HUNGARIAN JOURNAL
OF INDUSTRIAL CHEMISTRY
VESZPRÉM
Vol. 33(1-2). pp. 11-21. (2005)

NUMERICAL METHOD-INDEPENDENT STRUCTURAL SOLVABILITY ANALYSIS OF DAE MODELS*

A. LEITOLD[1] AND K. M. HANGOS[2]

[1]Department of Mathematics, University of Veszprém
H-8201 Veszprém, P.O.Box 158, HUNGARY
[2]Systems and Control Research Laboratory, Computer and Automation Institute, HAS
H-1518, Budapest, P.O.Box 63, HUNGARY

A graph-theoretical method [1,2] 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 of these models by using the so-called dynamic representation graph. The construction of the dynamic representation graphs that was originally proposed [2] for the most simple, single-step, explicit numerical methods, has been extended in this paper to higher order explicit and implicit solution methods, that are used more frequently and efficiently for numerical solution of DAE-systems. It is shown here that the representation graph for both higher order explicit and implicit solution methods has similar properties to the case of explicit numerical solution procedures both for index one and higher index models. Thus it is proven 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.

Keywords: Process models, 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 [3] 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 [2,4] based on the analysis tools developed by Murota, et al [1], 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. Besides of 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 [5,2].

* An extended version of the lecture presented in 4th MATHMOD Symposium, Vienna, Austria, February 5-7, 2003

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* [1] :

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

where x_j (j = 1,…, N) and u_k (k = 1,…,K) are unknowns, y_i (i = 1,…,M) are known parameters, f_i (i = 1,…,M) and g_k (k = 1,…,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 [1] :

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. So 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 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 can be seen in *Fig. 1.*

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 [6,2] :

- <*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.

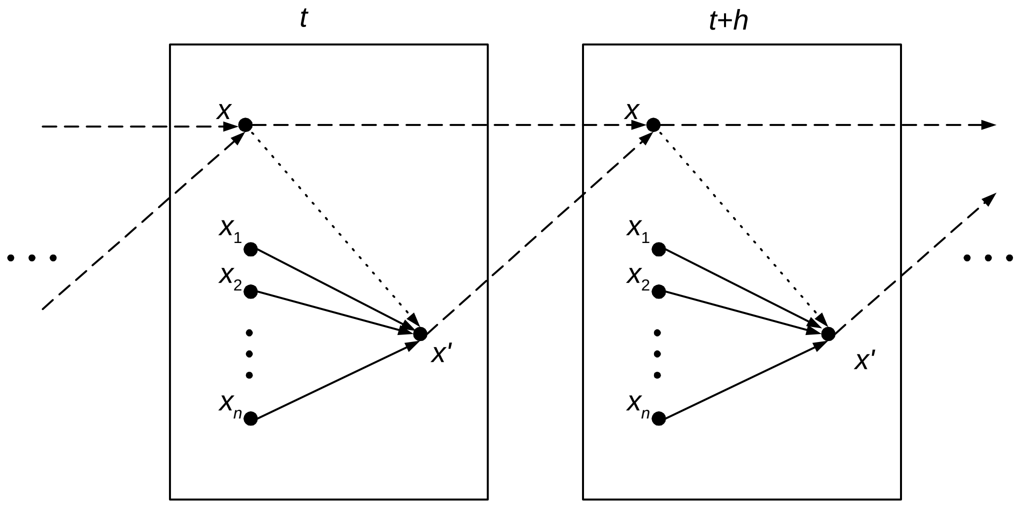

Fig.1 Dynamic representation graph assuming first order explicit solution method

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 unknown variable set *U*.

The *algebraic subgraph* belonging to any static graph in the dynamic representation graph can be obtained by considering the algebraic part of the model and taking its induced subgraph. The vertex set of the algebraic subgraph contains the variables that appear in the algebraic equations and the arc set that corresponds to the algebraic equations.

Differential index

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

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

The most important structural computational property of DAE models is the differential index. By definition [7], the differential index of the above semi-explicit DAE 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_{z2} is non-singular.

In our earlier work we have proved that *the differential index of the models investigated in* [5,2] *is equal to* 1 *if and only if there exists a Menger-type complete linking on the reduced graph at any time step t*.

An equivalent condition [5] of the above is that the differential index of a semi-explicit DAE model *M* is equal to one if and only if there exists a Menger-type complete linking on the reduced graph of the algebraic subgraph belonging to model *M*.

We have also proposed an algorithm using the structure of the representation graph for determination of the differential index of the underlying model.

The following two examples described in details in [5,2] are used throughout the paper to illustrate the notions and algorithms.

Example 1: Consider a liquid tank system with one inlet stream *F* and one exit stream *L*. Let the vessel be perfectly stirred. Heat is transferred to the liquid using a heater. The standard form of the model of the liquid tank system (M_1) is the following:

$$\begin{aligned}
M &= \int M' \\
U &= \int U' \\
M' &= -L + F \\
U' &= -L \cdot h_L + F \cdot h_F + Q \\
h_L &= U/M \\
h_L^* &= f_1(T_L, p) \\
h_F &= f_2(T_F, p_F) \\
s &= h_L - h_L^*, \quad s = 0 \\
L &= f_3(M)
\end{aligned}$$

Here *M* denotes the mass, *U* the internal energy, *Q* the heat transfer rate, h_L and h_L the specific internal enthalpy of inlet and outlet flow, respectively, T_F the temperature of inlet flow, p_F the pressure of inlet flow, T_L the temperature in the vessel, *p* the atmospheric pressure and f_1 and f_2 are given functions.

Let us consider the following two model specifications.

Specification 1.:

The variables F, T_F, p_F and Q are given as functions of time, the initial values M_0, U_0 and the pressure p are constants. The variables M, U, T_L, and L are to be calculated as functions of time.

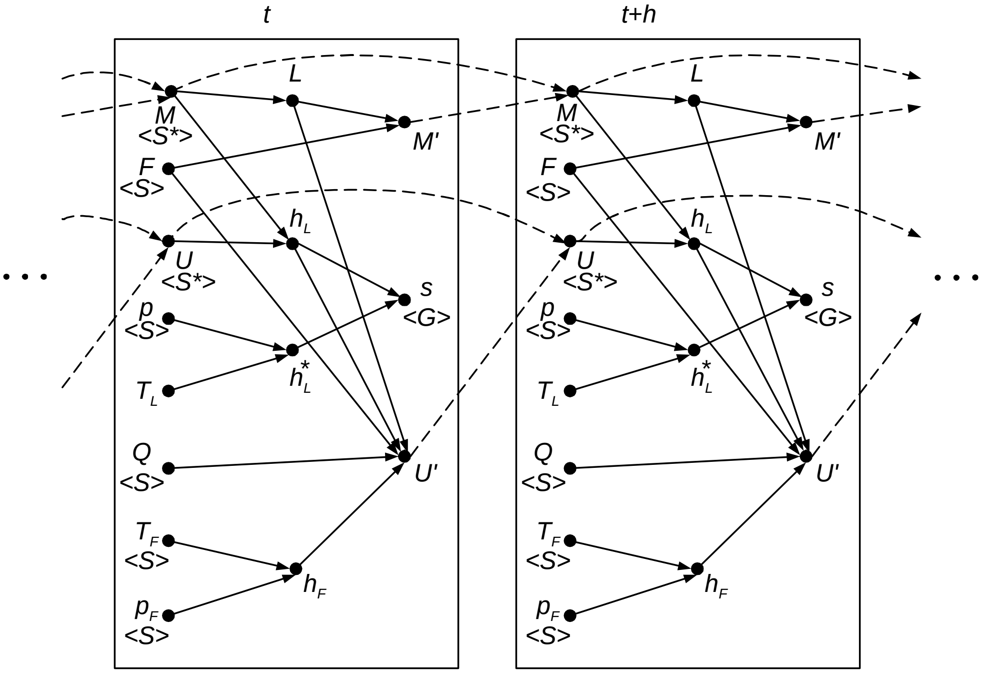

Fig.2 Representation graph of model M_1 with Specification 1 assuming first order explicit method

The representation graph of the liquid tank system and the assignment types of variables corresponding to the above specification are shown in *Fig.2*, while the reduced graph can be seen in *Fig.3*. It can be seen, that there exists a Menger-type complete linking on the reduced graph, which indicates that the differential index of the above model is equal to one.

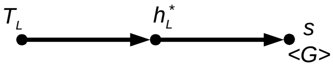

Fig.3 Reduced graph of model M_1 with Specification 1

Next let us consider the following specification.

Specification 2.:

The variables F, T_F, p_F and T_L, are given as functions of time, the initial values M_0, U_0 and the pressure p are constants. The variables M, U, Q and L are to be calculated as functions of time.

Representation graph of model M_1 with the assignment of vertices corresponding to Specification 2 are shown in *Fig.4*. It can be seen that there is no Menger-type complete linking on the graph, hence the standard form model is not structurally solvable. There is an overspecified part on the graph, which indicates the fact that the initial values of the model can not be chosen independently. The underspecified part $Q \rightarrow U$ indicates that Q cannot be calculated from the algebraic equations. This structure of the representation graph shows that the differential index of the model M_1 with Specification 2 is greater than 1. In our earlier work [2] we suggested an algorithm, which determines the differential index of the model using the structure of the representation graph and transform the model into a structurally solvable modified form.

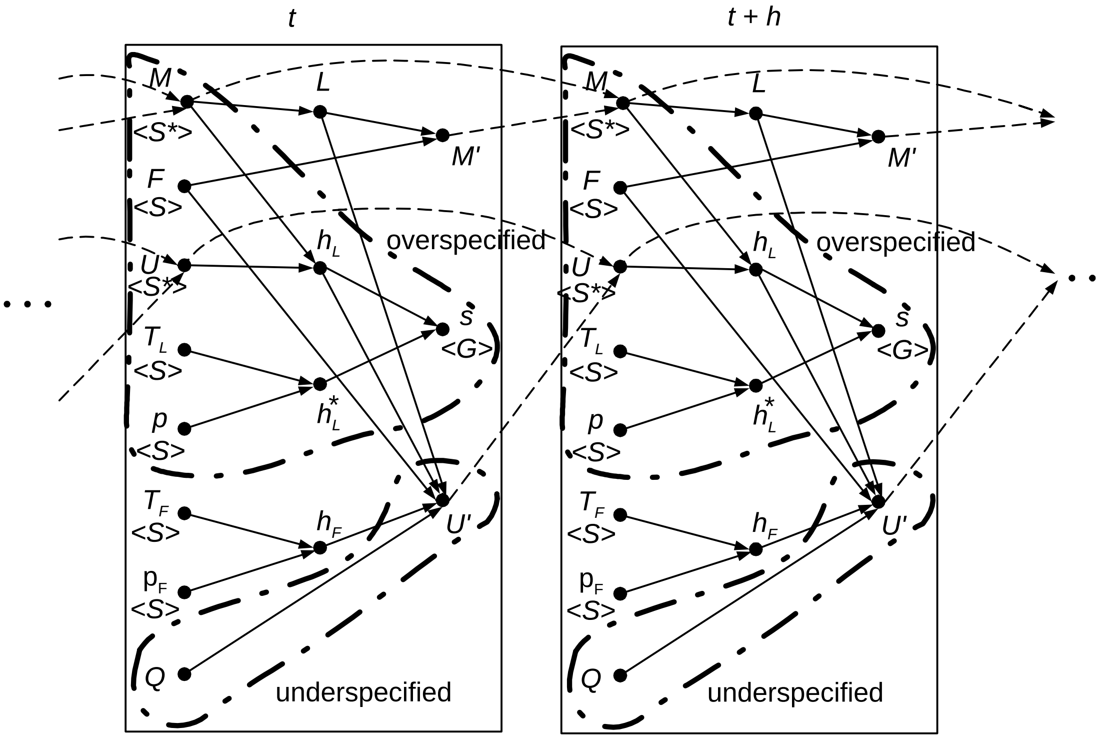

Fig.4 Representation graph of model M_1 with Specification 2 assuming first order explicit solution method

Example 2: Consider a simple tank system shown in *Fig.5*. The model (M_2) describing change of the level of liquid in the tank in standard form is the following [5]:

$$l = \int l'$$

$$l' = 1/A\cdot(F_1 - F_2)$$

$$F_1 = c_v\cdot (P_1 - P_2)^{1/2}$$

$$F_2 = c_v\cdot (P_2 - P_3)^{1/2}$$

$$s = P_2 - P_0 - \rho\, g\, l, \qquad s = 0$$

Here l denotes the level of the liquid in the tank, F_1 and F_2 the inlet and outlet flow rate respectively. P_0, P_1, P_2 and P_3 are pressures corresponding to *Fig.5*. The cross-section of the tank A, the valve parameter c_v, the density of the liquid ρ and the gravitational constant g are constant parameters.

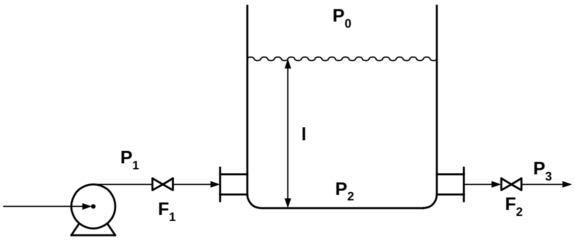

Fig.5 A simple tank system

The variables P_0, P_1, and P_3 are given as functions of time, the initial value l_0 is constant. The variables P_2, F_1, F_2 and l are to be calculated as functions of time.

The representation graph corresponding to above model and the variable type specifications are shown in *Fig.6*, while the reduced graph is shown in *Fig.7*. It can be seen that there exists a Menger-type complete linking on the reduced graph, which indicates that the differential index of the model is equal to one.

Representation of DAE-s assuming a higher-order explicit solution method

Higher order explicit solution methods, such as Runge-Kutta methods, are widely used for numerical solution of ordinary differential equations. As a characteristic example of such methods, let us consider a fourth order explicit Runge-Kutta method, where the value of the differential variable x at time $t+h$ is computed as

$$k_1 = f\,(t, x(t))$$

$$k_2 = f\,(t + h/2,\ x(t)+h/2\cdot k_1)$$

$$k_3 = f\,(t + h/2,\ x(t)+h/2\cdot k_2)$$

$$k_4 = f\,(t + h,\ \ x(t) + h\cdot k_3)$$

$$x(t + h) = x(t) + h/6\cdot (k_1 + 2\,k_2 + 2\,k_3 + k_4)$$

from the known value $x(t)$ at time t and from the right-hand side function f.

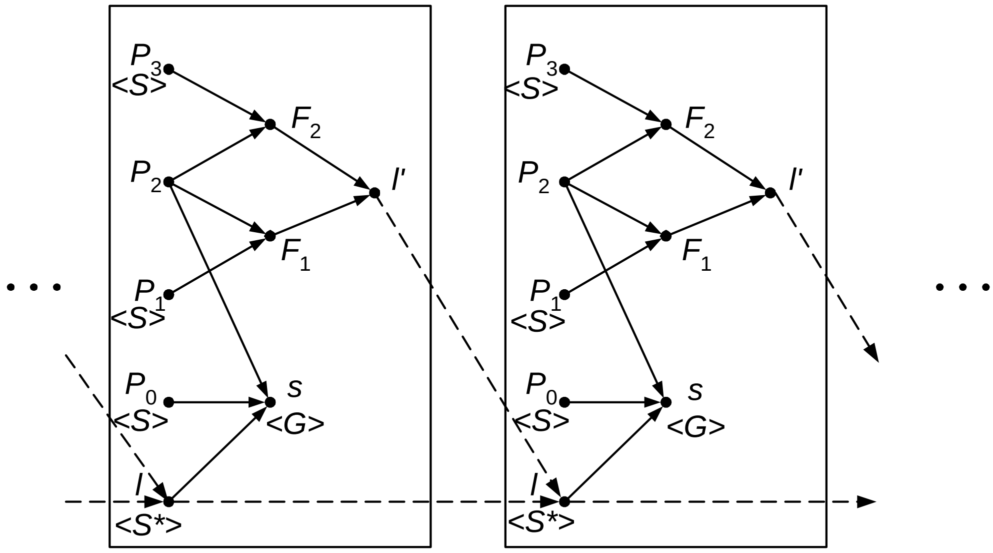

Fig.6 Representation graph of model M_2 assuming first order explicit solution method

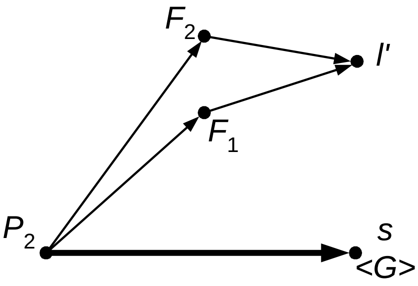

Fig.7 Reduced graph of model M_2

In order to construct the dynamic representation graph for this case, we construct three additional auxiliary graphs in between a pair of static graphs (main graphs) corresponding to an integration step from t to $t+h$. The internal structure of the auxiliary graphs is similar to that of the main graphs with the exception, that they do not contain the vertices associated to the differential variables and the arcs originating therefrom.

The arcs between the main and auxiliary graphs are determined by the numerical solution method, in the above fourth order Runge-Kutta method case the above set of equation generates the arcs shown in *Fig.8*. If there is an arc from the vertex of the differential variable x to another vertex of another variable x_k on the main graph corresponding to time t, then we direct further arcs from vertex x towards every auxiliary graph ending at their vertex x_k. The type labels of the vertices on the auxiliary graphs should be identical to that of the corresponding vertex on the main graphs. Futrhermore, we associate type <S*> labels to the vertices which have incoming arcs from only main or auxiliary graphs corresponding to previous time instances.

As an example, *Fig.9* shows the representation graph of model M_2 assuming a fourth order explicit Runge-Kutta method as its solution procedure.

Because of the way of the representation, the main graphs of a representation graph with higher order explicit solution method are identical with the static graphs of the earlier dynamic representation graph belonging to the Euler method. Only the connecting arcs between the main and the auxiliary graphs become more complicated as a consequence of the more complex numerical solution method. At the same time, the algebraic subgraphs of the main and auxiliary graphs are identical to the earlier algebraic subgraphs belonging to the Euler method. Therefore we can state the following propositions.

Proposition 1. The differential index of a semi-explicit DAE model M is equal to one if and only if there exists a Menger-type complete linking on the reduced algebraic subgraph of its dynamic representation graph independently of the fact if a first or higher order one-step explicit method is applied for the numerical solution.

The reduction of the main and auxiliary graphs can be performed following the same rules as we have seen earlier in the case of applying the explicit Euler method for the numerical solution. Then the following statement is valid.

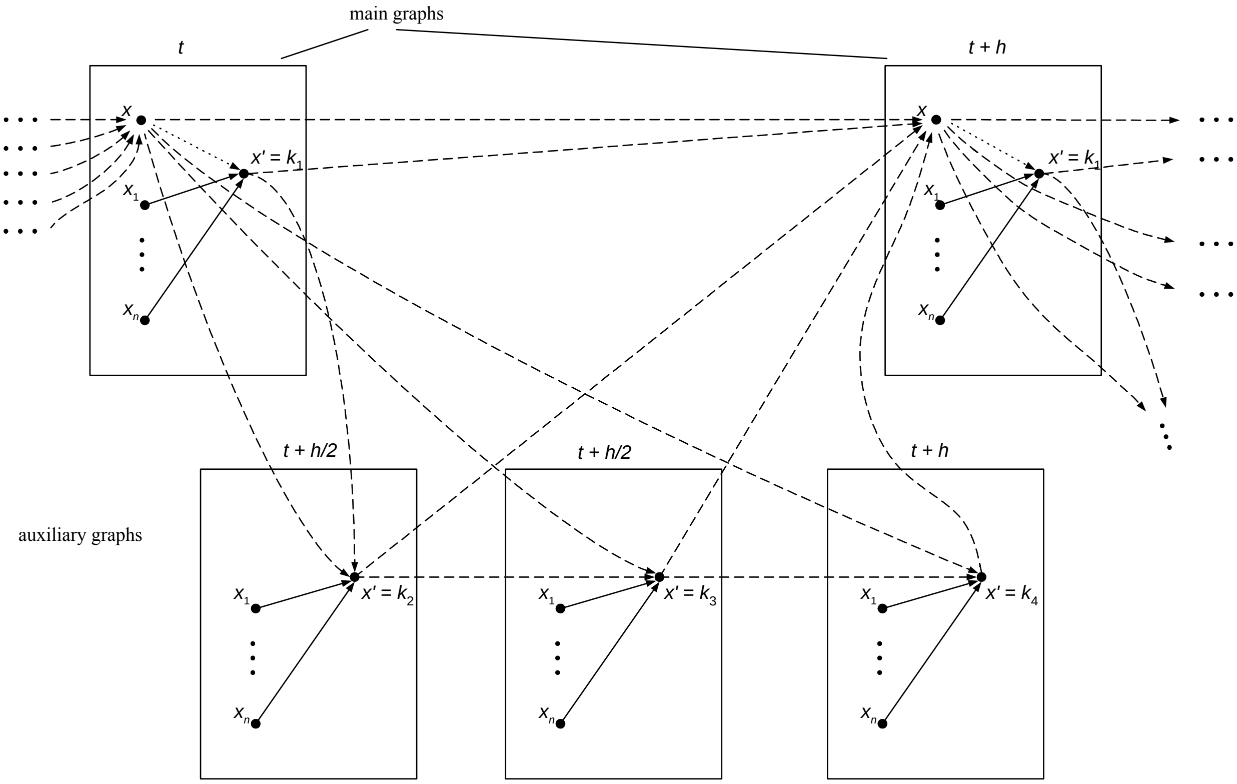

Fig.8 Dynamic representation graph assuming fourth order explicit Runge-Kutta method

Proposition 2. By performing the reduction of any of the main or auxiliary graphs, the same reduced graph is obtained, and it is identical to the reduced graph of the model when assuming the explicit Euler method as a numerical solution procedure.

Proof. The statement follows from the structure of the dynamic representation graph and from the way the type-labels are associated to the vertices on the main and auxiliary graphs.

In agreement of the above proposition, the reduced graphs of the main and auxiliary graphs of model M_2 shown in *Fig.9* are identical to the reduced graph seen in *Fig.7*.

It is important to note that there will be overspecified and underspecified subgraphs on each of the main and auxiliary graphs in the case of higher index models assuming higher order explicit numerical solution methods, similarly to the case when the simple explicit Euler method was assumed.

Representation of DAE-s assuming an implicit solution method

If we consider a single-step implicit solution method (for example the implicit Euler method) for the numerical solution procedure of the differential equations, then the value of differential variable x at time $t+h$ is calculated as follows:

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

where h is the step length.

In this case the modified structure of the dynamic representation graph is depicted in *Fig.10*. It can be seen that the algebraic part of the representation graph does not change compared to the case when an explicit solution method was assumed, hence the following proposition holds.

Proposition 3: The differential index of model M is equal to 1 if and only if, there is a Menger-type complete linking on the reduced graph of the algebraic part of the dynamic representation graph belonging to model M, independently of either an explicit or an implicit single step numerical solution procedure is applied for the solution of the differential equations.

Assume that the values of all derivative variables x' directly or indirectly depend on the corresponding differential variable x in the model. This assumption holds for stable models. As a result of this assumption a so called circle of calculation is obtained in the representation graph belonging to the differential variables according to the following definition.

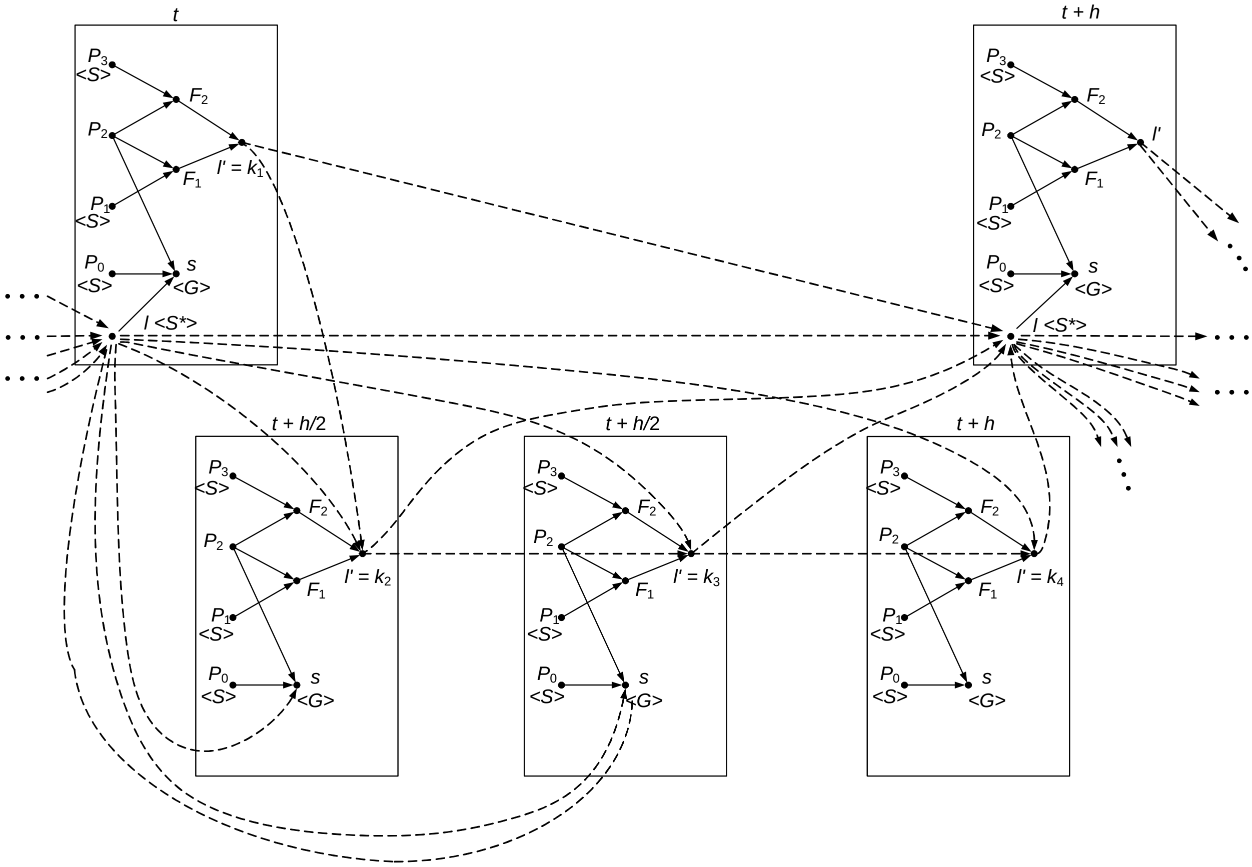

Fig.9 Representation graph of model M_2 assuming fourth order explicit Runge-Kutta method

Definition: *The circle of calculation belonging to the differential variable x* is a directed circle path, which contains the vertices *x* and *x*' and this directed circle is either

1. present on the representation graph (*type* 1 *circle of calculation*), or
2. if it is absent, then the basic undirected circle (when directions of arcs are not to be considered) can be found on the representation graph and the circle of calculation can be obtained by changing the direction of some arcs (*type* 2 *circle of calculation*). The direction of arc $x_1 \rightarrow x_2$ can be changed, if there exists a Menger type complete linking on the reduced algebraic subgraph, which contains it.

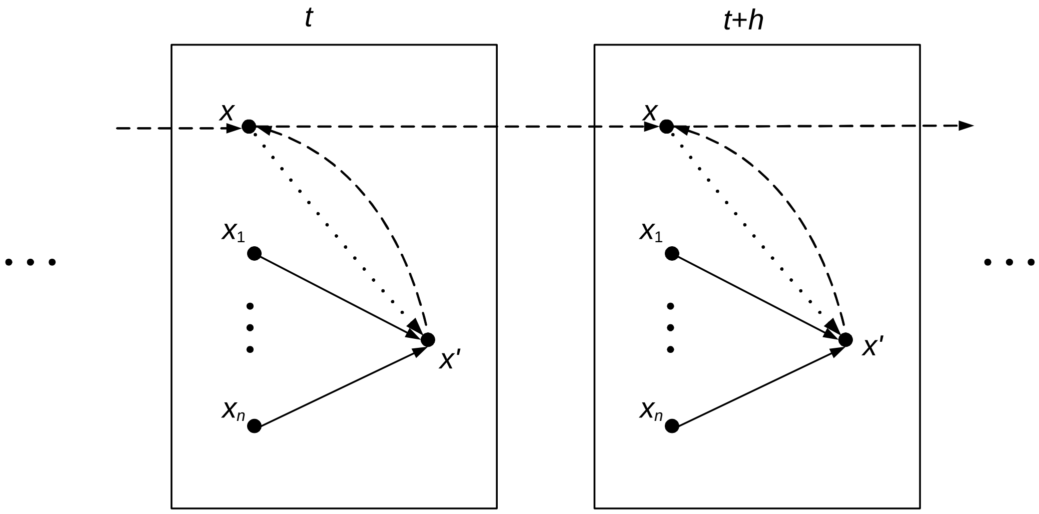

Fig.10 Dynamic representation graph assuming single implicit solution method

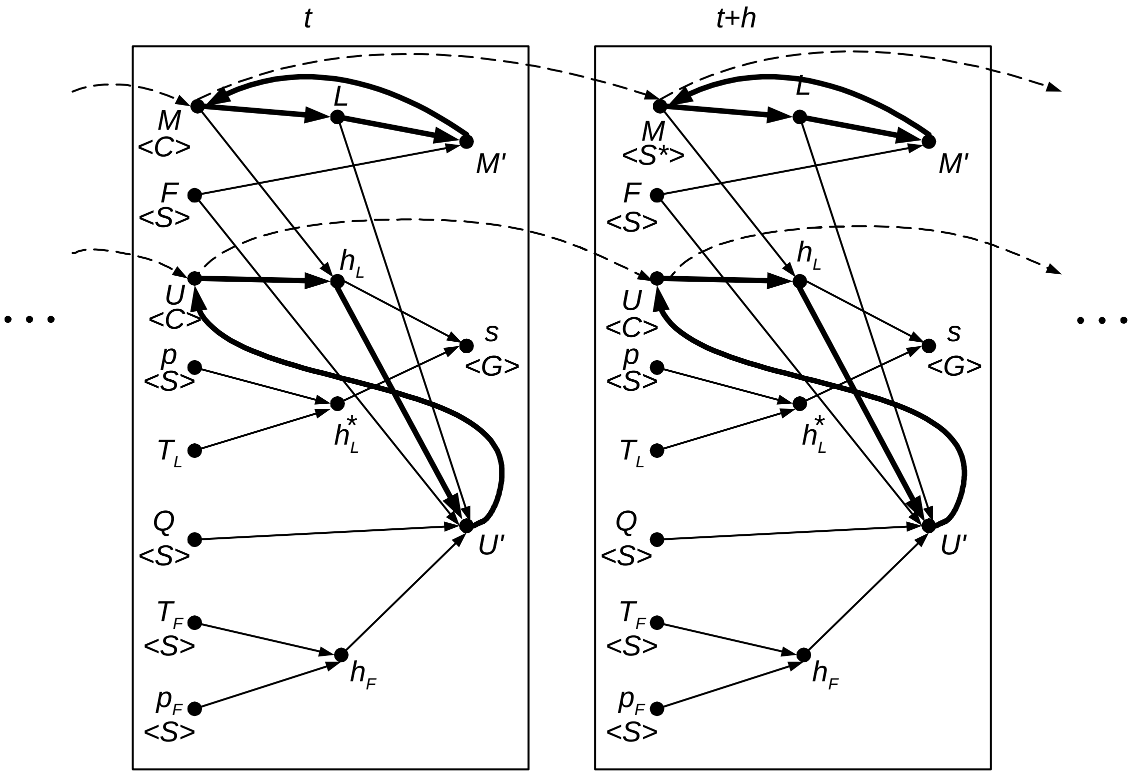

Fig.11 Representation graph of model M_1 with Specification 1 assuming an implicit solution method (→ circle of calculation)

The representation graphs of models in Examples 1. and 2. assuming implicit method for the numerical solution of differential equations can be seen in *Figs.11* and *12*. Type 1 circles of calculation can be seen in *Fig.11*, where the directed circles containing *M* and *M*' or *U* and *U*' *are present* in the representation graph. A Type 2 circle of calculation can be found in *Fig.12*, where the directed circle belonging to variable *l* can be obtained by *changing the direction* of arc $P_2 \to s$. According to the directed circle of calculation the values of some variables (for example *x* and *x*') can be calculated at any time *t* by iteration.

Therefore we propose a new type of assignment for all differential variables:

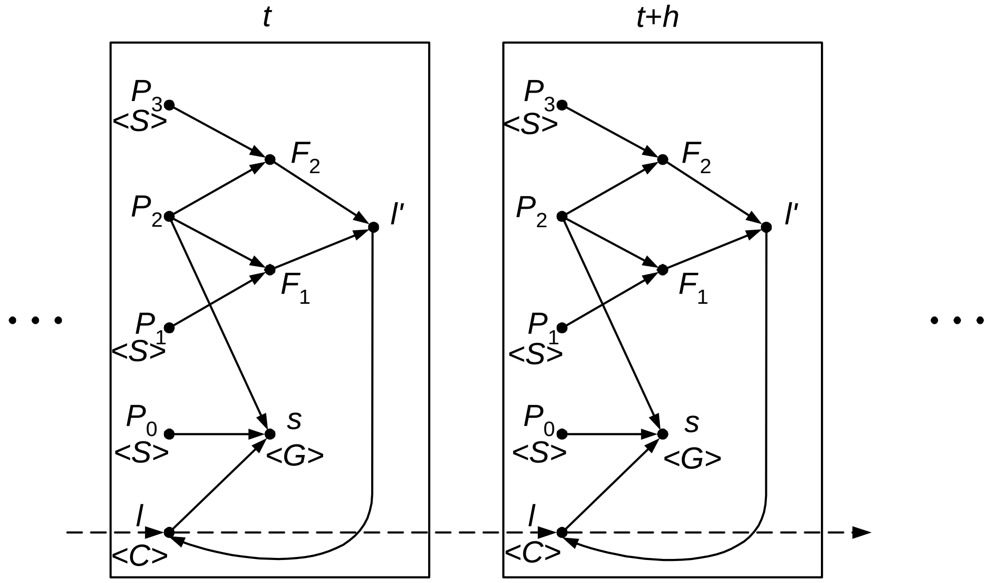

Fig.12 Representation graph of model M_2 assuming an implicit solution method

- <C> *circle-type variables*: Let <C> be the label of all differential variables in the representation graph, which indicates that the values of these variables can be calculated by iteration only. These labels can be seen in *Figs.11* and *12*.

Reduction of representation graphs assuming an implicit solution method

The reduction of representation graphs can be performed in this case *in two steps*:

The primary reduction is the same as in the case of explicit solution methods: *the primary reduced graph* can be obtained by omitting all vertices labelled primarily, secondarily, etc., by type <S> and all arcs starting from them.

The primary reduced graphs of the investigated models can be seen in *Figs.13* and *14*. The primary reduction is directed both to the implicit part of the model and to the iterations belonging to the differential variables. In the first case (*Fig.13*) iterations can be calculated independently of the implicit calculation and the sequence of the calculation steps can be seen, too. In the second case (*Fig.14*) the implicit calculation of the algebraic variable P_2 is a part of the iteration, hence the value of P_2 can be obtained iteratively, too.

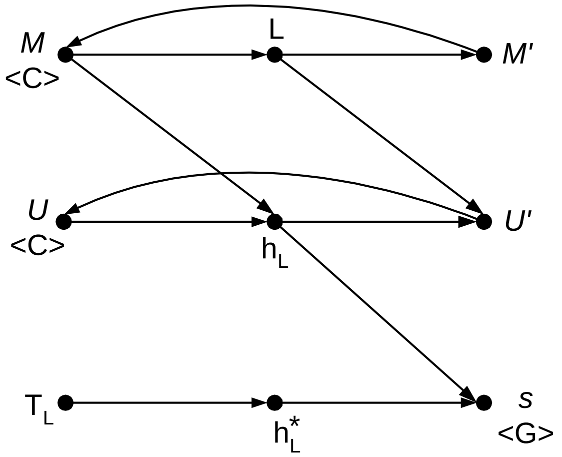

Fig.13 Primary reduced graph of model M_1 with Specification 1

The iterations belonging to differential variables come from the implicit numerical solution method and not from the structure of the models. If we do not want to consider these iterations, a *secondary reduction* can be constructed: the *secondary reduced graph* can be obtained if we omit the arcs $x' \rightarrow x$ belonging to all differential variables and the labels <*C*> are treated similarly to the labels <*S*> during the reduction. The secondary reduction is directed to the implicit part of the model, only.

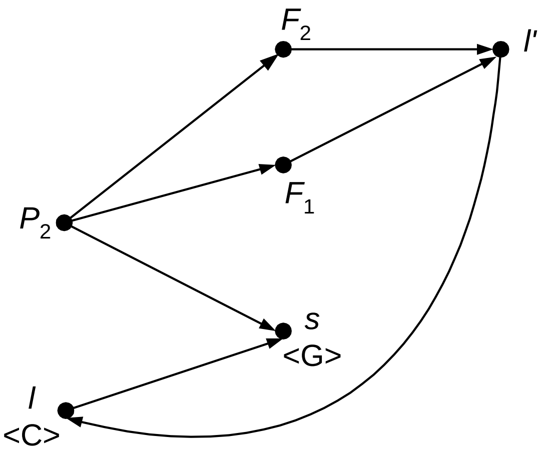

Fig.14 Primary reduced graph of model M_2

Proposition 4: The secondary reduction yields the same reduced graph as it can be obtained by assuming a single step explicit numerical solution procedure.

Proof: It follows directly from the structure of dynamic representation graphs and from the procedure of reduction.

In the case of higher index models with a single step implicit solution method, the representation graph contains an overspecified and underspecified subgraph, in the same way, as in the case of explicit solution methods. The algorithm for determination of the differential index can be performed similarly, too.

Conclusion

The effect of the numerical solution procedure on the structural analysis of dynamic lumped models described by semi-explicit differential and algebraic equations was investigated in this paper. The most important solvability properties of models were determined by graph-theoretical methods using the dynamic representation graph of the models. We compared the structure of dynamic representation graphs in the case of explicit and implicit solution methods for solving the differential equations.

It was shown that the representation graph with both higher order explicit and implicit solution methods has similar properties both in case of index 1 and higher index models as compared to first order explicit single-step numerical solution procedures.

Hence the properties of 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.

REFERENCES

1. MUROTA K.: Systems Analysis by Graphs and Matroids, Springer-Verlag, Berlin, 1987
2. LEITOLD A. and HANGOS K.M.: Computers and Chemical Engineering, *2001,* 25, 1633-1646
3. HANGOS K.M. and CAMERON I.T.: Process Modelling and Model Analysis, Academic Press, New York, London 2001
4. LEITOLD A. and HANGOS K.M.: Hungarian Journal of Industrial Chemistry, *2002,* 30(1), 61-71
5. LEITOLD A.: Analysis of Dynamic Process Models Using Computer Science Methods, PhD Dissertation, University of Eötvös Lóránd, Budapest, 2002
6. IRI M., TSUNEKAWA J. and YAJIMA K.: Information Processing, *1972*, 71. Vol. **2**. 1150-1155
7. BRENAN K.E., CAMPBELL S.L. and PETZOLD L.R.: Numerical Solution of Initial Value Problems in Differential – Algebraic Equations, North-Holland, New York, 1989