Proceeding of the 2004 American Control Conference
Boston, Massachusetts June 30 - July 2, 2004

Row by Row Structure Simplification

A. Yousefi B. Lohmann M. Buttelmann

***Abstract*— The system matrices optimization order reduction method calculates models of lower order but of high complexity. Secondary conditions can be formulated to calculate reduced systems with low complexity. One of the methods to find these secondary conditions is exploiting genetic algorithm in order to perform a global search within the search space and in advanced methods tabu search algorithm is used to perform a local search within the region found by the genetic algorithm. In this paper another solution is proposed to simplify the complexity of the reduced order system and to reduce the computation effort.**

I. INTRODUCTION

Most of the models based on physical phenomena have simple structures and consist of sparse matrices, but after applying the order reduction methods the reduced order system usually doesn't preserve the original simple structure. In this paper we deal with the system matrices optimization method [8], [7], which is exploited for a restricted type of nonlinear systems with following representation:

$$\mathcal{S}: \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{F}\mathbf{g}(\mathbf{x}(t), \mathbf{u}(t)) \\ \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) \end{cases} \tag{1}$$

where $\mathbf{g}(\mathbf{x}(t), \mathbf{u}(t))$ comprises the nonlinear part of the differential equations (for other famous methods of nonlinear order reduction refer to [4], [12], [11]). Starting from (1) the task of order reduction in system matrices optimization method is to find a system of lower order $\tilde{n}$ which delivers an approximation of the dominant state variables. These dominant state variables are chosen by the designer and are combined in the vector $\mathbf{x_{do}}$ which is related to the original vector x by $\mathbf{x_{do}} = R\mathbf{x}$. Based on the given system (1) and the dominant state variables, the system matrices optimization method calculates optimal matrices $\mathbf{E} = [\tilde{\mathbf{A}}, \tilde{\mathbf{B}}, \tilde{\mathbf{F}}]$ and $\mathbf{W}$ such that they optimally fit the snapshots of the dominant state variables of the original system in the sense of Euclidian norm [8], [7]. By assuming $\tilde{\mathbf{x}}$ as the approximation of $\mathbf{x_{do}}$, then the reduced system is set up as follow:

$$\mathcal{S}_{reduced}: \begin{cases} \dot{\tilde{\mathbf{x}}}(t) = \tilde{\mathbf{A}}\tilde{\mathbf{x}}(t) + \tilde{\mathbf{B}}\mathbf{u}(t) + \tilde{\mathbf{F}}\mathbf{g}(\mathbf{W}\tilde{\mathbf{x}}, \mathbf{u}) \\ \mathbf{y}(t) = \tilde{\mathbf{C}}\tilde{\mathbf{x}}(t) \end{cases} \tag{2}$$

Accordingly, the vector $\mathbf{g}$ of the nonlinearities is taken over from the original system (1) into the reduced order system and no additional nonlinearities are introduced. A disadvantage of this method is the fact that typically all elements of the matrices $\mathbf{E}$ and $\mathbf{W}$ are non-zero. So it is more desirable to not only reduce the order of the original system, but also keep the reduced order system as simple as possible. Our main goal in this paper is to simplify the reduced order nonlinear system in order to increase the sparsity (more number of zeros) of system matrices. For structure simplification also some methods have been developed that we review them in Section 2. In Section 3 we present a new simple method that enormously decreases the size of search space for structure simplification. In Section 4 we show the results of applying the new method and compare them to the old methods and Section 5 contains concluding remarks.

II. METHODS OF STRUCTURE SIMPLIFICATION

A. Secondary conditions in optimization problems

The general linear equality constrained minimization problem can be written as follow[1]:

Find X such that it minimizes $\|\mathbf{A}X - \mathbf{B}\|$ *and fulfills the equality* $\mathbf{C}X = \mathbf{D}$.

where $\mathbf{A}$ is an m-by-n matrix ($m \leq n$) and $\mathbf{C}X = \mathbf{D}$ defines a linear equality constraint. In [5], [3] some methods for solving this optimization problems are proposed and used in [8], [7], [6] to find the matrices $\mathbf{E}$ and $\mathbf{W}$ of the reduced order model (2). The ability to solve optimization problems with constraints can be used to combine some additional features to system matrices optimization method for order reduction and structure simplification. In [8] this basic idea is used for improving the steady state performance, and in [1], [2] this idea is exploited for structure simplification by introducing new secondary conditions.

B. Application of secondary conditions in structure simplification

As it is mentioned in the previous section, in [8], [7] it is shown that it is possible to force any element in system matrices to any desirable value by expressing some secondary constraints in the optimization problem. In [1] the complexity constraints on the reduced model is formulated and expressed by secondary conditions as follow:

$$\mathbf{k_E}\,\mathbf{E}\,\mathbf{h_E} - \mathbf{l_E} = \mathbf{0^T}, \qquad \mathbf{k_W}\,\mathbf{W}\,\mathbf{h_W} - \mathbf{l_W} = \mathbf{0^T} \tag{3}$$

with prescribed vectors $\mathbf{l}$, $\mathbf{k}$ and $\mathbf{h}$. Conditions of this type can be integrated into the optimization procedure in order to force some elements in system matrices deliberately to zero. For example the choice $\mathbf{h_E} = [1, 0, \ldots, 0]^T$ $\mathbf{k_E} = [0, 1, \ldots, 0]$ and $\mathbf{l_E} = [0]$ forces the first element in the second row of matrix $\mathbf{E}$ to zero. For solving this type of secondary constraints vectorization and Kronecker production is needed that results in very big matrices that restricts its applicability to large scale systems, for more details refer to [6].

After applying the system matrices optimization method, usually the system matrices of the reduced order system are full of nonzero elements and usually some of the elements are so small. Thus the conjecture is the question that "Are they really valid numbers or they are just results of some errors or round off in numerical computations?". The authenticity of this guess can be checked in two ways, first by replacing some of the small numbers by zero and check the performance of the resulted system, second by forcing the related elements to zero by solving the original optimization problem with appropriate constraints. It is trivial that the second way is more rational and yields to better results, but the problem is that often there are a large number of different choices to carry out this task. The number of different options is related directly to the size and complexity of the original system. For instance in our example (hydropneumatic vehicle suspension) there are $2^{99} \approx 6 \times 10^{29}$ different ways to replace the elements of matrix $\mathbf{E}$ with zero. In fact it is not usually possible to check every single option independently and find the best solution, therefore some methods for pioneered searching such as genetic algorithm or tabu search is demanded. In the succeeding subsections we elaborate these methods and existing techniques for structure simplification.

1) Genetic algorithm: With respect to the previous section, suitable choices of $\mathbf{l}$, $\mathbf{k}$ and $\mathbf{h}$ are needed as candidates for the optimal simplified reduced order system. In [1], [2] the genetic algorithm is used to search between different options. In this method each option is presented in form of a bit string (so-called an individual) which is a row vector that only consists of ones and zeros. In an individual ones

[1] The notation $\|\mathbf{A}\|$ in this paper is the square of the Euclidian (Frobenius) norm of matrix A and is defined as $\sum diag(\mathbf{A^T A})$

0-7803-8335-4/04/$17.00 ©2004 AACC

show the places, that zeros should be inserted in matrices $\mathbf{E}$ or $\mathbf{W}$. For instance suppose a 2 by 3 matrix E as follow:

$$E = \begin{pmatrix} 2 \times 10^{-17} & 2 & 10^{-16} \\ 5 & 6 \times 10^{-17} & -8 \end{pmatrix}$$

The constraint for the genetic algorithm that forces the three elements $e_{1,1}, e_{1,3}$ and $e_{2,2}$ of matrix E to zero is presented by the following row vector $\mathbf{g^T}$:

$$\mathbf{g^T} = [\ \underbrace{1 \quad 0 \quad 1}_{1_{st} \text{ row of } E} \qquad \underbrace{0 \quad 1 \quad 0}_{2_{nd} \text{ row of } E}\ ]$$

Consequently every row vector that has the length six and contains only ones and zeros corresponds to a simplified matrix E. The problem in here is that how many zeros and where they should be inserted. In [1], [2] the number and the positions of zeros in the matrices E and W are computed by the genetic algorithm and the non-zero elements are afterwards computed by the reduction method and the performance measurement (so-called *fitness function*) is calculated by simulation of each reduced model with P different typical inputs u(t) and by comparison with the behavior of the original system (1). This task is carried out by evaluating the area between the time curve of each dominant state variable $\mathbf{x_i}(t)$ in the original system and the reduced order system, which is an indicator for the quality of order reduction. The fitness function F is shown in (4).

$$F = \underbrace{\sum_{j=1}^{P} \sum_{i=1}^{n} \frac{\int_0^{t_{N_j}} (x_i(t) - \mathbf{w}_i\, \tilde{\mathbf{x}}(t))^2 dt}{\int_0^{t_{N_j}} x_i^2(t)}}_{\text{Approximation error}} - \underbrace{k \cdot (\textbf{no. of zero elements})}_{\text{Model-complexity}} \qquad (4)$$

where $\mathbf{w_i}$ is the i_{th} row of $\mathbf{W}$, t_{N_j} is the simulation time for the snapshots of the j_{th} typical input. Optionally, the model complexity of the reduced system is also considered in this fitness function. The number of zero-elements in E and W (multiplied with a weighting factor k) is subtracted from the value of the model approximation, so that systems with less complexity are ranked better than systems with the same approximation quality and higher complexity. In this method the starting population is selected randomly and it is also mentioned that sometimes some pre-specified structures from the engineer's know-how, can be formulated as an individual that can be added to the starting population. The tournament selection, two point cross over and normal mutation are used as genetic operators [9] and the genetic algorithm produces new generations with better and better individuals as long as the breaking condition which is the number of produced generations is not fulfilled.

2) Tabu search: Tabu search is a kind of iterative search and it is able to eliminate local minima and to search areas beyond a local minimum [10]. Therefore, it has the ability to find the global minimum of a multi-modal search space. In [2] this search method also used for structure simplification by calculating every solution in the neighborhood of the current solution and selecting the best one for the next iteration step. This means that the tabu search algorithm selects the way that produces the most improvement or the least deterioration. To avoid cycling problems if a current visited solution is selected again, a tabu list is used. The advantage of tabu search is the ability to find better solutions in local regions nearby the current solution, but because of the local search characteristic, to perform a global search for complex problems with many solutions in the neighborhood of the current solution, the calculation time is very high. Therefore in [2] it is recommended to solve the introduced problem of structure simplification in two steps by using the strong points of both search algorithms. First by finding the region containing the global optimum or at least a very good suboptimal in the large search space by genetic algorithm then finding the best solution in this local region by tabu search algorithm.

III. Row by row method

In this section we prove that the structure simplification problem can be broken into small independent subproblems. This idea will reduce the search space in size such that for systems which have less than twenty dominant state variables, it is not necessary to use search engines like the genetic algorithm or tabu search and the optimal answer can be evaluated with less computation effort. Assume that matrices $\chi_{\mathbf{do}}$, $\dot{\chi}_{\mathbf{do}}$, $\mathbf{\Psi}$ and $\mathbf{\Gamma}$ are the snapshots of the original system for typical inputs which respectively show the numerical values of dominant state variables, their derivatives, inputs and nonlinear part as it is shown in (5).

$$\begin{aligned} \chi_{\mathbf{do}} &= [\mathbf{x_{do}}(t_1) \cdots \mathbf{x_{do}}(t_N)],\ \mathbf{\Psi} = [\mathbf{u}(t_1) \cdots \mathbf{u}(t_N)] \\ \dot{\chi}_{\mathbf{do}} &= [\dot{\mathbf{x}}_{\mathbf{do}}(t_1) \cdots \dot{\mathbf{x}}_{\mathbf{do}}(t_N)],\ \mathbf{\Gamma} = [\mathbf{g}(t_1) \cdots \mathbf{g}(t_N)] \end{aligned} \qquad (5)$$

In the following theorem we show that the original optimization problem can be splitted into smaller optimization problems.

Theorem 1: In the system matrices optimization method solving the following optimization problem:

$$\min_{\mathbf{E}} \|\, \dot{\chi}_{\mathbf{do}} - \underbrace{[\tilde{A} \quad \tilde{B} \quad \tilde{F}]}_{\mathbf{E}} \underbrace{\begin{bmatrix} \chi_{do} \\ \Psi \\ \Gamma \end{bmatrix}}_{\mathbf{M}} \| \qquad (6)$$

is equivalent to solving $\tilde{n}$ independent optimization problems as follow:

$$\min_{\mathbf{e_i^T}} \|\, \dot{\mathbf{x}}_{\mathbf{do_i}}^{\mathbf{T}} - \underbrace{[\tilde{A}_i \quad \tilde{B}_i \quad \tilde{F}_i]}_{\mathbf{e_i^T}} \underbrace{\begin{bmatrix} \chi_{do} \\ \Psi \\ \Gamma \end{bmatrix}}_{\mathbf{M}} \|, \qquad i = 1, 2, \ldots, \tilde{n} \qquad (7)$$

where $\dot{\mathbf{x}}_{\mathbf{do_i}}^{\mathbf{T}}$ is the snapshots of derivative of the i_{th} state variable and $\tilde{A}_i,\ \tilde{B}_i\, and\ \ \tilde{F}_i$ are the i_{th} row of the reduced order system matrices $\tilde{A},\ \tilde{B}\, and\ \ \tilde{F}$ respectively, that should be evaluated.

In other words Theorem1 indicates that the reduced order system matrices can be evaluated completely using (6) or row by row using (7) and the results are exactly the same.

Proof: Suppose that the matrix $\mathbf{E_{opt}} = \begin{pmatrix} \mathbf{e_{opt_1}^T} \\ \vdots \\ \mathbf{e_{opt_n}^T} \end{pmatrix}$ is the optimal solution of least square problem (6) and $\tilde{\mathbf{e}}_{\mathbf{opt_i}}$ is the optimal solution of (7). Our assumption permits us to write:

$$\begin{aligned} \min_{E} \|\, \dot{\chi}_{\mathbf{do}} - \mathbf{EM}\| &= \|\, \dot{\chi}_{\mathbf{do}} - \mathbf{E_{opt}\, M}\| \\ &= \left\| \dot{\chi}_{\mathbf{do}} - \begin{pmatrix} \mathbf{e_{opt_1}^T} \\ \vdots \\ \mathbf{e_{opt_n}^T} \end{pmatrix} \mathbf{M} \right\| \\ &= \left\| \begin{pmatrix} \dot{\mathbf{x}}_{\mathbf{do_1}}^{\mathbf{T}} - \mathbf{e_{opt_1}^T M} \\ \vdots \\ \dot{\mathbf{x}}_{\mathbf{do_n}}^{\mathbf{T}} - \mathbf{e_{opt_n}^T M} \end{pmatrix} \right\| \\ &= \sum_{i=1}^{n} \|\, \dot{\mathbf{x}}_{\mathbf{do_i}}^{\mathbf{T}} - \mathbf{e_{opt_i}^T M}\| \\ &= \sum_{i=1}^{n} J_i, \qquad i = 1, \cdots, \tilde{n} \end{aligned} \qquad (8)$$

and

$$\begin{aligned}\min_{e_i^T} \| \dot{\mathbf{x}}_{\mathbf{do_i}}^{\mathbf{T}} - \mathbf{e_i^T M}\| &= \| \dot{\mathbf{x}}_{\mathbf{do_i}}^{\mathbf{T}} - \tilde{\mathbf{e}}_{\mathbf{opt_i}}^{\mathbf{T}}\, \mathbf{M}\| \\ &= \tilde{J}_i, \quad i = 1, \cdots, \tilde{n} \qquad (9)\end{aligned}$$

For proving Theorem1 it is sufficient to prove that:

$$\mathbf{e_{opt_i}^T} = \tilde{\mathbf{e}}_{\mathbf{opt_i}}^{\mathbf{T}} \quad \text{and} \quad J_i = \tilde{J}_i, \quad i = 1, \cdots, \tilde{n}$$

Now we suppose that $J_i \neq \tilde{J}_i$ and we show that it leads to contradiction. Thus we have two cases as follows:
If we suppose $J_i > \tilde{J}_i$ then we build the matrix

$$\hat{\mathbf{E}}_{\mathbf{opt}} = \begin{pmatrix} \mathbf{e_{opt_1}^T} \\ \vdots \\ \tilde{\mathbf{e}}_{\mathbf{opt_i}}^{\mathbf{T}} \\ \vdots \\ \mathbf{e_{opt_n}^T} \end{pmatrix}, \text{ then}$$

$$\begin{aligned}\| \dot{\chi}_{\mathbf{do}} - \hat{\mathbf{E}}_{\mathbf{opt}}\mathbf{M}\| &= J_1 + \cdots + \tilde{J}_i + \cdots + J_n \\ &< \sum_{i=1}^{n} J_i, \quad i = 1, \cdots, \tilde{n}\end{aligned}$$

that it contradicts (8).
Another case is that we suppose $J_i < \tilde{J}_i$ then similarly:

$$\| \dot{\mathbf{x}}_{\mathbf{do_i}}^{\mathbf{T}} - \mathbf{e_{opt_i}^T M}\| = J_i < \tilde{J}_i, \quad i = 1, \cdots, \tilde{n}$$

that it contradicts (9), therefore

$$J_i = \tilde{J}_i, \quad i = 1, \cdots, \tilde{n} \qquad (10)$$

According to the fact that the answer of (9) is unique (provided $\mathbf{M}$ is full rank) and we have proved that:

$$\begin{aligned}\min_{e_i^T} \| \dot{\mathbf{x}}_{\mathbf{do_i}}^{\mathbf{T}} - \mathbf{e_i^T M}\| &= \| \dot{\mathbf{x}}_{\mathbf{do_i}}^{\mathbf{T}} - \tilde{\mathbf{e}}_{\mathbf{opt_i}}^{\mathbf{T}}\mathbf{M}\| \\ &= \tilde{J}_i = J_i \\ &= \| \dot{\mathbf{x}}_{\mathbf{do_i}}^{\mathbf{T}} - \mathbf{e_{opt_i}^T M}\|, \quad i = 1, \cdots, \tilde{n}\end{aligned}$$

therefore

$$\mathbf{e_{opt_i}^T} = \tilde{\mathbf{e}}_{\mathbf{opt_i^T}}, \quad i = 1, \cdots, \tilde{n}$$

which complete the proof of Theorem1, `Q.E.D.`

With respect to Theorem1 the original optimization problem can be replaced by smaller optimization problems for each row and the optimal solution can be evaluated using (11).

$$\mathbf{e_{opt_i}^T} = \dot{\mathbf{x}}_{\mathbf{do_i}}^{\mathbf{T}} \mathbf{M^T}(\mathbf{MM^T})^{-\mathbf{1}} \qquad (11)$$

and the secondary condition can be simplified by changing the formulation for each row of matrices $\mathbf{E}$ and $\mathbf{W}$ as it is shown in (12).

$$\mathbf{e_i^T h_{e,i}} - \mathbf{l_{e,i}} = \mathbf{0^T}, \quad \mathbf{w_i^T h_{w,i}} - \mathbf{l_{w,i}} = \mathbf{0^T} \qquad (12)$$

where $\mathbf{e_i^T}$ is the i_{th} row of matrix $\mathbf{E}$ and $\mathbf{w_i^T}$ is the i_{th} row of matrix $\mathbf{W}$. The optimization problem (9) with secondary conditions of type (12) results in optimal solution (13), [6], [8]:

$$\begin{aligned}\mathbf{e_{opt_i}^T} = \dot{\mathbf{x}}_{\mathbf{do_i}}^{\mathbf{T}} \mathbf{M^T}(\mathbf{MM^T})^{-\mathbf{1}} + (\mathbf{l_{e,i}} - \dot{\mathbf{x}}_{\mathbf{do_i}}^{\mathbf{T}} \mathbf{M^T}(\mathbf{MM^T})^{-\mathbf{1}}\mathbf{h_{e,i}}) \cdot \\ \cdot (\mathbf{h_{e,i}^T}(\mathbf{MM^T})^{-\mathbf{1}}\mathbf{h_{e,i}})^{-\mathbf{1}}\mathbf{h_{e,i}^T}(\mathbf{MM^T})^{-\mathbf{1}} \\ (13)\end{aligned}$$

The advantages of this task are increment in computation accuracy and applicability of this method to larger systems. Another advantage of this replacement is its application in structure simplification. The same result as in (3) will be achieved just by choosing $\mathbf{h_{e,2}} = [1, 0, \ldots, 0]^T$, and $\mathbf{l_{e,2}} = [0]$, hence the corresponding element of matrix $\mathbf{E}$ becomes zero. The second method has less complexity and needs less computation effort in comparison with one introduced in [1]. For instance for a reduced system of order twenty, in the last method there are approximately $2^{500} \approx 3,27 \times 10^{150}$ different ways for simplification (forcing zeros in system matrices), but with our new method for each row there are approximately $2^{25} \approx 33,5 \times 10^6$ different ways, so that all different options can be examined (by normal computers) in acceptable time. The same optimization scheme as in (7) and (13) can be used to find the matrix $\mathbf{W}$.

For the subsequent considerations it is sufficient to summarize that if the complexity constraints $\mathbf{l}$ and $\mathbf{h}$, the original model (1), and the matrix R are given, the reduction method delivers optimal matrices $\mathbf{E}$ and $\mathbf{W}$ fulfilling the constraints and approximating the behavior of the original model.

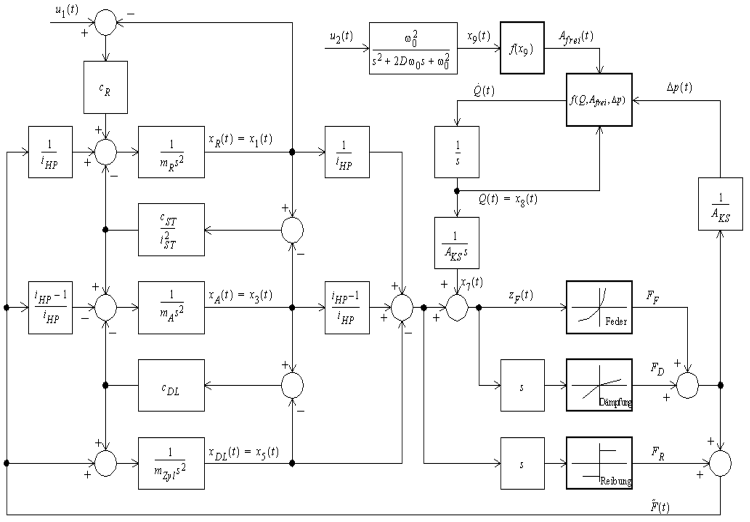

Fig. 1. block diagram of hydropneumatic vehicle suspension model

IV. Example

All above algorithms were tested on the example shown in Fig (1), which is described in more detail in [6], [7]. It is an active hydropneumatic vehicle suspension for passenger cars. The model of the original system is of order 10. The system matrices are shown below:

$$\mathbf{A} = \begin{bmatrix} 0 & 1,0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -4127 & 0 & 127,5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1,0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 12,75 & 0 & -1613 & 0 & 1600 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1,0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 533333 & 0 & -533333 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1,244 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -162 & 1,0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -8100 & 0 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} 0 & 0 \\ 4000 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 8100 \end{bmatrix}, \mathbf{F} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ -0,0148 & -0,0148 & -0,0148 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -0,000518 & -0,000518 & -0,000518 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0,667 & 0,667 & 0,667 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 17,5 & -71,43 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

There are seven dominant state variables ($x_1, x_2, x_3, x_4, x_7, x_8$ and x_9), so a reduced model of order 7 can be calculated. The order reduction method without structure simplification calculates the three system matrices as follow:

$$\tilde{\mathbf{A}} = \begin{bmatrix} 5{,}9 \cdot 10^{-13} & 1{,}0 \cdot 10^{0} & -3{,}8 \cdot 10^{-13} & 5{,}2 \cdot 10^{-14} & 6{,}7 \cdot 10^{-13} & 7{,}2 \cdot 10^{-14} & 2{,}5 \cdot 10^{-14} \\ -4{,}1 \cdot 10^{3} & -2{,}4 \cdot 10^{-11} & 1{,}3 \cdot 10^{2} & 1{,}2 \cdot 10^{-11} & -1{,}4 \cdot 10^{-9} & -7{,}9 \cdot 10^{-11} & 1{,}0 \cdot 10^{-11} \\ 3{,}1 \cdot 10^{-13} & -1{,}5 \cdot 10^{-13} & -8{,}5 \cdot 10^{-14} & 1{,}0 \cdot 10^{0} & 3{,}2 \cdot 10^{-13} & 6{,}9 \cdot 10^{-14} & 1{,}9 \cdot 10^{-14} \\ 1{,}3 \cdot 10^{1} & 7{,}6 \cdot 10^{-3} & -1{,}1 \cdot 10^{1} & -3{,}6 \cdot 10^{-2} & -1{,}3 \cdot 10^{0} & 8{,}8 \cdot 10^{-2} & -8{,}6 \cdot 10^{-2} \\ -4{,}9 \cdot 10^{-14} & 7{,}1 \cdot 10^{-15} & 2{,}9 \cdot 10^{-14} & 5{,}0 \cdot 10^{-15} & 4{,}9 \cdot 10^{-14} & 1{,}2 \cdot 10^{0} & -2{,}3 \cdot 10^{-15} \\ -2{,}3 \cdot 10^{-12} & -1{,}1 \cdot 10^{-14} & 1{,}9 \cdot 10^{-12} & 7{,}3 \cdot 10^{-14} & -2{,}5 \cdot 10^{-12} & 3{,}5 \cdot 10^{-13} & -7{,}8 \cdot 10^{-14} \\ -1{,}4 \cdot 10^{1} & 1{,}2 \cdot 10^{0} & 1{,}5 \cdot 10^{1} & -3{,}0 \cdot 10^{0} & -2{,}1 \cdot 10^{1} & -1{,}4 \cdot 10^{1} & -1{,}9 \cdot 10^{1} \end{bmatrix}$$

$$\tilde{\mathbf{B}} = \begin{bmatrix} -2{,}3 \cdot 10^{-14} & 4{,}7 \cdot 10^{-15} \\ 4{,}0 \cdot 10^{3} & -4{,}7 \cdot 10^{-13} \\ 6{,}1 \cdot 10^{-15} & 4{,}7 \cdot 10^{-15} \\ -1{,}3 \cdot 10^{0} & 1{,}8 \cdot 10^{-2} \\ 9{,}7 \cdot 10^{-15} & 9{,}4 \cdot 10^{-15} \\ 3{,}2 \cdot 10^{-13} & -3{,}5 \cdot 10^{-4} \\ 8{,}0 \cdot 10^{-1} & 2{,}8 \cdot 10^{1} \end{bmatrix} ,\tilde{\mathbf{F}} = \begin{bmatrix} 5{,}0 \cdot 10^{-17} & 4{,}1 \cdot 10^{-16} & 4{,}2 \cdot 10^{-17} & -8{,}4 \cdot 10^{-14} & -3{,}5 \cdot 10^{-14} \\ -1{,}5 \cdot 10^{-2} & -1{,}5 \cdot 10^{-2} & -1{,}5 \cdot 10^{-2} & -1{,}3 \cdot 10^{-11} & -8{,}4 \cdot 10^{-12} \\ 4{,}5 \cdot 10^{-17} & 3{,}6 \cdot 10^{-16} & 4{,}3 \cdot 10^{-17} & -7{,}3 \cdot 10^{-14} & -5{,}1 \cdot 10^{-14} \\ 1{,}6 \cdot 10^{-3} & 1{,}4 \cdot 10^{-3} & 1{,}5 \cdot 10^{-3} & 3{,}0 \cdot 10^{-2} & 2{,}0 \cdot 10^{-1} \\ 2{,}6 \cdot 10^{-18} & 5{,}3 \cdot 10^{-18} & -2{,}8 \cdot 10^{-18} & 6{,}7 \cdot 10{-}15 & 2{,}0 \cdot 10^{-14} \\ 8{,}8 \cdot 10^{-17} & -2{,}5 \cdot 10^{-16} & 1{,}8 \cdot 10^{-17} & 1{,}8 \cdot 10^{1} & -7{,}1 \cdot 10^{1} \\ 2{,}3 \cdot 10^{-4} & -2{,}1 \cdot 10^{-3} & -3{,}3 \cdot 10^{-4} & 3{,}2 \cdot 10^{0} & -6{,}9 \cdot 10^{1} \end{bmatrix}$$

The approximation of the original model is good, but the complexity of the reduced model is very high because each of the 98 elements of these matrices is non-zero.

In [2] the following results are found by using the genetic algorithm and it is shown that if the tabu search algorithm be used to search the local region found by the genetic algorithm, a solution with 81 zero elements can also be achieved.

$$\tilde{\mathbf{A}} = \begin{bmatrix} 0 & 1,0 & 0 & 0 & 0 & 0 & 0 \\ -4127,5 & 0 & 127,5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1,0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -16,8 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1,2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -31,6 \end{bmatrix}$$

$$\tilde{\mathbf{B}} = \begin{bmatrix} 0 & 0 \\ 4000 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 33,5 \end{bmatrix} , \tilde{\mathbf{F}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ -0,015 & -0,015 & -0,015 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0,0024 & 0,0013 & 0,0015 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 17,5 & -71,4 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Simulations of the original system and the reduced system proves that this simplification, found by the two search algorithms, is acceptable. To find the simple structure system, the genetic algorithm needs to calculate approximately 6000 solutions for the global search and the tabu search algorithm calculates approximately 500 solutions for the local search within a search space containing $2^{99} \approx 6.33 \times 10^{29}$possible solutions.

By using the row by row method the search area shrinks to $7 \times 2^{14} \approx 114,6 \times 10^{3}$ which is $18,1 \times 10^{-26}$ times smaller than the original search space, and the result is globally optimal according to the defined cost (fitness) function as follow:

$$\tilde{\mathbf{A}} = \begin{bmatrix} 0 & 1,0 & 0 & 0 & 0 & 0 & 0 \\ -4127,5 & 0 & 127,5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1,0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1,24 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -31,6 \end{bmatrix}$$

$$\tilde{\mathbf{B}} = \begin{bmatrix} 0 & 0 \\ 4000 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 33,5 \end{bmatrix}, \quad \tilde{\mathbf{F}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0.0018 & 0 & 0.0016 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 17,5 & -71,4 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

V. Concluding remarks

The primary method of system matrices optimization for order reduction of nonlinear systems and its variants for structure simplification may not be efficient for big nonlinear systems. Because the earliest method results in full matrices, which is not suitable for big systems and other variants that use e.g. the genetic algorithm or the tabu search need much computation effort and the results are not optimal for sure. In this paper we introduced the row by row method which contracts the search area and opens a new field to optimize the results. There are also a number of questions and open problems which would be interesting for further investigation, for instance:

- What are suitable cost functions instead of (4);
- an intelligent method for search in the reduced search space;
- numerical scaling the state variables and outputs in order to improve the approximation results.

References

[1] Buttelmann, M. and Lohmann, B. Model simplification and order reduction of nonlinear systems with genetic algorithms. In *Proceedings of the IMACS Symposium on Mathematical Modelling,3rd MATHMOD, Vienna*, pages 777–781, 2000.

[2] Buttelmann, M. and Lohmann, B. Non-linear model reduction by genetic algorithms with using a system structure related fitness function. In *European Control Conference (ECC), Porto, Portugal*, pages 1870–1875, 2001.

[3] R. Fletcher. *Practical Methods of Optimization, Vol. 1, Unconstrained Optimization, and Vol. 2, Constrained Optimization*. John Wiley and Sons., 1980.

[4] Kokotovic, P.V, O'Malley, R.E., and Sannuti, P. Singlar perturbation and order reduction in cnontrol theory - an overview. *Automatica*, 12:123–132, 1976.

[5] Lawson, C.L. and R.J. Hanson. *Solving Least Squares Problems*. Prentice-Hall, 1974.

[6] B. Lohmann. *Ordnungsreduktion und Dominanzanalyse nichtlinearer Systeme*, volume 8 of *VDI-Verlag*. VDI-Fortschrittsberichte, Dusseldorf, 1994.

[7] B. Lohmann. Application of model order reduction to a hydrpneumatic vehicle suspension. *IEEE Transactions on Control Systems Technology*, 3(2):102–109, March 1995.

[8] B. Lohmann. Order reduction and determination of dominant state varialbes for nonlinear systems. *Mathematical Modeling of Systems*, 1(2):77–90, 1995.

[9] M. Mitchell. *An Introduction to Genetic Algorithms*. MIT Press, Cambridge, 1996.

[10] Pham, D. T. and Karaboga, D. *Inteligent Optimisation Techniques - Genetic Algorithms, Tabu Search, Simulated Anealing and Neural Networks*. Springer, 2000.

[11] J. M. A. Scherpen. Balancing for nonlinear systems. *System & Control Letters*, 21:143–153, 1993.

[12] S. Volkwein. Proper orthogonal decomposition and singular value decomposition. Technical report, Graz University, 1999.