A DESCRIPTOR SYSTEMS PACKAGE FOR MATHEMATICA

A.I. Vardulakis, N. P. Karampetakis, E. Antoniou, P. Tzekis and S. Vologiannidis

Abstract-- We describe a recently developed integrated software package called Descriptor System Toolbox (DSP), implemented under Mathematica. This new package is fully compatible with two other packages of Mathematica: Control System Professional an add on toolbox of Mathematica that is already in the 1.1 Version and Polynomial Control Systems recently developed by N. Munro* [10]. *The DSP uses the functions developed in these two packages in order to provide new tools for the analysis and synthesis of descriptor system representations also known us generalized state space descriptions. Additional functions are also provided for the manipulation of rational matrices that are quite useful in Control theory applications .

Index terms — descriptor systems, Mathematica computer algebra system, computer aided control systems design

I. INTRODUCTION

Consider a system described by a set of linear differential and/or algebraic equations of the form:

$$E\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t) + Du(t)$$

where $E, A \in R^{n\times n}, B \in R^{n\times m}, C \in R^{p\times n}, D \in R^{p\times m}$ and E possibly non-singular with $\det|sE - A| \neq 0$. Systems of the above form are usually called singular systems, descriptor systems, generalized state space systems, semistate systems etc. It is easily seen that when E is non-singular the system may be rewritten as

$$\dot{x}(t) = E^{-1}Ax(t) + E^{-1}Bu(t)$$
$$y(t) = Cx(t) + Du(t)$$

which is the well-known state space representation. Therefore, descriptor systems constitute a more general class of linear systems than state space systems. Descriptor systems appear in the modelling of many physical phenomena, such as engineering systems (power systems, electrical networks, aerospace engineering, chemical processes), social economic systems, network analysis, biological systems, etc. An extended reference on descriptor systems may be found in [3], [5], [9].

Descriptor systems are part of a more general class of systems, named polynomial matrix descriptions (PMDs) described by the following differential and algebraic equations:

$$A(\rho)\beta(t) = B(\rho)u(t)$$
$$y(t) = C(\rho)\beta(t) + D(\rho)u(t)$$

where $\rho := d/dt$, $A(\rho) \in R[\rho]^{n\times n}$, $B(\rho) \in R[\rho]^{n\times m}$, $C(\rho) \in R[\rho]^{p\times n}$, and $D(\rho) \in R[\rho]^{p\times m}$.

Note that in case where

$$A(\rho) = \rho E - A, B(\rho) = B, C(\rho) = C, D(\rho) = D$$

we get the class of descriptor systems, whereas for

$$A(\rho) = \rho E - A, B(\rho) = B, C(\rho) = C, D(\rho) = D$$

we get the class of state space systems.

In case of state space systems, *Mathematica* has developed a well-known package named *Control Systems Professional* (CSP). CSP is powerful tool for the analysis and synthesis of linear MIMO (multi-input, multi-output) systems as well as SISO (single-input, single-output) systems in both time and frequency domains. CSP has been widely accepted in control engineering, mechanical engineering, aerodynamics, satellite instrumentation, etc. Professor N. Munro [10] has recently implemented an add-on package for CSP, named *Polynomial Control Systems* toolbox dealing with the general class of polynomial matrix descriptions. Some of the features of this new package are: a) Model Manipulation (transformation between various type of models, i.e. PMDs, state space systems, matrix fraction descriptions, b) System analysis i.e. study of the properties of the PMD such as controllability, observability, transmission zeros, invariant zeros, decoupling zeros etc., c) Multivariable system design, i.e. implementation of the Rosenbrock's Direct and Inverse Nyquist array design methods, characteristic locus etc..

On the other hand *MathWorks Inc*, has developed the *Control Systems* package that together with *Simulink* is a numerically oriented environment for the analysis and synthesis of MIMO and SISO systems. Varga [15] has extended the functionality of the *Control System Toolbox* of Matlab by allowing the manipulation of descriptor systems which in turn became known as the *Descriptor Systems Toolbox* for *Matlab*. Sebek [11] developed the *Polynomial System Toolbox* for *Matlab* for the study of systems, signals and their analysis and design, employing advanced polynomial methods.

In order to extend the functionality of the CSP toolbox for *Mathematica*, while maintaining compatibility to the *Polynomial Control Systems* toolbox, we have developed a toolbox that allows: a) manipulation of polynomial and

Department of Mathematics, Aristotle University of Thessaloniki, Greece, 54006, email : karampet@auth.gr

rational matrices i.e. solution of rational matrix Diophantine equations over several rings (polynomial, proper, proper and stable etc), b) manipulation of linear model descriptions i.e. descriptor systems and tools for transformation between various type of models such as state space systems, polynomial matrix descriptions etc., c) system analysis i.e. computations of various types of invariants such as decoupling zeros, system zeros and poles, controllability matrix, reachability matrix, observability matrix, etc. d) time domain responses i.e. state space and output responses to arbitrary input functions, e) synthesis and design techniques i.e. stabilizing compensator design, asymptotic tracking, pole assignment, etc. The methodology that has been used is known in the literature as the polynomial matrix approach. The "Polynomial matrix approach" for analysis and synthesis of linear multivariable control systems is a modern technique which is based on mathematical models of multivariable systems or processes which consist of sets of differential / difference and algebraic equations that govern the behavior of a system or process, relying heavily on the algebraic properties of polynomial matrices. Results and techniques on Polynomial Matrix Descriptions (PMDs) of linear systems can be found in [1, 2, 6, 8, 12, 14, 16 and 17].

II. MATRIX MANIPULATION

It is know that rational functions ring theory plays a crucial role in the Analysis and Synthesis of linear systems. Therefore in this section we introduce new functions for the study of rings of rational functions with poles in a prescribed region of the complex plane as well as for rational matrices with entries coming from these rings. We focus on a) the ring of rational functions with no poles in the complex plane (polynomials), b) the ring of rational functions with no poles at infinity (proper functions), c) the ring of rational functions with no poles in the extended right half complex plane (proper and Hurwitz stable rational functions) and d) the ring of rational functions with no poles outside the unit circle (proper and Schur stable rational functions). In the context of this section we propose a set of functions that a) determine whether a rational function (or a matrix) belongs to a particular ring b) calculate the quotient and remainder of a division between two rational functions over a particular ring, c) compute the greatest common divisor and least common multiple of functions (rational matrices) over a particular ring, d) check whether two rational functions (matrices) are coprime over a particular ring, e) compute structural invariants of a rational matrix such as the Smith form of a rational matrix in a particular ring, the Smith - McMillan form of a rational matrix using unimodular matrices over a particular ring, f) determine particular and general solutions of rational matrix Diophantine equations over specific rings (A(s)X(s)+B(s)Y(s)=C(s), X(s)A(s)+Y(s)B(s)=C(s), A(s)X(s)+Y(s)B(s)=C(s)), g) find generalized inverses of rational matrices (Moore – Penrose and Drazin inverse), i) compute finite and infinite Jordan pairs of polynomial matrices that are very useful in the analysis of polynomial matrix descriptions. In what follows we present some illustrative examples of the above

Example 1. Consider a PMD described by the Rosenbrock system matrix:

$$P(s)=\begin{bmatrix} T(s) & U(s) \\ -V(s) & W(s) \end{bmatrix} =$$

$$=\left[\begin{array}{cccc|c} s+1 & s^3+2s^2 & s^2+1 & 0 & 0 \\ s^2+3s+2 & s^4+4s^3+4s^2+s+2 & s^3+2s^2+s+3 & 0 & 0 \\ s^2+3s+1 & s^4+4s^3+4s^2-1 & s^3+2s^2+s+2 & 1 & 0 \\ 0 & 0 & -1 & 0 & 1 \\ \hline 0 & 0 & 0 & -1 & 0 \end{array}\right]$$

The above system matrix may be defined using the *Polynomial Control Systems* package as follows:

```
<<DescriptorControlSystems;
t = {{s + 1, s^3 + 2s^2, s^2 + 1, 0},
{s^2 + 3s + 2, s^4 + 4s^3 + 4s^2 + s + 2, s^3 +
2s^2 + s + 3, 0},
{s^2 + 3s + 1, s^4 + 4s^3 + 4s^2 - 1, s^3 + 2s^2 +
s + 2, 1},
{0, 0, -1, 0}};
u = {{0}, {0}, {0}, {1}};
v = {{0, 0, 0, 1}};
ss = SystemMatrix[s, t, u, v]
```

$$\left(\begin{array}{cccc|c} s+1 & s^3+2s^2 & s^2+1 & 0 & 0 \\ s^2+3s+2 & s^4+4s^3+4s^2+s+2 & s^3+2s^2+s+3 & 0 & 0 \\ s^2+3s+1 & s^4+4s^3+4s^2-1 & s^3+2s^2+s+2 & 1 & 0 \\ 0 & 0 & -1 & 0 & 1 \\ \hline 0 & 0 & 0 & -1 & 0 \end{array}\right)^{\mathcal{M}}$$

In order to investigate the infinite structure of the system we may use some of the functions of the Matrix Manipulation section i.e.:

Infinite system poles

```
RingMcMillanForm[t, s, ForbiddenPolesArea ->
InfinityPoint][[1]] // MatrixForm
```

$$\begin{pmatrix} s^4 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \frac{1}{s^2} \end{pmatrix}$$

Infinite transmission poles-zeros

```
Tf=v.Inverse[t].u
RingMcMillanForm[tf, s, ForbiddenPolesArea ->
InfinityPoint][[1]]
```

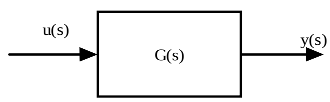
$$\begin{pmatrix} \frac{1}{s} \end{pmatrix}$$

Infinite input-decoupling zeros

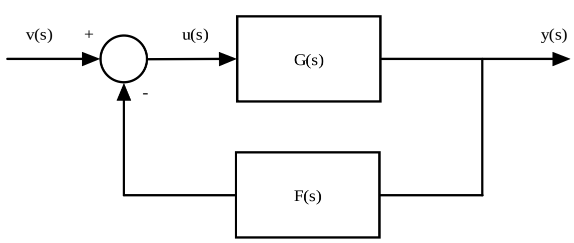
```
sc = AppendRows[t, u] ;
RingMcMillanForm[sc, s, ForbiddenPolesArea ->
InfinityPoint][[1]]
```

$$\begin{pmatrix} s^4 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{s^2} & 0 \end{pmatrix}$$

Example 2. Consider a SISO system described by



where $G(s) = \dfrac{1}{s(s-1)}$. In order to find a stabilizing compensator *F(s)* for the above system



we must first compute a left and a right coprime proper and Hurwitz stable matrix fraction description of $G(s)$. This can be easily done using the functions **`RingLeftMatrixFraction`** and **`RingRightMatrixFraction`** as follows

```
tf = {{1/(s(s-1))}};
{A1, B1} = RingLeftMatrixFraction[tf, s,
ForbiddenPolesArea -> HurwitzComplexPlane]
```

$$\left(\frac{9(s-1)s}{(s+1)^2} \right) \left(\frac{9}{(s+1)^2} \right)$$

```
{B2, A2} = RingRightMatrixFraction[tf, s,
ForbiddenPolesArea -> HurwitzComplexPlane]
```

$$\left(\frac{9}{(s+1)^2} \right) \left(\frac{9(s-1)s}{(s+1)^2} \right)$$

The next step is to compute a particular proper and Hurwitz stable solution X1,Y1 of the left matrix Diophantine equation X1 A2 + Y1 B2 = I, using the function **`LeftDiophantineSolve`.**

```
{D1, N1}=LeftDiophantineSolve[A2, B2,
IdentityMatrix[1], s, ForbiddenPolesArea ->
RightComplexPlane]
```

$$\left(\frac{s+4}{9(s+1)} \right) \quad \left(\frac{7s+1}{9(s+1)} \right)$$

Then the stabilizing compensator we are looking for is given by

```
W = Array[f, {1, 1}];
F = (D1 - W.B1).Inverse[N1 + W.A1] // Factor
```

$$\left(-\frac{81\, f(1,1)\, s^2 - s^2 - 81\, f(1,1)\, s - 5s - 4}{7s^2 + 8s + 81\, f(1,1) + 1} \right)$$

III. LINEAR MODEL DESCRIPTION

The aim of this section is to provide a set of functions for: a) the definition of descriptor systems in a way that conforms with CSP standards, b) the transformation between other objects such as state space system, polynomial matrix description, left or right matrix fraction description provided by CSP and PCS, c) the determination of special forms of descriptor systems such as Kalman controllable / observable form, Weierstrass canonical form, Kalman controllable / observable forms, strongly irreducible realizations and minimal realizations. We also provide right and left matrix fractions representations of a transfer function when the numerator and denominator matrices are either polynomial coprime matrices or proper coprime matrices or proper and Schur stable coprime matrices or proper and Hurwitz stable coprime matrices. This construction is very useful in synthesis problems such as stabilizing compensator design, asymptotic tracking etc. In what follows, we demonstrate the above functionality

Example 3. Consider the polynomial matrix description of example 1. A descriptor state space representation of this system is given by

```
Dss=DescriptorStateSpace[ss]
```

$$\left(\begin{array}{ccccccccc|ccccccccc|c}
0 & -\frac{1}{2\sqrt{2}} & 0 & \frac{\sqrt{\frac{5}{6}}}{2} & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -\frac{1}{4} & -\frac{\sqrt{3}}{2} & \frac{\sqrt{\frac{5}{3}}}{4} & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -\frac{\sqrt{\frac{3}{5}}}{4} & \frac{1}{2\sqrt{5}} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{17}{11} & -\frac{2\sqrt{\frac{21}{5}}}{11} & -\frac{3}{\sqrt{55}} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{5\sqrt{\frac{15}{7}}}{11} & -\frac{16}{11} & -\frac{2}{\sqrt{231}} \\
\hline
 & & & & & & & & & -2\sqrt{\frac{2}{3}} & -\frac{1}{2\sqrt{3}} & 4 & -\frac{\sqrt{5}}{2} & 0 & -1 & 0 & -8\sqrt{\frac{5}{11}} & \sqrt{\frac{21}{11}} & 0
\end{array}\right)^{\mathcal{D}}.$$

A minimal realization of the above system can be easily obtained by using the **`MinimalRealization`** function

```
MinimalRealization[dss]
```

$$\left(\begin{array}{cc|cc|c} 1 & 0 & -2 & 0 & \frac{7}{\sqrt{95}} \\ 0 & 1 & 0 & -1 & -\frac{1}{\sqrt{95}} \\ \hline & & \frac{\sqrt{95}}{7} & -\sqrt{95} & 0 \end{array}\right)^{\mathcal{D}}.$$

We can easily see that the above system is already in the Weierstrass canonical form.

Example 4. Define the descriptor state space system described by the following matrices

$$E=\begin{bmatrix}0&0&0\\0&1&1\\1&0&0\end{bmatrix};A=\begin{bmatrix}1&0&0\\0&2&0\\0&0&1\end{bmatrix};B=\begin{bmatrix}1\\0\\0\end{bmatrix};C=\begin{bmatrix}0&0&1\end{bmatrix}$$

The above system is defined as follows

```
Em={{-13,17,-1},{-10,13,-1},{4,-5,1}} ;
A=IdentityMatrix[3] ;
B={{0},{0},{1}} ;
Cm={{0,0,1}} ;
ds=DescriptorStateSpace[Em,A,B,Cm]
```

$$\left(\begin{array}{ccc|ccc|c} -13 & 17 & -1 & 1 & 0 & 0 & 0 \\ -10 & 13 & -1 & 0 & 1 & 0 & 0 \\ 4 & -5 & 1 & 0 & 0 & 1 & 1 \\ \hline & & & 0 & 0 & 1 & 0 \end{array}\right)^{\mathcal{D}}.$$

The Weierstrass canonical form of the above system is given by

```
WeierstrassCanonicalForm[ds]
```

$$\left(\begin{array}{ccc|ccc|c} 1 & 0 & 0 & 1 & 0 & 0 & \frac{1}{\sqrt{3}} \\ 0 & 0 & 1 & 0 & 1 & 0 & -\frac{9}{\sqrt{4579}} \\ 0 & 0 & 0 & 0 & 0 & 1 & \frac{5}{\sqrt{4579}} \\ \hline & & & 2\sqrt{3} & -\frac{\sqrt{4579}}{5} & -\frac{14\sqrt{4579}}{25} & 0 \end{array}\right)^{\mathcal{D}}.$$

Its transfer function matrix is

```
tf=TransferFunction[ds]
```

$$\left(\frac{s^2+1}{s-1}\right)^{\mathcal{T}}$$

A right proper matrix fraction description of the above transfer function is given by the pair of matrices

```
RingRightMatrixFraction[tf, s,
       ForbiddenPolesArea ->InfinityPoint]
```

$$(1) \qquad \left(\frac{s-1}{s^2+1}\right)$$

IV. System Analysis Properties

In this section we propose new functions for the determination of the structural invariants and properties of descriptor systems. Particularly a) we determine various invariants of the system such as controllability, reachability and observability matrices, finite and infinite decoupling zeros (input, output, input-output), finite and infinite system poles and zeros, finite and infinite invariant zeros, finite and infinite transmission poles and zeros, b) we check system properties i.e. whether the descriptor system is controllable / reachable, observable, stabilizable, detectable, stable, internally proper or internally stable.

Example 5. Consider the descriptor system presented in Example 4. The reachability matrix [14] of the above system is given by

```
Rm=ReachabilityMatrix[ds]
```

$$\begin{pmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & -\frac{9}{\sqrt{4579}} & \frac{5}{\sqrt{4579}} \\ 0 & \frac{5}{\sqrt{4579}} & 0 \end{pmatrix}$$

Then we can easily check whether its rank is equal or not to the dimension of the matrix *E* and therefore the reachability of the system.

```
Rank[rm]==3
```

True

Similarly the observability of the system can be easily checked as follows

```
om=ObservabilityMatrix[ds]
```

$$\begin{pmatrix} 2\sqrt{3} & 0 & 0 \\ 0 & -\frac{\sqrt{4579}}{5} & -\frac{14\sqrt{4579}}{25} \\ 0 & -\frac{14\sqrt{4579}}{25} & 0 \end{pmatrix}$$

```
Rank[om]==3
```

True

V. Time Domain Responses

In this section we compute symbolically the smooth and impulsive state/output response of the descriptor system given input and initial conditions.

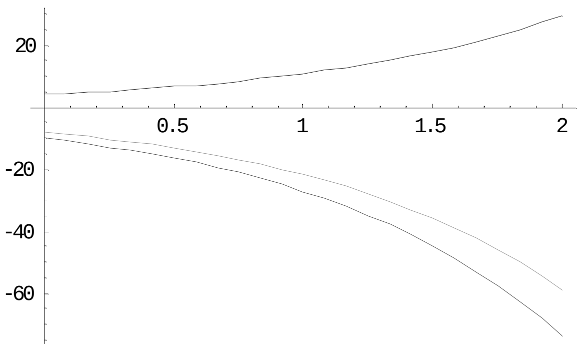
Example 6. Consider the system of Example 4. Given the following initial conditions and input

```
x0 = {1, 0,0};
ut = {DiracDelta[t]};
```

we get the following state space response and its plot

```
xd=StateResponse[ds,ut,t,InitialConditions->x0]
Plot[Evaluate[xd],{t,0,2},PlotStyle->
{RGBColor[1,0,0],RGBColor[0,1,0],
RGBColor[0,0,1]}]
```

$$\{3\,\delta(t) - 5\,e^{t}\,(\theta(t)+1) - 4\,\text{DiracDelta}'[t],\;
2\,\delta(t) - 4\,e^{t}\,(\theta(t)+1) - 3\,\text{DiracDelta}'[t],\;
-\delta(t) + 2\,e^{t}\,(\theta(t)+1) + \text{DiracDelta}'[t]\}$$



VI. DESIGN SYNTHESIS TECHNIQUES

This section is divided into three parts: a) implementation of classical design methods for MIMO systems such as stabilizing compensator design, asymptotic tracking, model matching and disturbance rejection, b) descriptor system interconnections such as series, parallel, feedback and generic interconnection, and c) pole assignment techniques for descriptor system.

Example 7. Consider the system in Example 4. The Smith McMillan form of the pencil is

```
RingMcMillanForm[s*Em -A, s][[1]]
```

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & s-2 \end{pmatrix}$$

and its Smith McMillan form at infinity is

```
RingMcMillanForm[s*Em -A, s, ForbiddenPolesArea ->
InfinityPoint][[1]]
```

$$\begin{pmatrix} s & 0 & 0 \\ 0 & s & 0 \\ 0 & 0 & \frac{1}{s} \end{pmatrix}$$

In order to assign the poles of the system to {-1, -2} by constant state feedback we use the following function:

```
StateFeedbackGains[ds,{ -1, -2}]
```

$$\begin{pmatrix} 0 & -3 & -2 \end{pmatrix}$$

Example 8. Consider the SISO system presented in Example 2. The Hurwitz stable stabilizing compensator for the above system is given by

```
StabilizingCompensator[H,…….]
```

$$\frac{1}{4}\,\frac{28W_{d,1,1}s + W_{d,1,1} + 52W_{d,1,1}s^2 + 16W_{n,1,1}s^2 - 16W_{n,1,1}s}{12W_{d,1,1}s + 5W_{d,1,1} + 4W_{d,1,1}s^2 - 4W_{n,1,1}}$$

where the functions $W_{d,1,1}, W_{n,1,1}$ are arbitrary Hurwitz and proper stable functions. Similarly the Schur stabilizing compensators are given by

```
StabilizingCompensator[H,…….]
```

$$\frac{1}{4}\,s\,\frac{36W_{d,1,1}s^2 + 36W_{d,1,1}s + 9W_{d,1,1} + 16W_{n,1,1}s^2 - 16sW_{n,1,1}}{12W_{d,1,1}s^2 + 9W_{d,1,1}s + 2W_{d,1,1} + 4W_{d,1,1}s^3 - 4sW_{n,1,1}}$$

where the functions $W_{d,1,1}, W_{n,1,1}$ are arbitrary Schur and proper stable functions

VII. CONCLUSIONS

The *Descriptor State Space* (*DSS*) package is a new package written in the *Mathematica* programming language and its primary aim is to extend the functionality of the *Control Systems Professional* package to handle descriptor state space representations. The *DSS* package is fully compatible to *Control Systems Professional* implemented by *Wolfram Research* and the *Polynomial Control Systems* written by Prof. Munro [10]. Most of the procedures used in *DSS* are symbolic and based on well established polynomial and algebraic theory results. Our package benefits from the accuracy of the symbolic manipulations at the cost of speed and memory efficiency. However, Mathematica provides a very powerful numerical engine, that we intend to employ in a future release of the product.

Acknowledgements: The help of Dr Igor Bakshee, of Wolfram Research Inc., during the preparation of the *Descriptor State Space* package is gratefully acknowledged.

VIII. REFERENCES

[1] Antsaklis P. J., Michel N.A., Linear Systems, The McGraw Hill Company In. 1977.
[2] Blomberg H. and Ylinen R., 1983, Algebraic Theory of Multivariable Linear Systems, Mathematics in Science and Engineering, Vol 166, Academic Press, London.
[3] S.L. Campbell, 1980, Singular Systems of Differential Equations, Research Notes in Mathematics, Pitman Advanced Publishing Program.
[4] Control System Professional from Wolfram Research, http://www.wolfram.com/products/applications/control/
[5] L. Dai, 1989, Singular Control Systems, Lecture Notes in Control and Information Sciences, Springer-Verlag.

[6] T. Kailath, Linear Systems, Englewood Cliffs: Prentice Hall, New York 1980.

[7] Kalman R.E. , Falb P.L. and Arbib M.A., Topics in Mathematical Systems Theory, International Series in Pure and Applied Mathematics, McGraw-Jill Book Company, New York 1969.

[8] Kucera V, Discrete Linear Control, the polynomial equation approach, John Wiley and Sons, 1979.

[9] F.L. Lewis, 1986, A survey of linear singular systems, Circuit Systems & Signal Process, Vol.5, No.1, pp.3-36.

[10] N. Munro, 2002, A polynomial control systems package, 11th IEEE Meditteranean Conference.

[11] Polynomial System Toolbox from POLYX, http://www.polyx.com/

[12] Rosenbrock H.H., State-Space and Multivariable Theory, Nelson, London 1970.

[13] Rosenbrock H.H., Computer-Aided Control System Design, Academic Press, London 1974.

[14] Vardulakis, A.I.G., Linear Multivariable Control, Algebraic Analysis and Synthesis Methods, John Wiley and Sons, Chichester, New York, 1991.

[15] Varga, A., 2000, A descriptor systems toolbox for MATLAB. Proc. of IEEE International Symposium on Computer Aided Control System Design, CACSD'2000, Anchorage, Alaska, 2000.

[16] Verghese G., Infinite Freequency Behaviour in Generalised Dynamically Systems, Ph.D. Dissertation, Stanford Univ., Stanford, CA. 1978.

[17] Vidyasagar M., Control System Synthesis, Cambridge MIT Press, 1985.

[18] Wolovich W.A., Linear Multivariable Systems, Springer Verlag, New York 1974.