off of the Department of Agriculture, Environment and Food of the University of Molise, Via Francesco de Sanctis, 86100 Campobasso, Italy; ing.filcat@gmail.com

<sup>2</sup> MD: Physics Department, Universidad de Las Palmas de Gran Canaria, 35000 Las Palmas, Spain; moises.diaz@ulpgc.es

<sup>3</sup> Department of Agriculture, Food, Natural Resources, and Engineering (DAFNE), University of Foggia, Via Napoli 25, 71122 Foggia, Italy

<sup>4</sup> Palazzo del Broletto, University School for Advanced Studies IUSS Pavia, Piazza della Vittoria 15, 27100 Pavia, Italy; gianfranco.semeraro@iusspavia.it

<sup>5</sup> Department of Computer Science, University of Bari Aldo Moro, Via Orabona 4, 70125 Bari, Italy; giuseppe.pirlo@uniba.it

\* Correspondence: roberto.romaniello@unifg.it

**Abstract:** The problem of simulating complex systems, such as production lines, industrial plants, food processing, etc., today represents an opportunity that brings with it the great advantage of limiting design costs. However, nowadays the designer, after defining and implementing the mathematical models of the studied process, may need to rebuild the whole simulation framework because he needs to modify the model of even just one subsystem. It is for this reason that in this paper, a new framework for the use of Individual Subsystem Models (ISM) for the modelling and simulation of interconnected systems has been studied and implemented. Furthermore, the study of the state of the art has revealed the lack of efficient and sufficiently general numerical algorithms, but, at the same time, it is simple to use to solve the algebraic-differential equations deriving from the ISM simulation. The proposed new approach follows the paradigm of co-simulation methods, including graph theory methods, to solve the general ISM simply and efficiently. In this approach, each subsystem is required to have its own representation independently of the other subsystems. In this way, it is always possible to replace any subsystem whenever an updated representation becomes available, making maintenance and evolution of the entire ISM flexible. Our framework calls each subsystem separately in an optimal (suboptimal) order based on the structure of the graph. Each calculated output is transferred to the input of the next subsystem in the chosen. The general procedure has been validated in the context of Linear and Time-Invariant ISMs: in these hypotheses, the stability conditions have been calculated and numerical tests have been performed which show the effectiveness of the proposed approach.

**Keywords:** individual subsystems model; solution scheme; graph structure; food industry



**Citation:** Catalano, F.; Diaz, M.; Romaniello, R.; Semeraro, G.; Pirlo, G. Interconnected Systems Modelling in Food Industry: General Solution Scheme and Stability Conditions for Linear Time-Invariant Systems. *Appl. Sci.* **2023**, *13*, 5740. <https://doi.org/10.3390/app13095740>

Academic Editor: Antonio Valero

Received: 14 April 2023

Revised: 30 April 2023

Accepted: 3 May 2023

Published: 6 May 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Control of complex interconnected systems has a wide range of usage in our society. Some applications can be found in [1,2] (fault diagnosis), [3] (unmanned vehicles), [4,5], (scheduling problems), [6,7] (hydraulic networks), [8–21] (power systems, mechanical systems), [22,23] (processing plants and supply chain management), [24] (Biological systems), and [25–27] (Information processing). For instance, control and fault diagnosis of large-scale interconnected networks are tasked with some design challenges due to each network's particular characteristics. An engineer designing a control or fault diagnosis scheme for such systems considers the number of equations and variables involved in the

process and its complexity due to non-linearities [28]. Mature energy prediction models, mainly Building Energy Modelling and Manufacturing Process Simulation, have been used extensively in their respective fields. However, they use different methods for each part of a building or a process, limiting the ability of the used frameworks to identify energy and yield improvements. Similarly, a holistic approach, such as in co-simulation (see below), “is more likely to achieve the greatest energy efficiency savings or reduction in energy use” [29,30].

In particular, the food sector is of great interest as there are many aspects that can be addressed through numerical models. Bianchi et al., 2021 [9] published a comprehensive review regarding the numerical modeling of food products and the numerical simulation of food processes.

They conclude that there are few examples of dynamic modeling and simulation of food processes that vary rapidly and unpredictably over time; for some types of processes, the authors propose the use of techniques typically used in other engineering fields. In fact, food processes are very complex and difficult to model. On the other hand, they are well represented by models used for the dynamic simulation of interconnected systems.

Moreover, “deviations in the proper operation of these interconnected systems or their parts are no longer merely technical difficulties; they pose a danger with a global security impact” [31]. These drawbacks are repetitively developed each time for the systems to be controlled and simulated, especially in the food industry. Another difficulty arises when dealing with extensive interconnected systems, such as the global iterative method, which is necessary to simulate non-linear systems and slow down the simulation process due to the huge computational work [32]. These systems are typically used in food and food processing modelling [9], such as biogas production from food processing residues [11,13], pasta factories [12,17], food freezing facilities, and installed cogeneration plants [14].

In contrast, one of the main advantages of modelling complex systems as interconnected ones are the facility of substituting one of the subsystems with an updated version. Even its portability characteristic allows substituting with a black box, furnished by another company. In this latter case, it is not appropriate to change the solving algorithms because the equations modelling the new subsystem are not known at worst. In fact, in this case, the black box model makes it hard to use any benefits from the knowledge of the governing equations.

What has just been said allows us to state that the techniques used in many engineering applications can be effectively exported to food applications, such as in the whole food supply chain. In all these cases, we find a common aspect: the presence of interconnected systems. In general, there is a lack of literature about the application of interconnected systems modelling to the food industry because of the lack of efficient algorithms needed.

The newly proposed method defining an a priori solving ordering of each subsystem based only on the whole system graph allows the carrying out of different simulations when varying the content of the N subsystem without altering the structure of the interconnections. In addition, the reliability of the proposed method is not altered when a black box is used as a subsystem. There is no need to iterate all the equations of the whole system, but the only required iterations are those implemented in each subsystem. Thus, the drawbacks mentioned above are overcome.

### 1.1. Related Works

Interconnection means an information transfer from one system or subsystem to another; this definition applies to a comprehensive set of systems: biological organisms, electronic devices, or even abstract concepts, such as equations [2].

The main elements representing the different subsystems of an interconnected system can be identified and schematized in subsystem diagrams. Each subsystem represents the cause–effect relationship between input variables and output variables of each subsystem. Such a system can be represented by a graph whose nodes are the system’s subsystems. A. Vandendorpe and P. Van Dooren [33] gave an effective definition of an interconnected sys-

tem: it can be viewed as a linear/non-linear system with a Multiple-Input-Multiple-Output (MIMO) transfer function composed of the interconnection of subsystems. Each subsystem is, in turn, characterized by a linear/non-linear MIMO transfer function. Therefore, by introducing the mathematical model of the elements into each subsystem, a quantitative evaluation of the whole system's evolution over time is possible.

According to Goetz and Shenoi [34], nowadays, the modern industry is heavily based on natural and/or generated resources (e.g., water, electricity, heat, etc.) and their distribution. Moreover, there is wide literature on this argument [35,36]. Therefore, there is no unique approach to the numeric solution of such systems.

There are several modelling approaches as reviewed in [29], but also in [37] for Time-driven and Event-driven approaches, [38,39] for Agent driven approach, and [40] for co-simulation:

- In the Time-driven approach, time is incremented with discrete time steps as in classical numerical simulation methods.
- In the Event-driven approach, discrete events occur in a sequence regardless of time between events ruling the model evolution; a mixed Time-Event driven approach is sometimes adopted.
- In the Agent-driven approach, different units can operate independently but communicate and cooperate with a defined environment. It can be either a Time-driven, Event-driven, or mixed approach.
- In the Co-simulation approach, a global simulation of interacting systems is carried out by composing its components' (independent) simulations. At each iteration of the global system simulation, each subsystem's inputs and outputs are transferred to the other applicable subsystems.

An interesting modelling and simulation method of interconnected systems composed of subsystems and suitable interconnections relies on the co-simulation approach and the Individual Subsystem Models (ISMs) paradigm. Conceptually, the ISMs were introduced by R.A. Di Perna [41], where a continuous or a discrete model for each subsystem is built, and interconnections between these models are defined to produce the overall system simulation. Despite this old definition, no general work has been carried out to solve ISM but only some subclasses. A class of a non-linear dynamical models is the Block-Oriented Model (BOM, a subclass of ISM). We can find a separately linear and memoryless non-linear models, each one represented as a single subsystem [42]. These models, also defined as causal models [43], are widely applied in numerical simulation (among many others: B. Shanshiashvili and N. Kavlashvili [44], M. Juneja et al. [45], and H. Ira-Ramírez et al. [46]). Another studied subclass of ISM is that of acausal models [43], consisting in defining instances of each subsystem model where only the relations among the different variables of each model are defined (e.g., thorough linear/non-linear algebraic/differential equations) and the relations among the different models too. Then, the whole model is solved as a whole or through reduction-model algorithms.

### 1.2. Aim and Organization of the Paper

Our method proposes to improve and generalize numerical algorithms for ISM simulation, overcoming the drawbacks mentioned above. Our hypothesis relies on the nature of each Individual System, making it possible to solve each subsystem independently of each other. Then, a graph of the whole ISM is realized, and an optimal (suboptimal) ordering of node numbering is automatically determined; it corresponds to the subsystem solution ordering. One of the significant components of our proposal is the feedback is finalized to control and stabilize the overall system. It is required in the modelling of such systems in order to transform the process described in terms of balance and constitutive equations. At each time step, after solving the  $i$ -th subsystem, all the calculated outputs are transferred to the inputs of the following connected subsystems, while the other inputs (not yet transferred because of the presence of feedback) remain with the values calculated during the previous time step. This allows optimizing the number of newly calculated inputs. Finally, stability conditions were established for Linear Time-Invariant (LTI) systems.

The paper is organized as follows. In Section 2, a brief review of the general framework of ISMs applies, then the proposed method is shown, and finally, the method is applied to linear ISM and stability conditions are obtained. In Section 3, some case studies of LTI ISM are discussed, then the newly proposed method is compared with the other methods.

## 2. Materials and Methods

### 2.1. A General Framework for Individual Subsystem Models

General ISMs give rise to a system of implicit Differential Algebraic Equations (DAEs) for each subsystem [47] of the form:

$$\Phi_i(t, x_i(t), \dot{x}_i(t), y_i(t), v_i(t), u_i(t), p_i) = 0, \quad (1)$$

where  $t$  is time,  $x_i$  are the differential variables,  $y_i$  are the subsystem output algebraic variables,  $v_i(t)$  is the subsystem input vector (that is, inputs whose values are equal to outputs of other subsystems in the ISM),  $u_i$  is the ISM external inputs vector (that is, the inputs, if any, coming from the outside environment), and  $p_i$  is the subsystem internal constant parameters vector. In DAEs, some of the equations could be purely algebraic, leading to more difficulties when combined with differential equations [48]. The interconnection of subsystems is achieved by connecting subsystem outputs to the inputs of other subsystems. Precisely, the  $i$ -th subsystem is connected to the  $j$ -th subsystem if there is a directed connection from one of the scalar components  $\tilde{y}_{ir}$  of the output vector  $y_i = (\tilde{y}_{ir})$  of the  $i$ -th subsystem to a scalar component  $\tilde{u}_{js}$  of the input vector  $u_j = (\tilde{u}_{js})$  of the  $j$ -th subsystem. Without loss of generality, we can neglect self-loops (i.e., we neglect the case  $j = i$ ) using the identity function as an intermediate auxiliary subsystem.

Appending vectors  $x_i, X = [x_i]_{i \in \mathbb{N}}$  and the same can be done  $Y = [y_i]$ ,  $V = [v_i]$ ,  $U = [u_i]$ ,  $P = [p_i]_{i \in \mathbb{N}}$   $\Phi = [\Phi_i]_{i \in \mathbb{N}}$ .

Therefore, combining all the subsystem equations, the result is a fully implicit system of DAEs that is generally solved with different available methods, including symbolic manipulation and numerical solvers:

$$\Phi(t, X, \dot{X}, Y, V, U, P) = 0. \quad (2)$$

On the other hand, without loss of generality, we can describe our method using the Ordinary Differential Equations ODE representation for better readability and using a suitable graphical representation. DAEs can be easily introduced without any change in the method. Moreover, if the Jacobian of  $\Phi$  with respect to  $\dot{X}$  is non-singular, all the following forms are equivalent to (2) [47]. In particular, for each subsystem  $N_i$  we have:

$$f_i[t, x_i, \dot{x}_i, v_i, u_i, p_i] = 0, \quad (3a)$$

$$y_i(t) = g_i[t, x_i, \dot{x}_i, v_i]. \quad (3b)$$

All the algebraic variables disappear and the  $x_i$  are all differential ones. Sometimes ODEs can be written in the explicit form:

$$\dot{x}_i = f_i(t, x_i(t), v_i(t), u_i(t), p_i). \quad (4)$$

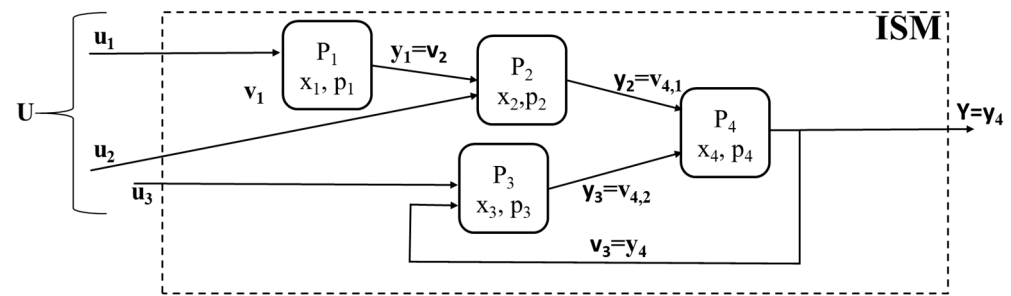
Such an interconnected ISM formed by all the subsystems  $N_i$  can be represented as shown in Figure 1. As said above, appending vectors  $v_i = [v_{ir}]$  allows to form a single column vector  $V$ :

$$V(t) = [v_1(t), v_2(t), \dots, v_N(t)]^T = [V_h], \quad (5)$$

where  $V_h$  is the  $h$ -th scalar component of  $V$ ; the same can be conducted with outputs  $y_i = [y_{is}]$ :

$$V = MY \quad (6)$$

where each element  $M_{hk}$  of the connection matrix,  $M$  is  $M_{hk} = 0$  or  $M_{hk} = 1$ .



**Figure 1.** An example of ISM scheme: Node  $P_i$ , represents a subsystem able to model an industrial Process and its internal state variables characterize it as  $x_i$  and parameters  $p_i$ , external inputs  $u_i$ , internal inputs  $v_i$ , and outputs  $y_i$ .

Last condition,  $M_{hk} = 1$  represents a connection from the scalar output  $Y_k$  to the internal scalar input  $V_h$ . If no self-loops are allowed, all the diagonal elements are null: this hypothesis can be accepted as it simplifies the calculations below without losing generality. Finally, we can assume that each scalar component of a subsystem output vector is connected to the one and only one scalar component of another subsystem internal input vector (multiple connections can be modelled with multiple identical scalar outputs). To this end, it helps in explaining the new approach in the next section. With this condition, each component of a subsystem output vector is connected to an input, meaning that each row and column of the matrix  $M$  have one and only one non-zero entry: matrix  $M$  is a permutation matrix.

As for any other method, numerical simulation of continuous ISMs systems first requires approximating each subsystem by a discrete model even if it is conducted only locally (i.e., only when the solution of each subsystem is required) and not for the whole DAEs system simultaneously. The numerical approximation of the dynamical evolution of an ISM of the form shown in Figure 1 is calculated at a sequence of  $n$  discrete time points,  $t_n = \sum_{k=0}^n \Delta t_k$ . Each step size  $\Delta t_k$  may be fixed or variable depending on the type of numerical method used due to the stiffness variability of the system requiring the change of  $\Delta t_k$  during simulation.

On the other hand, solving fully implicit DAEs is a popular method, but it requires a tremendous computational expense [49].

The main methods used to reduce DAES computational expense [50] are

- Casualization
- Tearing

Generally, they are carried out by means of symbolic processing. These methods essentially hide the algebraic variables from the numerical solver as much as possible, exposing only a minimal set of equations and variables: the state variables. The state is a minimal set of ISM variables. If their current values are known, and all future external ISM input values are known, then the future behaviour of the ISM is uniquely determined. Of course, the same definition of a state is also valid for each subsystem. If in the system of equations there are also pure ODEs (see (4)) in addition to the DAEs, then the vector of differential variables  $x_i$  appearing in each ODE naturally represents a suitable set of state variables for each ODE. On the other hand, for most DAEs, a subset of both differential and algebraic variables in Formulation (1) could still be used as a suitable state [49].

The first procedure described above, casualization, uses symbolical manipulation to obtain the state variables and convert a DAE model like Equation (1) into an iterative numerical model. The state derivatives are calculated for all the differential algebraical variables in proper sequential order. Casualization is divided into two steps: Matching and Subsystem-Lower Triangular (BLT) ordering. The most popular and widely used method is Tarjan's algorithm [21], still considered one of the most efficient. On the other hand, one



of the most significant drawbacks is symbolic manipulation: the computational cost moves from direct heavy numerical solving to heavy symbolic computation.

The second procedure, tearing [22,51,52], is a method for solving sparse systems of equations without any particular structure (e.g., banded or block-diagonal), and it is less computationally expensive than casualization. This is achieved by ordering variables and equations to divide the system into two partitions. The first one is highly structured, can be solved efficiently, and then utilized in the solution of the whole system by subsystem elimination.

While BLT ordering guarantees numerical stability, tearing does not. This well-known drawback of tearing is challenging to address when no reliable a priori information about the system structure is available or, if known, not used. However, when dealing with structured systems where each subsystem can be solved individually by calculating outputs from inputs and parameters (implicitly or explicitly does not matter), it might be tractable using numerical algorithms that guarantee numerical stability, under the assumption that the initial guess is sufficiently close to the solution; this is a relatively old result, although not as used as it should [53,54].

In both cases, numerical solving applied after casualization or tearing becomes less expensive as it accounts for computational complexity.

The discussion above derives that there are many available methods to model even large systems. However, each has its own: no global and comprehensive approach has been developed after the report of R.A. Di Perna [41]. Moreover, as said, some tools solve the system of DAEs as a whole. Other tools have a different approach to a preliminary determining of an ordering in the subsystem's execution. However, they are mainly based on each subsystem's nature rather than on the structure of the graph representing the ISM as a collection of subsystems interconnected with each other as described above. Therefore, the proposed method follows the paradigm of the co-simulation approach but with a new perspective, including graph theory methods for efficiently solving general ISM.

## 2.2. The New Approach to Individual Subsystems Models: General Aspects

As briefly inferred in the previous section, when dealing with ISMs, the structure of the problem can be directly defined by its model graph where the  $i$ -th subsystem,  $i = 1 \dots N$ , can be described by a subset of DAEs as in (7), here repeated for readability:

$$\Phi_i(t, x_i(t), \dot{x}_i(t), y_i(t), v_i(t), u_i(t), p_i) = 0. \quad (7)$$

All the subsystems are interconnected to each other by connecting their outputs to corresponding inputs of other subsystems. A graph  $G(V, E)$  represents this structure where  $V$  is the vertex set of the  $N$  nodes, and  $E \subseteq V \times V$  is the edge set (directed edges). Each node  $N_i$  ( $N_i \in V$ ,  $i = 1 \dots N$ ) corresponds to a single subsystem. A directed edge  $(N_i, N_j) \in E$  exists if a scalar output of the subsystem  $N_i$  is connected to a scalar input of subsystem  $N_j$ . As said above, in this study, without loss of generality, we do not consider self-loops; that is, every subsystem  $N_i$  is assumed not to be connected to itself, i.e.,  $(N_i, N_i) \notin E$ . Each subsystem  $N_i$  is assumed to admit a DAEs representation as in Equation (7). Thus, the graph incidence matrix can be considered the incidence matrix of the system of DAEs. The subset of DAEs (7) of each subsystem can be solved using the previously described methods applied directly to the subsystem's definition. This method allows joining different methods according to the context (e.g., causal or acausal methods, general numerical methods, etc.) to which each subsystem belongs.

This way, we describe each subsystem as representing independently of any other subsystem in the ISMs. The great advantage is that we can substitute it whenever needed with a different representation (i.e., different DAEs for that subsystem and different solution methods), making maintenance and evolution of the whole ISM easy to implement. Another advantage is that we do not need to know the content of each subsystem. It is sufficient to have a black box with known inputs and outputs: this becomes particularly helpful when we ask an external firm to model a particular component without necessarily asking for open access to its complete content. This representation belongs to the ISM framework

more generally than previously conducted, as each subsystem is modelled and solved individually and independently on other models. The graph structure coordinates all these solvers to determine the whole ISM's dynamical evolution efficiently. Therefore, one of the main novelties of our approach is that each subsystem (or a subgroup of them as shown below) could be solved separately in a proper order based on the structure of the graph, making it possible for the following procedure that simplifies ISM simulation. Precisely, each output of a subsystem (or group of them: T. Xu et al. [55]) becomes the input of the next subsystem in the chosen ordering. Matrix  $M$  can be structured as a block matrix to better describe what just said, each block representing a node (subsystem) and corresponding connections of the subsystem to another one, then represented as the sum of a strictly block lower triangular matrix  $M'$ , and a strictly block upper triangular matrix  $M''$ . This way, if no self-loops are considered then it is the diagonal blocks that are null and not the individual diagonal elements of  $M$ . The same obviously occurs for both  $M'$  and  $M''$ :  $M'$  represents all the feed-forward block connections in the considered node ordering, while  $M''$  represents all feedback block connections. This approach leads to a scheme middle way between pure explicit and pure implicit ones: using only known values (at time step  $t_n$ ) of the subsystems output variables, i.e.,  $y_{n+1}$ , for feed-forward edges (matrix  $M'$ ) and  $y_n$  for feedback edges (i.e., outputs not yet calculated at time step  $t_{n+1}$ : matrix  $M''$ ), the discretized version of the matrix Equation (3b) becomes:

$$v_{n+1} = M' y_{n+1} + M'' y_n \quad (8)$$

Therefore, it is desired to partition the subsystems (i.e., the nodes of the graph representing the ISM) into sequential groups [56], each group containing the minimum number of subsystems. Only feed forward of information exists from each group to the groups that follow (or as many as possible whenever a not empty minimum feedback arc set exists, see discussion below). With this partitioning, the equations of each group can be easily solved simultaneously. The result is used to solve the equations of the following groups in the selected ordering. A minimum number of subsystems is desired in each group because this minimizes the number of equations that must be solved simultaneously. For complex ISM, an algorithm [57,58] is necessary to accomplish the partitioning. No symbolic or other type of manipulation is required.

The partitioning process is only the first step toward a solution and remains to solve each partitioned group's coupled equations. If feedbacks are not altogether cancelled, the corresponding variables are such that it is possible to execute all the subsystems of the group in the selected order if these variables had known values (e.g., from previous time steps or from measured variables in the case of mixed simulation–measurement real-time system). If a group contains non-linear subsystem, then a definitive solution is not generally possible, and iteration procedures must be used separately for each group to reduce the computational effort. These groups can be considered the proper subsystems, while their sub-subsystems become auxiliary ones (see the second example in Section 3.2). Solving ISM without taking advantage of the structure of the equations of such interconnected systems would become highly computationally inefficient. On the other hand, the smallest possible set of variables should be chosen for each partitioned group to reduce the computational effort. The choice of a minimum number of such variables can be shown as equivalent to a common problem in the theory of directed graphs, determining a minimum feedback arc set for each group graph. The following definitions, taken from D.H. Younger [59], will help to make this identification:

- A feedback arc set is a set of arcs that leave the resultant graph circuit free if removed for a directed graph.
- A feedback arc set is a minimum if no other feedback arc set for that graph consists of a smaller number of arcs.
- A sequential ordering  $R$  of a graph of  $N$  nodes is a permutation of the sequence  $1 \dots N$

In general, a pure sequential evaluation of the above type (i.e., empty feedback arc set) is possible only if the graph is free of directed loops. The sorting procedure should act in two steps.

The first step is to partition the system graph nodes into groups as outlined above whenever possible. A preliminary simplification of the system graph is possible, reducing the complexity of the partitioning process: the source nodes and the arcs which leave them (external inputs  $U$ ) can never be iteration variables, as they represent known quantities. Therefore, the source nodes can be considered as the first group of the partition. Similarly, external outputs (e.g., “measured” outputs) can be considered the last group (sink nodes). If possible, the partitioning procedure is applied to the subgraph, called the Reduced ISM (RISM), obtained by deleting source and sink nodes from the original graph.

The second step consists of finding a minimum feedback arc set for each group (sources and sinks not included) which determines a set of iteration variables for each group of equations. Finally, if no partitioning is possible (i.e., there is only one group coincident with the whole ISMs), the node ordering procedure outlined below is applied directly to the RISM.

In regards to nodes ordering, D.H. Younger [59] shows that for any sequential ordering  $R$  of the nodes of the graph, the removal of those arcs  $(i, j)$  for which  $R(i) \geq R(j)$  must eliminate all directed loops, and if all these arcs are selected as iteration variables, then, since no directed loops remain, the RISM can be evaluated in the order determined by  $R$ . Hence, the chosen ordering determines the set of iteration variables. To find a minimum number of such variables, it is therefore desired to find a sequential ordering  $R$  which minimizes the number of arcs  $(i, j)$  for which  $R(i) \geq R(j)$ . This is precisely the problem of determining a minimum feedback arc set. Details can be found in D.H. Younger, 1963, but also other and more efficient algorithms can be found in the literature (e.g., Ref. [60]), giving sometimes optimal but often suboptimal ordering. The general problem is an NP-hard problem [61]. Many suboptimal ordering algorithms have the great advantage of fast execution even only directly proportional to  $N$ . The proposed method is analysed in detail for LTI systems in the next paragraph.

### 2.3. The New Approach to Individual Subsystems Models: Linear Time-Invariant Systems and State-Space Representation

The partitioning method outlined in the previous paragraph is now applied to LTI systems; Equations (3a), (3b) and (4) can be written explicitly. To distinguish vectors from matrices, we use uppercase letters for latter and lowercase letters for the former (e.g.,  $x$  instead of the  $X$  symbol used in the previous paragraph):

$$\dot{x} = Ax + Bv + Pu, \quad (9a)$$

$$y = Cx, \quad (9b)$$

$$v = My, \quad (9c)$$

where matrices  $A$ ,  $B$ , and  $C$  are block diagonal, each block representing each subsystem in the ISM: they can also depend on time and in this case the system is not autonomous, making the discussion only more complicated. Matrix  $P$  represents the dependence of the status vector  $x$  on the external input vector  $u$ . Moreover, the explicit dependence of internal outputs  $y$  on internal inputs  $v$  and external inputs  $u$  has been neglected to simplify the following calculations without loss of generality, but it can be easily reintroduced (for example for subsystems without ODEs but only algebraic input–output blocks:  $y_i = D_{ii}u_i + E_{ii}v_i$  for the  $i$ -th subsystem). Regarding external outputs, they can be chosen without loss of generality among the components of the internal output vector  $y$ .

Substituting the last two Equations (9b) and (9c), into the first one (9a) we obtain the following system of equations that can be solved by calculating  $x(t)$  as all the matrices and external inputs  $u$  are known

$$\dot{x} = Kx + Pu, \quad (10)$$



where the matrix  $K = A + BMC$  corresponds to the state transition matrix (precisely for LTI systems, the state transition matrix is  $e^{Kt}$ ). The system in Equation (10) is said asymptotically stable (AS) if for each  $\varepsilon > 0$  there is  $\delta(\varepsilon)$  so that for  $u = 0$  and a suitable norm  $\|\cdot\|$  we have

$$\|x(0)\| < \delta \Rightarrow \begin{cases} \|x(t)\| < \varepsilon \quad \forall t > 0 \\ \lim_{t \rightarrow \infty} \|x(t)\| = 0 \end{cases} \quad (11)$$

The system is AS if and only if the real parts of the eigenvalues  $\lambda(K)$  of matrix  $K$  are all negative:  $\lambda(K) < 0$ . If the sequence of discrete-time points  $t_n$  is equally spaced ( $t_n = n\Delta t$ ) the discrete version of system 10 can be obtained approximating  $\dot{x}$  as follows

$$\dot{x} \sim \frac{x_{n+1} - x_n}{\Delta t}, \quad (12)$$

where  $x_n = x(t_n)$ . With no loss of generality, other approximations may be used to improve convergence, especially when dealing with stiff problems. Moreover, variable time steps can also be used. There are different ways of treating 10: explicitly or implicitly [49]. Using the Euler's scheme, in the first case, we calculate the right side at time  $t_n$  obtaining

$$x_{n+1} = (I + \Delta t K)x_n + \Delta t P u_n. \quad (13)$$

This scheme is stable if the most significant absolute value  $\rho(I + \Delta t K)$  of the eigenvalues of the matrix  $I + \Delta t K$  (the spectral radius) is less than one for AS systems when

$$\Delta t < \frac{2}{\rho(K)}. \quad (14)$$

In the second case (implicit scheme), we calculate the right side of system (10) at time  $t_{n+1}$  obtaining:

$$x_{n+1} = (I - \Delta t K)^{-1} x_n + (I - \Delta t K)^{-1} \Delta t P u_n. \quad (15)$$

Setting  $H^{-1} = (I - \Delta t K)^{-1}$ , the one-step discrete-time transition matrix, the stability condition becomes  $\rho[H^{-1}] < 1$  and it is valid for any matrix  $K$  having  $\text{Re}[\lambda(K)] < 0$ ; therefore, the scheme is unconditionally stable for AS systems. The above schemes do not directly consider the structure of the system, and different procedures are generally used to reduce the computational effort as described in the second paragraph of this paper. The scheme sketched in the previous paragraph for non-linear systems that take advantage of the sorting procedure is now considered. As said in the previous section about general non-linear ISMs, Matrix  $M$  is the sum of a strictly block lower triangular matrix  $M'$ , and a strictly block upper triangular matrix  $M''$ :  $M'$  for all feed-forward connections in the considered node ordering and  $M''$  all feedback connections.

When optimal (or sub optimal) ordering is obtained,  $M''$  is as sparse as possible (or close enough), and a scheme middle way between pure explicit and pure implicit ones can be outlined, this time starting from system (9). Matrix Equation (8) is repeated here specialized for system (9)

$$v_{n+1} = M' y_{n+1} + M'' y_n. \quad (16)$$

Therefore, Equation (9b) can be written at time  $t_n$  using  $y_n = Cx_n$ , and at time  $t_{n+1}$  using  $y_{n+1} = Cx_{n+1}$ . Substituting these two conditions into (16), it becomes

$$v_{n+1} = M' C x_{n+1} + M'' C x_n. \quad (17)$$

Finally, the discrete version of Equation (9a) (Implicit Euler's scheme as regards the proper terms):

$$\frac{x_{n+1} - x_n}{\Delta t} = A x_{n+1} + B v_{n+1} + P u_{n+1} \quad (18)$$

becomes

$$\frac{x_{n+1} - x_n}{\Delta t} = Ax_{n+1} + BM'Cx_{n+1} + BM''Cx_n + Pu_{n+1} \quad (19)$$

which is solved to obtain

$$x_{n+1} = H_1^{-1}x_n + (I - \Delta tK')^{-1}\Delta tPu_{n+1} \quad (20)$$

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

### 3. Results and Discussion

#### 3.1. Stability of the Proposed Algorithm

The scheme in (20) is stable if two conditions are satisfied, one for each term ( $H_1^{-1}x_n$  and  $(I - \Delta tK')^{-1}\Delta tPu_{n+1}$ ) that is on  $H_1^{-1}$ , the one-step state-to-state transition matrix, and on  $(I - \Delta tK')^{-1}\Delta tP$ , the input-to-state transition matrix:

- $\rho[H_1^{-1}] < 1$ .
- $K'$  must have eigenvalues with negative real part; this means that each subsystem must be stable, being  $K'$  is the state transition matrix of the ISM with feedback edges deleted. If any one of the subsystems is unstable and is stabilized using an output-feedback stabilization method and deleting the corresponding feedback, the resulting feed-forward-only system becomes unstable again.

As regards the first condition, matrix  $BM''C$  is the product of block diagonal matrices with  $M''$ , a strictly upper block triangular matrix, giving a strictly upper block triangular matrix; such matrix is nilpotent,  $(BM''C)^r = 0$ ,  $r$  is its rank. This property allows for quickly calculating the inverse of  $I + \Delta tBM''C$ :

$$(I + \Delta tBM''C)^{-1} = I + \sum_{i=1}^{r-1} (-1)^i (\Delta tBM''C)^i. \quad (21)$$

Using (21) in  $H_1$  gives:

$$\begin{aligned} H_1 &= (I + \Delta tBM''C)^{-1}(I - \Delta tK') =, \\ &= I - \Delta tK' - \Delta tBM''C - \Delta t^2BM''CK' + [\sum_{i=2}^{n-1} (-1)^i (\Delta tBM''C)^i](I - \Delta tK') =, \\ &= I - \Delta t(A + BM'C + BM''C) - \Delta t^2BM''CK' + [\sum_{i=2}^{n-1} (-1)^i (\Delta tBM''C)^i](I - \Delta tK') =, \\ &= I - \Delta tK + O(\Delta t^2), \end{aligned} \quad (22)$$

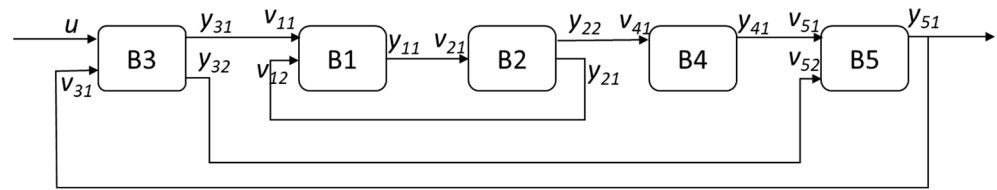
and finally,

$$H_1 = H + O(\Delta t^2). \quad (23)$$

Therefore, the eigenvalues of the perturbed matrix  $H_1$  are a perturbed version of those of  $H$  (Classic Euler implicit, Equation (13)), maintaining greater than one of their magnitudes (therefore less than one the magnitude of the eigenvalues of  $H_1^{-1}$ ) if the second-order perturbation is sufficiently small. In any case, Euler's implicit method, from which comes the method proposed here, is only of the first order and could be improved for accuracy reasons. The step size  $\Delta t$  should be anyway small enough to meet a certain error tolerance independently of stability conditions. For stiff systems, higher accuracy can be achieved using a higher-order methods and/or variable step size  $\Delta t_k$  while still maintaining stability [62].

#### 3.2. Numerical Tests on Linear Time-Invariant Individual System Models

As the first test, we consider a stable linear ISM made of five stable linear subsystems connected as in Figure 2 where the subsystems (nodes) are numbered casually.



**Figure 2.** Graphical scheme of the first LTI ISM tested: here, each subsystem is represented as a block (B1 ... B5) as they correspond to the submatrices blocks in the system of equations.

There is only one external input ( $u$ ) to subsystem B3 and only one external output coinciding with the output  $y_{51}$  from subsystem B5. There are two best node orderings, i.e., with the minimum feedback arc set: (B3, B1, B2, B4, and B5) and (B2, B4, B3, B5, and B1). The first node ordering will be analysed in the next example. In any case, because of the stability of all subsystems, the two ordering sets are equivalent. For now, let us choose the second node ordering: the corresponding matrices of the ISM, modified after nodes ordering, are shown below where each matrix has been partitioned into blocks, each representing one subsystem according to the minimum-feedback-arc-set ordering:

										$u$	
										1	
										0	
										0	
A =	-6	B2 -2.75	-1.5	0	0	0	0	0		P =	0
	4	0	0	0	0	0	0	0	B2		0
	0	1	0	0	0	0	0	0			0
	0	0	0	-1	0	0	0	0	B4		0
	0	0	0	0	-3	2	0	0			0
	0	0	0	0	2	-3	0	0	B3		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	
										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	
										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	
										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	
										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	
										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	
										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	
										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	
										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	
										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	
										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	
										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	
										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	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	

$$\begin{aligned}
 M &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} = \\
 &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\
 &= M' + M''
 \end{aligned}$$

$$K = A + B M C = \begin{bmatrix} & B2 & & B4 & & B3 & & B5 & & B1 \\ -6 & -2.75 & -1.5 & 0 & 0 & 0 & 0 & 0 & -4 & \\ 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & B2 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \\ -1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & B4 \\ 0 & 0 & 0 & 0 & -3 & 2 & -1 & 0 & 0 & \\ 0 & 0 & 0 & 0 & 2 & -3 & -1 & 0 & 0 & B3 \\ 0 & 0 & 0 & -1 & 2 & -1 & -2 & 0 & 0 & B5 \\ 0 & 0 & 2.6875 & 0 & -1 & 1 & 0 & -7 & 0 & B1 \end{bmatrix}$$

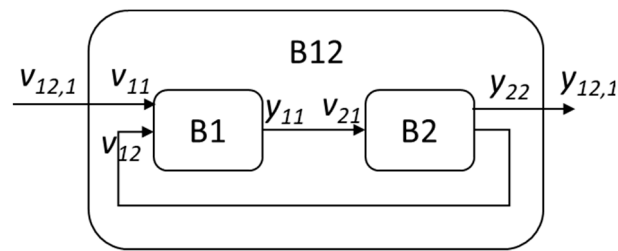
As said, both subsystems and the whole ISM are asymptotically stable as the real parts of the eigenvalues of matrix  $A$  (they coincide with the diagonal blocks' eigenvalues), and those of matrix  $K$  are negative. Considering the solution scheme (20), matrix  $K'$  has negative-real-part eigenvalues and the magnitude of all the eigenvalues of  $H_1^{-1}$  is less than one for sampling time  $\Delta t$  less than the most significant time constant of the original system.

A second example consists of a system very similar to the previous one but substituting subsystem B2 with an unstable one:

$$A = \begin{bmatrix} & B2 & & B4 & & B3 & & B5 & & B1 \\ -3 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \\ 2 & -3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & B2 \\ 0 & 0 & -7 & 0 & 0 & -5.375 & 0 & 0 & 0 & \\ 0 & 0 & 4 & -4 & -1 & 3 & 0 & 0 & 0 & B4 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & B3 \\ 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & B5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & B1 \end{bmatrix}$$

In this case, while the magnitude of all the eigenvalues of  $H^{-1}$  is undoubtedly less than one (the same occurs for  $H_1^{-1}$  for sampling time  $\Delta t < 0.151$ , that is the scheme that is stable for any initial condition and  $\Delta t < 0.151$ ), matrix  $K'$  has one eigenvalue with the positive real part; for any input  $u$ , the scheme becomes unstable. This is due, as said in the

previous paragraph about matrix  $K'$ , to the fact that the only-feed-forward system (i.e., with feedback arcs deleted) is unstable whenever we are involved with unstable subsystems. The feedback from the output  $y_{21}$  of subsystem B2 to input  $v_{12}$  of subsystem B1, and subsystem B1 itself stabilizes system B2: deleting this feedback it becomes unstable. Nevertheless, this is simply due to a modelling error: in fact, subsystems B1-B2 must be modelled as a one-subsystem as shown in Figure 3: system B12 represents a proper subsystem while subsystems B1 and B2 are only auxiliary ones used to simplify the graphical representation of B12 but not necessary in the whole ISMs.



**Figure 3.** The graphical scheme of the proper subsystem B12 obtained joining the auxiliary subsystems B1 and B2.

Inserting B12 in the ISM of Figure 2 instead of the couple B1-B2, the best ordering becomes (B3, B12, B4, B5). Matrix  $A$  becomes

		B3			B12			B4	B5	
A =		−3	2	0	0	0	0	0	0	B3
		2	−3	0	0	0	0	0	0	
		0	0	−7	0	0	−5.375	0	0	B12
		0	0	4	−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	B4
		0	0	0	0	0	0	0	−2	B5

It is easy to show that for this system, matrices  $K'$  and  $H_1^{-1}$  satisfy the two stability conditions.

The above results show how to effectively implement the proposed method both for stable and stabilized linear systems assessing and validating it at the same time. In fact, they represent a good canvas for any other kind of linear interconnected systems, especially for more complex ones as proposed in the following study of a refrigeration plant with heat recovery typically used in the food industry.

### 3.3. Study of an Industrial Refrigeration Plant

The studied process consists of a refrigeration plant, no matter the adopted process (in this case vapour compression cycle) with thermal waste recovery; the used Cold Fluid (e.g., glycol water, refrigerant, etc.) is refrigerated, sent to a cold process (e.g., a cold room, liquid fluid heat exchanger, etc.), and recirculated in the refrigeration system. The whole refrigeration plant consists of two main sections: a cold section (as briefly described above) and a warm section corresponding to the higher temperature level of the thermodynamic cycle. In this case, the cold section extracts heat from the Cold Process fluid (CPf) using water for refrigeration temperatures above 0 °C. Depending on the choice made on the thermodynamic cycle, the warm section must dissipate both the heat extracted from the cold fluid and the compressor work-equivalent heat: this is the waste heat to be recovered. It can be dissipated directly into the external environment through a fan coil unit or recovered into a secondary warm/hot fluid (HF), mainly water (W) (e.g., water preheating in liquid pasteurization as it is supposed consistently below 100 °C) used to heat the Hot Process



fluid (HPf). Both the CPf and the HPf are stored in insulated Cold and Hot Water Tanks (CWT, HWT). Moreover, warm water from RP and HP is stored in a Warm Water Tank (WWT) and recirculated: a control algorithm is supposed to ensure that the temperature  $T_{WT}$  of the WWT is always below the temperature of the refrigerant entering the expansion valve in the RP. As said the heat waste is used to pre-heat the warm fluid so an auxiliary Boiler is needed between the water tank and a final hot process to rise the water temperature to desired final temperature. In Figure 4, an outline of the chosen plant is shown with the primary input–output variables, pipes and pumps included in the model description below.

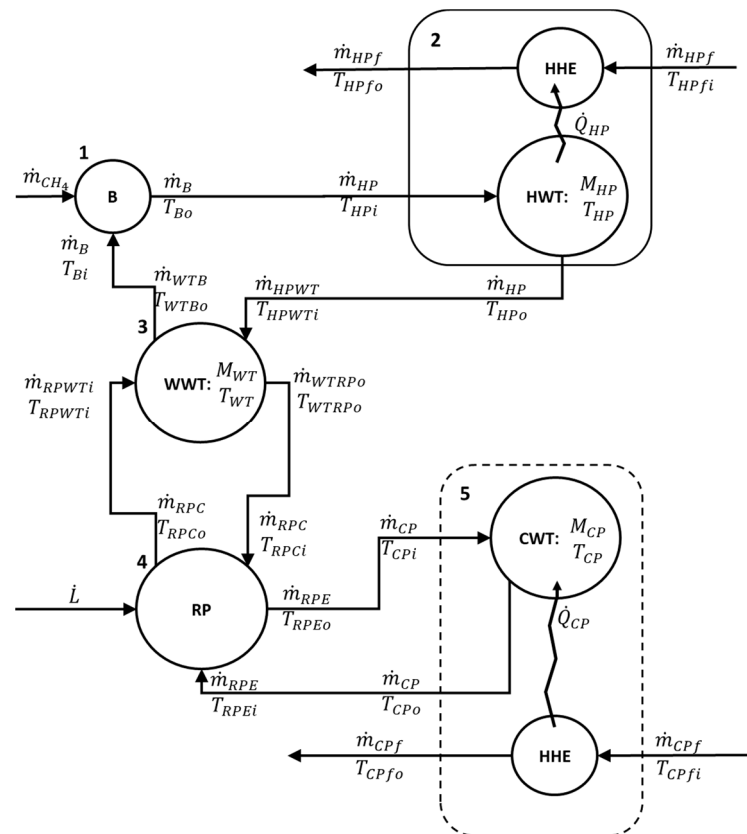


Figure 4. Outline of the simulated plant.

External inputs:

- $T_E$ : the external environment temperature; it is repeated for all subsystems,
- $\dot{m}_{HPf}$  and  $T_{HPfi}$ : flow rate and temperature of the HP fluid entering in the hot process,
- $\dot{L}$ : the power furnished to the refrigerant compressor in the refrigeration plant (remember that we are supposing a vapor compression cycle),
- $\dot{m}_{CPf}$   $T_{CPfi}$ : flow rate and temperature of the CP fluid entering in the cold process,
- $\dot{m}_{CH_4}$ : the methane flow rate feeding the Boiler (B).

External outputs:

- $\dot{T}_{HPfo}$ : flow rate and temperature of the HP fluid leaving the hot process,
- $T_{CPfo}$ : flow rate and temperature of the CP fluid leaving the cold process.

All the processes are interconnected through directed lines, representing mass and energy flows. Thus, the whole plant can be represented as a directed graph where the processes are the nodes and flows are the directed edges of the graph. Output pipes are included in the model equations.

Moreover, all tanks (HWT, WWT, and CWT) are controlled such that the mass in each tank remains constant during the whole process. When the hypothesis of constant level in

the three tanks is removed, some of the process equations become non-linear Differential Algebraic Equations (DAEs).

Now we are ready to assess equations and process variables for each subsystem independently on how they are connected.

### 3.3.1. Mathematical Models of Each Machine in the Chosen Plant System Boiler (B) and Corresponding Output Pipe

The water coming into the boiler from the WWT is heated by burning methane. The governing equation is

$$\text{BoilerB} : T_{Bo} = (1 - \eta_{PB}) \left( T_{Bi} + \frac{\eta_B h_{fCH_4}}{\dot{m}_{Bi} c_w} \dot{m}_{CH_4} \right) + \eta_{PB} T_E$$

where  $\dot{m}_B$ ,  $T_{Bi}$  and  $T_{Bo}$  are, respectively, the mass flow rates and the temperatures of the water entering the boiler and leaving the output pipe,  $\eta_{PB}$  is the pipe heat exchange efficiency (the smallest possible if the pipe is well insulated; this is true for all pipes and the same efficiency is used here),  $\eta_B$  is the boiler efficiency,  $h_{fCH_4}$  is the Lower Heating Value (LHV) of methane,  $c_w$  is the specific heat of water, and  $T_E$  is the external temperature.

In terms of internal input  $v_1 = T_{Bi}$ , external inputs  $u_1 = T_E$  and  $u_2 = \dot{m}_{CH_4}$ , and output  $y_1 = T_{Bo}$  (in this case there are no ODEs, so no state variable is needed; anyway is useful to add a dummy constant state  $x_1 = x_1^{(0)}$ ), the above equations become:

$$\text{Dummystateequation} : \dot{x}_1 = 0$$

$$\text{BoilerB} : y_1 = (1 - \eta_{PB})v_1 + \eta_{PB}u_1 + (1 - \eta_{PB})\frac{\eta_B h_{fCH_4}}{\dot{m}_B c_w}u_2$$

### Hot Process (HP) and Corresponding Hot Water Tank (HWT) and Output Pipe

In the HP, the HPf enters the process (we can think it as a heat exchanger even if here it is not considered the heat transfer mechanism) and it is heated (transferred heat:  $\dot{Q}_{HP}$ ) using the hot water coming from the Boiler. This water is, therefore, cooled (it resides in HWT of total mass  $M_{HP}$ ) and it leaves the HP going to the output pipe. The HWT is considered a perfectly insulated tank (as well as the CWT, see below). The governing equations are

$$\text{HWT} : \frac{dT_{HP}}{dt} = -\frac{\dot{m}_{HP}}{M_{HP}}T_{HP} + \frac{\dot{m}_{HP}}{M_{HP}}T_{HPi} - \frac{\dot{Q}_{HP}}{M_{HP}c_w},$$

$$\text{HP} : \eta_{HPf}\dot{m}_{HPf}c_{HPf}(T_{HP} - T_{HPfi}) = \dot{Q}_{HP} = \dot{m}_{HPf}c_{HPf}(T_{HPfo} - T_{HPfi}),$$

$$\text{HWT output pipe} : T_{HPo} = (1 - \eta_{PHP})T_{HP} + \eta_{PHP}T_E,$$

where  $\dot{Q}_{HP}$  is the heat exchanged between the hot water and the HPf,  $\dot{m}_{HP}$ ,  $T_{HPi}$  and  $T_{HPo}$  are respectively the mass flow rate and the temperatures of the water entering the HP and leaving the output pipe,  $T_{HP}$  the temperature of the water leaving the HP (equal to that of the water in the HWT) and entering the output pipe,  $\dot{m}_{HPf}$ ,  $c_{HPf}$ ,  $T_{HPfi}$  and  $T_{HPfo}$  are respectively the mass flow rate, the specific heat and the temperatures of the HPf entering and leaving the HP,  $\eta_{PHP}$  the output pipe heat exchange efficiency,  $\eta_{HPf}$  the HP heat exchanger efficiency.

In terms of state  $x_2 = T_{HP}$ , internal input  $v_2 = T_{HPi}$ , external inputs  $u_1 = T_E$  and  $u_3 = T_{HPfi}$  and outputs  $y_2 = \begin{bmatrix} T_{HPfo} \\ T_{HPo} \end{bmatrix}$  the above equations become

$$\text{HWT} : \dot{x}_2 = -\left( \frac{c_w \dot{m}_{HP} + \eta_{HPf} \dot{m}_{HPf} c_{HPf}}{M_{HP} c_w} \right) x_2 + \frac{\dot{m}_{HP}}{M_{HP}} v_2 + \frac{\eta_{HPf} \dot{m}_{HPf} c_{HPf}}{M_{HP} c_w} u_3.$$

$$HP : y_{21} = \eta_{HPf}x_2 + (1 - \eta_{HPf})u_3.$$

$$HWT \text{ output pipe} : y_{22} = (1 - \eta_{PHP})x_2 + \eta_{PHP}u_1.$$

### Warm Water Tank (WWT) and Corresponding Output Pipes

In the WWT (of total mass  $M_{WT}$  and temperature  $T_{WT}$ ), there are two streams of water, respectively, on HP and RP sides; the water coming from the HP side is cooler than water in the WT, so it is a cooling fluid for the WWT, while the water coming from the RP is warmer than the water in the WWT, and it is a heating fluid for the WT. Differently from the HWT and the CWT, the WWT is considered not perfectly insulated. The governing equations are

$$\begin{aligned} WWT : \frac{d(T_{WT})}{dt} &= \frac{\dot{m}_{HPWT}}{M_{WT}} T_{HPWTi} + \frac{\dot{m}_{RPWTi}}{M_{WT}} T_{RPWTi} - \frac{\dot{m}_{WTB}}{M_{WT}} T_{WT} - \frac{\dot{m}_{WTRP0}}{M_{WT}} T_{WT} + \frac{U_{WT}A_{WT}}{M_{WT}c_w} (T_E - T_{WT}) = \\ &= -\frac{c_w\dot{m}_{WTB} + c_w\dot{m}_{WTRP0} + U_{WT}A_{WT}}{M_{WT}c_w} T_{WT} + \frac{\dot{m}_{HPWT}}{M_{WT}} T_{HPWTi} + \frac{\dot{m}_{RPWTi}}{M_{WT}} T_{RPWTi} + \frac{U_{WT}A_{WT}}{M_{WT}c_w} T_E \end{aligned}$$

$$\text{Output pipes} \begin{cases} T_{WTB0} = (1 - \eta_{PWTB})T_{WT} + \eta_{PWTB}T_E \\ T_{WTRP0} = (1 - \eta_{PWTRP})T_{WT} + \eta_{PWTRP}T_E \end{cases}.$$

where  $\dot{m}_{HPWT}$ ,  $T_{HPWTi}$  and  $T_{WTB0}$  are respectively the mass flow rate and the temperatures of the water entering the WWT and leaving the output pipe (Boiler and Hot Process side),  $\dot{m}_{RPWT}$ ,  $T_{RPWTi}$  and  $T_{WTRP0}$  are respectively the mass flow rate and the temperatures of the water entering the WWT and leaving output pipe (Refrigeration Plant side),  $\eta_{PWTB}$  and  $\eta_{PWTRP}$  the heat exchange efficiencies of the two output pipes,  $U_{WT}$  and  $A_{WT}$  the global heat exchange coefficient and external surface of the WWT, respectively.

In terms of state  $x_3 = T_{WT}$ , internal inputs  $v_3 = \begin{bmatrix} T_{HPWTi} \\ T_{RPWTi} \end{bmatrix}$ , external input  $u_1 = T_E$

and outputs  $y_3 = \begin{bmatrix} T_{WTB0} \\ T_{WTRP0} \end{bmatrix}$  from the above equations become:

$$WWT : \dot{x}_3 = -\left(\frac{c_w\dot{m}_{WTB} + c_w\dot{m}_{WTRP0} + U_{WT}A_{WT}}{M_{WT}c_w}\right)x_3 + \frac{\dot{m}_{HPWT}}{M_{WT}}v_{31} + \frac{\dot{m}_{RPWTi}}{M_{WT}}v_{32} + \frac{U_{WT}A_{WT}}{M_{WT}c_w}u_1.$$

$$\text{Output pipes} : \begin{cases} y_{31} = (1 - \eta_{PWTB})x_3 + \eta_{PWTB}u_1 \\ y_{32} = (1 - \eta_{PWTRP})x_3 + \eta_{PWTRP}u_1 \end{cases}$$

### Refrigeration Plant (RP) and Corresponding Output Pipe

In the RP, a vapor compression cycle is operated to refrigerate the water coming from the CP. The heat transferred from water to the RP refrigerant (not specified here; the detailed process is not considered here) and the work is furnished to the compression cycle heat in the water coming from the WWT. The governing equations are

$$\text{Condenser side} : T_{RPC0} = (1 - \eta_{PC})\left(T_{RPCi} + \frac{(1 + COP)\dot{L}}{\dot{m}_{RPC}c_w}\right) + \eta_{PC}T_E.$$

$$\text{Evaporator side} : T_{RPE0} = (1 - \eta_{PE})\left(T_{RPEi} - \frac{COP}{\dot{m}_{RPE}c_w}\dot{L}\right) + \eta_{PE}T_E,$$

where  $\dot{m}_{RPC}$ ,  $T_{RPCi}$  and  $T_{RPC0}$  are respectively the mass flow rate and the temperatures of the water entering the RP and leaving the output pipe (condenser side),  $\dot{m}_{RPE}$ ,  $T_{RPEi}$  and  $T_{RPE0}$  are respectively the mass flow rate and the temperatures of the water entering and leaving the RP (evaporator side),  $\eta_{PC}$  and  $\eta_{PE}$  the heat exchange efficiencies of the output pipes,  $COP$  the Coefficient of Production of the RP.

In terms of internal input  $v_4 = \begin{bmatrix} T_{RPCi} \\ T_{RPEi} \end{bmatrix}$ , external inputs  $u_1 = T_E$  and  $u_4 = \dot{L}$  and outputs  $y_4 = \begin{bmatrix} T_{RPCo} \\ T_{RPEo} \end{bmatrix}$  (in this case there are no ODEs, so no state variable is needed; anyway, it is useful to add a dummy constant state  $x_4 = x_4^{(0)}$ ) from the above equations become

$$\text{Dummy state equation : } \dot{x}_4 = 0.$$

$$\text{Condenser side : } y_{41} = (1 - \eta_{PC})v_{41} + \frac{(1 + \text{COP})}{\dot{m}_{RPC}c_w}u_4 + \eta_{PC}u_1.$$

$$\text{Evaporator side : } y_{42} = (1 - \eta_{PE})v_{42} - \frac{\text{COP}}{\dot{m}_{RPE}c_w}u_4 + \eta_{PE}u_1.$$

### Cold Process (CP) and Output Pipe

In the CP, the cold water refrigerates the CPf (transferred heat:  $\dot{Q}_{CP}$ ). This water is, therefore, heated (it resides in a tank of the total mass  $M_{CP}$  and temperature  $T_{CP}$ ) and it leaves the CP going to the output pipe. The governing equations are

$$\text{CWT : } \frac{dT_{CP}}{dt} = -\frac{\dot{m}_{CP}}{M_{CP}}T_{CP} + \frac{\dot{m}_{CP}}{M_{CP}}T_{CPi} + \frac{\dot{Q}_{CP}}{M_{CP}c_w}.$$

$$\text{CP : } \eta_{CPf}\dot{m}_{CPf}c_{PCPf}(T_{CP} - T_{CPfi}) = -\dot{Q}_{CP} = \dot{m}_{CPf}c_{PCPf}(T_{CPfo} - T_{CPfi}).$$

$$\text{CWT output pipe : } T_{CPo} = (1 - \eta_{PCP})T_{CP} + \eta_{PCP}T_E.$$

where  $\dot{m}_{CP}$ ,  $T_{CPi}$  and  $T_{CPo}$  are, respectively, the mass flow rate and the temperatures of the water entering the CP and leaving the output pipe (RP side),  $\dot{m}_{CPf}$ ,  $T_{CPfi}$  and  $T_{CPfo}$  are respectively the mass flow rate and the temperatures of the CPf entering and leaving the CP (CPf side),  $\eta_{PCP}$  the output pipe heat exchange efficiency (RP side) and  $\eta_{CPf}$  CP heat exchanger efficiency (CPf side).

In terms of state  $x_5 = T_{CP}$ , internal input  $v_5 = T_{CPi}$ , external inputs  $u_1 = T_E$  and  $u_5 = T_{CPfi}$  and outputs  $y_5 = \begin{bmatrix} T_{CPfo} \\ T_{CPo} \end{bmatrix}$  from the above equations become

$$\text{HWT : } \dot{x}_5 = -\left(\frac{c_w\dot{m}_{CP} + \eta_{CPf}\dot{m}_{CPf}c_{PCPf}}{M_{CP}c_w}\right)x_5 + \frac{\dot{m}_{CP}}{M_{CP}}v_5 + \frac{\eta_{CPf}\dot{m}_{CPf}c_{PCPf}}{M_{CP}c_w}u_5.$$

$$\text{HP : } y_{51} = \eta_{CPf}x_5 + (1 - \eta_{CPf})u_5.$$

$$\text{HWT output pipe : } y_{52} = (1 - \eta_{PCP})x_5 + \eta_{PCP}u_1.$$

### Solution of the Whole Process Equations

Another condition is generally hypothesized as the no-delay condition; pumps' response time is small compared with controlled parameters variation, and the work to operate them is neglected. With the hypothesis above, we conclude that the flow rate does not vary within each circuit, so we have  $\dot{m}_{HPWT} = \dot{m}_{WTB}$  and  $\dot{m}_{RPWTi} = \dot{m}_{WTRPo}$  leading to

$$\dot{m}_{HPWT} = \dot{m}_{WTB} = \dot{m}_B = \dot{m}_{HP} = \dot{m}_H,$$

$$\dot{m}_{RPWTi} = \dot{m}_{WTRPo} = \dot{m}_{RPC} = \dot{m}_W,$$

$$\dot{m}_{RPE} = \dot{m}_{RPC} = \dot{m}_{CP} = \dot{m}_C.$$

Moreover, the presence of Algebraic Equations leads to the following system of DAEs:

$$\begin{cases} \dot{x} = Ax + Bv + Pu \\ y = Cx + Du + Ev, \\ v = My \end{cases}$$

where:

$$x = \begin{pmatrix} x_1^{(0)} \\ T_{HP} \\ T_{WT} \\ x_4^{(0)} \\ T_{CP} \end{pmatrix} \quad u = \begin{pmatrix} T_E \\ \dot{m}_{CH_4} \\ T_{HPfi} \\ \dot{L} \\ T_{CPfi} \end{pmatrix},$$

$$v = \begin{pmatrix} v_1 \\ v_2 \\ v_{31} \\ v_{32} \\ v_{41} \\ v_{42} \\ v_5 \end{pmatrix} = \begin{pmatrix} T_{Bi} \\ T_{HPi} \\ T_{HPWTi} \\ T_{RPWTi} \\ T_{RPCi} \\ T_{RPEi} \\ T_{CPi} \end{pmatrix} \quad y = \begin{pmatrix} y_1 \\ y_{21} \\ y_{22} \\ y_{31} \\ y_{32} \\ y_{41} \\ y_{42} \\ y_{51} \\ y_{52} \end{pmatrix} = \begin{pmatrix} T_{Bo} \\ T_{HPfo} \\ T_{HPo} \\ T_{WTBo} \\ T_{WTRPo} \\ T_{RPCo} \\ T_{RPEo} \\ T_{CPfo} \\ T_{CPo} \end{pmatrix}.$$

With the constant external input  $u$ , we have the following analytical solution

$$\begin{cases} x = e^{Kt}(x_0 + K^{-1}P_1u) - K^{-1}P_1u \\ y = (I - EM)^{-1}(Cx + Du) \end{cases},$$

with:

$$K = A + BM(I - EM)^{-1}C \\ P_1 = P + BM(I - EM)^{-1}D.$$

$K$  contains two null rows and columns as two subsystems do not contain ODEs:

	B1	B2	B3	B4	B5	
	0	0	0	0	0	B1
	0	$-\left(\frac{c_w \dot{m}_H + \eta_{PHPF} \dot{m}_{HPf} c_{HPf}}{\dot{M}_{HP} c_w}\right)$	$\frac{\dot{m}_H (1 - \eta_{PWTB}) (1 - \eta_{PB})}{\dot{M}_{HP}}$	0	0	B2
K =	0	$\frac{\dot{m}_H (1 - \eta_{PHF})}{\dot{M}_{WT}}$	$\frac{\dot{m}_W (1 - \eta_{PWTRP}) (1 - \eta_{PC})}{\dot{M}_{WT}} - \left(\frac{c_w \dot{m}_H + c_w \dot{m}_W + U_{WT} A_{WT}}{\dot{M}_{WT} c_w}\right)$	0	0	B3
	0	0	0	0	0	B4
	0	0	0	0	$\frac{\dot{m}_C (1 - \eta_{PCP}) (1 - \eta_{PE})}{\dot{M}_{CP}} - \left(\frac{c_w \dot{m}_C + \eta_{CPf} \dot{m}_{CPf} c_{CPf}}{\dot{M}_{CP} c_w}\right)$	B5

To solve the system of DAEs, it must be reduced to a system of three ODEs in three state variables ( $x_r = T_{HP}, T_{WT}, T_{CP}$ ). This is achieved by just deleting the null rows and columns, but this is a small system, or by symbolic manipulation. Further manipulation ensures that the  $3 \times 3$  reduced matrix  $K_r$  has negative real eigenvalues.

In Table 1, the values of all the constants used in the simulation are shown, while the state variables have the same initial value:  $T_{HP} = T_{WT} = T_{CP} = 40^\circ\text{C}$ .



**Table 1.** Constants used in the simulation.

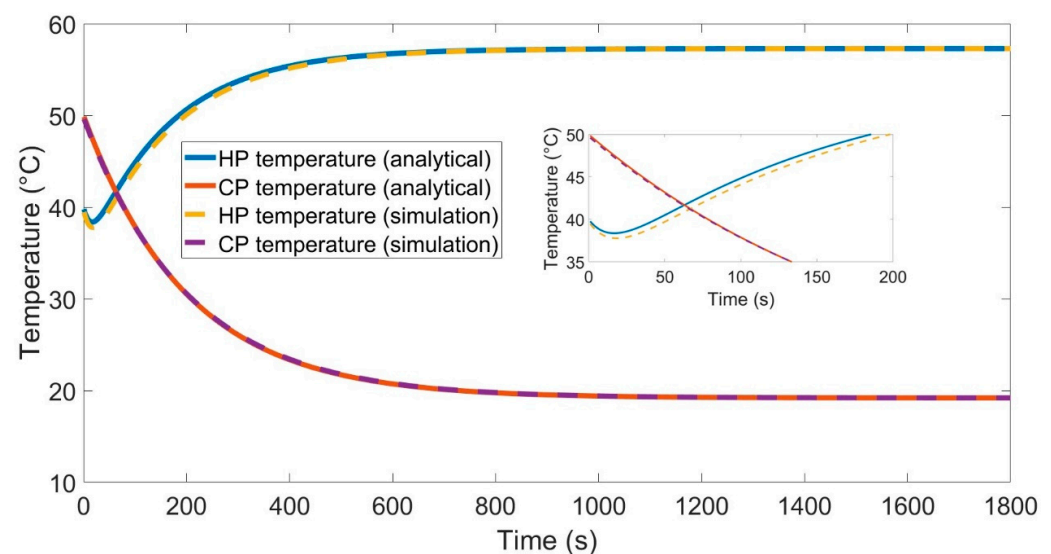
Parameters	Values	Parameters	Values
$\dot{m}_H$	2 kg/s	$c_w = c_{CPf} = c_{HPf}$	4.187 kJ/kgK
$\dot{m}_W$	2 kg/s	$h_{fch4}$	50,000 kJ/kg
$\dot{m}_C$	2 kg/s	$\eta_B = \eta_{CPf} = \eta_{HPf}$	0.98
$M_{HP}$	100 kg	$COP$	3
$M_{WT}$	50 kg	$T_E$	25 °C
$M_{CP}$	200 kg	$\dot{m}_{CH4}$	0.0001 kg/s
$\eta_P$	0.01	$\dot{L}$	43 kW
$U_{WT}$	0 kW/m <sup>2</sup> K	$A_{WT}$	-

In this case, the explicit form of the analytical solution is

$$x = \begin{cases} T_{HP} = 6.03 - 23.31 e^{-0.00628 t} + 6.032 e^{-0.0637 t} \\ T_{WT} = 77.81 - 27.64 e^{-0.00628 t} - 10.17 e^{-0.0637 t} \\ T_{CP} = 19.19 + 30.81 e^{-0.005 t} \end{cases}$$

### 3.3.2. Simulation Results

The simulation was executed using a time step  $\Delta t = 1$  s, while the number of time steps  $n_t = 1800$ . Figure 5 shows the trend of the solutions (analytical and simulated), respectively, for  $T_{HPf0}$  and  $T_{CPf0}$

**Figure 5.** Trend of the temperatures in the Hot and Cold Processes: analytical and simulated results are compared.

It must be highlighted that the solution obtained with the proposed numerical technique is very close to the analytical one, even just a little less close to it if compared with the fully implicit one. The absolute error is less than 2% for  $T_{HPf0}$  and less than 0.5% for  $T_{WT}$ .

### 3.4. Performance Analysis

Now we are ready to compare the other methods (Full implicit and Full explicit schemes) regarding stability conditions when unstable subsystems are present (but, of course, stabilized by suitable feedback), the possibility of a block substitution, and the possibility of choosing a particular node ordering in solving each subsystem.

In particular, we have seen that even if Full Implicit schemes are unconditionally stable (see Table 2), they do not allow the substitution of a subsystem. We may wish to substitute the unstable subsystem B2 of the second example in the previous paragraph,

e.g., only modifying some internal parameters, with the stable system B2 of the first example. However, this needs to change the iterative algorithm as it must consider all the equations together that change at every substitution. A fully Explicit algorithm indeed allows both node ordering (but with no preference for optimality) and block substitution as subsystems are solved separately. However, it becomes unstable in the presence of unstable subsystems as feedback is entirely cancelled. On the other hand, the proposed new method allows both optimal (suboptimal) ordering, speeding up the solving algorithm if compared with Full Implicit methods, and block substitution as done in the second example where the two subsystems B1 and B2 are substituted with the block B12 without altering the global solution algorithm.

**Table 2.** Comparative performance of the standard methods and the proposed new one.

Method	Convergence	Block Substitution	Equation Independency	Further Manipulation
Full Implicit	Yes	NO	NO	Needed
Full Explicit	Not Always	Yes	Yes	Needed
New	Yes	Yes	Yes	Not Needed

Therefore, no symbolical manipulation is needed to obtain the state variables and convert a DAE model into an iterative numerical model, nor to solve the system of DAEs as a whole. As said the most common and effective approach consists in preliminary determining an ordering in the subsystem's execution. However, they are mainly based on the system's nature rather than on its structure. On the other hand, when dealing with ISMs, the structure of the problem can be directly defined by its model graph, and this is what is achieved with the proposed technique.

Finally, the following steps allow to schematize the procedure:

1. decompose the entire system into individual subsystems (all stable when studied individually);
2. identify the status, input, and output variables of each single subsystem;
3. model each individual subsystem, including an individual numerical solver;
4. identify the connections between the output variables of each subsystem with the input variables of other subsystems;
5. represent in a graph all the subsystems (nodes) and connections (arcs);
6. search for the optimal (or suboptimal) order of the nodes of the graph;

solve each subsystem in the order identified, transferring the values of the output variables to the input variables to which they are connected.

#### 4. Conclusions

This paper presents a new method for dealing with the study and simulation of Individual Subsystems Models (ISMs). It is based on different methods that, once joined in a comprehensive framework, allow for a larger ISMs class than before. In particular, the proposed method relies on the framework of the co-simulation approach but with the novelty of integrating consolidated graph theory methods, such as the minimum feedback set of a graph and the corresponding node ordering. This approach leads to obtaining a sequential subsystem solving, regardless of the specific method used for each subsystem, reducing the complexity of the whole system's solver, and speeding up the solving procedure thanks to the optimal (suboptimal) ordering of the subsystems solution. Moreover, it is based only on the interconnections structure and not on the equations themselves. First, the general procedure of the newly proposed method has been outlined, then a stability condition was obtained for Linear Time-Invariant systems, and some examples are shown.

In particular, not only a stable system has been analysed but also an only feed-forward unstable system (i.e., with feedback arcs deleted) and this occurs whenever we are involved with stabilized unstable subsystems; feedback arcs stabilize these kinds of systems.

Moreover, it was also simulated a refrigeration plant with a vapour compression cycle and thermal waste recovery using also an intermediate warm water tank, all typical of food industry processes. In this case the result of applying the proposed numerical technique gives a solution very close to the analytical one with an absolute error less than 2% for the temperature of the hot process side and 0.5% for the temperature of the warm water in the tank.

Future research should address two different objectives. Firstly, a more general stability condition should be searched for non-linear or quasi-nonlinear systems. Secondly, a new simulation framework could be developed where the proposed method can be embedded.

**Author Contributions:** Conceptualization, F.C. and G.P.; methodology, M.D., F.C. and G.S.; investigation, F.C., M.D., R.R. and G.S.; data curation, F.C., M.D., R.R. and G.S.; writing—original draft preparation, F.C., R.R. and G.S.; writing—review and editing, M.D. and G.P.; visualization, F.C. and G.S.; supervision, M.D. and G.P. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

## References

1. Zhang, Y.; Li, Y.; Tomsovic, K.; Djouadi, S.; Yue, M. Review on Set-Theoretic Methods for Safety Verification and Control of Power System. *arXiv* **2020**, arXiv:2001.00080. [\[CrossRef\]](#)
2. Martínez-Villegas, C.T.; Theilliol, D.; Torres, L. Review of two noncentralized observer-based diagnosis schemes for interconnected systems. In Proceedings of the 10th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes, SAFEPROCESS 2018, Warsaw, Poland, 29–31 August 2018.
3. Zohdi, T.I. Multiple UAVs for Mapping: A Review of Basic Modeling, Simulation, and Applications. *Annu. Rev. Environ. Resour.* **2018**, *43*, 523–543. [\[CrossRef\]](#)
4. Dolgui, A.; Ivanov, D.; Sethi, S.P.; Sokolov, B. Scheduling in production, supply chain and Industry 4.0 systems by optimal control: Fundamentals, state-of-the-art and applications. *Int. J. Prod. Res.* **2019**, *57*, 411–432. [\[CrossRef\]](#)
5. Aversano, L.; De Lucia, A.; Gaeta, M.; Stefanucci, S.; Villani, M.L. Managing coordination and cooperation in distributed software processes: The GENESIS environment. *Softw. Process Improv. Pract.* **2004**, *9*, 239–263. [\[CrossRef\]](#)
6. Galdiero, E.; De Paola, F.; Fontana, N.; Giugni, M.; Savic, D. Decision support system for the optimal design of district metered areas. *J. Hydroinform.* **2016**, *18*, 49–61. [\[CrossRef\]](#)
7. Pöschacker, M.; Khatib, T.; Elmenreich, W. The Microgrid Simulation Tool RAPSIm: Description and Case Study. In Proceedings of the IEEE Innovative Smart Grid Technologies Asia (ISGT-ASIA 2014), Kuala Lumpur, Malaysia, 20–23 May 2014.
8. Perone, C.; Bianchi, B.; Catalano, F.; Orsino, M. Experimental evaluation of functional and energy performance of pneumatic oenological presses for high quality white wines. *Sustainability* **2022**, *14*, 8033. [\[CrossRef\]](#)
9. Bianchi, B.; Catalano, F.; Oliveto, R.; Ricciardi, S. Dynamic simulation driven design and management of production facilities in agricultural/food industry. *Acta Hortic.* **2021**, *1311*, 241–248. [\[CrossRef\]](#)
10. Catalano, F.; Leone, A.; Bianchi, B.; Tamborrino, A. A new tool for food industrial plant simulation and IoT control. *Chem. Eng. Trans.* **2021**, *87*, 367–372.
11. Catalano, F.; Bianchi, B.; Berardi, A.; Leone, A.; Tamborrino, A. Experimental trials and dynamical simulation of the potential biogas production in a frozen food industry. *Chem. Eng. Trans.* **2021**, *87*, 295–300.
12. Tamborrino, A.; Catalano, F.; Berardi, A.; Bianchi, B. New modelling approach for the energy and steam consumption evaluation in a fresh pasta industry. *Chem. Eng. Trans.* **2021**, *87*, 409–414. [\[CrossRef\]](#)
13. Tamborrino, A.; Catalano, F.; Leone, A.; Bianchi, B. A real case study of a full-scale anaerobic digestion plant powered by olive by-products. *Foods* **2021**, *10*, 1946. [\[CrossRef\]](#) [\[PubMed\]](#)
14. Catalano, F.; Perone, C.; Iannacci, V.; Leone, A.; Tamborrino, A.; Bianchi, B. Energetic analysis and optimal design of a CHP plant in a frozen food processing factory through a dynamical simulation model. *Energy Convers. Manag.* **2020**, *225*, 113444. [\[CrossRef\]](#)
15. Satin, A.; Savelev, R.; Smagin, D.; Napreenko, K.; Neveshkina, A. Application SimInTech Software for Optimization Fuel System Parameters of the Perspective Helicopter. *MATEC Web Conf.* **2019**, *304*, 04016. [\[CrossRef\]](#)

16. Tamborrino, A.; Perone, C.; Catalano, F.; Squeo, G.; Caponio, F.; Bianchi, B. Modelling energy consumption and energy-saving in high-quality olive oil decanter centrifuge: Numerical study and experimental validation. *Energies* **2019**, *12*, 2592. [\[CrossRef\]](#)
17. Perone, C.; Catalano, F.; Giametta, F.; Tamborrino, A.; Bianchi, B.; Ayr, U. Study and analysis of a cogeneration system with microturbines in a food farming of dry pasta. *Chem. Eng. Trans.* **2017**, *58*, 499–504.
18. Castro, R.D.; Cellier, F.E.; Fischlin, A. Sustainability analysis of complex dynamic systems using embodied energy flows: The eco-bond graphs modeling and simulation framework. *J. Comput. Sci.* **2015**, *10*, 108–125. [\[CrossRef\]](#)
19. McPhee, J.; Schmitke, C.; Redmond, S. Dynamic Modelling of Mechatronic Multibody Systems with Symbolic Computing and Linear Graph Theory. *Math. Comput. Model. Dyn. Syst.* **2004**, *10*, 1–23. [\[CrossRef\]](#)
20. Shi, P.; McPhee, J. Dynamics of Flexible Multibody Systems Using Virtual Work and Linear Graph Theory. *Multibody Syst. Dyn.* **2000**, *4*, 355–381. [\[CrossRef\]](#)
21. Tarjan, R.E. Depth-first search and linear graph algorithms. *SIAM J. Comput.* **1972**, *1*, 146–160. [\[CrossRef\]](#)
22. Eghbal, A.; Gerber, A.G.; Aubanel, E. Acceleration of unsteady hydrodynamic simulations using the parareal algorithm. *J. Comput. Sci.* **2017**, *19*, 57–76. [\[CrossRef\]](#)
23. Perea-López, E.; Ydstie, B.E.; Grossmann, I.E. A model predictive control strategy for supply chain optimization. *Comput. Chem. Eng.* **2003**, *27*, 1201–1218. [\[CrossRef\]](#)
24. Muniyandi, R.C.; Mohd Zin, A. Modeling framework for membrane computing in biological systems: Evaluation with a case study. *J. Comput. Sci.* **2014**, *5*, 137–143. [\[CrossRef\]](#)
25. Aversano, L.; Cerulo, L.; Di Penta, M. Relationship between design patterns defects and crosscutting concern scattering degree: An empirical study. *IET Softw.* **2009**, *3*, 395–409. [\[CrossRef\]](#)
26. Aversano, L.; Bruno, M.; Canfora, G.; Di Penta, M.; Distanto, D. Using concept lattices to support service selection. *Int. J. Web Serv. Res.* **2006**, *3*, 32–51. [\[CrossRef\]](#)
27. Aversano, L.; Bodhuin, T.; Tortorella, M. Assessment and impact analysis for aligning business processes and software systems. In Proceedings of the ACM Symposium on Applied Computing, Santa Fe, NM, USA, 13–17 March 2005; Volume 2, pp. 1338–1343.
28. Langbort, C.; Chandra, R.S.; D’Andrea, R. Distributed control design for systems interconnected over an arbitrary graph. *IEEE Trans. Autom. Control* **2004**, *49*, 1502–1519. [\[CrossRef\]](#)
29. Garwood, T.L.; Hughes, B.R.; Oates, M.R.; O’Connor, D.; Hughes, R. A review of energy simulation tools for the manufacturing sector. *Renew. Sustain. Energy Rev.* **2018**, *81*, 895–911. [\[CrossRef\]](#)
30. Rahimifard, S.; Seow, Y.; Childs, T. Minimising embodied product energy to support energy efficient manufacturing. *CIRP Ann.—Manuf. Technol.* **2010**, *59*, 25–28. [\[CrossRef\]](#)
31. Galinec, D.; Luić, L. Design of Conceptual Model for Raising Awareness of Digital Threats. *WSEAS Trans. Environ. Dev.* **2020**, *16*, 493–504. [\[CrossRef\]](#)
32. Falade, K.; Tiamiyu, A.T. Numerical Solution of Partial Differential Equations with Fractional Variable Coefficients Using New Iterative Method (NIM). *Int. J. Math. Comput. Sci.* **2020**, *3*, 12–21. [\[CrossRef\]](#)
33. Vandendorpe, A.; Van Dooren, P. Model Reduction of Interconnected Systems. In *Model Order Reduction: Theory, Research Aspects and Applications*; Mathematics in Industry (The European Consortium for Mathematics in Industry); Schilders, W.H.A., van der Vorst, H.A., Rommes, J., Eds.; Springer: Berlin/Heidelberg, Germany, 2008; Volume 13.
34. Goetz, E.; Sheno, S. *Critical Infrastructure Protection*; Springer: Boston, MA, USA, 2008.
35. Deb, C.; Schlueter, A. Review of data-driven energy modelling techniques for building retrofit. *Renew. Sustain. Energy Rev.* **2021**, *144*, 110990. [\[CrossRef\]](#)
36. Feng, B.; Zhuo, L.; Xie, D.; Mao, Y.; Gao, J.; Xie, P.; Wu, P. A quantitative review of water footprint accounting and simulation for crop production based on publications during 2002–2018. *Ecol. Indic.* **2021**, *120*, 106962. [\[CrossRef\]](#)
37. Chiacchio, F.; Iacono, A.; Compagno, L.; D’Urso, D. A general framework for dependability modelling coupling discrete-event and time-driven simulation. *Reliab. Eng. Syst. Saf.* **2020**, *199*, 106904. [\[CrossRef\]](#)
38. Iannino, V.; Mocci, C.; Vannocci, M.; Colla, V.; Caputo, A.; Ferraris, F. An Event-Driven Agent-Based Simulation Model for Industrial Processes. *Appl. Sci.* **2020**, *10*, 4343. [\[CrossRef\]](#)
39. Meyer, R. Event-driven multi-agent simulation. In *Multi-Agent-Based Simulation XV, Proceedings of the International Workshop, MABS 2014, Paris, France, 5–6 May 2014*; Grimaldo, F., Norling, E., Eds.; Springer International Publishing: Berlin/Heidelberg, Germany, 2015.
40. Gomes, C.; Thule, C.; Broman, D.; Gorm Larsen, P.; Vangheluwe, H. CoSimulation: A Survey. *ACM Comput. Surv.* **2018**, *51*, 33.
41. Di Perna, R.A. *Computational Methods for Digital Simulation of Continuous Systems*; Technical Report No. 123; Columbia University: New York, NY, USA, 1970.
42. Wills, A.; Schön, T.B.; Ljung, L.; Ninness, B. Identification of Hammerstein–Wiener Models. *Automatica* **2013**, *49*, 70–81. [\[CrossRef\]](#)
43. Kofránek, J.; Kulhánek, J.; Matejak, M.; Ježek, F.; Silar, J. Integrative physiology in Modelica. In Proceedings of the 12th International Modelica Conference, Prague, Czech Republic, 15–17 May 2017; pp. 589–603.
44. Shanshiashvili, B.; Kavlashvili, N. Parameter Identification of Nonlinear Dynamic Systems of Industrial Processes. *Biol. Chem. Res.* **2020**, *7*, 21–34.
45. Juneja, M.; Mohanty, S.R.; Naga, S.K. Robust optimisation-based order reduction and stability analysis of autonomous DC microgrid with consideration of non-linearity. *Int. Trans. Electr. Energy Syst.* **2019**, *30*, e12228. [\[CrossRef\]](#)

46. Ira-Ramírez, H.; Gao, Z.; Cuevas-Ramírez, L. Tracking in Interconnected Gantry Crane Systems: A Decentralized Active Disturbance Rejection Control. In Proceedings of the 33rd Chinese Control Conference, Nanjing, China, 28–30 July 2014; pp. 4342–4347.
47. Falgout, R.D.; Lecouvez, M.; Woodward, C.S. A parallel-in-time algorithm for variable step multistep methods. *J. Comput. Sci.* **2019**, *37*, 101029. [\[CrossRef\]](#)
48. Mattsson, S.E.; Söderlind, G. Index Reduction in Differential-Algebraic Equations Using Dummy Derivatives. *SIAM J. Sci. Comput.* **1993**, *14*, 677–692. [\[CrossRef\]](#)
49. Ascher, U.M.; Petzold, L.R. *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*; SIAM Bookstore: University City, PA, USA, 1998; Available online: <https://my.siam.org/Store/Product/viewproduct/?ProductId=961> (accessed on 23 November 2022).
50. Cellier, F.E.; Kofman, E. *Continuous System Simulation*; Springer: Berlin/Heidelberg, Germany, 2006.
51. Täuber, P.; Ochel, L.; Braun, W.; Bachmann, B. Practical realization and adaptation of cellier’s tearing method. In Proceedings of the 6th International Workshop on Equation-Based Object-Oriented Modeling Languages and Tools, EOOLT’14, Berlin, Germany, 10 October 2014; pp. 11–19.
52. Otter, M.; Elmqvist, H. Transformation of Differential Algebraic Array Equations to Index One Form. In Proceedings of the Modelica Conference, Prague, Czech Republic, 15–17 May 2017.
53. Westerberg, A.W.; Edie, F.C. Computer-aided design, Part 1 Enhancing Convergence Properties by the Choice of Output Variable Assignments in the Solution of Sparse Equation Sets. *Chem. Eng. J.* **1971**, *2*, 9–16. [\[CrossRef\]](#)
54. Gupta, P.K.; Westerberg, A.W.; Hendry, J.E.; Hughes, R.R. Assigning output variables to equations using linear programming. *AIChE J.* **1974**, *20*, 397–399. [\[CrossRef\]](#)
55. Xu, T.; Wang, G.; Yang, J. Finding strongly connected components of simple digraphs based on granulation strategy. *Int. J. Approx. Reason.* **2020**, *118*, 64–78. [\[CrossRef\]](#)
56. Gerbner, D.; Keszegha, B.; Palmer, C.; Pálvölgyi, C. Topological orderings of weighted directed acyclic graphs. *Inf. Process. Lett.* **2016**, *116*, 564–568. [\[CrossRef\]](#)
57. Bloem, R.; Gabow, H.N.; Somenzi, F. An Algorithm for Strongly Connected Component Analysis in  $n \log n$  Symbolic Steps. *Form. Methods Syst. Des.* **2006**, *28*, 37–56. [\[CrossRef\]](#)
58. Xie, A.; Beere, A. Implicit Enumeration of Strongly Connected Components and an Application to Formal Verification. *IEEE Trans. Comput. -Aided Des. Integr. Circuits Syst.* **2000**, *19*, 10.
59. Younger, D.H. Minimum Feedback Arc Sets for a Directed Graph. *IEEE Trans. Circuit Theory* **1963**, *10*, 229–245. [\[CrossRef\]](#)
60. Rajasingh, I.; Rajan, B.; Little Joice, S.L. Feedback Arc Set in Oriented Graphs. *J. Comp. Math. Sci.* **2011**, *2*, 804–811.
61. Charbit, P.; Thomassé, S.; Yeo, A. The Minimum Feedback Arc Set Problem is NP-Hard for Tournaments. *Comb. Probab. Comput.* **2007**, *16*, 1–4. [\[CrossRef\]](#)
62. Nwachukwu, G.C.; Mokwunyei, N.E. Generalized Adams-Type Second Derivative Methods for Stiff Systems of ODEs. *Int. J. Appl. Math.* **2018**, *48*, 1–11.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.