STRUCTURAL ANALYSIS OF PROCESS MODELS USING THEIR REPRESENTATION GRAPH

A. Leitold[1⊠], M. Gerzson[2]

University of Pannonia, Department of Mathematics, P.O.B. 158, Veszprém, H-8201, HUNGARY
⊠E-mail: leitolda@almos.vein.hu
University of Pannonia, Department of Electrical Engineering and Information Systems, P.O.B. 158,
Veszprém, H-8201, HUNGARY

A graph-theoretical method for the structural analysis of dynamic lumped process models described by differential and algebraic equations (DAEs) is applied in this paper in order to determine the most important solvability properties (degree of freedom, structural solvability, model decomposition, dynamic degree of freedom, differential index, e.g.) of these models by using the so-called dynamic representation graph. The structure of the dynamic representation graph is suitable for the determination of the mentioned solvability properties. The most common methods in the modelling practice for the construction of models of complex systems are the union of submodels and hierarchical modelling. Our goal is to investigate the effect of the model union to the solvability properties, especially to the differential index. We show how the representation graph of a complex model can be built up from the representation graphs of submodels. The effect of the structure of submodels and their joining points to the structure of the complex graph and the conclusions drawn from the complex graph structure to the solvability properties are also investigated. The representation graph of the complex model can be easily built up from the representation graphs of the simple models according to the linking of the technological subsystems. If one of the submodels has greater than one differential index then the under and overspecified subgraphs referring to this higher index can be found in the representation graph of the complex model, too. The change in the relative position of the underspecified and the overspecified subgraphs has an effect to the value of differential index. If these subgraphs move further from their original positions the value of the differential index increases. If their relative positions do not change during the built up process then the value of the differential index of the complex system is equal to the value of the differential index of the subsystem having the higher value.

Keywords: process models, model composition, DAE-models, differential index, solvability, structural analysis

Introduction

The structural analysis of dynamic lumped process models forms an important step in the model building procedure [1] and it is used for the determination of the solvability properties of the model, too. This analysis includes the determination of the degree of freedom, structural solvability, differential index and the dynamic degrees of freedom. As a result of the analysis, the decomposition of the model is obtained and the calculation path can be determined. This way the appropriate numerical method for solving the model can be chosen efficiently. Moreover, advice on how to improve the computational properties of the model by modifying its form or its specification can also be given.

Effective graph-theoretical methods have been proposed in the literature [2, 3] based on the analysis tools developed by [4], for the determination of the most important solvability property of lumped dynamic models: the differential index. The properties of the dynamic representation graph of process models described by semi-explicit DAE-systems have also

been analysed there in case of index 1 and higher index models. Beside the algorithm of determining the differential index by using the representation graph, a model modification method has also been proposed in the literature, which results in a structurally solvable model even in the case of higher index models [2].

Basic notions

Structural solvability

As a first step, we consider a system of linear or non-linear algebraic equations in its so called *standard form* [4]:

$$y_i = f_i(x, u), \qquad i = 1, \ldots, M$$
$$u_k = g_k(x, u), \qquad k = 1, \ldots, K$$

where x_j ($j = 1, \ldots, N$) and u_k ($k = 1, \ldots, K$) are unknowns, y_i ($i = 1, \ldots, M$) are known parameters, f_i ($i = 1, \ldots, M$) and g_k ($k = 1, \ldots, K$) are assumed to be sufficiently smooth real-valued functions. The system of equations

above is structurally solvable, if the Jacobian matrix $J(x, u)$ referring to the above model is non-singular.

Consider a system of equations in standard form. We construct a directed graph to represent the structure of the set of equations in the following way. The vertex-set corresponding to unknowns and parameters is partitioned as $X \cup U \cup Y$, where $X = \{x_1, \ldots, x_N\}$, $U = \{u_1, \ldots, u_K\}$ and $Y = \{y_1, \ldots, y_M\}$. The functional dependence described by an equation is expressed by arcs coming into y_i or u_k respectively from those x_j and u_l, which appear on its right-hand side. This graph is called the *representation graph* of the system of equations.

A *Menger-type linking* from X to Y is a set of pair-wise vertex-disjoint directed paths from a vertex in X to a vertex in Y. The size of a linking is the number of directed paths from X to Y contained in the linking. In case $|X| = |Y|$, $(M = N)$, a linking of size $|X|$ is called a *complete linking*. The graphical condition of the structural solvability is then the following [4]:

Linkage theorem: Assume that the non-vanishing elements of partial derivatives f_i and g_k in the standard form model are algebraically independent over the rational number field Q. *Then the model is structurally solvable if and only if there exists a Menger-type complete linking from X to Y on the representation graph.*

We can adapt the graphical techniques to DAE-systems, as well. An ordinary differential equation of a DAE-system can be described by the following equation:

$$x' = f(x_1, \ldots, x_n)$$

Here x denotes an arbitrary variable depending on time, x' denotes the derivative of x with respect to time and $x_1, \ldots, x_n$ are those variables which have effect on variable x' according to the differential equation.

In DAE-systems there are two types of variables. *Differential variables* are the variables with their time derivative present in the model. Variables, which do not have their time derivative present, are called *algebraic variables*. The derivative x' is called *derivative* (velocity) *variable*.

Dynamic representation graph

The value of differential variables is usually computed by using a numerical integration method. Therefore a system of equations including also differential equations can be represented by a *dynamic graph*. A dynamic graph is a sequence of static graphs corresponding to each time step of the integration. On a dynamic graph there are directed arcs attached from the previous static graph to the succeeding static graph that are determined by the method applied for solving the ordinary differential equations. In case of a single step explicit method, the value of a differential variable at time $t+h$ is computed using the corresponding differential value and its value at a previous time t. For example, when the explicit Euler method is used:

$$x(t+h) = x(t) + h \cdot x'(t)$$

where h denotes the step length during the numerical integration. The structure of a dynamic graph assuming explicit Euler method for solving differential equations is shown in *Fig. 1*.
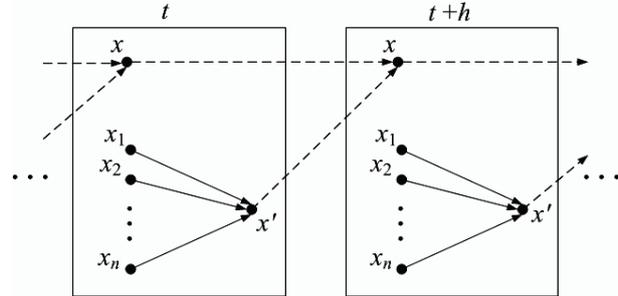
Figure 1: Dynamic representation graph assuming first order explicit solution method

The structural analysis based on graph theoretical technique is carried out in steps performed sequentially. The first step is to rewrite the model into its standard form. The second step is the assignment of types to vertices in the representation graph. The important types of vertices determined by the model specification are the following [2, 5]:

- $<S>$(*set*)-*type variables*: These represent variables, which are assigned to the specified given values. In the case of a dynamic representation graph assuming explicit method for solving the differential equations, the differential variables will be labelled by type $<S^*>$ because their initial value can be obtained from the initial values, and then their values can be calculated step by step by numerical integration. Labels $<S>$ and $<S^*>$ are treated the same way during the analysis.

- $<G>$(*given*)-*type variables*: A variable assigned to a specific value of a left hand side is a $<G>$-type variable. Unlike the $<S>$-type variables, the values of the right hand side variables will be suitably adjusted so as to preserve the equality of the two sides.

According to the representation graph, the value of every variable which has incoming arcs only from vertices labelled by type $<S>$ can be calculated by simple substitution into the corresponding equation. These variables become *secondarily labelled by type* $<S>$, and this process can be repeated if necessary. Omitting all vertices labelled primarily, secondarily, etc. by type $<S>$ and all arcs starting from them from the representation graph we obtain the *reduced graph*. The classification of vertices of a reduced graph is as follows:

- all initial vertices form the unknown variable set X,
- all terminal vertices labelled by type $<G>$ constitute the known variable (parameter) set Y,
- all other vertices constitute the unkown variable set U.

Differential index

Dynamic process models can be described by *semi-explicit DAEs* as follows:

$$z_1' = f(z_1, z_2, t), \quad z_1(t_0) = z_{10} \tag{1}$$
$$0 = g(z_1, z_2, t) \tag{2}$$

The most important structural computational property of DAE models is the differential index [6]. By definition [7] the differential index of the semi-explicit DAE (Equations (1)-(2)) is one if one differentiation is sufficient to express z_2' as a continuous function of z_1, z_2 and t. One differentiation is sufficient if and only if the Jacobian matrix g_{z_2} is non-singular.

In our earlier work we have proved that the differential index of the models investigated in [2] is equal to 1 if and only if there exists a Menger-type complete linking on the reduced graph at any time step *t*.

If the differential index of the investigated model is greater than 1 then there is no Menger-type complete linking on the static graph at any time step *t*. The properties of a static graph of a dynamic model, which has differential index >1 are as follows.

1. The fact that the initial values of differential variables cannot be chosen independently results in an *over-specified* part on the graph. This situation can be easy shown by assignment of types to vertices corresponding to the model specification. There is an overspecified part on the graph if a vertex labelled by type <S*> or <G> can also be labelled preliminary, secondarily, tertiarily or etc. by type <S>.

2. Non-singularity of g_{z_2} results in an *underspecified* part on the graph. In this part those algebraic variables appear, which cannot be calculated from algebraic equations and those derivative variables, which we want to calculate from them.

We have also proposed an algorithm using the structure of the representation graph for determination of the differential index of the underlying model. The main steps of this algorithm are the following:

1. Let us form the following variable sets.

   I_0 is the set of the differential variables belonging to the overspecified subgraph,

   D_0 is the set of the derivative variables referring to the differential variables of set I_0,

   I_1 is the set of differential variables from which directed paths lead to the derivative variables in the set D_0,

   D_1 is the set of derivative variables referring to the differential variables of set I_1, … ,

   I_k is the set of differential variables from which directed paths lead to the derivative variables in the set D_{k-1},

   D_k is the set of derivative variables referring to the differential variables of set I_k, …

2. Let *n* be the smallest natural number for which the set D_n contains some derivative variables of the underspecified subgraph. Then the differential index of the model is

$$v_d = n+2$$

If there is no such number *n* then the model is not structurally solvable.

In our earlier work we have shown that the important properties of the representation graph including the differential index of the models are independent of the assumption whether a single-step, explicit or implicit numerical method is used for the solution of the differential equations [8].

Structural analysis of simple models using their representation graphs

In this section, simple, small sized, dynamic models are investigated using their representation graphs. We show the influence of the change of the modelling goal (and so the model specification) and the modelling conditions to the differential index. The examples used in this and next sections are based on examples of [9].

Example 1 – Perfectly stirred tank reactor

Suppose a perfectly stirred tank reactor and let the concentration of its inlet flow be denoted by c_0. The change of concentration in the tank can be described by the following equation:

$$c' = \frac{q}{V}(c_0 - c) \tag{3}$$

where *c* is the concentration in the tank, *q* is the outlet flow rate and *V* is the volume of the tank.

Case a) Let us assume that we know the concentration of the inlet flow in the function of time: $c_0 = c(t)$, and we want to determine the concentration of the outlet flow. The standard form model consists of the following equations:

$$c = \int c'$$
$$c' = \frac{q}{V}(c_0 - c)$$
$$c_0 = c_0(t)$$

Given: $c(t_0)$, $c_0(t)$;
Constant: q, V;
To be calculated: *c* as a function of time.

Since the structural properties of the model described by representation graph can be investigated based on the structure of the static graphs, and these properties are independent from the arcs connecting individual static graphs to each other, we illustrate only one static graph as a representation graph of models for the sake of simplicity.

The representation graph of this simple model is shown in *Fig. 2a*. The reduced graph is an empty graph in this case indicating the differential index is equal to 1.

We remark that the substitution of the condition $c_0 = c_0(t)$ into the Equation (3) results in a model of

technological system with only one differential equation, so the differential index would be equal to 0.

Case b) Let us assume now that the modelling goal is the dynamic design of the same system, i.e. the determination of the necessary inlet flow concentration in order to ensure the required outlet concentration $c = c(t)$. The standard form model is the following:
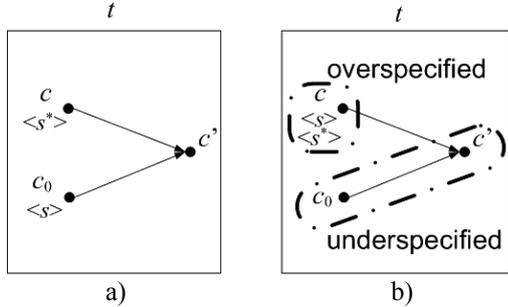
Figure 2: The representation graphs of the Example 1

$$c = \int c'$$

$$c' = \frac{q}{V}(c_0 - c)$$

$$c = c(t)$$

Given: $c(t_0)$, $c(t)$;
Constant: q, V;
To be calculated: c_0 as a function of time.

In this case, there are an underspecified and an overspecified subgraphs on the representation graph (see *Fig. 2b*) referring to the differential index greater than 1 value. The differential index can be calculated based on the structure of the representation graph:

$$I_0 = \{c\}$$
$$D_0 = \{c'\}$$

Since the vertex referring to the derivative variable c' can be found in the underspecified subgraph, therefore $n = 0$ and $v_d = n + 2 = 2$.

Example 2a – Liquid mixer model

Suppose a liquid mixer tank having one inlet and one outlet flow (see *Fig. 3*) The inlet flow consists of two components A and B. The two components have different density. There is a certain amount of liquid in tank at $t = t_0$. The feed is perfectly mixed with the tank liquid. The density of the liquid in tank, the flow rates and the mol fractions of the components are functions of time. The number of moles (N_i) of components A and B can be described by the following equation:

$$N_i' = F_0 x_{i0} - F x_i \qquad \text{where } i = \{A, B\}$$

where F_0 and F are the inlet and outlet flow rate, and x_{i0} and x_i are the mol fraction of the component i in the inlet and outlet flows, resp. Let p_L denote the pressure of the liquid at the bottom of the tank, a the area of the tank and M_{Wi} the molar weight of the component i. The

outlet flow rate depends on the liquid pressure (p_L) and the valve constant (k).
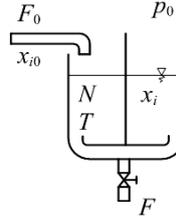
Figure 3: Liquid mixer tank with variable volume

The modelling goal is to calculate the liquid composition in the tank. The standard form model consists of the following equations:

$$N_A = \int N_A' \qquad\qquad N_B = \int N_B'$$
$$N_A' = F_0 x_{A0} - F x_A \qquad N_B' = F_0 x_{B0} - F x_B$$
$$N = N_A + N_B$$
$$x_A = N_A/N \qquad\qquad x_B = N_B/N$$
$$M_w = M_{wA} x_A + M_{wB} x_B$$
$$p_L = p_0 + (M_w \cdot N)/a$$
$$F = k(p_L - p_0)^{1/2}$$

Given: $N_A(t_0)$, $N_B(t_0)$, x_{A0}, x_{B0}, F_0;
Constant: M_{wA}, M_{wB}, a, k, p_0;
To be calculated: N_A, N_B, F as functions of time.

The representation graph of the model is shown in *Fig. 4*. The reduced graph is an empty graph because there is no implicit equation in the model, therefore the differential index (v_d) is equal to 1.
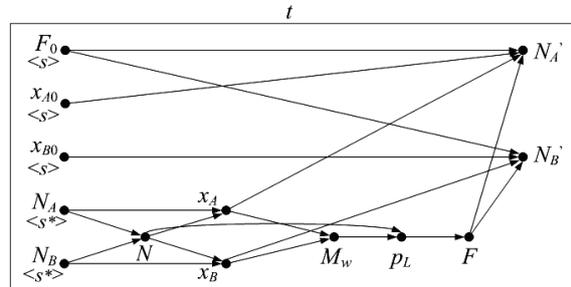
Figure 4: The representation graph of the Example 2a

Example 2b – Liquid mixer model with constant tank volume

Suppose a liquid mixer tank as in Example 2a but let the volume of the liquid in the tank V be constant in this case (*Fig. 5*). Let v_A and v_B be the molar specific volumes of components A and B, resp. The other assumptions are the same as in Example 2a. The modelling goal is to calculate the liquid composition in the tank, again, but the modified volume condition must be taken into account.

The standard form model consists of the following equations:

$$N_A = \int N_A' \qquad\qquad N_B = \int N_B'$$
$$N_A' = F_0 x_{A0} - F x_A \qquad N_B' = F_0 x_{B0} - F x_B$$

$$N = N_A + N_B$$
$$x_A = N_A/N \qquad\qquad x_B = N_B/N$$
$$v = v_A x_A + v_B x_B$$
$$V = N \cdot v$$

Given: $\qquad N_A(t_0), N_B(t_0), x_{A0}, x_{B0}, F_0, V$;

Constant: $\qquad\qquad v_A, v_B$;

To be calculated: N_A, N_B, F as functions of time.
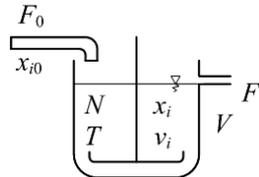
Figure 5: Liquid mixer tank with constant volume

An overspecified and an underspecified subgraph can be found on the representation graph (see *Fig. 6*) and the differential index can be determined based on their structures:

$$I_0 = \{N_A, N_B\}$$
$$D_0 = \{N_A{'}, N_B{'}\}$$

Since the vertices referring to derivative variables $N_A{'}, N_B{'}$ can be found in the underspecified subgraph, $n = 0$ and $v_d = n + 2 = 2$.
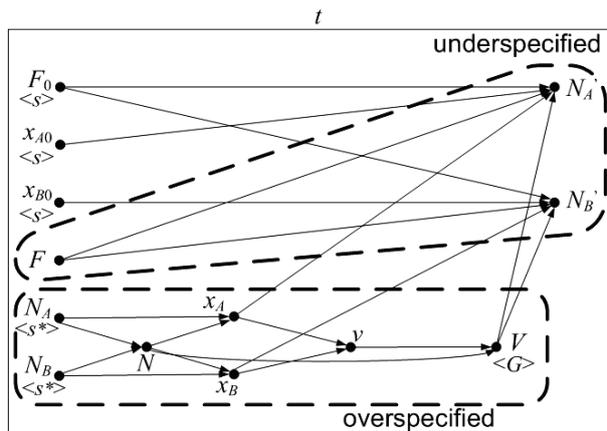
Figure 6: The representation graph of the Example 2b

Structural analysis of composite models

In this section, more complex composite models are built from the simple dynamic models of the previous section. The goal is to investigate the effect of this "build up process" of simple, small sized models, i.e the effect of the model composition to the structural properties of the composite models.

Example 3 – Cascade of perfectly stirred tank reactors

Suppose a system consists of k perfectly stirred tank reactors. A feed of concentration c_0 is fed into the first tank. The concentrations in the tanks are described by the following equation:

$$c_i{'} = \frac{q}{V_i}(c_{i-1} - c_i) \qquad i = 1,2,\dots,k$$

where c_i is the concentration in the tank i, q is the flow rates from tank to tank and V_i is the volume of the tank i.

Two different specifications can be added to these equations according to modelling goal:
a) in dynamic simulation studies the feed concentration c_0 is given by $c_0 = c_0(t)$;
b) in dynamic design the product concentration c_k is given by $c_k = c_k(t)$.

The representation graphs referring to these specifications can be seen in *Figs 7a* and *b*. These graphs can be considered as multiplications of the representation graphs in Figs 2a and b. In the first case, the reduced graph is an empty graph, therefore the differential index is equal to 1. In the second case, there are under and overspecified subgraphs on the representation graph and based on their structures:

$$I_0 = \{c_k\}$$
$$D_0 = \{c_k{'}\}$$

$$I_1 = \{c_{k-1}\}$$
$$D_1 = \{c_{k-1}^2\}$$

$$I_{k-1} = \{c_1\}$$
$$D_{k-1} = \{c_1^?\}$$

Since the vertex referring to the derivative variable $c_1{'}$ can be found in the underspecified subgraph, $n = k - 1$ and $v_d = n + 2 = k + 1$.
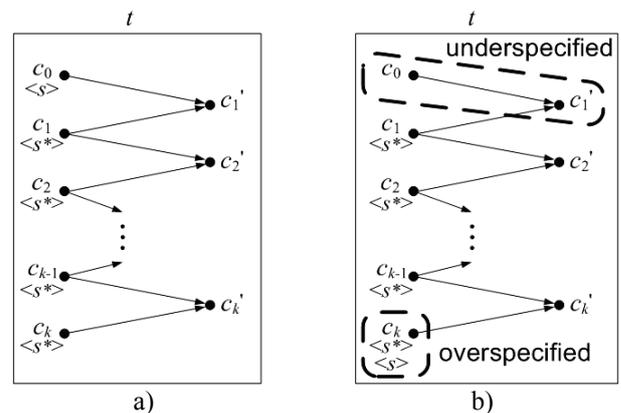
Figure 7: The representation graphs of the Example 3

The effect of the increasing differential index of the cascade model can be followed on the representation graph: the underspecified and the overspecified subgraphs move increasingly further from each other as the cascade elements are inserted. The path between the derivative variable $c_1{'}$ of the underspecified subgraph and the differential variable c_k of the overspecified subgraph is increasingly longer (the direction is not taken into account) and along this path the differential and derivative variables are located alternately.

Example 4 – Sequence of mixing tanks

Suppose that a system consist of a sequence of k mixer tanks (see *Fig. 8*). Let the volume of liquid in the tank j be constant while in the other tanks the liquid volumes are variables. The model of the constant volume tank is described in Example 2b, while the models of the other tanks are the same as the model in Example 2a. The following assumptions are held: A liquid feed stream is fed into the first tank. The feed consists of two components A and B. The liquid flows from the first tank through the system. The other assumptions are the same as in Example 2a and 2b.

The model of this cascade system using the models of Example 2. is the following:

$$N_{Ai} = \int N_{Ai}'$$
$$N_{Ai}' = F_{i-1}x_{Ai-1} - F_i x_{Ai}$$
$$N_i = N_{Ai} + N_{Bi}$$
$$x_{Ai} = N_{Ai}/N_i$$
$$M_{wi} = M_{wA}x_{Ai} + M_{wB}x_{Bi}$$
$$p_{Li} = p_0 + (M_{wi} \cdot N)/a$$
$$F_i = k (p_{Li} - p_0)^{1/2}$$

$$N_{Bi} = \int N_{Bi}'$$
$$N_{Bi}' = F_{i-1}x_{Bi-1} - F_i x_{Bi}$$
where $i = 1, \ldots, k$
$$x_{Bi} = N_{Bi}/N_i$$

where $i = 1, \ldots, k, \quad i \neq j$
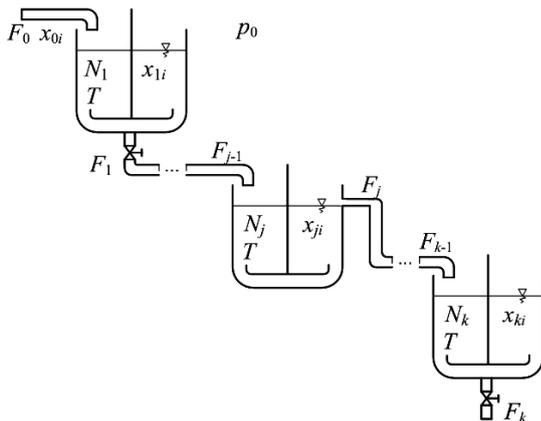
Figure 8: Sequence of liquid tanks

$$v_j = v_A x_{Aj} + v_B x_{Bj}$$
$$V_j = N_j \cdot v_j$$

Given: $\quad N_{Ai}(t_0), N_{Bi}(t_0), i = 1, \ldots, k$
$\qquad\qquad x_{A0}, x_{B0}, F_0, V_j;$

Constant: $\quad M_{wA}, M_{wB}, a, k, p_0, v_A, v_B;$

To be calculated: N_{Ai}, N_{Bi}, F_k as functions of time.

This model is built up from $k–1$ differential index 1 models and one differential index 2 model according to the liquid mixing system. The representation graph of the whole system can be constructed easily from the representation graphs of the simple models (see Figs 4 and 6). The resulted graph can be seen in *Fig. 9*.

An overspecified and an underspecified subgraph can be found on the representation graph and the differential index can be determined based on their structures:
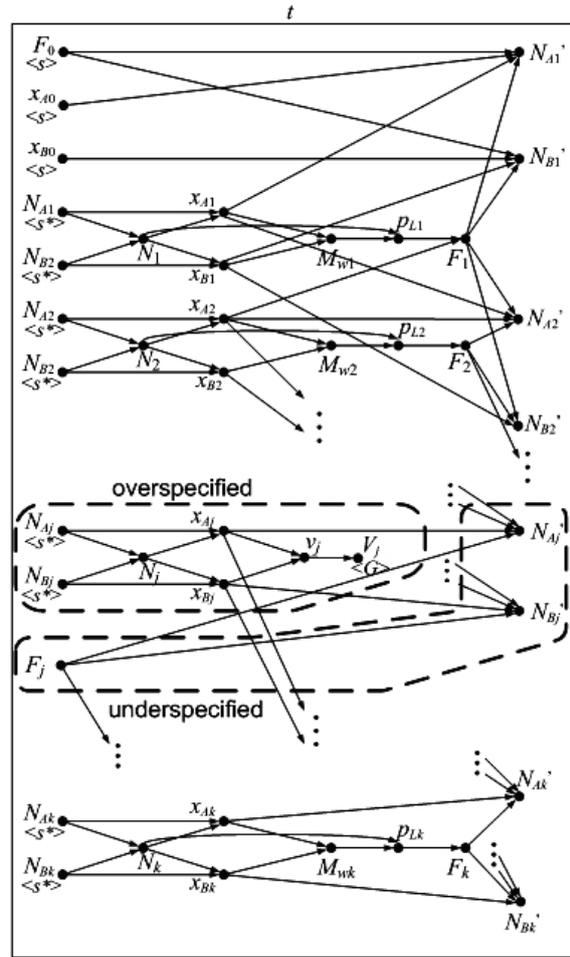
Figure 9: The representation graph of the Example 4

$$I_0 = \{N_{Aj}, N_{Bj}\}$$
$$D_0 = \{N_{Aj}', N_{Bj}'\}$$

Since the vertices referring to differential variables N_{Aj}', N_{Bj}' can be found in the underspecified subgraph, $n = 0$ and $v_d = n + 2 = 2$.

In this example the union of the representation graphs of submodels has been created in such a way that *the position of the underspecified and overspecified subgraphs referring to the higher differential index in the extended graph is unvaried to their original position, therefore the differential index of the complex model is the same as of the model of Example 3b.*

Conclusion

In this paper we investigated the solvability properties of complex dynamic systems when they are built up from simple models. We have shown that the representation graph can be used efficiently for the investigation of the differential index during the model composition process,

too. The representation graph of the complex model can be easily built up from the representation graphs of the simple models according to the linking of the technological subsystems. If one of the submodels has greater than one differential index then the under and overspecified subgraphs referring to this higher index can be found in the representation graph of the complex model, too. The change in the relative position of the underspecified and the overspecified subgraphs has an effect to the value of differential index.

REFERENCES

1. HANGOS K. M., CAMERON I. T.: *Process Modelling and Model Analysis*. Academic Press, London (2001)

2. LEITOLD A., HANGOS K. M.: Structural Solvability Analysis of Dynamic Process Models, *Computers chem. Engng* 25, (2001), 1633–1646

3. LEITOLD A., HANGOS K. M.: Effect of Steady State Assumption on the Structural Solvability of Dynamic Process Models, *Hung. J. of Ind. Chem.* 30 1 (2002), 61–71

4. MUROTA K.: *Systems Analysis by Graphs and Matroids*. Springer Verlag, Berlin (1987)

5. IRI M., TSUNEKAWA J., YAJIMA K.: The Graphical Techniques Used for a Chemical Process Simulator 'JUSE GIFS', *Information Processing*, 71. (1972) Vol. 2, 1150–1155

6. GEAR C. W., PETZOLD L. R.: ODE Methods for the Solution of Differential/Algebraic Systems, *SIAM J. Numer. Anal.* (1984) Vol. 21, No. 4, 716–728

7. BRENAN K. E., CAMPBELL S. L., PETZOLD L. R.: *Numerical Solution of Initial Value Problems in Differential – Algebraic Equations*. North-Holland, New York (1989)

8. LEITOLD A., HANGOS K. M.: Numerical Method-independent Structural Solvability Analysis of DAE Models, *Hung. J. of Ind. Chem.* 33 (2005) 1-2, 11–21

9. MOE H. I.: *Dynamic Process Simulation. PhD Thesis* University of Trondheim, Dept. of Chemical Eng. (1995)