Wiesław Krajewski; Umberto Viaro
Routh-type L_2 model reduction revisited

Kybernetika, Vol. 54 (2018), No. 3, 557–575

Persistent URL: http://dml.cz/dmlcz/147436

KYBERNETIKA — VOLUME *54* (2018), NUMBER 3, PAGES 557–575

ROUTH–TYPE L_2 MODEL REDUCTION REVISITED

Wiesław Krajewski and Umberto Viaro

A computationally simple method for generating reduced–order models that minimise the L_2 norm of the approximation error while preserving a number of second–order information indices as well as the steady–state value of the step response, is presented. The method exploits the energy–conservation property peculiar to the Routh reduction method and the interpolation property of the L_2–optimal approximation. Two examples taken from the relevant literature show that the suggested techniques may lead to approximations that are not worse than those afforded by popular more cumbersome techniques.

Keywords: model reduction, L_2 norm, Routh approximation, steady–state response

Classification: 93C05, 93A15, 93B11

1. INTRODUCTION

The model reduction problem has aroused a continual interest in the engineering community since the dawn of control and system theory [40, 71], its importance being evident not only in system simulation and controller synthesis but also in many problems related to robustness and uncertainty issues. Indeed, despite the dramatic increase of computing capabilities that make the need for simplified models less compelling, the new challenges facing the control engineer have led to a revival of studies on this topic with particular emphasis on optimisation and algorithmic efficiency (see, e. g., [1, 2, 4, 7, 11, 12], [14]–[59], [61, 65, 69, 70]).

Besides the reduction methods based on the conservation of first–order information indices (e. g., coefficients of suitable series expansions), such as the classic Padé technique and its numerous variants [8, 9] that are characterised by remarkable computational simplicity and ease of implementation, the methods based on second–order information indices (e. g., principal components, Hankel singular values, impulse–response energies) [16]–[19], [27, 31, 32, 43, 45, 60, 63], and on suitable quadratic criteria, such as the L_2 norm of the error [7, 14], [21]–[24], [29], [33]–[35], [49, 62], [66]–[68], [70, 72], have enjoyed an increasing popularity since the late Seventies and early Eighties, and dedicated software has been developed for their implementation.

The advantages of the aforementioned methods are certainly related to an intuitively appealing definition of the reduction criterion and to the possibility of determining bounds on some error norms [18]. However, their computational complexity increases

DOI: 10.14736/kyb-2018-3-0557

rapidly with the dimensionality of the original system [20], which has stimulated research on efficient numerical algorithms [30], especially in view of the very high order of certain circuits and structures [10, 17].

This paper presents a computationally efficient model reduction technique that combines the advantages of the Routh approximation [25, 36] in terms of conservation of resolvent–kernel energies [6] with those of the L_2–optimal rational approximation. Essentially, the reduction algorithm requires: (i) the construction of a Routh table, (ii) the solution of an algebraic equation of degree equal to the order of the approximating model, and (ii) the satisfaction of a set of interpolation conditions (conditions for L_2 optimality [21], [28, 29, 35]). Moreover, the algorithm can be adapted to ensure the desired asymptotic behaviour in the response to step inputs, a characteristic that is not exhibited by most popular reduction techniques such as Hankel– and L_2–norm approximations and pure balanced truncation.

Among the numerous choices for generating *stable* reduced–order transfer functions, it seems reasonable to form their denominators in such a way that the approximating models share some essential characteristics with the original system. The choice made in this paper is to generate these denominators by means of Routh's algorithm because it leads to the conservation of some resolvent kernel energies which play a fundamental role in the determination of all forced responses, as shown in the next sections. Examples do not defy expectations.

The rest of this paper is organised as follows. Section 2 recalls the basic recursion of the Routh algorithm and the energy–conservation property of the Routh approximation, while Section 3 recalls the interpolation conditions that are satisfied by the L_2–optimal reduced models. Section 4 presents the suggested reduction algorithm and discusses its computational complexity. Section 5 shows the results of the application of the algorithm to a pair of examples taken from the literature on model reduction.

2. ROUTH'S ALGORITHM AND ITS USE

This section briefly illustrates the properties of the Routh algorithm that are relevant to the reduction procedure described in Section 4 (for other interesting properties of this remarkable algorithm see [3, 6, 26, 50] and [64]).

The Routh algorithm generates a sequence of polynomials of descending degree starting from the even and odd parts of a given real polynomial of degree n

$$P_n(s) = Q_n(s) + Q_{n-1}(s) \tag{1}$$

according to the recursion

$$Q_{i-2}(s) = Q_i(s) - q_{i-1}\, s\, Q_{i-1}(s), \tag{2}$$

where

$$Q_i(s) = \sum_{k=0}^{\lfloor i/2 \rfloor} r_{i,i-2k} s^{i-2k}, \quad 0 \le i \le n, \tag{3}$$

and q_{i-1} is the ratio of the leading coefficients of $Q_i(s)$ and $Q_{i-1}(s)$, respectively, i. e.,

$$q_{i-1} = \frac{r_{i,i}}{r_{i-1,i-1}}, \quad 1 \le i \le n, \tag{4}$$

which ensures that the degree of the right–hand side of (2) is $i-2$ like its left–hand side. The entries of the row of order i in the standard Routh table for $P_n(s)$, shown in Table 1, are precisely the coefficients of the decreasing powers of s in (3).

n	$r_{n,n}$	$r_{n,n-2}$	$r_{n,n-4}$	$r_{n,n-6}$	$\cdots$
$n-1$	$r_{n-1,n-1}$	$r_{n-1,n-3}$	$r_{n-1,n-5}$	$r_{n-1,n-7}$	$\cdots$
$n-2$	$r_{n-2,n-2}$	$r_{n-2,n-4}$	$r_{n-2,n-6}$	$\cdots$	
$n-3$	$r_{n-3,n-3}$	$r_{n-3,n-5}$	$r_{n-3,n-7}$	$\cdots$	
.	$\cdots$	$\cdots$	$\cdots$		
.	$\cdots$	$\cdots$	$\cdots$		
i	$r_{i,i}$	$r_{i,i-2}$	$\cdots$		
.	$\cdots$	$\cdots$			
.	$\cdots$	$\cdots$			
3	$r_{3,3}$	$r_{3,1}$			
2	$r_{2,2}$	$r_{2,0}$			
1	$r_{1,1}$				
0	$r_{0,0}$				

Tab. 1. Routh table for polynomial $P_n(s) = Q_n(s) + Q_{n-1}(s)$ generated according to recursion (2). Its entries correspond to the coefficients of polynomials (3).

As is well known, if and only if $P_n(s)$ is a Hurwitz polynomial, the leading coefficients $r_{j,j}$, like all the other coefficients in the table, are different from zero and have the same sign, so that the entire sequence of $n+1$ polynomials $Q_j(s)$, $j = n, n-1, \ldots, 1, 0$, containing only even or only odd powers of s, can be constructed and all the n quotients (4) are positive (Routh criterion).

Now, a *complete* polynomial $P_i(s)$ can be associated with every pair of consecutive polynomials in this sequence according to

$$P_i(s) = Q_i(s) + Q_{i-1}(s), \tag{5}$$

thus forming a sequence of complete real polynomials $\{P_j(s), j = n, \ldots, 1\}$. Clearly, two consecutive polynomials $P_i(s)$ and $P_{i-1}(s)$ share the same even or odd part $Q_{i-1}(s)$, and the Routh table for $P_{i-1}(s)$ coincides with the "tail" of the Routh tables for $P_j(s)$, $j = i, i+1, \ldots, n$. As a consequence, all polynomials in the sequence are Hurwitz if $P_n(s)$ is so. From (2) and (5), the following recursive relation between two consecutive complete polynomials is immediately obtained:

$$P_{i-1}(s) = \left(1 + \frac{q_{i-1}}{2}\, s\right) P_i(s) - (-1)^i \frac{q_{i-1}}{2}\, s\, P_i(-s) \tag{6}$$

which is the so–called two–term form of Routh's algorithm as opposed to the usual three–term (or split) form (2) [38]. It is also called "step–down" form because it generates polynomials of descending degree.

The Routh approximation method (cf., e. g., [36]) uses as the denominator of the ith–order reduced transfer function the polynomial $P_i(s)$ in the sequence generated from the

original denominator $P_n(s)$. Quite interestingly, besides stability, the reduced–order model obtained in this way keeps a number of original second–order information indices (impulse–response energies). To show this, consider the function

$$K_i(s) = \frac{1}{P_i(s)},\ i < n, \tag{7}$$

and denote by $k_i^{(h)}(t)$ the hth derivative of its impulse response $k_i(t)$, which is the so–called *resolvent kernel* of the convolution integral that determines the forced response $y_{f,i}(t)$ to an input $u(t)$ of an ith–order LTI system with transfer function

$$G_i(s) = \frac{N_i(s)}{P_i(s)} = N_i(s)\, K_i(s), \tag{8}$$

where $N_i(s) = n_{i,0} + n_{i,1}s + \ldots + n_{i,i-1}s^{i-1}$, that is,

$$y_{f,i}(t) = \int_0^t k_i(t-\tau)\, n_i(\tau)\, \mathrm{d}\tau, \tag{9}$$

where

$$n_i(t) = n_{i,0}u(t) + n_{i,1}\frac{\mathrm{d}u}{\mathrm{d}t} + \ldots + n_{i,i-1}\frac{\mathrm{d}^{i-1}u}{\mathrm{d}t^{i-i}} \tag{10}$$

is the right–hand side of the standard form of the differential equation describing the input–output behaviour of (8).

Denoting by

$$E_{i,h} = \int_0^\infty [k_i^{(h)}(t)]^2 \mathrm{d}t \tag{11}$$

the energy of the kth derivative $k_i^{(h)}(t)$ of $k_i(t)$, it may be proved [6] that

$$E_{i,h} = E_{n,h},\ h = 0, 1, \ldots, i-1. \tag{12}$$

In other words, the ith–order model whose denominator is formed from the original denominator $P_n(s)$ according to the Routh recursion (6) preserves the first i *kernel energies* of the nth order (original) system with transfer function $G_n(s)$. Also, these energies may be computed recursively using only the entries of the Routh table for $P_n(s)$ [6]. From the kernel energies and the coefficients of $N_i(s)$, the impulse–response energy for the system with transfer function $G_i(s)$ can easily be determined and equated to the impulse–response energy of $G_n(s)$ (see, e. g., [32]). Of course, matching these energies does not entail minimising the L_2 norm of the difference between the original and reduced impulse responses (approximation error).

Remark 1. Another interesting link among the polynomials generated via Routh's algorithm is provided by relation (5) which shows that the roots of every complete Hurwiz polynomial $P_i(s)$ belong to the root locus for the equation $Q_i(s) + \lambda\, Q_{i-1}(s)$, $\lambda \in \mathcal{R}$, whose departure and arrival points are the roots of it even and odd parts $Q_i(s)$ and $Q_{i-i}(s)$ which alternate along the imaginary axis according to the Hermite–Biehler theorem [58]. In turn, equation (2) shows that the roots of $Q_{i-2}(s)$, which separate those of

$Q_{i-i}(s)$ along the same axis, belong to the root locus for the equation $Q_i(s)+\lambda\, s\, Q_{i-1}(s)$, $\lambda \in \mathcal{R}$, whose departure and arrival points are the roots of $Q_i(s)$ and $s\, Q_{i-1}(s)$. It follows that the roots of $Q_{i-2}(s)$ are contained in the convex hull of the roots of $Q_i(s)$, which allows us to establish some root clustering properties of Routh's algorithm [37].

The next section reviews briefly the conditions under which an ith–order transfer function minimises the L_2 norm of the approximation error subject to the conservation of the energies (12).

3. CONDITIONS FOR L_2 OPTIMALITY

The L_2–optimal rational approximation satisfies a set of interpolation conditions that have been known to the control community for quite some time in the s–domain SISO case [40] and have more recently been extended to MIMO systems represented by transfer function matrices in [29]. By exploiting these interpolation conditions, some efficient reduction algorithms that avoid the direct computation of the gradient of the objective function (L_2 norm of the error) have been developed (see, e. g., [21, 22, 35, 62, 66] in a state–space setting and [33] in an input–output setting). However, these procedures are intrinsically nonlinear, strongly depend on the initial conditions, do not even retain the steady–state value of the step response and, in some cases, might give rise to unstable models of stable systems [70]. These drawbacks justify the search for alternative simpler and more robust techniques, even if they lead to constrained optima or near–optima in the L_2 sense [24, 49, 68]. Such a path is followed in this paper by suitably combining some classic control–theory tools. Specifically, the transform of the reduced–model kernel is chosen as in (7) with $P_i(s)$ obtained from the original denominator using the Routh recursion (6), thus ensuring the retention of a number of kernel energies, and then the numerator parameters are determined so as to minimise the L_2 norm of either the impulse–response error or, when the original steady–state value of the step response must be reproduced, the transient component of the step–response error (see Section 4).

To this purpose, consider first the impulse response (the step response will be considered later) and denote the difference between the impulse responses $g_n(t)$ and $g_i(t)$ of the systems characterised by the strictly proper transfer functions $G_n(s)$ (original system) and $G_i(s)$ (reduced–order model), respectively, by

$$d_i(t) = g_n(t) - g_i(t) \tag{13}$$

whose Laplace transform, according to (8), is

$$D_i(s) = G_n(s) - G_i(s) = \frac{P_i(s)N_n(s) - P_n(s)N_i(s)}{P_n(s)P_i(s)}. \tag{14}$$

The squared L_2 norm of (13) induced by the usual scalar product is

$$||d_i(t)||^2 = \int_0^\infty d_i(t)\, d_i^*(t)\, \mathrm{d}t, \tag{15}$$

where the asterisk denotes complex conjugation. Assuming for simplicity that the i roots $p_{i,h}$, $h = 1, 2, \ldots, i$, of $P_i(s)$ (poles of $G_i(s)$) are distinct, and indicating with

$$\mathcal{F}_i = \mathrm{span}\big\{e^{p_{i,1}t}, e^{p_{i,2}t}, \ldots, e^{p_{i,i}t}\big\} \tag{16}$$

the ith dimensional vector space generated by the modes of the reduced–order system, index (15) is minimum if, and only if, for *any* function $f_i(t) \in \mathcal{F}_i$, the following orthogonality condition holds [39]:

$$\int_0^\infty d_i(t) f_i^*(t)\,\mathrm{d}t = 0. \tag{17}$$

Denoting by $F_i(s)$ the Laplace transform of $f_i(t)$ and recalling that the Laplace transform of $f_i^*(t)$ is $F_i^*(s^*)$, from (17) and Parseval's theorem we obtain

$$\int_0^\infty d_i(t) f_i^*(t)\,\mathrm{d}t = \frac{1}{2\pi\jmath}\int_{-\jmath\infty}^{+\jmath\infty} D_i(s) F_i^*(-s^*)\,\mathrm{d}s = 0. \tag{18}$$

Therefore, by Cauchy's integral formula, all the poles of $D_i(s)F_i^*(-s^*)$ must lie in the left half–plane like those of $D_i(s)$. Since the poles $-p_{i,h}^*$ of all functions $F_i^*(-s^*)$ are in the right half–plane, it follows that they must be cancelled by the zeros of $D_i(s) = G_n(s) - G_i(s)$, that is, $G_i(s)$ must *interpolate* $G_n(s)$ at the negatives of its own poles $p_{i,h}$ (which are real or in conjugate pairs).

Taking (14) into account, the aforementioned optimality condition can be expressed in the compact form of a polynomial identity as:

$$P_i(s)N_n(s) - P_n(s)N_i(s) = M_{n-1}(s)\prod_{h=1}^{i}(s + p_{i,h}), \tag{19}$$

where $M_{n-1}(s)$ is a (real) polynomial of degree equal, at most, to $n-1$. By equating the coefficients of the equal powers of s on both sides of (19) a system of $n+i$ equations linear in the same number of unknown coefficients of $N_i(s)$ and $M_{n-1}(s)$ can be formed. Polynomial $M_{n-1}(s)$ can then be used, if necessary, to compute the value of the index (15) by resorting again to Cauchy's integral formula and Parseval's theorem. Alternatively, by setting $s = -p_{i,h}$, $h = 1, 2, \ldots, i$, in (19), the following smaller set of i equations *linear* in the i unknown coefficients of $N_i(s)$ is formed:

$$N_i(-p_{i,h}) = P_i(-p_{i,h})\,\frac{N_n(-p_{i,h})}{P_n(-p_{i,h})},\quad h = 1, 2, \ldots, i. \tag{20}$$

In both cases, it is necessary to preliminarily determine the roots $p_{i,h}$ of polynomial $P_i(s)$, which can easily be done using readily available algorithms.

4. REDUCTION PROCEDURE

On the basis of the previous considerations, the following reduction algorithm can be used to find the L_2–optimal approximation of given order i subject to the conservation of i kernel energies.

Algorithm A — Impulse–response–based algorithm

1. Generate, according to (6), a sequence of polynomials of descending degree from the original denominator polynomial $P_n(s)$ down to the polynomial $P_i(s)$ of the desired reduced degree $i < n$.

2. Find the roots $p_{i,h}$, $h = 1, 2, \ldots, i$, of the polynomial equation $P_i(s) = 0$.

3. Determine the i coefficients of the $(i-1)$th degree numerator polynomial $N_i(s)$ of the transfer function $G_i(s) = N_i(s)/P_i(s)$ approximating $G_n(s) = N_n(s)/P_n(s)$ by solving the system (20) of i linear equations.

Remark 2. The impulse–response approximation error, whose constrained L_2–norm is minimised by Algorithm A, tends asymptotically to zero because the reduced–order model is guaranteed asymptotically stable like the original system, which is a necessary condition for the existence of a minimum. Its uniqueness is also guaranteed as long as the i interpolation points are distinct, so that the number of independent equations in (20) equals the number of unknown coefficients to be determined.

Remark 3. Algorithm A cannot directly be extended to persistent inputs and corresponding outputs, not even step responses, because the related approximation error wouldn't tend to zero without additional constraints. Instead, the algorithm can be extended to filtered transfer functions, provided that the filter is stable. Note, by the way, that popular reduction algorithms, such as balanced truncation and Hankel–norm approximation, apply to stable systems only.

The above procedure is not computationally demanding. In particular: (i) the construction of the entire Routh table for a polynomial of degree n requires $\mathcal{O}(n^2/2)$ elementary operations [6] (but the aforementioned algorithm can be arrested at the ith row), (ii) the computational complexity of the Gauss elimination procedure to solve a system of i linear equation is $\mathcal{O}(i^3)$ [53], while (iii) the solution of polynomial equations up to degree 20 does not pose any particular problem in terms of both numerical robustness and efficiency [46] (note that, usually, $i \ll n$)). It is also worth mentioning that *fraction–free* Routh tests that increase considerably the numerical accuracy of the classical Routh algorithm have been proposed recently [5].

As already said, the reduced–order model obtained according to the aforementioned procedure keeps i kernel energies and minimises index (15) subject to the Hurwitz denominator $P_i(s)$. However, since index (15) refers to the impulse response, the asymptotic response to any other input $u(t)$ is not equal, in general, to that of the original system. A suggestion as to how the method can be adapted to the case in which the steady–state response to step inputs must be preserved, at the expense of the number of parameters left for optimisation, is outlined next. The procedure could be extended to reproduce the asymptotic response to more complicated inputs. However, for simplicity such an extension is not pursued here.

Assume again that the reduced–order model denominator $P_i(s)$ is obtained from the Routh algorithm (6), thus preserving stability and a number of kernel energies, and denote by

$$\hat{G}_i(s) = \frac{\hat{N}_i(s)}{P_i(s)} \tag{21}$$

the transfer function of the strictly–proper reduced–order model whose numerator $\hat{N}_i(s)$ must be determined in such a way that the steady–state response to a step input is preserved.

The step–response transform of the stable original system can be decomposed [13] as

$$Y_{f,n}(s) = \frac{N_n(s)}{P_n(s)}\,\frac{1}{s} = \frac{T_n(s)}{P_n(s)} + \frac{K}{s}, \tag{22}$$

where $T_n(s)/P_n(s)$ is the Laplace transform of the transient response and K is the steady–state value. In order for $\hat{G}_i(s)$ to exhibit the same steady state, the transform of its step response should be decomposable as

$$Y_{f,i}(s) = \frac{\hat{N}_i(s)}{P_i(s)}\,\frac{1}{s} = \frac{T_i(s)}{P_i(s)} + \frac{K}{s}, \tag{23}$$

where the transient component $T_i(s)/P_i(s)$ is strictly proper and the steady–state component K/s matches the one of (22). From (23) it follows that

$$\hat{N}_i(s) = T_i(s)\,s + K\,P_i(s). \tag{24}$$

Since the degree of this polynomial identity is i, by equating the coefficients of the same powers of s on both sides of (24), a system of $i+1$ equations is obtained. Therefore, to admit a unique solution, the number of unknowns must also be $i+1$. Now, if $T_i(s)$ is completely determined by minimising the L_2 norm of the difference between the transient terms in the step response, i.e:

$$\hat{d}_i(t) = y_{tr,n}(t) - y_{tr,i}(t), \tag{25}$$

where $y_{tr,n}(t) = LT^{-1}[T_n(s)/P_n(s)]$ and $y_{tr,i}(t) = LT^{-1}[T_i(s)/P_i(s)]$, then the number of unknowns in (24) is only i (number of coefficients of $\hat{N}_i(s)$) and no solution exists.

To overcome this problem, a further unknown should be introduced. One way to do this, is to replace $T_i(s)/P_i(s)$ by the sum of the best approximation (in the L_2 sense) of immediately lower order $i-1$, i. e., $T_{i-1}(s)/P_{i-1}(s)$, and an "auxiliary" stable first–order term $x/(s-q)$ with unknown gain x and pre–specified pole q. Not to influence appreciably the system dynamics, this pole could be located far to the left of the roots of $P_{i-1}(s)$, but other choices are of course possible (and even advisable). Taking (21) into account, the Laplace transform of the step response of the reduced–order model then becomes:

$$Y_{f,i}(s) = \hat{G}_i(s)\,\frac{1}{s} = \frac{\hat{N}_i(s)}{P_{i-1}(s)(s-q)}\,\frac{1}{s} = \frac{T_{i-1}(s)}{P_{i-1}(s)} + \frac{x}{s-q} + \frac{k}{s} \tag{26}$$

leading directly to the polynomial identity:

$$\hat{N}_i(s) = T_{i-1}(s)\,(s-q)\,s + x\,P_{i-1}(s)\,s + k\,P_{i-1}(s)\,(s-q). \tag{27}$$

In this way the number of unknowns (the i coefficients of $\hat{N}_i(s)$ plus x) matches the number of $i+1$ equations obtained by equating the coefficients of the equal powers of s on both sides of (27). In particular, using the notation:

$$T_{i-1}(s) = b_{i-1,i-2}s^{i-2} + b_{i-1,i-3}s^{i-3} + \ldots + b_{i-1,0}, \tag{28}$$

$$P_{i-1}(s) = a_{i-1,i-1}s^{i-1} + a_{i-1,i-2}s^{i-2} + \ldots + a_{i-1,0}, \tag{29}$$

the unknown parameter x is obtained from the coefficients of s^i only as:

$$x = -\frac{b_{i-1,i-2}}{a_{i-1,i-1}} - k. \tag{30}$$

Once x has been determined, the computation of the coefficients of $\hat{N}_i(s)$ is straightforward since all terms at the right–hand side of (27) become known. In conclusion, the algorithm for finding an ith–order reduced model that keeps the steady–state value of the original step response can be presented as follows.

Algorithm B — Step–response–based algorithm

1. Decompose the original step response transform as in (22).
2. Find the transient component $T_{i-1}(s)/P_{i-1}(s)$ that minimises the L_2 norm of $\hat{d}_{i-1}(t) = y_{tr,n}(t) - y_{tr,i-1}(t)$ with $P_{i-1}(s)$ obtained from the original denominator $P_n(s)$ via Routh's algorithm (6).
3. Choose q.
4. Compute x according to (30).
5. Determine the coefficients of polynomial $\hat{N}_i(s)$ from (27).
6. Form the ith–order approximating transfer function as

$$\hat{G}_i(s) = \frac{\hat{N}_i(s)}{P_{i-1}(s)(s-q)}.$$

The only demanding step of *Algorithm B* is clearly the second. It entails the same operations as *Algorithm A* (referred, however, to the transient component of the step response instead of the impulse response), namely, the construction of a (part of a) Routh table, the solution of a polynomial equation of degree $i-1$, and the solution of the system of $i-1$ linear equations similar to (20):

$$T_{i-1}(-p_{i-1,h}) = P_{i-1}(-p_{i-1,h})\,\frac{T_n(-p_{i-1,h})}{P_n(-p_{i-1,h})}, \quad h = 1, 2, \ldots, i-1, \tag{31}$$

which correspond to a set of $i-1$ interpolation conditions at the negatives of the roots $p_{i-1,h}$, $h = 1, 2, \ldots, i-1$, of $P_{i-1}(s)$. Therefore the computational complexity of *Algorithm B* is not much greater than that of *Algorithm A*, at least if the auxiliary pole q is arbitrarily placed to the left of the other poles of $Y_{f,i}(s)$, as previously suggested. Alternatively, q may be chosen so as to minimise $\|G_n(s) - \hat{G}_i(s)\|$. This result can be obtained by using a classic grid–search method that implies: (i) repeating Steps 3 through 6 for a number of different auxiliary poles q, (ii) computing the related values of the aforementioned norm, and (iii) picking up the pole q that ensures the least value of this norm. Clearly, this alternative choice increases the computational complexity of the procedure but might be worthwhile.

5. EXAMPLES

Two examples taken from the literature on model reduction are worked out in this section. The results obtained from the application of *Algorithms A* and *B* are compared with those obtained using the popular balanced truncation method as well as the methods employed by the authors who considered the same examples most recently. As usual, the comparison is based on the visual inspection of the responses to the impulse and step inputs, on the Bode plots and on the value of the L_2 norm of the respective impulse–response errors.

5.1. Example 1

Consider first the following 9th–order original transfer function put forth in [44]:

$$G(s) = \frac{s^4 + 35s^3 + 291s^2 + 1093s + 1700}{s^9 + 9s^8 + 66s^7 + 294s^6 + 1029s^5 + 2541s^4 + 4684s^3 + 5856s^2 + 4620s + 1700}, \quad (32)$$

whose poles are $-1, -1 \pm \jmath, -1 \pm \jmath 2, -1 \pm \jmath 3, -1 \pm \jmath 4$. The same original system has been used in [11] to find a third–order approximating model by means of a "biased stability–equation" technique.

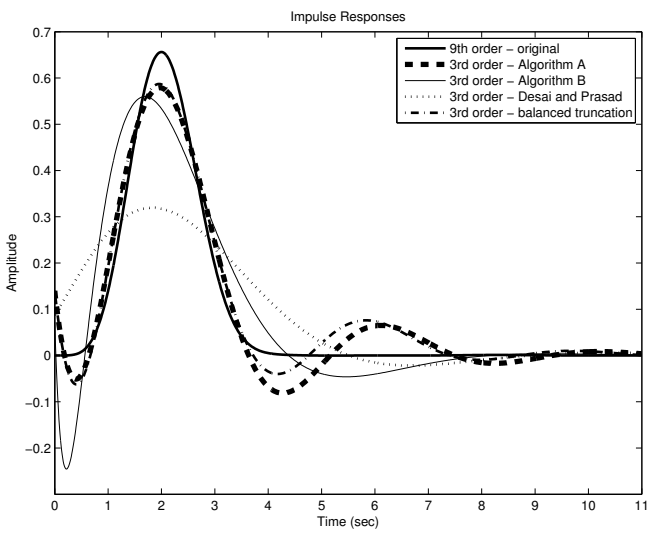

Fig. 1. Impulse responses of: (i) the original model (32) (bold solid line), (ii) the third–order approximation (33) obtained using *Algorithm A* (bold dashed line), (iii) the third–order approximation (34) obtained using *Algorithm B* (thin solid line), (iv) the third–order model derived in [11] (dotted line), and (v) the third–order model obtained via balanced truncation (dashdotted line).

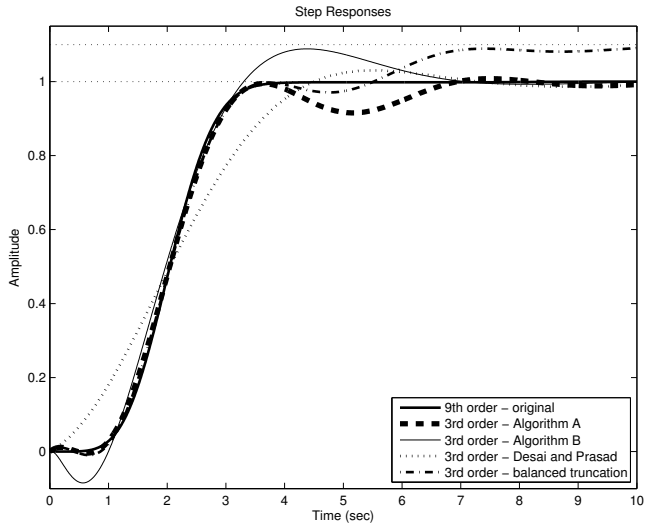

Fig. 2. Step responses of: (i) the original model (32) (bold solid line), (ii) the third–order approximation (33) obtained using *Algorithm A* (bold dashed line), (iii) the third–order approximation (34) obtained using *Algorithm B* (thin solid line), (iv) the third–order model derived in [11] (dotted line), and (v) the third–order model obtained via balanced truncation (dashdotted line).

Algorithm A in Section 4 has been applied to find a third–order transfer function based on the impulse response error only. According to Step 1 of the procedure, its denominator has been formed from the denominator of (32) using Routh's recursion leading to $P_3(s) = s^3 + 1.6412s^2 + 3.3077s + 1.8601$, whose roots (Step 2) turn out to be $-0.7024, -0.4694 \pm \jmath 1.5582$. Finally, according to Step 3 the coefficients of the numerator of the third–order transfer function have been found by solving the system of three interpolation equations corresponding to (20). The resulting model is

$$G_3(s) = \frac{0.1399s^2 - 0.8022s + 1.8554}{s^3 + 1.6412s^2 + 3.3077s + 1.8601}. \tag{33}$$

The squared L_2 norm of the related impulse–response error turns out to be 0.0184, whereas the squared error norm for the model obtained in [11] is 0.1348 and that for the third–order model obtained from balanced truncation is 0.0158.

In order to reproduce the steady–state value of the original step response, resort has been made to *Algorithm B*. According to Step 1, the original step response has first been decomposed as in (22). Next, using *Algorithm A*, a second–order approximation of the transient component of this response has been determined (Step 2). According to Step 3, the far–off pole q of the auxiliary term in (26) has been chosen by means of the iterative search outlined at the end of Section 4, leading to $q = -5.2$ (note, however, that this choice is not critical). The related residue x has been found (Step 4) according to (30).

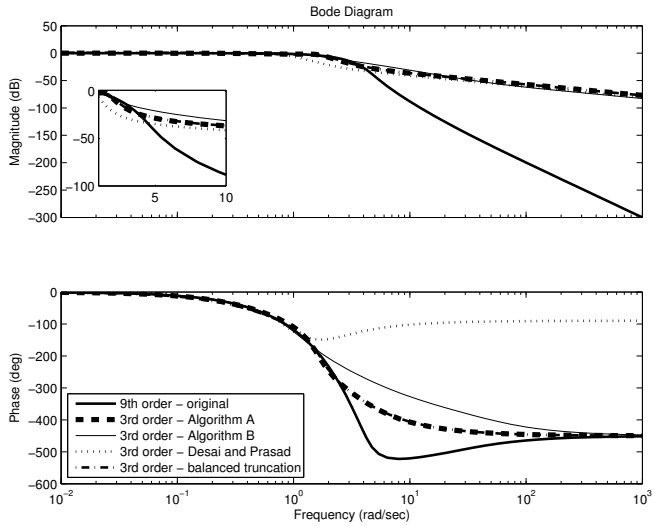

Fig. 3. Bode plots of: (i) the original model (32) (bold solid line), (ii) the third–order approximation (33) obtained using *Algorithm A* (bold dashed line), (iii) the third–order approximation (34) obtained using *Algorithm B* (thin solid line), (iv) the third–order model derived in [11] (dotted line), and (v) the third–order model obtained via balanced truncation (dashdotted line).

The numerator of the third–order transfer function has been computed, according to Step 5, by particularising the polynomial identity (27). Finally, the transfer function of the approximating model has been determined according to Step 6; it turns out to be

$$\hat{G}_3(s) = \frac{0.0724s^2 - 3.1780s + 5.8933}{s^3 + 6.5248s^2 + 8.0224s + 5.8933} \tag{34}$$

whose poles are $-5.2, -0.6624 \pm \jmath 0.8334$. In this case the squared L_2 error norm turns out to be 0.0662. Figures 1, 2 and 3 compare, respectively, the impulse responses, the step responses and the Bode plots of (33) and (34) with those obtained using the method suggested in [11] and the balanced truncation method.

5.2. Example 2

Consider the 10th–order system described by the transfer function:

$$G(s) = \frac{540.70748 \times 10^{17}}{\prod_{i=1}^{10}(s + b_i)}, \tag{35}$$

where $b_1 = 2.04, b_2 = 18.3, b_3 = 50.13, b_4 = 95.15, b_5 = 148.85, b_6 = 205.16, b_7 = 257.21, b_8 = 298.03, b_9 = 320.97, b_{10} = 404.16$. The same system has been considered in [48, 41].

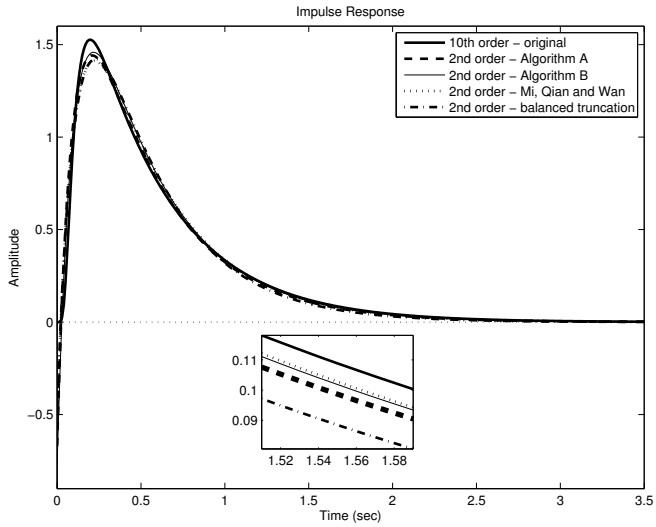

Fig. 4. Impulse responses of: (i) the original model (35) (bold solid line), (ii) the second–order approximation (36) obtained using *Algorithm A* (bold dashed line), (iii) the second–order approximation (37) obtained using *Algorithm B* (thin solid line), (iv) the second–order model derived in [41] (dotted line), and (v) the second–order model obtained via balanced truncation (dashdotted line).

Assume, as in [41], that a second–order approximating model is needed.

Algorithm A, based on the impulse response without consideration of the steady–state response to step inputs, leads to

$$G_2(s) = \frac{-0.6687s + 23.2918}{s^2 + 13.0793s + 23.6262} \tag{36}$$

whose poles are $-10.9147, -2.1646$. The squared L_2 norm of the related impulse–response error turns out to be 0.0082, whereas the squared error norm for the model obtained in [41] and for the one obtained via balanced truncation is 0.0074.

Algorithm B, which reproduces the steady–state value of the step response, for $q = -19.1$ ((found using the iterative search outlined at the end of Section 4)) leads to

$$\hat{G}_2(s) = \frac{-0.3521s + 34.5019}{s^2 + 20.9064s + 34.5019} \tag{37}$$

whose poles are $-19.1, -1.8064$. In this case the squared L_2 error norm turns out to be 0.0398. Figures 4, 5 and 6 compare, respectively, the impulse responses, the step responses and the Bode plots of (36) and (37) with those obtained using the method suggested in [41] and the balanced truncation method.

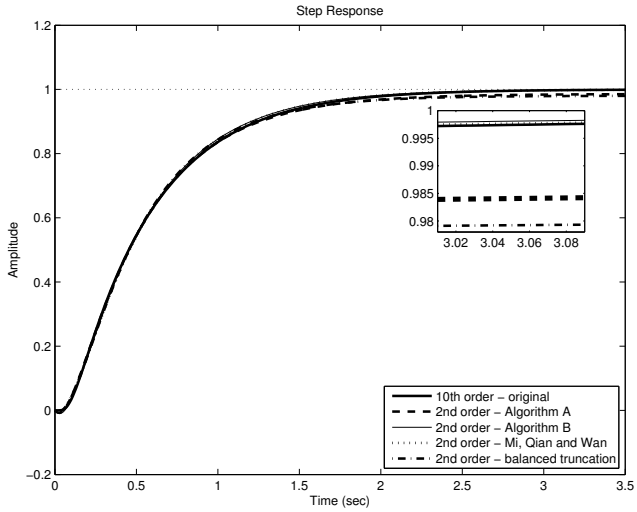

Fig. 5. Step responses of: (i) the original model (35) (bold solid line), (ii) the second–order approximation (36) obtained using *Algorithm A* (bold dashed line), (iii) the second–order approximation (37) obtained using *Algorithm B* (thin solid line), (iv) the second–order model derived in [41] (dotted line), and (v) the second–order model obtained via balanced truncation (dashdotted line).

6. CONCLUSIONS

By exploiting some remarkable properties of classic control–theory tools, a model reduction method has been proposed that: (i) preserves stability, (ii) ensures the conservation of a number of second–order indices, namely the energies of the kernel (7) characterising the transient component of all forced responses, and (iii) minimises the L_2 norm of the approximation error for the reduced–order model's denominator generated via Routh's algorithm. Also, by resorting to the decomposition of the forced response into a transient and a steady–state component [13], the method can easily be adapted to match the original asymptotic response to step inputs.

The procedure, which can be implemented using standard and readily available programs, is computationally very simple and may lead to approximations that are not worse than those afforded by alternative more cumbersome techniques that preserve second–order information indices, as shown by two examples taken from the literature on model reduction.

Even if this paper has been concerned mainly with approximation criteria, model accuracy and procedural complexity, these are by no means the only aspects to be taken into account in the construction of a reduced–order model. For instance, if it is required to design a feedback controller on the basis of a simplified model of a complex plant, care must be taken that the resulting controller is practically implementable and satisfies the design specifications when applied to the actual original plant.

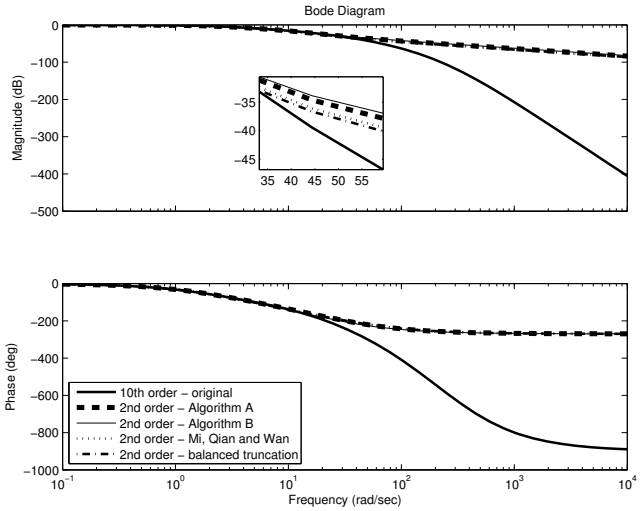

Fig. 6. Bode plots of: (i) the original model (35) (bold solid line), (ii) the second–order approximation (36) obtained using *Algorithm A* (bold dashed line), (iii) the second–order approximation (37) obtained using *Algorithm B* (thin solid line), (iv) the second–order model derived in [41] (dotted line), and (v) the second–order model obtained via balanced truncation (dashdotted line).

(Received July 11, 2016)

REFERENCES

[1] O.M.K. Alsmadi and Z.S. Abo–Hammour: A robust computational technique for model order reduction of two–time–scale discrete systems via genetic algorithms. Computational Intelligence and Neuroscience *2015* (2015), 1–9. DOI:10.1155/2015/615079

[2] A. Astolfi: Model reduction by moment matching for linear and nonlinear systems. IEEE Trans. Automat. Contr. *55* (2010), 2321–2336. DOI:10.1109/tac.2010.2046044

[3] S. Barnett and D. D. Šiljac: Routh's algorithm: a centennial survey. SIAM Review *19* (1977), 472–489. DOI:10.1137/1019070

[4] U. Baur, C. Beattie, P. Benner, and S. Gugercin: Interpolatory projection methods for parameterized model reduction. SIAM J. Scientific Computing *33* (2011), 2489–2518. DOI:10.1137/090776925

[5] Y. Bistritz: Optimal fraction–free Routh tests for complex and real integer polynomials. IEEE Trans. Circuits Syst. I *60* (2013), 2453–2464. DOI:10.1109/tcsi.2013.2246232

[6] A. Beghi, A. Lepschy, and U. Viaro: A property of the Routh table and its use. IEEE Trans. Automat. Control *39* (1994), 2494–2496. DOI:10.1109/9.362839

[7] P. Benner, S. Grundel, and N. Hornung: Parametric model order reduction with a small H_2–error using radial basis functions. Adv. Comput. Math. *41* (2015), 5, 1231–1253. DOI:10.1007/s10444-015-9410-7

[8] A. Bultheel and B. De Moor: Rational approximation in linear systems and control. J. Comput. Appl. Math. *121* (2000), 355–378. DOI:10.1016/s0377-0427(00)00339-3

[9] A. Bultheel and M. Van Barel: Padé techniques for model reduction in linear system theory: a survey. J. Comput. Appl. Math. *14* (1986), 401–438. DOI:10.1016/0377-0427(86)90076-2

[10] Y. Chahlaoui and P. Van Dooren: A collection of benchmark examples for model reduction of linear time invariant dynamical systems. SLICOT Working Note 2002-2: http://slicot.org/20-site/126-benchmark-examples-for-model-reduction. (2002).

[11] S. R. Desai and R. Prasad: A new approach to order reduction using stability equation and big bang big crunch optimization. Systems Science Control Engineering: An Open Access J. *1* (2013), 20–27. DOI:10.1080/21642583.2013.804463

[12] S. R. Desai and R. Prasad: Generating lower order systems using modified truncation and PSO. Int. J. Computer Appl. *12* (2013), 17–21.

[13] P. Dorato, A. Lepschy, and U. Viaro: Some comments on steady-state and asymptotic responses. IEEE Trans. Education *37* (1994), 264–268. DOI:10.1109/13.312135

[14] A. Ferrante, W. Krajewski, A. Lepschy, and U. Viaro: Convergent algorithm for L_2 model reduction. Automatica *35* (1999), 75–79. DOI:10.1016/s0005-1098(98)00142-3

[15] L. Fortuna, G. Nunnari, and A. Gallo: Model Order Reduction Techniques with Applications in Electrical Engineering. Springer–Verlag, London 1992. DOI:10.1007/978-1-4471-3198-4

[16] P. A. Fuhrmann: A polynomial approach to Hankel norm and balanced approximations. Linear Algebra Appl. *146* (1991), 133–220. DOI:10.1016/0024-3795(91)90025-r

[17] W. K. Gawronski: Dynamics and Control of Structures: A Modal Approach. Springer, New York, 1998. DOI:10.1016/0024-3795(91)90025-r

[18] K. Glover: All optimal Hankel–norm approximations of linear multivariable systems and their L_∞–error bounds. Int. J. Control *39* (1984), 1115–1193. DOI:10.1080/00207178408933239

[19] G. Gu: All optimal Hankel–norm approximations of linear multivariable systems and their L_∞–error bounds in discrete time. Int. J. Control *78* (2005), 408–423. DOI:10.1080/00207170500110988

[20] S. Gugercin and A. Antoulas: A survey of model reduction by balanced truncation and some new results. Int. J. Control *77* (2004), 748–766. DOI:10.1080/00207170410001713448

[21] S. Gugercin, A. Antoulas, and C. Beattie: A rational Krylov iteration for optimal H_2 model reduction. Proc. 17th Int. Symp. Mathematical Theory of Networks and Systems, Kyoto 2006, pp. 1665–1667.

[22] S. Gugercin, A. Antoulas, and C. Beattie: H_2 model reduction for large–scale linear dynamical systems. SIAM J. Matrix Analysis Appl. *30* (2008), 609–638. DOI:10.1137/060666123

[23] S. Gugercin, R. V. Polyuga, C. Beattie, and A. van der Schaft: Structure–preserving tangential interpolation for model reduction of port–Hamiltonian systems. Automatica *48* (2012), 1963–1974. DOI:10.1016/j.automatica.2012.05.052

[24] X. X. Huang, W. Y. Yan, and K. L. Teo: H_2 near–optimal model reduction. IEEE Trans. Automat. Control *46* (2001), 1279–1284. DOI:10.1109/9.940934

[25] M. F. Hutton and B. Friedland: Routh approximations for reducing order of linear, time-invariant systems. IEEE Trans. Automat. Control *20* (1975), 329–337. DOI:10.1109/tac.1975.1100953

[26] R. Jeltsch and M. Mansour, eds: Stability Theory: Hurwitz Centenary Conference 1995. Birkhäuser, Basel, Switzerland, 1996. DOI:10.1007/978-3-0348-9208-7

[27] A. M. King, U. B. Desai, and R. E. Skelton: A generalized approach to q–Markov covariance equivalent realizations for discrete systems. Automatica *24* (1988), 507–515. DOI:10.1016/0005-1098(88)90095-7

[28] W. Krajewski, A. Lepschy, and U. Viaro: Compact form of the optimality conditions for multivariable L_2 model reduction. Atti Ist. Veneto SS.LL.AA.: Classe di Scienze Fisiche, Matematiche e Naturali *Tomo CL (1991/92)* (1992), 119–128.

[29] W. Krajewski, A. Lepschy, G. A. Mian, and U. Viaro: Optimality conditions in multivariable L_2 model reduction. J. Franklin Inst. *330* (1993), 431–439. DOI:10.1016/0016-0032(93)90090-h

[30] W. Krajewski, A. Lepschy, and U. Viaro: Remarks on algorithms for L_2 model reduction. Atti Ist. Veneto SS.LL.AA.: Classe di Scienze Fisiche, Matematiche e Naturali *Tomo CLII (1993/1994)* (1994), 99–106.

[31] W. Krajewski, A. Lepschy, and U. Viaro: Reduction of linear continuous–time multivariable systems by matching first– and second–order information. IEEE Trans. Automat. Control *39* (1994), 2126–2129. DOI:10.1109/9.328814

[32] W. Krajewski, A. Lepschy, and U. Viaro: Model reduction by matching Markov parameters, time moments, and impulse–response energies. IEEE Trans. Automat. Contr. *40*, 949–953. DOI:10.1109/9.384238

[33] W. Krajewski, A. Lepschy, M. Redivo–Zaglia, and U. Viaro: A program for solving the L2 reduced–order problem with fixed denominator degree. Numerical Algorithms *9* (1995), 355–377. DOI:10.1007/bf02141596

[34] W. Krajewski and U. Viaro: On MIMO model reduction by the weighted equation–error approach. Numerical Algorithms *44* (2007), 83–98. DOI:10.1007/s11075-007-9086-2

[35] W. Krajewski and U. Viaro: Iterative–interpolation algorithms for L_2 model reduction. Control and Cybernetics *38* (2009), 543–554.

[36] V. Krishnamurthy and V. Seshadri: Model reduction using the Routh stability criterion. IEEE Trans. Automat. Contr. *23* (1978), 729–731. DOI:10.1109/tac.1978.1101805

[37] A. Lepschy, G. A. Mian, and U. Viaro: A geometrical interpretation of the Routh test. J. Franklin Inst. *325* (1988, 695–703. DOI:10.1016/0016-0032(88)90003-8

[38] A. Lepschy, G. A. Mian, and U. Viaro: Splitting of some s–domain stability test algorithms. Int. J. Control *50* (1989), 2237–2247. DOI:10.1080/00207178908953495

[39] D. G. Luenberger: Optimization by Vector Space Methods. Wiley, New York 1969. DOI:10.1137/1012072

[40] L. Meier and D. G. Luenberger: Approximation of linear constant systems. IEEE Trans. Automat. Contr. *12* (1967), 585–588. DOI:10.1109/tac.1967.1098680

[41] W. Mi, T. Qian, and F. Wan: A fast adaptive model reduction method based on Takenaka–Malmquist systems. Systems Control Lett. *61* (2012), 223–230. DOI:10.1016/j.sysconle.2011.10.016

[42] S. K. Mittal, D. Chandra, and B. Dwivedi: A computer–aided approach for Routh–Padé approximation of SISO systems based on multi–objective optimization. Int. J. Eng. Technol. *2* (2010), 204–210.

[43] B. C. Moore: Principal component analysis in linear systems: controllability, observability, and model reduction. IEEE Trans. Automat. Control *26* (1981), 17–32. DOI:10.1109/tac.1981.1102568

[44] S. Mukherjee, Satakshi, and R .C. Mittal: Model order reduction using response–matching technique. J. Franklin Inst. *342* (2005), 503–519. DOI:10.1016/j.jfranklin.2005.01.008

[45] C. T. Mullis and R. A. Roberts: The use of second–order data in the approximation of discrete–time linear systems. IEEE Trans. Acoust. Speech Signal Process. *24* (1976), 226–238. DOI:10.1109/tassp.1976.1162795

[46] V.Y. Pan: Solving a polynomial equation: some history and recent progress. SIAM Rev. *39* (1997), 187–220. DOI:10.1137/s0036144595288554

[47] S. Panda, S. K. Tomar, R. Prasad, and C. Ardil: Reduction of linear time-invariant systems using Routh-approximation and PSO. Int. J. Electrical Computer Electronics and Communication Eng. *3* (2009), 20–27.

[48] G. Parmar, S. Mukherjee, and R. Prasad: System reduction using factor division algorithm and eigen spectrum analysis. Appl. Math. Modelling *31* (2007), 2542–2552. DOI:10.1016/j.apm.2006.10.004

[49] D. Petersson and J. Löfberg: Model reduction using a frequency–limited H_2 cost. Systems Control Lett. *67* (2012), 32–39. DOI:10.1016/j.sysconle.2014.02.004

[50] L. Qiu: What can Routh table offer in addition to stability?. J. Contr. Theory Appl. *1* (2003), 9–16. DOI:10.1007/s11768-003-0003-5

[51] K. Ramawat and A. Kumar: Improved Padé–pole clustering approach using genetic algorithm for model order reduction. Int. J. Computer Appl. *114* (2015), 24–285. DOI:10.5120/19943-1737

[52] J. S. Rana, R. Prasad, and R. Singh: Order reduction using modified pole clustering and factor division method. Int. J. Innovative Tech. Exploring Eng. *3* (2014), 134–136.

[53] V. S. Ryaben'kii and S. V. Tsynkov: A Theoretical Introduction to Numerical Analysis. Chapman and Hall/CRC (Taylor and Francis Group), Boca Raton 2006.

[54] M. Rydel and W. Stanisławski: Selection of fitness functions for evolutionary algorithms and their influence on the properties of a reduced mathematical MIMO model. In: Proc. IEEE Int. Conf. Methods and Models in Automation and Robotics, Miedzyzdroje 2015. DOI:10.1109/mmar.2015.7283941

[55] D. K. Saini and R. Prasad: Order reduction of linear interval systems using genetic algorithm. Int. J. Eng. Technol. *2* (2010), 316–319.

[56] A. Sikander and R. Prasad: A novel order reduction method using cuckoo search algorithm. IETE J. Research *61* (2015), 83–90. DOI:10.1080/03772063.2015.1009396

[57] A. Sikander and R. Prasad: Linear time invariant system reduction using mixed method approach. Appl. Math. Modelling *39* (2015), 4848–4858. DOI:10.1016/j.apm.2015.04.014

[58] C. B. Soh: Generalization of the Hermite–Biehler theorem and applications. IEEE Trans. Automat. Control *35* (1990), 222–225. DOI:10.1109/9.45186

[59] H. N. Soloklo and M. M. Farsangi: Order reduction by minimizing integral square error and H_∞ norm of error. J. Adv. Computer Res. *5* (2014), 29–42.

[60] V. Sreeram and P. Agathoklis: Model reduction of linear discrete systems via weighted impulse response Gramians. Int. J. Control *53* (1991), 129–144. DOI:10.1080/00207179108953613

[61] N. Tanguy, N. Iassamen, M. Telescu, and P. Cloastre: Parameter optimization of orthonormal basis functions for efficient rational approximations. Appl. Math. Modelling *39* (2015), 4963–4970. DOI:10.1016/j.apm.2015.04.017

[62] P. Van Dooren, K. A. Gallivan, and P. A. Absil: H_2–optimal model reduction of MIMO systems. Appl. Math. Lett. *21* (2008), 1267–1273. DOI:10.1016/j.aml.2007.09.015

[63] A. Varga and B. D. O. Anderson: Accuracy–enhancing methods for balancing-related frequency-weighted model and controller reduction. Automatica *39* (2003), 919–927. DOI:10.1016/s0005-1098(03)00030-x

[64] U. Viaro: Stability tests revisited. In: A Tribute to Antonio Lepschy (G. Picci and M. E. Valcher, eds.), Edizioni Libreria Progetto, Padova 2007, pp. 189–199.

[65] C.B. Vishwakarma and R. Prasad: Order reduction using the advantages of differentiation method and factor division algorithm. Indian J. Engineering & Materials Sciences *15* (2008), 447–451.

[66] Y. Xu and T. Zeng: Optimal H_2 model reduction for large scale MIMO systems via tangential interpolation. Int. J. Numerical Analysis and Modeling *8* (2011), 174–188.

[67] Y. Wang, D. S. Bernstein, and L. T. Watson: Probability–one homotopy algorithms for solving the coupled Lyapunov equations arising in reduced–order H_2/H_∞ modeling, estimation, and control. Appl. Math Comput. *123* (2001), 155–185. DOI:10.1016/s0096-3003(00)00059-x

[68] W. Y. Yan and J. Lam: An approximate approach to H^2 optimal model reduction. IEEE Trans. Automat. Control *44* (1999), 1341–1358. DOI:10.1109/9.774107

[69] C. Zeng and Y. Q. Chen: Global Padé approximations of the generalized Mittag–Leffler function and its inverse. Fractional Calculus and Applied Analysis (arXiv:1310.5592) *18* (2015), 1–15. DOI:10.1515/fca-2015-0086

[70] T. Zeng and C. Lu: Two–sided Grassmann manifold algorithm for optimal H_2 model reduction. Int. J. Numerical Methods Engrg. *104* (2015), 10, 928–943. DOI:10.1515/fca-2015-0086

[71] J. G. Ziegler and N. B. Nichols: Optimum settings for automatic controllers. Trans. ASME *64* (1942), 759–765. DOI:10.1109/tit.1972.1054906

[72] D. Žigić, L. T. Watson, E. G. Collins, Jr., and D. S. Bernstein: Homotopy methods for solving the optimal projection equations for the H_2 reduced order model problem. Int. J. Control *56* (1992), 173–191. DOI:10.1080/00207179208934308

Wiesław Krajewski, Systems Research Institute, Polish Academy of Sciences, ul. Newelska 6, 01–447 Warsaw. Poland.
e-mail: krajewsk@ibspan.waw.pl

Umberto Viaro, Polytechnic Department of Engineering and Architecture, University of Udine, Via delle Scienze 206, 33100 Udine. Italy.
e-mail: umberto.viaro@uniud.it