



ΑΡΙΣΤΟΤΕΛΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΟΝΙΚΗΣ

ΤΜΗΜΑ ΜΑΘΗΜΑΤΙΚΩΝ

ΜΕΤΑΠΤΥΧΙΑΚΟ ΠΡΟΓΡΑΜΜΑ ΣΠΟΥΔΩΝ

«ΘΕΩΡΗΤΙΚΗ ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΘΕΩΡΙΑ ΣΥΣΤΗΜΑΤΩΝ ΚΑΙ ΕΛΕΓΧΟΥ»

Ανάπτυξη και Εφαρμογή μέσω της Γλώσσας Συμβολικής  
Επεξεργασίας Mathematica Αλγορίθμων και Διαδραστικού  
Υπολογιστικού Περιβάλλοντος με Χρήση Γραφικών  
(Graphics User Interface) για τον Υπολογισμό της  
Παραμετρικής Οικογένειας Σταθεροποιητικών  
Αντισταθμιστών Γραμμικών Πολυμεταβλητών Συστημάτων

## ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Μιχάλης Μ. Κασιδιάρης

**Επιβλέπων:** Αντώνιος Ι. Γ. Βαρδουλάκης  
Καθηγητής τμ. Μαθηματικών Α.Π.Θ.

Θεσσαλονίκη, Φεβρουάριος 2009





ΑΡΙΣΤΟΤΕΛΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΟΝΙΚΗΣ  
ΤΜΗΜΑ ΜΑΘΗΜΑΤΙΚΩΝ  
ΜΕΤΑΠΤΥΧΙΑΚΟ ΠΡΟΓΡΑΜΜΑ ΣΠΟΥΔΩΝ  
«ΘΕΩΡΗΤΙΚΗ ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΘΕΩΡΙΑ ΣΥΣΤΗΜΑΤΩΝ ΚΑΙ ΕΛΕΓΧΟΥ»

Ανάπτυξη και Εφαρμογή μέσω της Γλώσσας Συμβολικής  
Επεξεργασίας Mathematica Αλγορίθμων και Διαδραστικού  
Υπολογιστικού Περιβάλλοντος με Χρήση Γραφικών  
(Graphics User Interface) για τον Υπολογισμό της  
Παραμετρικής Οικογένειας Σταθεροποιητικών  
Αντισταθμιστών Γραμμικών Πολυμεταβλητών Συστημάτων

## ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**Μιχάλης Μ. Κασιδιάρης**

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή.

.....  
Α. Ι. Γ. Βαρδουλάκης  
Καθηγητής Α.Π.Θ.

.....  
Μ. Γουσίδου Κουτίτα  
Αν. Καθηγήτρια Α.Π.Θ.

.....  
Ν. Π. Καραμπετάκης  
Επ. Καθηγητής Α.Π.Θ.

Θεσσαλονίκη, Φεβρουάριος 2009

.....  
Μιχάλης Μ. Κασιδιάρης  
Πτυχιούχος Μαθηματικός Α.Π.Θ.

Copyright © Μιχάλης Μ. Κασιδιάρης, 2009.  
Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευτεί ότι εκφράζουν τις επίσημες θέσεις του Α.Π.Θ.

## ΠΕΡΙΛΗΨΗ

Δεδομένου γραμμικού συστήματος αυτομάτου ελέγχου πολλών εισόδων και εξόδων το οποίο εκφράζεται από ένα πίνακα αυστηρώς κανονικών συναρτήσεων μεταφοράς, αναζητούνται όλοι οι κανονικοί αντισταθμιστές (δηλαδή που περιγράφονται από πίνακα κανονικών συναρτήσεων μεταφοράς) που σε διάταξη μοναδιαίας ανάδρασης σταθεροποιούν το αρχικό σύστημα, τοποθετώντας επιθυμητές ρίζες στον πίνακα παρονομαστή του κλειστού συστήματος. Η παρούσα εργασία υλοποιεί στον Η/Υ, μέσω της συμβολικής γλώσσας προγραμματισμού Mathematica, συγκεκριμένο αλγόριθμο επίλυσης του παραπάνω προβλήματος, δημιουργώντας όλα τα προαπαιτούμενα εργαλεία καθώς και ένα γραφικό περιβάλλον χρήσης μέσω του οποίου εισάγονται τα δεδομένα και διεκπεραιώνονται όλοι οι υπολογισμοί. Τέλος, κατά τον σχεδιασμό της εφαρμογής, δόθηκε έμφαση στην δυνατότητα χρήσης της, με όσο το δυνατόν μικρότερες απαιτήσεις γνώσης σύνταξης εντολών της Mathematica και απομνημόνευσης των ενδιάμεσων βημάτων του υπολογισμού.

## ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ

Λογισμικό Συστημάτων Ελέγχου, Γραμμικά Πολυμεταβλητά Συστήματα, Σύνθεση Συστημάτων Ελέγχου με τη Βοήθεια Η/Υ.

## **ABSTRACT**

Given a linear, multi-input, multi-output system, described by a strictly proper plant, all proper compensators are sought which, when employed in a unity feedback configuration, stabilize the initial system by assigning specific roots to the resulting closed loop system denominator matrix. The objective of the present thesis is the realization to computer software, using the symbolic programming language Mathematica, of a specific algorithm resolving the problem described above, by creating all the necessary computational tools along with a graphical user interface through which all data is entered and all computational tasks are performed. Finally, the possibility of using the application without the necessity of prior experience with Mathematica, as well as of the need to memorize the exact computational steps involved has been of special importance throughout its development.

## **KEYWORDS**

Control Systems Software, Linear Multivariable Systems, Computer – aided Control Systems Synthesis.

## ΠΡΟΛΟΓΟΣ

Θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα Καθηγητή κ. Βαρδουλάκη Αντώνιο Ιωάννη, κατ' αρχήν για την εμπιστοσύνη του, η οποία αποτέλεσε ιδιαίτερο κίνητρο για την παρακολούθηση από μέρος μου του μεταπτυχιακού προγράμματος σπουδών, καθώς και για την καθοδήγηση και τις υποδείξεις του κατά τη διάρκεια της εκπόνησης της παρούσας εργασίας.

Ευχαριστώ, επίσης, θερμά τον Επίκουρο Καθηγητή κ. Καραμπετάκη Νικόλαο και την Αναπληρώτρια Καθηγήτρια κα. Γουσίδου Μαρία, μέλη της τριμελούς εξεταστικής επιτροπής, για τις συμβουλές και το χρόνο που αφιέρωσαν για τη μελέτη και αξιολόγηση της διπλωματικής μου εργασίας, όπως επίσης και για το γενικότερο ενδιαφέρον τους προς τους φοιτητές.

Ιδιαίτερος, θα ήθελα να ευχαριστήσω την οικογένεια μου για την εμπιστοσύνη και την αμέριστη υποστήριξή της καθ' όλη τη διάρκεια των σπουδών μου.





*Στη μνήμη του παππού μου,  
Μιχάλη*



## ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

<b>ΠΕΡΙΛΗΨΗ.....</b>	<b>5</b>
<b>ABSTRACT.....</b>	<b>6</b>
<b>ΠΡΟΛΟΓΟΣ.....</b>	<b>7</b>
<b>ΚΕΦΑΛΑΙΟ 1 ΠΡΟΑΠΑΙΤΟΥΜΕΝΑ.....</b>	<b>13</b>
1.1 Εισαγωγή .....	13
1.2 Βασικοί Συμβολισμοί .....	14
1.3 Γενικευμένη Wolovich resultant .....	14
<b>ΚΕΦΑΛΑΙΟ 2 ΑΛΓΟΡΙΘΜΟΣ ΥΠΟΛΟΓΙΣΜΟΥ ΟΙΚΟΓΕΝΕΙΑΣ</b>	
<b>ΣΤΑΘΕΡΟΠΟΙΗΤΙΚΩΝ ΑΝΤΙΣΤΑΘΜΙΣΤΩΝ .....</b>	<b>17</b>
2.1 Εφαρμογή των Διοφαντικών εξισώσεων πολυωνυμικών πινάκων .....	17
2.2 Ανάλυση αλγορίθμου εύρεσης των λύσεων της Διοφαντικής εξίσωσης για τις οποίες προκύπτουν κανονικοί αντισταθμιστές .....	18
2.3 Σύνθεση αλγορίθμου .....	23
2.4 Παράδειγμα .....	24
<b>ΚΕΦΑΛΑΙΟ 3 ΥΛΟΠΟΙΗΣΗ ΑΛΓΟΡΙΘΜΟΥ ΚΑΙ ΕΦΑΡΜΟΓΗ ΣΤΟ</b>	
<b>ΜΑΤΗΜΑΤΙΚΑ .....</b>	<b>29</b>
3.1 Γενικά .....	29
3.2 Σημαντικές επιμέρους συναρτήσεις και αλγόριθμοι .....	30
3.3 Χρήση της εφαρμογής .....	47
<b>ΒΙΒΛΙΟΓΡΑΦΙΑ .....</b>	<b>63</b>
<b>ΠΑΡΑΡΤΗΜΑ .....</b>	<b>65</b>

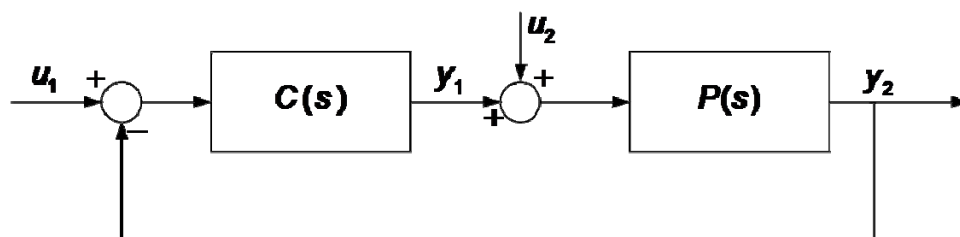


## ΚΕΦΑΛΑΙΟ 1 ΠΡΟΑΠΑΙΤΟΥΜΕΝΑ

### 1.1 Εισαγωγή

Η ανάπτυξη μεθόδων σταθεροποίησης συστημάτων που χαρακτηρίζονται από ανεπιθύμητη συμπεριφορά (αστάθεια) αποτελεί κύριο πεδίο εφαρμογών της Θεωρίας Ελέγχου. Είναι γνωστό ότι, στην περίπτωση που ένα σύστημα περιγράφεται από τη συνάρτηση μεταφοράς του, η ευστάθεια προσδιορίζεται από την διάταξη των πόλων της συνάρτησης μεταφοράς πάνω στο μιγαδικό επίπεδο. Επομένως, η σταθεροποίηση ενός ασταθούς συστήματος ισοδυναμεί με σχεδιασμό ενός νέου τροποποιημένου συστήματος, η συνάρτηση μεταφοράς του οποίου έχει επιθυμητούς πόλους. Η παρούσα εργασία έχει ως αντικείμενο την κατασκευή προγράμματος στην συμβολική γλώσσα προγραμματισμού Mathematica, το οποίο υλοποιεί συγκεκριμένο αλγόριθμο υπολογισμού αντισταθμιστών για τη σταθεροποίηση γραμμικών πολυμεταβλητών συστημάτων και, ταυτοχρόνως, παρέχει διαδραστικό περιβάλλον χρήσης μέσω του οποίου πραγματοποιείται ο υπολογισμός.

Ο αλγόριθμος υπολογισμού είναι αυτός που δημοσιεύτηκε με την εργασία “On the computation and parametrization of proper denominator assigning compensators for strictly proper plants” των Ε. Ν. Αντωνίου και Α. Ι. Γ. Βαρδουλάκη και εφαρμόζεται σε γραμμικά, πολυμεταβλητά, χρονικώς αναλλοίωτα συστήματα τα οποία εκφράζονται από αυστηρώς κανονικές συναρτήσεις μεταφοράς και υπολογίζει την παραμετροποιημένη οικογένεια όλων των αντισταθμιστών εκείνων που εκφράζονται από κανονικές συναρτήσεις μεταφοράς. Τα επαγόμενα κλειστά συστήματα έχουν τη γενική μορφή του σχήματος 1.1.



Σχήμα 1.1

Στη συνέχεια του Κεφαλαίου 1 εισάγονται οι βασικοί συμβολισμοί που συναντώνται στα επόμενα, ο ορισμός της γενικευμένης Wolovich resultant που αποτελεί

το κύριο εργαλείο ανάλυσης και υπολογισμού της μεθόδου που χρησιμοποιείται, καθώς και σχετικά αποτελέσματα που χρησιμοποιούνται για τη σύνθεση του αλγορίθμου. Στο Κεφάλαιο 2 παρουσιάζεται συνοπτικά ο υλοποιούμενος από το πρόγραμμα αλγόριθμος ενώ στο Κεφάλαιο 3 παρουσιάζεται η εφαρμογή στην Mathematica, η οποία αποτελεί και το αντικείμενο της παρούσας εργασίας. Τέλος, στο Παράρτημα, περιέχεται ο πλήρης κώδικας της εφαρμογής.

## 1.2 Βασικοί Συμβολισμοί

Στα παρακάτω χρησιμοποιούμε τους συμβολισμούς  $\mathbb{R}$  για το σώμα των πραγματικών αριθμών,  $\mathbb{C}$  για το σώμα των μιγαδικών αριθμών,  $\mathbf{m}$  για το σύνολο των φυσικών αριθμών  $\{1, 2, \dots, m\}$ ,  $\mathbb{R}[s]$  για να συμβολίσουμε τον *δακτύλιο των πολυωνύμων μίας μεταβλητής με πραγματικούς συντελεστές*,  $\mathbb{R}[s]^{p \times m}$  για το σύνολο των  $p \times m$  πινάκων με στοιχεία πολυώνυμα του  $\mathbb{R}[s]$  και τους οποίους για συντομία θα αποκαλούμε  $p \times m$  πολυωνυμικούς πίνακες,  $\mathbb{R}(s)^{p \times m}$  για το σύνολο των  $p \times m$  πινάκων με στοιχεία ρητές εκφράσεις στοιχείων του  $\mathbb{R}[s]$  και τους οποίους για συντομία θα αποκαλούμε  $p \times m$  ρητούς πίνακες,  $\mathbb{R}_{pr}(s)^{p \times m}$  για το σύνολο των  $p \times m$  κανονικών ρητών πινάκων,  $\mathbb{R}_{po}(s)^{p \times m}$  για το σύνολο των  $p \times m$  αυστηρώς κανονικών ρητών πινάκων. Τέλος, με  $\delta_M[\bullet]$  συμβολίζουμε τον βαθμό McMillan του  $[\bullet]$ .

## 1.3 Γενικευμένη Wolovich resultant

Στην παράγραφο αυτή ορίζεται η γενικευμένη Wolovich resultant (Antoniou & Vardulakis, 2005), η οποία αποτελεί και κύριο εργαλείο στο οποίο βασίζεται ο αλγόριθμος υπολογισμού που χρησιμοποιείται στην παρούσα εργασία, καθώς και τα απαραίτητα, για τη σύνθεση του συγκεκριμένου αλγορίθμου, αποτελέσματα.

Ας είναι  $N_R(s) \in \mathbb{R}[s]^{p \times m}$ ,  $D_R(s) \in \mathbb{R}[s]^{m \times m}$ , ένα ζεύγος πολυωνυμικών πινάκων, με τον  $D_R(s)$  να είναι αντιστρέψιμος για σχεδόν όλες τις τιμές του  $s \in \mathbb{C}$ . Ορίζουμε τον block πίνακα  $F(s) := \begin{bmatrix} D_R^\top(s), N_R^\top(s) \end{bmatrix}^\top \in \mathbb{R}[s]^{(m+p) \times m}$  και συμβολίζουμε με  $k_i = \deg_{s_i} F(s)$ ,  $i \in \mathbf{m}$  τους βαθμούς των στηλών του  $F(s)$ . Ακολούθως, ορίζουμε για  $k \geq 1$ , τον  $(m+p)k \times m$  πολυωνυμικό πίνακα  $X_k(s)$  ως εξής:

$$X_k(s) := S_k(s) \begin{bmatrix} D_R(s) \\ N_R(s) \end{bmatrix} = \underbrace{\begin{bmatrix} I_{m+p} \\ sI_{m+p} \\ \vdots \\ s^{k-1}I_{m+p} \end{bmatrix}}_{(m+p)k \times (m+p)} \underbrace{\begin{bmatrix} D_R(s) \\ N_R(s) \end{bmatrix}}_{(m+p) \times m} = \underbrace{\begin{bmatrix} F(s) \\ sF(s) \\ \vdots \\ s^{k-1}F(s) \end{bmatrix}}_{(m+p)k \times m} \quad (1.1)$$

Ο  $X_k(s)$  μπορεί να γραφτεί με τον παρακάτω τρόπο:

$$X_k(s) = M_k^{F(s)} \underbrace{\text{block diag}_{i \in \mathbf{m}} \left\{ \begin{bmatrix} 1 \\ s \\ \vdots \\ s^{k_i+k-1} \end{bmatrix} \right\}}_{\left( \sum_{i=1}^m k_i + mk \right) \times m} =: M_k^{F(s)} S_k^{F(s)}(s) \quad (1.2)$$

όπου ο πίνακας  $M_k^{F(s)} \in \mathbb{R}^{(m+p)k \times (mk + \sum_{i=1}^m k_i)}$  ονομάζεται γενικευμένη Wolovich resultant του πίνακα  $F(s)$ , τάξης  $k$ .

Χρήσιμη, για τη συνέχεια, είναι η περιγραφή του αριστερού πυρήνα (left kernel) του πίνακα  $M_k^{F(s)}$  ο οποίος ορίζεται:

$$\text{Ker}\left(M_k^{F(s)}\right)^\top = \left\{ x^\top \in \mathbb{R}^{1 \times (m+p)k} : x^\top M_k^{F(s)} = \mathbf{0}_{1 \times (mk + \sum_{i=1}^m k_i)} \right\} \quad (1.3)$$

Το επόμενο θεώρημα προσδιορίζει τη διάσταση του  $\text{Ker}\left(M_k^{F(s)}\right)^\top$ .

**Θεώρημα 1.3.1 (Antoniou & Vardoulakis, 2005)** Έστω ζεύγος πολυωνυμικών πινάκων  $N_R(s) \in \mathbb{R}[s]^{p \times m}$ ,  $D_R(s) \in \mathbb{R}[s]^{m \times m}$ , ώστε να ισχύει  $\text{rank}_{\mathbb{R}(s)} \left[ D_R^\top(s), N_R^\top(s) \right]^\top = m$ . Αν είναι  $\mu_i, i \in \mathbf{p}$  οι (αναλλοίωτοι) ελάχιστοι δυναμικοί δείκτες γραμμών (invariant row minimal dynamical indices) του ρητού πίνακα  $P(s) = N_R(s)D_R^{-1}(s) \in \mathbb{R}(s)^{p \times m}$  και  $M_k^{\left[ D_R^\top(s), N_R^\top(s) \right]^\top} \in \mathbb{R}^{(m+p)k \times (mk + \sum_{i=1}^m k_i)}$  η γενικευμένη Wolovich resultant τάξης  $k$  για τον πίνακα  $\left[ D_R^\top(s), N_R^\top(s) \right]^\top$ , τότε ισχύει:

$$\dim_{\mathbb{R}} \ker M_k^{\left[ D_R^\top(s), N_R^\top(s) \right]^\top} = \sum_{i: \mu_i \leq k} (k - \mu_i).$$

**Πρόταση 1.3.2 (Antonίου & Vardulakis, 2005)** Υπό τις προϋποθέσεις του Θεωρήματος 1.3.1, ισχύει

$$\text{rank } M_k^{\left[ D_R^\top(s), N_R^\top(s) \right]^\top} = (p+m)k - \sum_{i: \mu_i \leq k} (k - \mu_i).$$

Επιπροσθέτως, αν  $k \geq \mu$ ,  $\mu = \max_{i \in \mathbf{p}} \{\mu_i\}$  τότε ισχύει

$$\text{rank } M_k^{\left[ D_R^\top(s), N_R^\top(s) \right]^\top} = mk + \delta_M P(s).$$

**Πρόταση 1.3.3 (Antonίου & Vardulakis, 2005)** Έστω ζεύγος πολυωνυμικών πινάκων  $N_R(s) \in \mathbb{R}[s]^{p \times m}$ ,  $D_R(s) \in \mathbb{R}[s]^{m \times m}$  και ο  $F(s) = \left[ D_R^\top(s), N_R^\top(s) \right]^\top$  είναι κανονικός ως προς τις στήλες του, με βαθμούς στηλών  $\deg_{ci} F(s) = k_i$ ,  $i \in \mathbf{m}$ . Τότε οι  $N_R(s)$ ,  $D_R(s)$  είναι δεξιά πρώτοι μεταξύ τους στο  $\mathbb{C}$  αν και μόνο αν η γενικευμένη Wolovich resultant  $M_k^{F(s)}$  έχει πλήρη τάξη στηλών για  $k \geq \mu$ . Ισοδύναμα, αν ο  $F(s) = \left[ D_R^\top(s), N_R^\top(s) \right]^\top$  είναι κανονικός ως προς τις στήλες του, τότε οι  $N_R(s)$ ,  $D_R(s)$  είναι δεξιά πρώτοι μεταξύ τους, αν και μόνο αν, για  $k \geq \mu$ ,  $\text{rank } M_k^{F(s)} = mk + \delta_M F(s)$ .



## ΚΕΦΑΛΑΙΟ 2

### ΑΛΓΟΡΙΘΜΟΣ ΥΠΟΛΟΓΙΣΜΟΥ ΟΙΚΟΓΕΝΕΙΑΣ ΣΤΑΘΕΡΟΠΟΙΗΤΙΚΩΝ ΑΝΤΙΣΤΑΘΜΙΣΤΩΝ

#### 2.1 Εφαρμογή των Διοφαντικών εξισώσεων πολυωνυμικών πινάκων

Θεωρούμε ένα γραμμικό, πολυμεταβλητό σύστημα που περιγράφεται από έναν πίνακα αυστηρώς κανονικών συναρτήσεων μεταφοράς (strictly proper, linear, multivariable plant),  $P(s) \in \mathbb{R}_{po}(s)^{p \times m}$  με  $m$  εισόδους και  $p$  εξόδους. Έστω

$$P(s) = N_R(s)D_R(s)^{-1} = D_L(s)^{-1}N_L(s) \quad (2.1)$$

η πρώτη δεξιά κλασματική (right coprime MFD) και πρώτη αριστερή κλασματική περιγραφή (left coprime MFD) του  $P(s)$ , αντιστοίχως, με τους  $N_R(s) \in \mathbb{R}[s]^{p \times m}$ ,  $D_R(s) \in \mathbb{R}[s]^{m \times m}$  κανονικό ως προς τις στήλες (column proper) με βαθμούς στηλών  $\deg_{ci} D_R(s) = k_i, i \in \mathbf{m}$  και  $N_L(s) \in \mathbb{R}[s]^{p \times m}$  και  $D_L[s] \in \mathbb{R}[s]^{p \times p}$  κανονικό ως προς τις γραμμές (row proper) με βαθμούς γραμμών  $\deg_{ri} D_L(s) = \mu_i, i \in \mathbf{p}$ . Ορίζουμε  $\mu := \max_{i \in \mathbf{p}} \{\mu_i\}$  (ο δείκτης παρατηρησιμότητας – observability index του  $P(s)$ ).

Το πρόβλημα της ανάθεσης του παρανομαστή του συστήματος κλειστού βρόγχου (closed-loop system) που προκύπτει αν στο αρχικό σύστημα προσθέσουμε δυναμικό αντισταθμιστή (dynamic precompensator) μέσω μοναδιαίας ανάδρασης (unity feedback), ανάγεται στο πρόβλημα της επίλυσης της Διοφαντικής εξίσωσης πολυωνυμικών πινάκων της μορφής:

$$X_L(s)D_R(s) + Y_L(s)N_R(s) = D_C(s) \quad (2.2)$$

όπου  $D_C(s) \in \mathbb{R}[s]^{m \times m}$  είναι ο επιθυμητός πίνακας παρανομαστής του κλειστού συστήματος. Οι πίνακες  $X_L(s) \in \mathbb{R}[s]^{m \times m}$ ,  $Y_L(s) \in \mathbb{R}[s]^{m \times p}$  αποτελούν μία – όχι απαραίτητα πρώτη – αριστερή κλασματική περιγραφή (left MFD) του αντισταθμιστή  $C(s)$ , ώστε:

$$C(s) = X_L(s)^{-1}Y_L(s) \in \mathbb{R}(s)^{m \times p}. \quad (2.3)$$

Είναι γνωστό ότι η (2.2) έχει λύση για αυθαίρετο  $D_C(s)$  αν και μόνο αν οι πίνακες  $N_R(s), D_R(s)$  είναι από δεξιά πρώτοι μεταξύ τους (right coprime). Επιπροσθέτως, αν μία συγκεκριμένη λύση της (2.2) είναι η  $\bar{X}_L(s), \bar{Y}_L(s)$ , τότε και κάθε ζεύγος της μορφής  $X_L(s) = \bar{X}_L(s) + T(s)N_L(s)$ ,  $Y_L(s) = \bar{Y}_L(s) - T(s)D_L(s)$ , για τυχόντα πολυωνυμικό πίνακα  $T(s) \in \mathbb{R}[s]^{m \times p}$ , αποτελεί επίσης λύση της (2.2).

Στη συγκεκριμένη περίπτωση, αναζητούνται και οι συνθήκες εκείνες που πρέπει να πληροί η (2.2) ώστε τα ζεύγη πολυωνυμικών πινάκων  $X_L(s), Y_L(s)$  που προκύπτουν να οδηγούν στη σύνθεση κανονικών (proper) αντισταθμιστών, δηλαδή να είναι τέτοια ώστε  $C(s) = X_L(s)^{-1}Y_L(s) \in \mathbb{R}_{pr}(s)^{m \times p}$ . Όπως θα δούμε στη συνέχεια, οι περιορισμοί που προκύπτουν αφορούν στην επιλογή των δυνάμεων των γραμμών (row powers) του πίνακα παρανομαστή του κλειστού συστήματος  $D_C(s)$ . Στον αλγόριθμο που χρησιμοποιείται, ο επιθυμητός παρανομαστής  $D_C(s)$  επιλέγεται να είναι κανονικός ως προς τις γραμμές και τις στήλες του (row – column reduced) με σκοπό να απλουστευθεί ο έλεγχος των βαθμών του πίνακα αριθμητή και πίνακα παρανομαστή του αντισταθμιστή  $C(s)$ .

## 2.2 Ανάλυση αλγορίθμου εύρεσης των λύσεων της Διοφαντικής εξίσωσης για τις οποίες προκύπτουν κανονικοί αντισταθμιστές

Στην ανάλυση που ακολουθεί, η Διοφαντική εξίσωση (2.2) οδηγεί στη διατύπωση συστημάτων αριθμητικών εξισώσεων των οποίων η επίλυση μας επιτρέπει τον υπολογισμό ολόκληρης της παραμετρικής οικογένειας των αντισταθμιστών εκείνων που οδηγούν σε κλειστό σύστημα με τον επιθυμητό πίνακα παρανομαστή.

Ας είναι  $X_L(s), Y_L(s)$  μία λύση της (2.2) για έναν συγκεκριμένο επιθυμητό πίνακα παρανομαστή του κλειστού συστήματος  $D_C(s)$ . Έστω  $k-1$  ο μέγιστος βαθμός του  $s$  που προκύπτει μεταξύ των στοιχείων του πίνακα  $\Omega(s) := [X_L(s), Y_L(s)] \in \mathbb{R}[s]^{m \times (m+p)}$ . Έτσι, μπορούμε να γράψουμε:

$$\Omega(s) = \bar{\Omega}_k S_k(s) \quad (2.4)$$

όπου  $\bar{\Omega}_k \in \mathbb{R}^{m \times k(p+m)}$  και ο  $S_k(s)$  ορίζεται όπως στην (1.1).

Η (2.2) μπορεί να γραφτεί

$$\Omega(s)F(s) = D_C(s) \quad (2.5)$$

όπου  $F(s) = [D_R^\top(s), N_R^\top(s)]^\top \in \mathbb{R}[s]^{(m+p) \times m}$ . Η (2.5) από την (2.4) γράφεται:

$$\bar{\Omega}_k S_k(s)F(s) = D_C(s) \quad (2.6)$$

Η τελευταία σχέση σε συνδυασμό με τις (1.1), (1.2) συνεπάγεται:

$$\bar{\Omega}_k M_k^{F(s)} S_k^{F(s)}(s) = D_C(s) \quad (2.7)$$

Παρατηρώντας την (2.7) και συγκρίνοντας τους βαθμούς του  $s$  και στα δύο της μέλη, εύκολα διαπιστώνεται ότι  $\deg_{ci} D_C(s) \leq k_i + k - 1$ ,  $i \in \mathbf{m}$  και επομένως ο  $D_C(s)$  μπορεί να

γραφτεί  $D_C(s) = \bar{D}_k S_k^{F(s)}$ ,  $\bar{D}_k \in \mathbb{R}^{m \times (\sum_{i=1}^m k_i + mk)}$  και η (2.7) γίνεται:

$$\bar{\Omega}_k M_k^{F(s)} S_k^{F(s)}(s) = \bar{D}_k S_k^{F(s)} \quad (2.8)$$

Η (2.8) πρέπει να ισχύει για κάθε  $s \in \mathbb{C}$  και επομένως η επόμενη εξίσωση προκύπτει ισοδύναμα:

$$\bar{\Omega}_k M_k^{F(s)}(s) = \bar{D}_k \quad (2.9)$$

Δεδομένου, λοιπόν, του μέγιστου βαθμού του  $\Omega(s)$  και επιλέγοντας καταλλήλως το  $k$ , είναι δυνατό να υπολογίσουμε κάθε λύση της (2.2) από ένα σύστημα αριθμητικών εξισώσεων της μορφής (2.9). Στη συνέχεια καθορίζεται η συνθήκη εκείνη που επιτρέπει την επίλυση της (2.2) ώστε οι προκύπτουσες λύσεις να οδηγούν σε κανονικούς αντισταθμιστές.

**Λήμμα 2.2.1 (Kucera & Zagalak, 1999)** Θεωρούμε τη Διοφαντική εξίσωση πολυωνυμικών πινάκων της μορφής (2.2) υπό τις εξής προϋποθέσεις:

1. Ο  $D_R(s)$  είναι κανονικός ως προς τις στήλες του, με βαθμούς στηλών  $k_i = \deg_{ci} D_R(s)$ ,  $i \in \mathbf{m}$ ,
2. Οι  $D_R(s), N_R(s)$  είναι από δεξιά πρώτοι μεταξύ τους,
3. Το  $P(s) = N_R(s)D_R^{-1}(s) = D_L^{-1}(s)N_L(s)$  είναι αυστηρώς κανονικό,
4. Οι  $N_L(s), D_L(s)$  είναι από αριστερά πρώτοι μεταξύ τους,

5. Ο  $D_L(s)$  είναι κανονικός ως προς τις γραμμές του , με βαθμούς γραμμών  $\mu_i = \deg_{ri} D_L(s), i \in \mathbf{p}$  και ορίζουμε  $\mu := \max_{i \in \mathbf{p}} \{\mu_i\}$ ,
6. Ο  $D_C(s)$  είναι ταυτοχρόνως κανονικός ως προς τις στήλες και τις γραμμές του με  $\deg_{ci} D_C(s) = \deg_{ri} D_C(s) = k_i + \xi_i, i \in \mathbf{m}$ , όπου  $\xi_i$  ακέραιοι τέτοιοι ώστε  $\xi_i \geq \mu - 1, i \in \mathbf{m}$ .

Αν  $X_L(s), Y_L(s)$  είναι μία λύση της (2.2) και  $C(s) = X_L^{-1}(s)Y_L(s) \in \mathbb{R}_{pr}(s)$  τότε ο  $X_L(s)$  είναι κανονικός ως προς τις γραμμές του, με βαθμούς γραμμών  $\deg_{ri} X_L(s) = \xi_i, i \in \mathbf{m}$ .

Λαμβάνοντας υπ' όψιν ότι οι βαθμοί των γραμμών του  $Y_L(s)$  δεν μπορεί να υπερβαίνουν τους βαθμούς των αντίστοιχων γραμμών του  $X_L(s)$ , δηλαδή ότι  $\deg_{ri} Y_L(s) \leq \xi_i, i \in \mathbf{m}$ , αφού  $X_L^{-1}(s)Y_L(s) \in \mathbb{R}_{pr}^{m \times p}(s)$  (Vardulakis, 1991; Kailath, 1980; Callier & Desoer, 1982), η  $i$ -οστή γραμμή του πίνακα  $\Omega(s)$  θα έχει βαθμό  $\xi_i$ . Στη συνέχεια συμβολίζουμε τις γραμμές του  $\Omega(s)$  με  $\omega_i^\top(s) \in \mathbb{R}[s]^{1 \times (m+p)}, i \in \mathbf{m}$ , οι οποίες γράφονται αναλυτικά:

$$\omega_i^\top(s) = \sum_{j=0}^{\xi_i} \omega_{ij}^\top s^j, \quad \omega_{ij}^\top \in \mathbb{R}^{1 \times (m+p)}, j \in \mathbf{m} \quad (2.10)$$

Ορίζουμε τα διανύσματα γραμμές  $\bar{\omega}_i^\top = [\omega_{i0}^\top, \omega_{i1}^\top, \dots, \omega_{i\xi_i}^\top] \in \mathbb{R}^{1 \times (p+m)(\xi_i+1)}, i \in \mathbf{m}$ .

Ας είναι τώρα  $d_i^\top(s), i \in \mathbf{m}$  οι γραμμές του  $D_C(s)$ . Χρησιμοποιώντας την υπόθεση 6 του Λήμματος 2.2.1, ορίζονται διανύσματα γραμμές  $\bar{d}_i^\top \in \mathbb{R}^{1 \times m(\xi_i+1) + \sum_{j=1}^m k_j}, i \in \mathbf{m}$  που προκύπτουν από την ισότητα:

$$d_i^\top(s) = \bar{d}_i^\top S_{\xi_i+1}^{F(s)}(s), i \in \mathbf{m} \quad (2.11)$$

όπου  $S_{\xi_i+1}^{F(s)}(s)$  είναι ο πίνακας διαστάσεων  $(m(\xi_i+1) + \sum_{j=1}^m k_j) \times m$  ο οποίος ορίζεται (για  $k = \xi_i + 1$ ) στην (1.2).

**Θεώρημα 2.2.2 (Antoniu & Vardulakis, 2005)** Δεδομένων των συνθηκών 1 – 6 του Λήμματος 2.2.1, κάθε ζεύγος πινάκων  $X_L(s), Y_L(s)$  που αποτελεί λύση της Διοφαντικής

Εξίσωσης (2.2) τέτοια ώστε  $X_L^{-1}(s)Y_L(s) = C(s) \in \mathbb{R}_{pr}^{m \times p}(s)$  μπορεί να υπολογιστεί από μία λύση των συστημάτων αριθμητικών εξισώσεων

$$\bar{\omega}_i^\top M_{\xi_i+1}^{F(s)} = \bar{d}_i^\top, \quad i \in \mathbf{m} \quad (2.12)$$

και αντιστρόφως, δηλαδή, από κάθε λύση  $\bar{\omega}_i^\top, i \in \mathbf{m}$  των εξισώσεων (2.12) προκύπτει και ένα ζεύγος  $X_L(s), Y_L(s)$  λύση της Διοφαντικής εξίσωσης (2.2), τέτοιο ώστε  $X_L^{-1}(s)Y_L(s) = C(s) \in \mathbb{R}_{pr}^{m \times p}(s)$ .

Το προηγούμενο θεώρημα εξασφαλίζει την «ένα προς ένα» αντιστοιχία κανονικών αντισταθμιστών με τις λύσεις των εξισώσεων (2.12), όταν πληρούνται οι 1 – 6 προϋποθέσεις του Λήμματος 2.2.1. Συνεπώς, συνοψίζοντας την έως αυτό το σημείο ανάλυση, μπορούμε να πούμε ότι η επίλυση των εξισώσεων (2.12) μας οδηγεί σε όλους εκείνους τους αντισταθμιστές  $C(s)$  οι οποίοι σε διάταξη *unity feedback* με ένα δεδομένο αυστηρώς κανονικό πίνακα συναρτήσεων μεταφοράς  $P(s) = N_R(s)D_R^{-1}(s)$  με  $D_R(s), N_R(s)$  από δεξιά πρώτους μεταξύ τους,  $k_i = \deg_{ci} D_R(s), i \in \mathbf{m}$ , και δείκτη παρατηρησιμότητας  $\mu$ , δημιουργούν σύστημα κλειστού βρόγχου  $\Sigma(P, C)$  του οποίου ο πίνακας παρονομαστής  $D_C(s)$  ορίζεται με μόνους περιορισμούς (α) την κανονικότητα ως προς τις γραμμές και τις στήλες του, (β) οι βαθμοί των στηλών και των γραμμών του να ικανοποιούν την  $\deg_{ci} D_C(s) = \deg_{ri} D_C(s) = k_i + \xi_i, i \in \mathbf{m}$  με τα  $\xi_i$  ακέραιους επιλεγμένους έτσι ώστε όλοι να είναι μεγαλύτεροι από  $\mu - 1$ .

Το παρακάτω θεώρημα προσδιορίζει το πλήθος των απαιτούμενων ελεύθερων παραμέτρων για την παραμετροποίηση της οικογένειας των ζητούμενων αντισταθμιστών.

**Πρόταση 2.2.3 (Antonίου & Vardoulakis, 2005)** Αν ισχύουν οι υποθέσεις 1-6 του Λήμματος 2.2.1, τότε, το πλήθος των ελεύθερων παραμέτρων στην παραμετροποιημένη έκφραση της οικογένειας των αντισταθμιστών που προκύπτει από την επίλυση των (2.12) ισούται με

$$\nu = m(p - \delta_M P(s)) + p \sum_{i=1}^m \xi_i.$$

Η διαδικασία επίλυσης των συστημάτων (2.12) ενδέχεται να απαιτεί τον υπολογισμό πολλών διαφορετικών τάξεων της γενικευμένης Wolovich resultant (όταν τα  $\xi_i$  έχουν

πολλές διαφορετικές τιμές). Στη συνέχεια υποδεικνύεται πως μπορεί να αξιοποιηθεί η ανηγμένη κλιμακωτή κατά στήλες μορφή (column echelon form) της γενικευμένης Wolovich resultant. Συμβολίζουμε με  $\{i_1, i_2, \dots, i_m\}$  την διάταξη των δεικτών, ώστε  $\xi_{i_1} \leq \xi_{i_2} \leq \dots \leq \xi_{i_m}$ . Επίσης, συμβολίζουμε  $\xi := \max_{i \in \mathbf{m}} \{\xi_i\}$ . Για να λύσουμε την (2.12) για  $i = i_1$  μπορούμε να εφαρμόσουμε την μέθοδο απαλοιφής του Gauss στις στήλες του πίνακα  $M_{\xi_{i_1}+1}^{F(s)}$ , οπότε θα προκύψει η ανηγμένη κλιμακωτή ως προς τις στήλες μορφή (reduced column echelon form)  $R_{\xi_{i_1}+1}^{F(s)}$ . Αφού η resultant έχει την ιδιότητα να διατηρεί τον αριστερό πυρήνα της αναλλοίωτο στους στοιχειώδεις μετασχηματισμούς στηλών, οι στήλες του  $M_{\xi_{i_1}+1}^{F(s)}$  εμφανίζονται στις πρώτες  $(p+m)\xi_{i_1}$  γραμμές του  $M_{\xi_{i_2}+1}^{F(s)}$  (περιέχονται μαζί με  $m(\xi_{i_2} - \xi_{i_1})$  μηδενικές στήλες). Εφόσον η  $M_{\xi_{i_1}+1}^{F(s)}$  έχει πλήρη τάξη στηλών, η ανηγμένη κλιμακωτή ως προς τις στήλες μορφή της  $M_{\xi_{i_2}+1}^{F(s)}$  θα έχει την εξής block τριγωνική μορφή

$$R_{\xi_{i_2}+1}^{F(s)} = \begin{bmatrix} R_{\xi_{i_1}+1}^{F(s)} & \mathbf{0} \\ \mathcal{Q}_{11} & \mathcal{Q}_{12} \end{bmatrix}$$

Συνεχίζοντας επαγωγικά, η ανηγμένη κλιμακωτή ως προς τις στήλες μορφή της  $M_{\xi_{i_{j+1}}+1}^{F(s)}$  θα έχει την όμοια block τριγωνική μορφή

$$R_{\xi_{i_{j+1}}+1}^{F(s)} = \begin{bmatrix} R_{\xi_{i_j}+1}^{F(s)} & \mathbf{0} \\ \mathcal{Q}_{j1} & \mathcal{Q}_{j2} \end{bmatrix}$$

για  $j = 1, 2, \dots, m-1$ . Άρα, από την  $M_{\xi+1}^{F(s)}$  και τον υπολογισμό της ανηγμένης κλιμακωτής ως προς τις στήλες μορφής της,  $R_{\xi+1}^{F(s)}$ , προκύπτουν οι λύσεις των (2.12) για κάθε  $i \in \mathbf{m}$ , χωρίς να είναι αναγκαίος ο υπολογισμός και ο μετασχηματισμός των στηλών των resultants μικρότερης τάξης, επιλέγοντας σε κάθε περίπτωση κατάλληλο αριθμό γραμμών και στηλών της  $R_{\xi+1}^{F(s)}$ .

## 2.3 Σύνθεση αλγορίθμου

Η ανάλυση, στην προηγούμενη παράγραφο, της μεθόδου εύρεσης των κανονικών αντισταθμιστών που οδηγούν σε κλειστά συστήματα με επιθυμητό πίνακα παρονομαστή, οδηγεί στην σύνθεση του παρακάτω αλγορίθμου:

- **Βήμα 1.** Για δεδομένο πίνακα αυστηρώς κανονικών ρητών συναρτήσεων  $P(s) \in \mathbb{R}_{po}(s)^{p \times m}$ , υπολογίζουμε μία δεξιά πρώτη κλασματική περιγραφή του (right coprime MFD)  $N_R(s) \in \mathbb{R}[s]^{p \times m}$ ,  $D_R(s) \in \mathbb{R}[s]^{m \times m}$ , με τον  $D_R(s)$  κανονικό ως προς τις στήλες του. Θέτουμε  $k_i = \deg_{ci} D_R(s)$ ,  $i \in \mathbf{m}$ .
- **Βήμα 2.** Προσδιορίζουμε τον ελάχιστο φυσικό αριθμό  $k$  για τον οποίο η  $k$  τάξης γενικευμένη Wolovich resultant  $M_k^{F(s)}$ , όπου  $F(s) := [D_R^\top(s), N_R^\top(s)]^\top \in \mathbb{R}[s]^{(m+p) \times m}$ , έχει πλήρη τάξη στηλών. Θέτουμε  $\mu = k$ .
- **Βήμα 3.** Επιλέγουμε  $\xi_i \geq \mu - 1$ ,  $i \in \mathbf{m}$ .
- **Βήμα 4.** Για  $\xi := \max_{i \in \mathbf{m}} \{\xi_i\}$  υπολογίζουμε την γενικευμένη Wolovich resultant  $M_{\xi+1}^{F(s)}$  τάξης  $\xi + 1$  από την (1.2).
- **Βήμα 5.** Ορίζουμε τον επιθυμητό πίνακα παρονομαστή  $D_C(s) \in \mathbb{R}[s]^{m \times m}$  ώστε να είναι κανονικός και ως προς τις γραμμές και ως προς τις στήλες του, με βαθμούς γραμμών και στηλών  $\deg_{ci} D_C(s) = \deg_{ri} D_C(s) = k_i + \xi_i$ ,  $i \in \mathbf{m}$  (τα παραπάνω υλοποιούνται εύκολα αν επιλέξουμε τον  $D_C(s)$  διαγώνιο). Δημιουργούμε τον  $\bar{D}_{\xi+1} \in \mathbb{R}^{m \times (\sum_{i=1}^m k_i + m(\xi+1))}$  γράφοντας τον  $D_C(s) = \bar{D}_{\xi+1} S_{\xi+1}^{F(s)}$  με τρόπο ανάλογο της (2.8).
- **Βήμα 6.** Συνθέτουμε τον block πίνακα  $\bar{M}_{\xi+1}^{F(s)} = \begin{bmatrix} M_{\xi+1}^{F(s)} \\ \bar{D}_{\xi+1} \end{bmatrix}$ .
- **Βήμα 7.** Υπολογίζουμε την ανηγμένη κλιμακωτή ως προς τις στήλες μορφή (reduced column echelon form)  $\bar{R}_{\xi+1}^{F(s)}$  του  $\bar{M}_{\xi+1}^{F(s)}$ , ώστε  $\bar{R}_{\xi+1}^{F(s)} = \begin{bmatrix} R_{\xi+1}^{F(s)} \\ \Delta_{\xi+1} \end{bmatrix}$ .
- **Βήμα 8.** Για κάθε  $i \in \mathbf{m}$ , κατασκευάζουμε τον πίνακα  $R_{\xi_i+1}^{F(s)}$  από τις πρώτες  $(\xi_i + 1)(m + p)$  γραμμές του  $R_{\xi+1}^{F(s)}$  και τον πίνακα - γραμμή  $\Delta_{\xi_i+1}$  από την  $i$ -οστή

γραμμή του  $\Delta_{\xi+1}$ , αγνοώντας τις τελευταίες  $(\xi - \xi_i)m$  μηδενικές στήλες και στους δύο πίνακες.

- **Βήμα 9.** Για κάθε  $i \in \mathbf{m}$ , υπολογίζουμε την γενική λύση των  $\bar{\omega}_i^\top R_{\xi_i+1}^{F(s)} = \Delta_{\xi_i+1}$ .
- **Βήμα 10.** Από τα  $\bar{\omega}_i^\top$  και την (2.10) προκύπτουν οι γραμμές  $\omega_i^\top(s) \in \mathbb{R}[s]^{1 \times (m+p)}$ ,  $i \in \mathbf{m}$  του  $\Omega(s)$

## 2.4 Παράδειγμα

Έστω

$$P(s) = \begin{bmatrix} \frac{s+1}{s(s-2)} & 0 \\ \frac{1}{s(s-1)} & \frac{1}{s-1} \end{bmatrix}$$

με  $D_R(s) = \begin{bmatrix} s^2 - 2s & 0 \\ 1 & s-1 \end{bmatrix}$ ,  $N_R(s) = \begin{bmatrix} s+1 & 0 \\ 1 & 1 \end{bmatrix}$

ώστε  $k_1=2, k_2=1$  και ο βαθμός McMillan είναι  $\delta_M P(s) = k_1 + k_2 = 3$ . Ο δείκτης παρατηρησιμότητας του  $P(s)$  είναι  $\mu=2$ . Αυτό, γιατί υπολογίζοντας διαδοχικά τις γενικευμένες Wolovich resultants του  $F(s) = [D_R^\top(s), N_R^\top(s)]^\top \in \mathbb{R}[s]^{4 \times 2}$  τάξης 1 και 2 διαπιστώνουμε ότι η resultant τάξης 1 δεν έχει πλήρη τάξη στηλών, ενώ η resultant τάξης 2 έχει πλήρη τάξη στηλών:

$$M_1^{F(s)} = \begin{bmatrix} 0 & -2 & 1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \end{bmatrix} \in \mathbb{R}^{4 \times 5} \Rightarrow \text{rank } M_1^{F(s)} = 4$$

$$M_2^{F(s)} = \begin{bmatrix} 0 & -2 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -2 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \in \mathbb{R}^{8 \times 7} \Rightarrow \text{rank } M_2^{F(s)} = 7$$



Ας είναι

$$D_C(s) = \begin{bmatrix} s^3 + 8s^2 + 24s + 32 & 0 \\ 0 & s^3 + 15s^2 + 62s + 48 \end{bmatrix}$$

ο επιθυμητός πίνακας παρονομαστής του κλειστού συστήματος, έχοντας επιλέξει  $\xi_1 = 1 \geq \mu - 1 = 1$ ,  $\xi_2 = 2 \geq \mu - 1 = 1$ . Περιμένουμε ότι ο υπολογισμός της έκφρασης της οικογένειας όλων των κανονικών αντισταθμιστών που στο κλειστό σύστημα αναθέτουν τον επιθυμητό παρονομαστή  $D_C(s)$  θα περιέχει

$m(p - \delta_M P(s)) + p \sum_{i=1}^m \xi_i = 2(2 - 3) + 2(1 + 2) = 4$  ελεύθερες παραμέτρους. Στη συνέχεια

δημιουργούμε την γενικευμένη Wolovich resultant τάξης  $k = \max_{i=1,2} \{\xi_i\} + 1 = 3$

$$M_3^{F(s)} = \begin{bmatrix} 0 & -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \in \mathbb{R}^{12 \times 9}$$

Γράφουμε τον  $D_C(s)$  με βάση τους συντελεστές του ως εξής:

$$D_C(s) = \bar{D}_3 S_3^{F(s)}$$

$$= \begin{bmatrix} 32 & 24 & 8 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 48 & 62 & 15 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ s & 0 \\ s^2 & 0 \\ s^3 & 0 \\ s^4 & 0 \\ 0 & 1 \\ 0 & s \\ 0 & s^2 \\ 0 & s^3 \end{bmatrix}$$

Στη συνέχεια ορίζουμε τον επαυξημένο πίνακα  $\bar{M}_3^{F(s)} = \begin{bmatrix} M_3^{F(s)} \\ \bar{D}_3 \end{bmatrix}$  και με τη Gauss μέθοδο

απαλοιφής στις στήλες του, προκύπτει η ανηγμένη ως προς τις στήλες μορφή:

$$\bar{R}_3^{F(s)} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ -1 & 1 & -2 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 2 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 & 1 & 1 & 0 & 0 \\ 6 & 0 & 32 & 0 & 1 & 0 & 4 & 0 & 0 \\ -63 & 78 & -204 & 126 & 0 & 16 & 62 & 0 & 1 \end{bmatrix} = \begin{bmatrix} R_3^{F(s)} \\ \Delta_3 \end{bmatrix}$$

όπου  $R_3^{F(s)} \in \mathbb{R}^{12 \times 9}$  και  $\Delta_3 \in \mathbb{R}^{2 \times 9}$ .

Για τον προσδιορισμό του  $\omega_1^\top(s)$  χρειάζονται οι πρώτες  $(p+m)(\xi_1+1) = 8$  γραμμές του  $\bar{R}_3^{F(s)}$  καθώς και η πρώτη γραμμή του  $\Delta_3$ , ενώ παραλείπουμε τις δύο τελευταίες στήλες και στους δύο πίνακες. Έτσι προκύπτει η ισοδύναμη, ανηγμένη μορφή των εξισώσεων (2.12) για  $i=1$ . Δηλαδή:

$$\bar{\omega}_1^\top \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ -1 & 1 & -2 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} = [6 \ 0 \ 32 \ 0 \ 1 \ 0 \ 4]$$

η γενική λύση της οποίας είναι

$$\bar{\omega}_1^\top = [6 \ 0 \ 32 \ 0 \ 1 \ 0 \ 4 \ 0] + t_1 [1 \ -1 \ 2 \ -1 \ 0 \ 0 \ -1 \ 1], \quad t_1 \in \mathbb{R}$$

και από την (2.10) έχουμε

$$\omega_1^\top(s) = [s + 6 + t_1 \quad -t_1 \quad (4 - t_1)s \quad t_1 s - t_1].$$

Με ανάλογο τρόπο, για να προσδιορίσουμε το  $\omega_2^\top(s)$ , χρειάζονται οι  $(p+m)(\xi_2+1)=12$  γραμμές του  $\bar{R}_3^{F(s)}$  καθώς και η δεύτερη γραμμή του  $\Delta_3$ . Έτσι προκύπτει η ισοδύναμη, ανηγμένη μορφή των εξισώσεων (2.12) για  $i=2$ . Δηλαδή:

$$\bar{\omega}_2^\top \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ -1 & 1 & -2 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 2 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 & 1 & 1 & 0 & 0 \end{bmatrix} = [-63 \ 78 \ -204 \ 126 \ 0 \ 16 \ 62 \ 0 \ 1]$$

η γενική λύση της οποίας είναι

$$\bar{\omega}_2^\top = [-63 \ 78 \ -204 \ 126 \ 0 \ 16 \ 62 \ 0 \ 0 \ 1 \ 0 \ 0] + [t_2 \ t_3 \ t_4] \begin{bmatrix} 1 & -1 & 2 & -1 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & -1 & 0 & -2 & 0 & 0 & 0 & 1 & 0 \\ 0 & -1 & 2 & -1 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

όπου  $t_2, t_3, t_4 \in \mathbb{R}$ . Έτσι από την (2.10) έχουμε

$$\bar{\omega}_2^\top(s) = [-t_3s + t_2 - t_3 - 63 \quad s^2 + (16 - t_4)s - t_2 - t_4 + 78 \\ t_3s^2 + (62 - t_2 - 2t_3 - t_4)s + 2t_2 + 2t_4 - 204 \quad t_4s^2 + t_2s - t_2 - t_4 - 126]$$

Τέλος, αφού  $\mathcal{Q}(s) = [X_L(s), Y_L(s)] = [\omega_1(s), \omega_2(s)]^\top$ , η παραμετροποίηση όλων των κανονικών αντισταθμιστών είναι η

$$X_L(s) = \begin{bmatrix} s + 6 + t_1 & -t_1 \\ -t_3s + t_2 - t_3 - 63 & s^2 + (16 - t_4)s - t_2 - t_4 + 78 \end{bmatrix} \\ Y_L(s) = \begin{bmatrix} (4 - t_1)s + 32 + 2t_1 & t_1s - t_1 \\ t_3s^2 + (62 - t_2 - 2t_3 - t_4)s + 2t_2 + 2t_4 - 204 & t_4s^2 + t_2s - t_2 - t_4 - 126 \end{bmatrix}$$

για ελεύθερες παραμέτρους  $t_1, t_2, t_3, t_4 \in \mathbb{R}$ . Προφανώς ο  $C(s) = X_L^{-1}(s)Y_L(s)$  είναι κανονικός για κάθε τετράδα πραγματικών τιμών των παραμέτρων  $t_1, t_2, t_3, t_4$ .

## ΚΕΦΑΛΑΙΟ 3

### ΥΛΟΠΟΙΗΣΗ ΑΛΓΟΡΙΘΜΟΥ ΚΑΙ ΕΦΑΡΜΟΓΗ ΣΤΟ MATHEMATICA

#### 3.1 Γενικά

Η δυνατότητα χρήσης H/Y για την διεκπεραίωση των υπολογισμών του αλγορίθμου της παραγράφου 2.3 μειώνει σε πολύ μεγάλο βαθμό τον χρόνο και την προσπάθεια που απαιτείται για την εφαρμογή του. Ταυτοχρόνως, μειώνεται σε καθοριστικό βαθμό η πιθανότητα εκτέλεσης λανθασμένων υπολογισμών που οφείλονται στον ανθρώπινο παράγοντα. Στο κεφάλαιο αυτό παρουσιάζεται η εφαρμογή – αντικείμενο της παρούσας διπλωματικής εργασίας που έχει ως σκοπό την υλοποίηση στον H/Y του αλγορίθμου στον οποίο αναφέρεται το προηγούμενο κεφάλαιο. Επιπρόσθετος στόχος της είναι να δημιουργεί ένα, παραθυρικού τύπου, γραφικό περιβάλλον μέσα από το οποίο διεκπεραιώνονται οι διαδικασίες εισαγωγής και ελέγχου της ορθότητας των δεδομένων, εκτελούνται όλοι οι απαιτούμενοι υπολογισμοί και παρουσιάζεται το αποτέλεσμα τους, με τέτοιο τρόπο ώστε να μην απαιτείται από τον χρήστη οποιαδήποτε γνώση χρήσης και σύνταξης συναρτήσεων της Mathematica.

Αν και η εφαρμογή, για λόγους ευκολίας στην εκτέλεση και τη χρήση της, αποτελεί ένα ενιαίο πρόγραμμα, ο προγραμματισμός της περιέχει ουσιαστικά ένα αυτόνομο τμήμα, την ομάδα των συναρτήσεων που επιτελούν όλους τους απαιτούμενους υπολογισμούς για τον εύρεση των ζητούμενων αντισταθμιστών. Στο υπόλοιπο του προγράμματος υλοποιείται το γραφικό περιβάλλον χρήσης. Αποτελείται, δηλαδή, από τις εκφράσεις των «παραθύρων» και των γραφικών στοιχείων ελέγχου που περιέχονται σε αυτά, τις συναρτήσεις που αυτά καλούν (callbacks) καθώς και τις μεταβλητές που απαιτούνται για τον έλεγχο της ροής της διαδικασίας μέσω του γραφικού περιβάλλοντος (π.χ. να μην επιτρέπεται στο χρήστη να ορίζει με λάθος σειρά τα δεδομένα του προβλήματος). Για παράδειγμα, το γραφικό στοιχείο (button) που χρησιμοποιεί ο χρήστης για να ορίσει τον πίνακα συναρτήσεων μεταφοράς του αρχικού συστήματος αποτελεί στοιχείο του προγραμματισμού του γραφικού περιβάλλοντος. Μέρος της λειτουργίας του είναι και η κλήση της συνάρτησης ‘ColumnProperDenominatorRightCoprimeMFD’ που δημιουργήθηκε για να υπολογίζει δεξιές πρώτες κλασματικές περιγραφές ρητών πινάκων, με κανονικούς ως προς τις στήλες παρονομαστές. Η συνάρτηση

‘ColumnProperDenominatorRightCoprimeMFD’ ανήκει στο αυτόνομο υπολογιστικό τμήμα του προγράμματος. Το τμήμα αυτό αποτελεί ουσιαστικά ένα πακέτο συναρτήσεων, που θα μπορούσε να χρησιμοποιεί ο χρήστης και εκτός του γραφικού περιβάλλοντος της εφαρμογής για να εκτελεί όλους τους απαιτούμενους υπολογισμούς, καλώντας τις συναρτήσεις που το αποτελούν από τη γραμμή εντολών του Mathematica. Θα μπορούσε επίσης να καλεί μεμονωμένες συναρτήσεις του πακέτου για να εκτελεί συγκεκριμένους υπολογισμούς που συναντώνται συχνά στη Θεωρία Ελέγχου γραμμικών πολυμεταβλητών συστημάτων. Στην παράγραφο 3.2 περιγράφονται οι σημαντικότερες συναρτήσεις τέτοιου είδους που δημιουργήθηκαν στα πλαίσια της υλοποίησης του κύριου αλγόριθμου υπολογισμού των αντισταθμιστών.

### 3.2 Σημαντικές επιμέρους συναρτήσεις και αλγόριθμοι

Το μεγαλύτερο μέρος των αλγορίθμων που χρησιμοποιούνται ως βάση για την πραγματοποίηση των παρακάτω επιμέρους συναρτήσεων, περιέχονται ή προκύπτουν από τη μελέτη του πρώτου κυρίως κεφαλαίου του (Vardulakis, 1991). Στον πίνακα 3.1 παρουσιάζονται συνοπτικά οι συναρτήσεις αυτές.

Όνομα Συνάρτησης	Περιγραφή
SmithForm	Υπολογισμός Smith μορφής πολυωνυμικού πίνακα
McMillanForm	Υπολογισμός Smith/McMillan μορφής πολυωνυμικού/ρητού πίνακα
LeftStructureMatrix (RightStructureMatrix)	Υπολογισμός αριστερού (δεξιού) δομικού πίνακα και του πίνακα μετασχηματισμού
PMLeftGCD (PMRightGCD)	Υπολογισμός αριστερού (δεξιού) ΜΚΔ πολυωνυμικών πινάκων
LeftMFD (RightMFD)	Υπολογισμός αριστερής (δεξιάς) κλασματικής περιγραφής ρητού πίνακα
LeftCoprimeMFD (RightCoprimeMFD)	Υπολογισμός πρώτης αριστερής (δεξιάς) κλασματικής περιγραφής ρητού πίνακα
MatrixReductionToRowProper (MatrixReductionToColumnProper)	Υπολογίζει τον πίνακα συντελεστή κατά βαθμούς γραμμών(στηλών) για πολυωνυμικό πίνακα
MatrixReductionToRowProper (MatrixReductionToColumnProper)	Επιστρέφει πίνακα μετασχηματισμένο σε κανονικό ως προς τις γραμμές (στήλες) μαζί με τον πίνακα μετασχηματισμού.
RowProperDenominatorLeftCoprimeMFD (ColumnProperDenominatorRightCoprimeMFD)	Υπολογισμός πρώτης αριστερής (δεξιάς) κλασματικής περιγραφής ρητού πίνακα με κανονικό ως προς τις γραμμές (στήλες) παρονομαστή
GWResultant	Υπολογίζει την $k$ -τάξης γενικευμένη Wolovich resultant πολυωνυμικού πίνακα

Πίνακας 3.1. Κυριότερες συναρτήσεις που ορίζονται στο «υπολογιστικό μέρος» του προγράμματος και παρουσιάζονται στην Κλασική Θεωρία Ελέγχου γραμμικών πολυμεταβλητών συστημάτων.

Ακολουθεί η αναλυτικότερη παρουσίαση τους. Σημειώνεται ότι ο κώδικας για τον προγραμματισμό τους παρατίθεται στο παράρτημα, το οποίο περιέχει τον πλήρη κώδικα της εφαρμογής.

- **Συνάρτηση *SmithForm***

**Χρήση – Σύνταξη** *SmithForm[A]*, επιστρέφει τη Smith μορφή του πολυωνυμικού πίνακα A.

**Αλγόριθμος**

**Βήμα 1:** Θέτω p τον αριθμό των γραμμών, m τον αριθμό των στηλών και r την τάξη του πίνακα A.

**Βήμα 2:** Θέτω  $m[0]=1$  και  $m[i]$  τον Μ.Κ.Δ. όλων των υποοριζουσών τάξης i του A για κάθε i από 1 έως r.

**Βήμα 3:** Υπολογίζω τα  $\varepsilon[i]=m[i]/m[i-1]$

**Βήμα 4:** Η Smith μορφή του A είναι ο πίνακας με p γραμμές και m στήλες και μηδενικά σε όλες τις θέσεις εκτός των r πρώτων στοιχείων της διαγωνίου του όπου περιέχει διαδοχικά τα στοιχεία  $\varepsilon[i]$ , με i από 1 έως r.

**Παράδειγμα**

α)

$A = \{(s+2)(s-1), s+9, 0\}, \{s(s-2), 2, 1\}\};$

**SmithForm[A]**

$\{(1, 0, 0), \{0, 1, 0\}\}$

**MatrixForm[%]**

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

β)

$A = \{(s+2)(s-1), s+9, 0\}, \{s(s-2), 2, 1\}, \{s-1, s-2, s(s-1)(s-3)\}, \{1, 0, (s+1)(s+2)\}\};$

**SmithForm[A]**

$\{(1, 0, 0), \{0, 1, 0\}, \{0, 0, 1\}, \{0, 0, 0\}\}$

**MatrixForm[%]**

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

- **Συνάρτηση *McMillanForm***

**Χρήση – Σύνταξη** *McMillanForm[A]* , επιστρέφει τη *McMillan* μορφή του ρητού (πάνω στα πολυώνυμα) πίνακα *A*.

**Αλγόριθμος**

**Βήμα 1:** Θέτω *d* το Ε.Κ.Π όλων των παρονομαστών, όλων των στοιχείων του πίνακα *A*.

**Βήμα 2:** Γράφω τον πίνακα *A* με τη μορφή  $A=(1/d)*N$  με τον *N* πολυωνυμικό.

**Βήμα 3:** Υπολογίζω τη *Smith* μορφή του *N*.

**Βήμα 4:** Η *McMillan* μορφή του *A* προκύπτει από τη διαίρεση κάθε στοιχείου της *Smith* μορφής του *N* με το *d* και αφού πραγματοποιηθούν όλες οι απλοποιήσεις ( $McMillan[A] = (1/d)*SmithForm[N]$ ).

**Παράδειγμα**

$$A=\left\{\left\{\frac{1}{s+1},0\right\},\left\{\frac{1}{s+2},\frac{1}{s+3}\right\}\right\};$$

**McMillanForm[A]**

**MatrixForm[%]**

$$\left( \begin{array}{cc} \frac{1}{(1+s) (2+s) (3+s)} & 0 \\ 0 & 2 + s \end{array} \right)$$

- **Συνάρτηση *LeftStructureMatrix***

**Χρήση – Σύνταξη** *LeftStructureMatrix[A , var]* , επιστρέφει έναν αριστερό δομικό (*left structure*) πίνακα του ρητού, πάνω στα πολυώνυμα της μεταβλητής *var*, πίνακα *A* (με βαθμό μικρότερο ή ίσο του πλήθους των γραμμών του) καθώς και τον πίνακα του μετασχηματισμού του.

**Αλγόριθμος**

**Βήμα 1:** Θέτω *p* τον αριθμό των γραμμών, *m* τον αριθμό των στηλών και *r* την τάξη του πίνακα *A*. Είναι  $r \leq p$ . Θέτω αρχικά τον πίνακα μετασχηματισμού *T* ίσο με τον μοναδιαίο πίνακα διάστασης  $m \times m$  και  $LS=A$ .

**Βήμα 2:** Φέρνω στη θέση (1,1) το στοιχείο της πρώτης γραμμής του *LS* με το ελάχιστο  $\delta(\cdot)$  ( $\delta(a(s)) = \text{βαθμός αριθμητή } a(s) - \text{βαθμός παρονομαστή } a(s)$ ). Η αντιμετάθεση πραγματοποιείται πολλαπλασιάζοντας από δεξιά τον *LS* με κατάλληλο πίνακα *E* (που προκύπτει από μοναδιαίο με αντιμετάθεση των κατάλληλων στηλών). Θέτω  $LS \rightarrow LS * E$  και  $T \rightarrow T * E$  (γινόμενο πινάκων).



**Βήμα 3:** Κάθε στοιχείο της πρώτης γραμμής γράφεται:  $ls[1,j] = q[1,j]*ls[1,1]+u[1,j]$  ( όπου  $ls[i,j]$  το  $(i,j)$  στοιχείο του πίνακα LS,  $q[1,j]$  κατάλληλο πολυώνυμο και  $u[1,j]$  ρητή έκφραση με  $\delta(u[1,j]) \leq \delta(ls[1,1])$  ).

**Βήμα 4:** Αφαιρώ την πρώτη στήλη  $q[1,j]$  φορές από την  $j$  στήλη, για κάθε  $j$  από 2 έως  $m$ . Ορίζω εκ νέου τον  $E$  (θα διαφέρει από τον  $m \times m$  μοναδιαίο μόνο στην πρώτη γραμμή η οποία θα έχει τη μορφή  $[1, -q[1,2], \dots, -q[1,m]]$ ) ώστε να εκφράζει τον συγκεκριμένο στοιχειώδη μετασχηματισμό. Θέτω  $LS \rightarrow LS * E$  και  $T \rightarrow T * E$ .

**Βήμα 5:** Το  $\delta(.)$  των στοιχείων της πρώτης γραμμής έχει ελαττωθεί. Επαναλαμβάνω τα βήματα 2 έως 4 μέχρι να μηδενιστούν τα  $m-1$  τελευταία στοιχεία της γραμμής 1 (δηλαδή όλα εκτός του  $(1,1)$ )

**Βήμα 6:** Εφαρμόζω τα ανάλογα των βημάτων 2 έως 5 για τις υπολειπόμενες γραμμές του πίνακα. Δηλαδή, για τη γραμμή  $i$ , φέρνω στη θέση  $(i,i)$  το στοιχείο της  $i$  γραμμής με το μικρότερο  $\delta$ , επιλέγοντας πάντα μεταξύ των στοιχείων  $(i,j)$  με  $j \geq i$  (από τη διαγώνιο και δεξιότερα). Κατόπιν μειώνω το  $\delta(.)$  των στοιχείων της γραμμής που βρίσκονται μετά τη διαγώνιο, όπως στα βήματα 3,4. Επαναλαμβάνω μέχρι να μηδενιστούν τα τελευταία  $m-i$  στοιχεία της γραμμής  $i$ .

**Βήμα 7:** Η διαδικασία ολοκληρώνεται όταν τα παραπάνω εκτελεστούν για όλες τις γραμμές. Ένας αριστερός δομικός πίνακας του  $A$  είναι ο LS. Ο αντίστοιχος πίνακας μετασχηματισμού είναι ο T ( $A * T = LS$ ).

**Παράδειγμα**

$$A = \left\{ \left\{ \frac{1}{s+1}, 0, \frac{s+1}{s-3} \right\}, \left\{ \frac{1}{s+2}, \frac{1}{s+3}, s^2 - 3s + 9 \right\} \right\}$$

$$\{LS, T\} = \text{LeftStructureMatrix}[A, s];$$

$$\text{MatrixForm}[LS]$$

$$\begin{pmatrix} \frac{16}{(-3+s)(1+s)} & 0 & 0 \\ \frac{13+2s-s^2+s^3}{2+s} & \frac{1}{16(2+s)(3+s)} & 0 \end{pmatrix}$$

$$\text{MatrixForm}[T]$$

$$\begin{pmatrix} -5 - s & \frac{1}{16}(1+s)^2 & -(1+s)^2(2+s) \\ 0 & \frac{1}{16}(-82 + 24s - s^2 - 3s^3 + s^4) & -(3+s)(-55 + 7s + 5s^2 - 4s^3 + s^4) \\ 1 & \frac{3-s}{16} & (-3+s)(2+s) \end{pmatrix}$$

$$\text{FullSimplify}[A.T-LS]$$

(\*Επαλήθευση για τον μετασχηματισμό \*)

$$\{\{0,0,0\}, \{0,0,0\}\}$$

- **Συνάρτηση *RightStructureMatrix***

**Χρήση – Σύνταξη** *RightStructureMatrix*[A , var] , επιστρέφει έναν δεξιό δομικό (right structure) πίνακα του ρητού, πάνω στα πολυώνυμα της μεταβλητής var, πίνακα A (με βαθμό μικρότερο του πλήθους των στηλών του) καθώς και τον πίνακα του μετασχηματισμού.

**Αλγόριθμος**

Ο αλγόριθμος είναι όμοιος με τον προηγούμενο. Οι μετασχηματισμοί εφαρμόζονται στις γραμμές αντί για τις στήλες του πίνακα. Έτσι:

**Βήμα 1:** Εφαρμόζω την *LeftStructureMatrix* για τον ανάστροφο του πίνακα.

**Βήμα 2:** Αναστρέφω τον αριστερό δομικό πίνακα για τον ανάστροφο του A και ο πίνακας RS που προκύπτει είναι δεξιός δομικός πίνακας του A. Ομοίως, αναστρέφοντας τον πίνακα μετασχηματισμού που επιστρέφει η *LeftStructureMatrix* προκύπτει ο πίνακας μετασχηματισμού του A (δηλ.  $T^*A=RS$ ).

**Παράδειγμα**

$$A = \left\{ \left\{ \frac{1}{(1+s)}, \frac{1}{(s+2)} \right\}, \left\{ 0, \frac{1}{(3+s)} \right\}, \left\{ \frac{(1+s)}{(s-3)}, 9-3s+s^2 \right\} \right\};$$

$$\{RS, T\} = \text{RightStructureMatrix}[A, s];$$

**MatrixForm**[RS]

$$\begin{pmatrix} \frac{16}{(-3+s)(1+s)} & \frac{13+2s-s^2+s^3}{2+s} \\ 0 & \frac{1}{16(2+s)(3+s)} \\ 0 & 0 \end{pmatrix}$$

**MatrixForm**[T]

$$\begin{pmatrix} -5-s & 0 & 1 \\ \frac{1}{16}(1+s)^2 & \frac{1}{16}(-82+24s-s^2-3s^3+s^4) & \frac{3-s}{16} \\ -(1+s)^2(2+s) & -(3+s)(-55+7s+5s^2-4s^3+s^4) & (-3+s)(2+s) \end{pmatrix}$$

**FullSimplify**[T.A-RS] (\*Επαλήθευση για τον μετασχηματισμό\*)

$$\{\{0,0\}, \{0,0\}, \{0,0\}\}$$

- **Συνάρτηση *PMLeftGCD***

**Χρήση – Σύνταξη** *PMLeftGCD*[{A,B} , var] , επιστρέφει τον πολωνυμικό ως προς τη μεταβλητή var αριστερό Μ.Κ.Δ. των πολωνυμικών πινάκων A, B διαστάσεων  $p \times k$  και  $p \times t$  αντιστοίχως, με  $m=k+t \geq p$  αν ο βαθμός του block πίνακα [A , B] είναι p.

**Αλγόριθμος**

**Βήμα 1:** Συνθέτω τον block πίνακα  $T = [A, B]$  (με αριθμό γραμμών  $p$ ).

**Βήμα 2:** Υπολογίζω τον αριστερό δομικό πίνακα LST του  $T$ .

**Βήμα 3:** Ο πίνακας αριστερός Μ.Κ.Δ. προκύπτει από τις πρώτες  $p$  στήλες του LST.

**Παράδειγμα**

$$A = \{ \{s+1, 0\}, \{s(s+2), s+1\} \};$$

$$B = \{ \{ (s-3)(s-1)(s+1), (s+1) \}, \{ s+3, (s+1)s \} \};$$

$$PMLeftGCD[\{A, B\}, s]$$

$$\{ \{1+s, 0\}, \{s(2+s), 1\} \}$$

$$MatrixForm[\%]$$

$$\begin{pmatrix} 1 + s & 0 \\ s(2 + s) & 1 \end{pmatrix}$$

- *Συνάρτηση  $PMRightGCD$*

**Χρήση – Σύνταξη**  $PMRightGCD[\{A, B\}, var]$ , επιστρέφει τον πολυωνυμικό ως προς τη μεταβλητή  $var$  δεξιό Μ.Κ.Δ. των πολυωνυμικών πινάκων  $A, B$  διαστάσεων  $k \times m$  και  $t \times m$  αντιστοίχως, με  $p=k+t \geq m$  αν ο βαθμός του κατακόρυφου block πίνακα  $\{A, B\}$  είναι  $m$ .

**Αλγόριθμος**

**Βήμα 1:** Συνθέτω τον κατακόρυφο block πίνακα  $T = [VerticalBlock\{A, B\}]$  (με αριθμό στηλών  $m$ ).

**Βήμα 2:** Υπολογίζω τον δεξιό δομικό πίνακα PST του  $T$ .

**Βήμα 3:** Ο πίνακας δεξιός Μ.Κ.Δ. προκύπτει από τις πρώτες  $m$  στήλες του LST.

**Παράδειγμα**

$$A = \{ \{s+1, 0\}, \{s(s+2), s+1\} \};$$

$$B = \{ \{ (s-3)(s-1)(s+1), (s+1) \}, \{ s+3, (s+1)s \} \};$$

$$PMLeftGCD[\{A, B\}, s]$$

$$\{ \{1+s, 0\}, \{s(2+s), 1\} \}$$

$$MatrixForm[\%]$$

$$\begin{pmatrix} 1 + s & 0 \\ s(2 + s) & 1 \end{pmatrix}$$

- **Συνάρτηση *RightMFD***

**Χρήση – Σύνταξη** *RightMFD[A]*, επιστρέφει μία δεξιά κλασματική περιγραφή του ρητού πίνακα A.

**Αλγόριθμος**

**Βήμα 1:** Θέτω p τον αριθμό των γραμμών, m τον αριθμό των στηλών του A. Υπολογίζω όλα τα στοιχεία d[i] ως τα Ε.Κ.Π. των παρονομαστών όλων των στοιχείων της στήλης i του A, για i από 1 έως m.

**Βήμα 2:** Δημιουργώ τον διαγώνιο πίνακα DR με διαγώνια στοιχεία τα d[1], ..., d[m].

**Βήμα 3:** Το ζεύγος πολυωνυμικών πινάκων  $NR=A*DR$ , NR αποτελεί δεξιά κλασματική περιγραφή του A, με πίνακα παρονομαστή τον DR.

**Παράδειγμα**

$A=\{1/s,0\},\{(s+1)/(s-1),1/(s(s-2))\}$ ;

$\{NR,DR\}=RightMFD[A]$ ;

**MatrixForm[NR]**

$$\begin{pmatrix} -1 + s & 0 \\ s(1 + s) & 1 \end{pmatrix}$$

**MatrixForm[DR]**

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

**NR.Inverse[DR]-A//Simplify** (\*Επαλήθευση\*)

$\{0,0\},\{0,0\}$

- **Συνάρτηση *RightCoprimeMFD***

**Χρήση – Σύνταξη** *RightCoprimeMFD[A, var]*, επιστρέφει δεξιά κλασματική περιγραφή του ρητού πίνακα A, με τους πίνακες αριθμητή και παρονομαστή από δεξιά πρώτους μεταξύ τους ως προς τα πολυωνυμικά στοιχεία της μεταβλητής var.

**Αλγόριθμος**

**Βήμα 1:** Υπολογίζω δεξιά κλασματική περιγραφή NR1, DR1 του πίνακα A με χρήση της *RightMFD*.

**Βήμα 2:** Υπολογίζω τον δεξιό Μ.Κ.Δ. RG των NR1, DR1. Υπολογίζω τον αντίστροφο του, InvRG.

**Βήμα 3:** Υπολογίζω τους πίνακες  $NR=NR1*InvRG$  και  $DR=DR1*InvRG$ .

Το ζεύγος NR , DR αποτελεί δεξιά κλασματική περιγραφή του A, με τον αριθμητή και παρονομαστή από δεξιά πρώτους μεταξύ τους.

**Παράδειγμα**

**{NR,DR}=RightCoprimeMFD[A,s];**

**MatrixForm[NR]**

$$\begin{pmatrix} 0 & 2 + s \\ 1 & (1 + s) (3 + s) \end{pmatrix}$$

**MatrixForm[DR]**

$$\begin{pmatrix} -1 - s & 0 \\ 2 + s & (1 + s) (2 + s)^2 \end{pmatrix}$$

**NR.Inverse[DR]-A//Simplify** (\* Επαλήθευση \*)

**{{0,0},{0,0}}**

**PMRightGCD[{NR,DR},s]** (\* Ο δεξιός Μ.Κ.Δ. των NR,DR \*)

**{{1,(1+s) (3+s)},{0,1}}**

**Det[%]** (\* Μ.Κ.Δ. μονομετρικός, NR,DR από δεξιά πρώτοι μεταξύ τους \*)

1

- **Συνάρτηση LeftMFD**

**Χρήση – Σύνταξη** *LeftMFD[A]* , επιστρέφει μία αριστερή κλασματική περιγραφή του ρητού πίνακα A.

**Αλγόριθμος**

**Βήμα 1:** Θέτω p τον αριθμό των γραμμών, m τον αριθμό των στηλών του A. Υπολογίζω όλα τα στοιχεία d[i] ως τα Ε.Κ.Π. των παρονομαστών όλων των στοιχείων της γραμμής i του A, για i από 1 έως p.

**Βήμα 3:** Δημιουργώ τον διαγώνιο πίνακα DL με διαγώνια στοιχεία τα d[1], ..., d[p].

**Βήμα 4:** Το ζεύγος πολυωνυμικών πινάκων DL, NL=DL\*A αποτελεί αριστερή κλασματική περιγραφή του A, με πίνακα παρονομαστή τον DL.

**Παράδειγμα**

**A={{1/(s+1)^2,1/(s+2)/(s+1)},{1/(s+2)/(s+1),(s+3)/(s+2)^2}};**

**{DL,NL}=LeftMFD[A];**

**MatrixForm[DL]**

$$\begin{pmatrix} (1 + s)^2 (2 + s) & 0 \\ 0 & (1 + s) (2 + s)^2 \end{pmatrix}$$

**MatrixForm[DL]**

$$\begin{pmatrix} (1+s)^2(2+s) & 0 \\ 0 & (1+s)(2+s)^2 \end{pmatrix}$$

- **Συνάρτηση LeftCoprimeMFD**

**Χρήση – Σύνταξη** *LeftCoprimeMFD[A, var]*, επιστρέφει αριστερή κλασματική περιγραφή του ρητού πίνακα A, με τους πίνακες παρονομαστή και αριθμητή από αριστερά πρώτους μεταξύ τους ως προς τα πολυωνυμικά στοιχεία της μεταβλητής var.

**Αλγόριθμος**

**Βήμα 1:** Υπολογίζω αριστερή κλασματική περιγραφή DL1, NL1 του πίνακα A με χρήση της *LeftMFD*.

**Βήμα 2:** Υπολογίζω τον αριστερό Μ.Κ.Δ. LG των DL1, NL1. Υπολογίζω τον αντίστροφό του, InvLG.

**Βήμα 3:** Υπολογίζω τους πίνακες DL=InvLG\*DL1 και NL=InvLG\*NL1.

Το ζεύγος DL, NL αποτελεί αριστερή κλασματική περιγραφή του A, με τον παρονομαστή και αριθμητή από αριστερά πρώτους μεταξύ τους.

**Παράδειγμα**

**A**={{1/(s+1)^2,1/(s+2)/(s+1)},{1/(s+2)/(s+1),(s+3)/(s+2)^2}};

**{DL,NL}**=LeftCoprimeMFD[A,s];

**MatrixForm[DL]**

$$\begin{pmatrix} -1-s & 2+s \\ 0 & (1+s)(2+s)^2 \end{pmatrix}$$

**MatrixForm[NL]**

$$\begin{pmatrix} -1-s & 2+s \\ 0 & (1+s)(2+s)^2 \end{pmatrix}$$

**Inverse[DL].NL-A//Simplify** (\* Επαλήθευση \*)

{{0,0},{0,0}}

**PMLeftGCD[{DL,NL},s]** (\* Ο αριστερός Μ.Κ.Δ. των DL,NL \*)

{{1,0},{(1+s)(3+s),1}}

**Det[%]** (\* Μ.Κ.Δ. μονομετρικός, DL,NL από αριστερά πρώτοι μεταξύ τους \*)

1

τους \*)

- **Συνάρτηση *HighestRowDegreeCoefficientMatrix***

**Χρήση – Σύνταξη** *HighestRowDegreeCoefficientMatrix[A , var]* , επιστρέφει τον πίνακα συντελεστή κατά βαθμούς γραμμών (highest row degree coefficient matrix) ως προς τη μεταβλητή var του πολυωνυμικού πίνακα A.

**Αλγόριθμος**

**Βήμα 1:** Θέτω p τον αριθμό των γραμμών, m τον αριθμό των στηλών του A. Υπολογίζω τους βαθμούς γραμμών dg[i] του A για i από 1 έως p.

**Βήμα 2:** Κατασκευάζω τον p×m πραγματικό πίνακα HRA ώστε το (i,j) στοιχείο του, hra[i,j], να είναι ίσο με τον συντελεστή της δύναμης της μεταβλητής var με εκθέτη dg[i] που εμφανίζεται στο αντίστοιχο (i,j) πολυωνυμικό στοιχείο του πίνακα A, για i από 1 έως p και j από 1 έως m. Ο πίνακας συντελεστής κατά βαθμούς γραμμών του A είναι ο HRA.

**Παράδειγμα**

A={{s+1,(s-2)2s},{1,2(s-3)(s-4)}};

HighestRowDegreeCoefficientMatrix[A,s]

{{0,2},{0,2}}

MatrixForm[%]

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

- **Συνάρτηση *HighestColumnDegreeCoefficientMatrix***

**Χρήση – Σύνταξη** *HighestColumnDegreeCoefficientMatrix[A , var]* , επιστρέφει τον πίνακα συντελεστή κατά βαθμούς στηλών (highest column degree coefficient matrix) ως προς τη μεταβλητή var του πολυωνυμικού πίνακα A.

**Αλγόριθμος**

**Βήμα 1:** Υπολογίζω τον ανάστροφο του πίνακα A. Για τον ανάστροφο υπολογίζω τον πίνακα συντελεστή κατά βαθμούς γραμμών HRT.

**Βήμα 2:** Θέτω HCA τον ανάστροφο του HRT. Ο HCA είναι ο πίνακας συντελεστής κατά βαθμούς στηλών του A.

**Παράδειγμα**

A={{s+1,(s-2)2s},{1,2(s-3)(s-4)}};

HighestColumnDegreeCoefficientMatrix[A,s]

{{1,2},{0,2}}

**MatrixForm[%]**

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

- **Συνάρτηση *MatrixReductionToRowProper***

**Χρήση – Σύνταξη** *MatrixReductionToRowProper*[A , var] , επιστρέφει τον ισοδύναμο κανονικό ως προς τις γραμμές (row proper, row reduced) πίνακα RRA του πολυωνυμικού ως προς τη μεταβλητή var πίνακα A, καθώς και τον πίνακα μετασχηματισμού T, με την προϋπόθεση ότι ο A έχει πλήρη τάξη γραμμών ή στηλών.

### Αλγόριθμος

**Βήμα 1:** Θέτω p τον αριθμό των γραμμών, m τον αριθμό των στηλών του A και υπολογίζω την τάξη r του πίνακα A.

**Βήμα 2:** Υπολογίζω τον πίνακα συντελεστή κατά βαθμούς γραμμών HR του A.

Υπολογίζω την τάξη rHR του πίνακα HR.

**Βήμα 3:** Ελέγχω αν rHR = min{p,m}. Στην περίπτωση που αυτό ισχύει, ο A είναι ήδη κανονικός ως προς τις γραμμές. Επιστρέφω τον A και τον μοναδιαίο p×p πίνακα ως πίνακα μετασχηματισμού και δεν συνεχίζω στα υπόλοιπα βήματα. Διαφορετικά, ελέγχω αν r = min{p,m}. Αν αυτό δεν ισχύει, ο A δεν μπορεί να μετασχηματιστεί σε κανονικό ως προς τις γραμμές πίνακα και δεν συνεχίζω.

**Βήμα 4:** Θέτω RRA=A και T ίσο με τον μοναδιαίο p×p πίνακα. Επίσης θέτω V=HR.

**Βήμα 5:** Υπολογίζω διάνυσμα γραμμή a=[ a[1], ... , a[p] ] ώστε a\*V=0 (a στοιχείο του αριστερού πυρήνα του V).

**Βήμα 6:** Υπολογίζω τους βαθμούς (μέγιστους εκθέτες ως προς var) γραμμών του RRA, r[1], ..., r[p]. Θέτω r0 τον μέγιστο από αυτούς και i0 το δείκτη γραμμής στην οποία αντιστοιχεί.

**Βήμα 7:** Σχηματίζω το διάνυσμα γραμμή: as = [ a[1]\*var<sup>(r0-r[1])</sup>, ... , a[i0], ... , a[p]\*var<sup>(r0-r[p])</sup> ].

**Βήμα 8:** Κατασκευάζω τον πίνακα στοιχειώδους μετασχηματισμού E ώστε να είναι ίσος με μοναδιαίο p×p πίνακα του οποίου η i0 γραμμή έχει αντικατασταθεί από το διάνυσμα γραμμή as.

**Βήμα 9:** Θέτω T → E\*T , RRA → E\*RRA.

**Βήμα 10:** Υπολογίζω εκ νέου τον V ως πίνακα συντελεστή κατά βαθμούς γραμμών του RRA. Υπολογίζω την τάξη rV του πίνακα V. Αν rV < min{p,m}, επαναλαμβάνω τη



διαδικασία από το βήμα 5 και μετά, διαφορετικά  $rV = \min\{p,m\}$  και επιστρέφω τον RRA ως ισοδύναμο του A και κανονικό ως προς τις γραμμές και τον T ως τον πίνακα του μετασχηματισμού.

**Παράδειγμα**

$A = \{\{s+1, (s-2)2s\}, \{1, 2(s-3)(s-4)\}\};$

**MatrixRank[A]**

2

**HighestRowDegreeCoefficientMatrix[A,s]**

$\{\{0,2\}, \{0,2\}\}$

**MatrixRank[%]** (\* Ο A δεν είναι κανονικός ως προς τις γραμμές του \*)

1

**{RRA,T}=MatrixReductionToRowProper[A,s];**

**MatrixForm[RRA]**

$\begin{pmatrix} -s & -2 & (-12 + 5s) \\ 1 & 2 & (-4 + s) & (-3 + s) \end{pmatrix}$

**MatrixRank[HighestRowDegreeCoefficientMatrix[RRA,s]]**

2 (\* Ο RRA είναι κανονικός ως προς τις γραμμές του \*)

**MatrixForm[T]**

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

**T.A-RRA//Simplify** (\* Επαλήθευση για τον μετασχηματισμό \*)

$\{\{0,0\}, \{0,0\}\}$

- **Συνάρτηση *MatrixReductionToColumnProper***

**Χρήση – Σύνταξη** *MatrixReductionToColumnProper[A, var]*, επιστρέφει τον ισοδύναμο κανονικό ως προς τις στήλες (column proper, column reduced) πίνακα CRA του πολυωνυμικού ως προς τη μεταβλητή var πίνακα A καθώς και τον πίνακα μετασχηματισμού T, με την προϋπόθεση ότι ο A έχει πλήρη τάξη γραμμών ή στηλών.

**Αλγόριθμος**

**Βήμα 1:** Θέτω p τον αριθμό των γραμμών, m τον αριθμό των στηλών του A και υπολογίζω την τάξη r του πίνακα A.

**Βήμα 2:** Υπολογίζω τον πίνακα συντελεστή κατά βαθμούς στηλών HC του A.

Υπολογίζω την τάξη  $r_{HC}$  του πίνακα  $HC$ .

**Βήμα 3:** Ελέγχω αν  $r_{HC} = \min\{p,m\}$ . Στην περίπτωση που αυτό ισχύει, ο  $A$  είναι ήδη κανονικός ως προς τις στήλες. Επιστρέφω τον  $A$  και τον μοναδιαίο  $m \times m$  πίνακα ως πίνακα μετασχηματισμού και δεν συνεχίζω στα υπόλοιπα βήματα. Διαφορετικά, ελέγχω αν  $r = \min\{p,m\}$ . Αν αυτό δεν ισχύει, ο  $A$  δεν μπορεί να μετασχηματιστεί σε κανονικό ως προς τις στήλες πίνακα και δεν συνεχίζω.

**Βήμα 4:** Θέτω  $CRA=A$  και  $T$  ίσο με τον μοναδιαίο  $m \times m$  πίνακα. Επίσης θέτω  $V=HC$ .

**Βήμα 5:** Υπολογίζω διάνυσμα  $a=[a[1], \dots, a[p]]^T$  ώστε  $V*a=0$  ( $a$  στοιχείο του δεξιού πυρήνα του  $V$ ).

**Βήμα 6:** Υπολογίζω τους βαθμούς (μέγιστους εκθέτες ως προς  $\text{var}$ ) στηλών του  $CRA$ ,  $c[1], \dots, c[m]$ . Θέτω  $c_0$  τον μέγιστο από αυτούς και  $i_0$  το δείκτη στήλης στην οποία αντιστοιχεί.

**Βήμα 7:** Σχηματίζω το διάνυσμα:  $as = [a[1]*\text{var}^{(c_0-c[1])}, \dots, a[i_0], \dots, a[m]*\text{var}^{(c_0-c[m])}]^T$ .

**Βήμα 8:** Κατασκευάζω τον πίνακα στοιχειώδους μετασχηματισμού  $E$  ώστε να είναι ίσος με μοναδιαίο  $m \times m$  πίνακα του οποίου η  $i_0$  στήλη έχει αντικατασταθεί από το διάνυσμα  $as$ .

**Βήμα 9:** Θέτω  $T \rightarrow T*E$ ,  $RRA \rightarrow RRA*E$ .

**Βήμα 10:** Υπολογίζω εκ νέου τον  $V$  ως πίνακα συντελεστή κατά βαθμούς στηλών του  $CRA$ . Υπολογίζω την τάξη  $r_V$  του πίνακα  $V$ . Αν  $r_V < \min\{p,m\}$ , επαναλαμβάνω τη διαδικασία από το βήμα 5 και μετά, διαφορετικά  $r_V = \min\{p,m\}$  και επιστρέφω τον  $CRA$  ως ισοδύναμο του  $A$  και κανονικό ως προς τις στήλες και τον  $T$  ως τον πίνακα του μετασχηματισμού.

### Παράδειγμα

$A=\{\{1+s,1\},\{2(-2+s),2(-4+s),(-3+s)\}\}$ ;

**MatrixRank[A]**

2

**HighestColumnDegreeCoefficientMatrix[A,s]**

$\{\{0,0\},\{2,2\}\}$

**MatrixRank[%]** (\* Ο  $A$  δεν είναι κανονικός ως προς τις στήλες του \*)

1

**{CRA,T}=MatrixReductionToColumnProper[A,s];**

**MatrixForm[CRA]**

$$\begin{pmatrix} -s & 1 \\ -2(-12+5s) & 2(-4+s)(-3+s) \end{pmatrix}$$

**MatrixRank[HighestColumnDegreeCoefficientMatrix[CRA,s]]**

2 (\* Ο CRA είναι κανονικός ως προς τις στήλες του \*)

**MatrixForm[T]**

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

**CRA-A.T//Simplify** (\* Επαλήθευση για τον μετασχηματισμό \*)

{{0,0},{0,0}}

- **Συνάρτηση *RowProperDenominatorLeftCoprimeMFD***

**Χρήση – Σύνταξη** *RowProperDenominatorLeftCoprimeMFD* [A , var] , επιστρέφει αριστερή κλασματική περιγραφή του ρητού πίνακα A, με τους πίνακες παρονομαστή και αριθμητή από αριστερά πρώτους μεταξύ τους ως προς τα πολυωνυμικά στοιχεία της μεταβλητής var και επιπλέον με το πίνακα παρονομαστή κανονικό ως προς τις γραμμές του.

**Αλγόριθμος**

**Βήμα 1:** Υπολογίζω μία αριστερή κλασματική περιγραφή του ρητού πίνακα A, DL1 , NL1 με τους πίνακες παρονομαστή DL1 και αριθμητή NL1 από αριστερά πρώτους μεταξύ τους, με χρήση της *LeftCoprimeMFD*.

**Βήμα 2:** Από τον DL1, υπολογίζω τον ισοδύναμο του, κανονικό ως προς τις γραμμές του, πίνακα DL, κάνοντας χρήση της *MatrixReductionToRowProper*. Επίσης, θέτω T τον πίνακα μετασχηματισμού που επιστρέφει η προηγούμενη συνάρτηση.

**Βήμα 3:** Υπολογίζω τον NL=T\*NL1. Το ζεύγος DL , NL αποτελεί αριστερή κλασματική περιγραφή του ρητού πίνακα A, με τους DL , NL από αριστερά πρώτους μεταξύ τους. Ο DL είναι ταυτοχρόνως κανονικός ως προς τις γραμμές του.

**Παράδειγμα**

A={{(s+1)/(s(s-2)),0},{1/(s(s-1)),1/(s-1)}};

{DL,NL}=RowProperDenominatorLeftCoprimeMFD[A,s]

{{{2-s,-1+s},{0,(-1+s) s}},{-1,1},{1,s}}

**MatrixForm[DL]**

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

**MatrixForm[NL]**

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

**MatrixRank[HighestRowDegreeCoefficientMatrix[DL,s]]**

2 (\* Ο DL είναι κανονικός ως προς τις γραμμές του \*)

**PMLeftGCD[{DL,NL},s]//Det**

-1 (\* Οι DL, NL είναι από αριστερά πρώτοι μεταξύ τους \*)

**Inverse[DL].NL-A//Simplify**

{{0,0},{0,0}} (\* Επαλήθευση: DL<sup>(-1)</sup>\*NL=A \*)

- **Συνάρτηση ColumnProperDenominatorRightCoprimeMFD**

**Χρήση – Σύνταξη** *ColumnProperDenominatorRightCoprimeMFD [A , var]* , επιστρέφει δεξιά κλασματική περιγραφή του ρητού πίνακα A, με τους πίνακες αριθμητή και παρονομαστή από δεξιά πρώτους μεταξύ τους ως προς τα πολυωνυμικά στοιχεία της μεταβλητής var και επιπλέον με το πίνακα παρονομαστή κανονικό ως προς τις στήλες του.

### Αλγόριθμος

**Βήμα 1:** Υπολογίζω μία δεξιά κλασματική περιγραφή του ρητού πίνακα A, NR1 , DR1 με τους πίνακες αριθμητή NR1 και παρονομαστή DR1 από δεξιά πρώτους μεταξύ τους, με χρήση της *RightCoprimeMFD*.

**Βήμα 2:** Από τον DR1, υπολογίζω τον ισοδύναμο του, κανονικό ως προς τις στήλες του, πίνακα DR, κάνοντας χρήση της *MatrixReductionToColumnProper*. Επίσης, θέτω T τον πίνακα μετασχηματισμού που επιστρέφει η προηγούμενη συνάρτηση.

**Βήμα 3:** Υπολογίζω τον NR=NR1\*T. Το ζεύγος NR , DR αποτελεί αριστερή κλασματική περιγραφή του ρητού πίνακα A, με τους NR , DR από δεξιά πρώτους μεταξύ τους. Ο DR είναι ταυτοχρόνως κανονικός ως προς τις γραμμές του.

### Παράδειγμα

$A = \left\{ \left\{ \frac{s+1}{s(s-2)}, 0 \right\}, \left\{ \frac{1}{s(s-1)}, \frac{1}{s-1} \right\} \right\};$

$\{NR, DR\} = \text{ColumnProperDenominatorRightCoprimeMFD}[A, s]$

$\left\{ \left\{ 1+s, 0 \right\}, \left\{ 1, 1 \right\} \right\}, \left\{ \left\{ -2+s, s, 0 \right\}, \left\{ 1, -1+s \right\} \right\}$

**MatrixForm[NR]**

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

**MatrixForm[DR]**

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

**MatrixRank[HighestColumnDegreeCoefficientMatrix[DR,s]]**

2 (\*Ο DR είναι κανονικός ως προς τις στήλες του \*)

**PMRightGCD[{NR,DR},s]//Det**

1 (\* Οι NR,DR είναι από δεξιά πρώτοι μεταξύ τους \*)

**NR.Inverse[DR]-A//Simplify**

{{0,0},{0,0}} (\* Επαλήθευση: A=NR\*DR<sup>(-1)</sup> \*)

- **Συνάρτηση *GWResultant***

**Χρήση – Σύνταξη** *GWResultant [A ,B, var,k]* , επιστρέφει τη γενικευμένη Wolovich resultant τάξης k (k ακέραιος μεγαλύτερος ή ίσος της μονάδας) δύο πολυωνυμικών ως προς μεταβλητή var πινάκων A , B που αποτελούνται από ισάριθμες στήλες.

### **Αλγόριθμος**

**Βήμα 1:** Συνθέτω τον κατακόρυφο block πίνακα  $F=[\text{VerticalBlock}\{A,B\}]$  (Το block A πάνω και ακριβώς από κάτω το block B). Θέτω r το πλήθος των γραμμών του F και c το πλήθος των στηλών. Υπολογίζω τους βαθμούς  $d[1], d[2], \dots, d[c]$  των στηλών 1 , 2 , ... , c του F αντιστοίχως.

**Βήμα 2:** Συμβολίζω με  $f[i,j]$  το (i,j) πολυωνυμικό στοιχείο του πίνακα F και με  $\mathbf{0}[n]$  το μηδενικό διάνυσμα γραμμής μήκους n (το  $\mathbf{0}[0]$  ορίζεται ως το κενό διάνυσμα). Ορίζω την απεικόνιση  $\varphi_k[t,i,j] : f[i,j] \rightarrow [ \mathbf{0}[t], f[i, j, 0], f[i, j, 1], \dots, f[i, j, d[j]], \mathbf{0}[k-1-t] ]$ , όπου  $f[v, u, w]$  είναι ο συντελεστής της δύναμης  $\text{var}^w$  που εμφανίζεται στο (v,u) πολυωνυμικό στοιχείο (δηλ. στο  $f[v, u]$ ).

**Βήμα 3:** Κατασκευάζω τον block πίνακα

$$M_k = \begin{bmatrix} \varphi_k[0,1,1] & \cdots & \varphi_k[0,1,c] \\ \vdots & \ddots & \vdots \\ \varphi_k[0,r,1] & \cdots & \varphi_k[0,r,c] \\ \varphi_k[1,1,1] & \cdots & \varphi_k[1,1,c] \\ \vdots & \ddots & \vdots \\ \varphi_k[1,r,1] & \cdots & \varphi_k[1,r,c] \\ \vdots & \ddots & \vdots \\ \varphi_k[k-1,1,1] & \cdots & \varphi_k[k-1,1,c] \\ \vdots & \ddots & \vdots \\ \varphi_k[k-1,r,1] & \cdots & \varphi_k[k-1,r,c] \end{bmatrix}$$

Ο  $M_k$  είναι η γενικευμένη Wolovich resultant τάξης  $k$  των πολυωνυμικών πινάκων  $A, B$ .

**Παράδειγμα**

$A = \{ \{(-2+s) s, 0\}, \{1, -1+s\} \};$

$B = \{ \{1+s, 0\}, \{1, 1\} \};$

**GWResultant[A,B,s,1]** (\* Η γεν. Wolovich resultant τάξης 1 των A,B \*)

$\{ \{0, -2, 1, 0, 0\}, \{1, 0, 0, -1, 1\}, \{1, 1, 0, 0, 0\}, \{1, 0, 0, 1, 0\} \}$

**MatrixForm[%]**

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

**GWResultant[A,B,s,2]** (\* Η γεν. Wolovich resultant τάξης 2 των A,B \*)

$\{ \{0, -2, 1, 0, 0, 0, 0\}, \{1, 0, 0, 0, -1, 1, 0\}, \{1, 1, 0, 0, 0, 0, 0\}, \{1, 0, 0, 0, 1, 0, 0\}, \{0, 0, -2, 1, 0, 0, 0\}, \{0, 1, 0, 0, 0, -1, 1\}, \{0, 1, 1, 0, 0, 0, 0\}, \{0, 1, 0, 0, 0, 1, 0\} \}$

**MatrixForm[%]**

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

**GWResultant[A,B,s,3]** (\* Η γεν. Wolovich resultant τάξης 3 των A,B \*)

$\{\{0,-2,1,0,0,0,0,0,0\},\{1,0,0,0,0,-1,1,0,0\},\{1,1,0,0,0,0,0,0,0\},\{1,0,0,0,0,1,0,0,0\},\{0,0,-2,1,0,0,0,0,0\},\{0,1,0,0,0,0,-1,1,0\},\{0,1,1,0,0,0,0,0,0\},\{0,1,0,0,0,0,1,0,0\},\{0,0,0,-2,1,0,0,0,0\},\{0,0,1,0,0,0,0,-1,1\},\{0,0,1,1,0,0,0,0,0\},\{0,0,1,0,0,0,0,1,0\}\}$

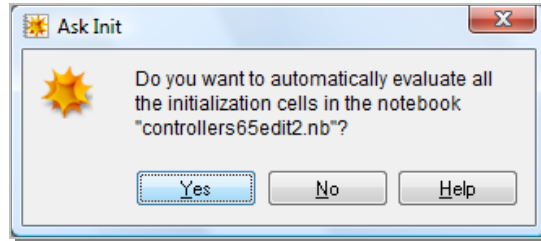
**MatrixForm[%]**

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

### 3.3 Χρήση της εφαρμογής

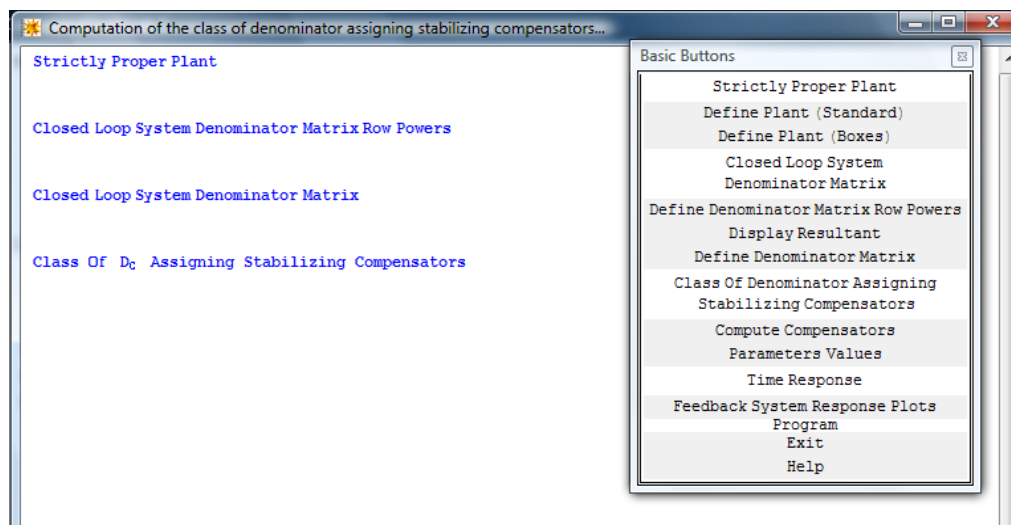
Η χρήση της εφαρμογής απαιτεί Η/Υ με εγκατεστημένη έκδοση τις σειράς 5 (5.0 – 5.2) της Mathematica. Οι πιο πρόσφατες εκδόσεις της δεν είναι προς τα πίσω συμβατές, κυρίως σε ό,τι σχετίζεται με τον προγραμματισμό των γραφικών στοιχείων ελέγχου. Συνεπώς η εκτέλεση της εφαρμογής δεν είναι εφικτή σε εκδόσεις πιο πρόσφατες από την 5. Πέραν της Mathematica, δεν χρησιμοποιείται κάποιο άλλο πακέτο συναρτήσεων ή λογισμικού και η εφαρμογή μπορεί να εκτελεστεί από οποιοδήποτε μέσο χωρίς να χρειάζεται η τοποθέτηση του αρχείου σε συγκεκριμένη μονάδα δίσκου ή φάκελο.

Για την εκτέλεση της εφαρμογής, είτε εκτελούμε απ’ ευθείας το αρχείο “compensators.nb” με διπλό κλικ, είτε το ανοίγουμε, μέσα από το περιβάλλον της Mathematica, μέσω της επιλογής “File / Open”. Εμφανίζεται το παράθυρο – διαλόγου της εικόνας 3.1 στο οποίο απαντούμε καταφατικά για την αρχικοποίηση των συναρτήσεων της εφαρμογής.



Εικόνα 3.1

Στη συνέχεια εμφανίζεται το περιβάλλον χρήσης της εφαρμογής, με τα δύο κύρια παράθυρα, τα οποία έχουν ως τίτλους “*Basic Buttons*” και “*Computation of the denominator assigning stabilizing compensators*” (εικόνα 3.2). Το παράθυρο “*Basic Buttons*” αποτελείται από πλήκτρα τα οποία ξεκινούν όλες τις απαραίτητες διαδικασίες που σχετίζονται με εισαγωγή δεδομένων και εκτέλεση απαραίτητων υπολογισμών. Στο παράθυρο “*Computation of the denominator assigning stabilizing compensators*” εμφανίζονται όλα τα ήδη εισηγμένα δεδομένα και, στο τέλος, αφού έχει πραγματοποιηθεί ο υπολογισμός, η έκφραση της παραμετρικής οικογένειας των αντισταθμιστών.



Εικόνα 3.2

Ξεκινώντας τη διαδικασία, απαιτείται η εισαγωγή του αρχικού συστήματος το οποίο πρέπει να περιγράφεται από αυστηρώς κανονικό πίνακα συναρτήσεων μεταφοράς. Αυτό επιτυγχάνεται, είτε πατώντας το πλήκτρο “*Define Plant (Standard)*”, είτε το πλήκτρο “*Define Plant (Boxes)*” από το παράθυρο “*Basic Buttons*”. Το “*Define Plant (Standard)*” επιλέγεται στην περίπτωση που επιθυμούμε να χρησιμοποιήσουμε τον τυπικό τρόπο εισαγωγής πινάκων της Mathematica (nested lists). Εμφανίζεται παράθυρο διαλόγου, μέσα στο οποίο εισάγουμε την έκφραση του πίνακα, χρησιμοποιώντας την τυπική σύνταξη για

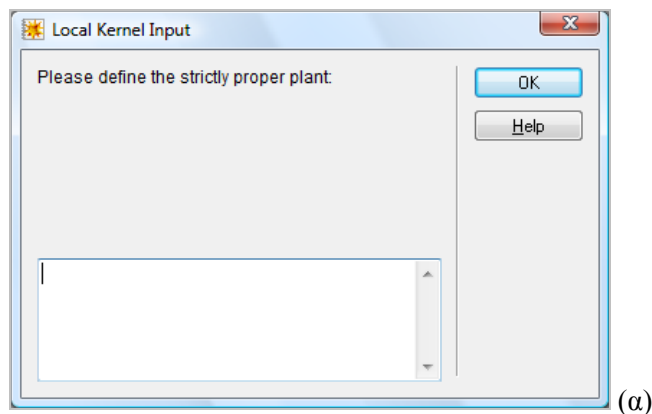


πίνακες στη Mathematica. Στην περίπτωση που δεν επιθυμείται η χρήση τυπικής σύνταξης της Mathematica για την εισαγωγή του πίνακα, επιλέγουμε τη δεύτερη μέθοδο εισαγωγής, πατώντας το πλήκτρο “*Define Plant (Boxes)*” από το παράθυρο “*Basic Buttons*”. Κατά τη διαδικασία αυτή εμφανίζεται αρχικά ένα παράθυρο διαλόγου όπου ορίζουμε τις διαστάσεις του πίνακα που πρόκειται να εισάγουμε (για παράδειγμα, γράφουμε ‘{3,2}’ για να δηλώσουμε ότι ο πίνακας θα έχει 3 γραμμές και 2 στήλες). Κατόπιν, εμφανίζεται νέο παράθυρο όπου απεικονίζεται ένας πίνακας με τις επιθυμητές διαστάσεις. Κάθε στοιχείο του πίνακα είναι ενεργό (πλήκτρο). Πατώντας πάνω σε κάποιο στοιχείο θα μας ζητηθεί να ορίσουμε την τιμή του. Επαναλαμβάνοντας την διαδικασία για κάθε στοιχείο, ορίζουμε τον πίνακα πλήρως. Σε περίπτωση σφάλματος εισαγωγής κάποιου στοιχείου του πίνακα, μπορούμε, κάνοντας «κλικ» με το ποντίκι πάνω του, να επαναπροσδιορίσουμε την τιμή του. Ανεξαρτήτως της διαδικασίας εισαγωγής του πίνακα που χρησιμοποιείται (“*Define Plant (Boxes)*” ή “*Define Plant (Standard)*”) πρέπει τα στοιχεία του πίνακα να είναι αυστηρώς κανονικές ρητές συναρτήσεις. Σε αντίθετη περίπτωση θα είναι αναγκαία η εκ νέου εισαγωγή πίνακα και στο παράθυρο “*Computation of the denominator assigning stabilizing compensators*”, στο πεδίο “*Strictly Proper Plant*” θα εμφανιστεί μήνυμα μη εγκυρότητας του πίνακα συναρτήσεων μεταφοράς: ‘*The Plant Definition is Invalid. Please redefine the system's transfer function matrix.*’ κάτω από το πεδίο “*Strictly Proper Plant*”.

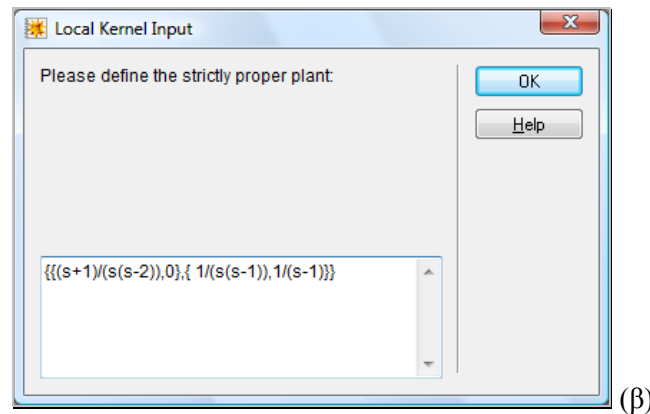
### Παράδειγμα εισαγωγής πίνακα συναρτήσεων μεταφοράς αρχικού συστήματος.

Θα εισάγουμε τον πίνακα συναρτήσεων μεταφοράς του παραδείγματος της παραγράφου 2.4, χρησιμοποιώντας και τις δύο δυνατές μεθόδους:

1. Πατώντας στο “*Define Plant (Standard)*” εμφανίζεται το παράθυρο διαλόγου της εικόνας 3.3(α). Εισάγουμε σε αυτό την έκφραση ‘ $\{(s+1)/(s(s-2)), 0\}, \{1/(s(s-1)), 1/(s-1)\}$ ’ και πατάμε το πλήκτρο ‘*Enter*’ ή με το ποντίκι στο ‘*OK*’ (εικόνα 3.3(β)).

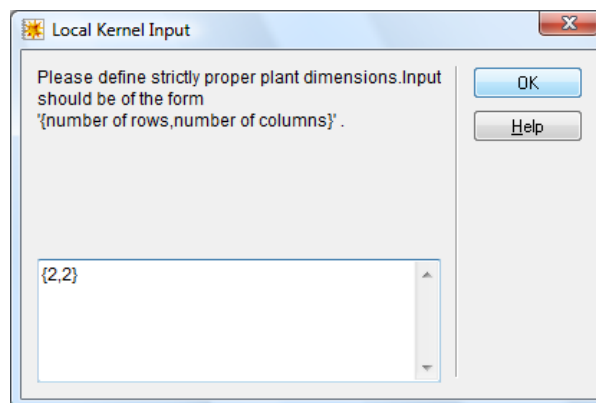


(α)



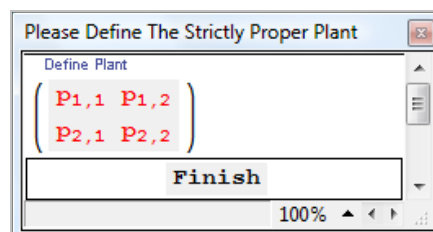
Εικόνα 3.3

2. Αν προτιμήσουμε την εναλλακτική μέθοδο εισαγωγής του πίνακα, πατώντας στο “*Define Plant (Standard)*” εμφανίζεται το παράθυρο διαλόγου της εικόνας 3.4.



Εικόνα 3.4

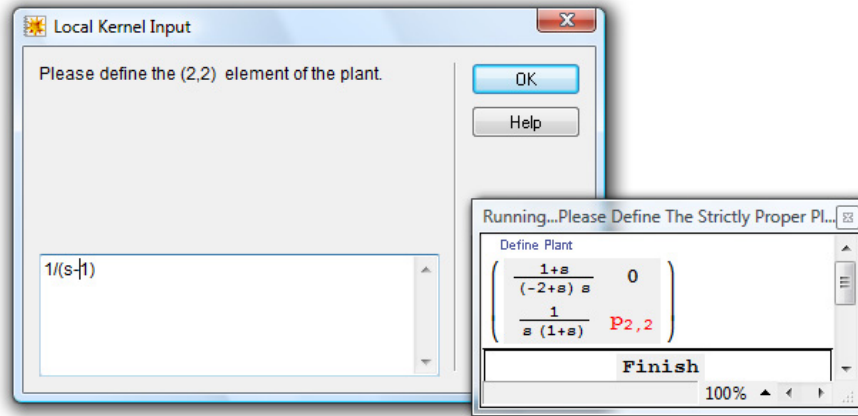
Εισάγουμε τις διαστάσεις του πίνακα, δηλαδή ‘{2,2}’. Θα δημιουργηθεί το παράθυρο της εικόνας 3.5, το οποίο περιέχει έναν «άδειο» πίνακα με δύο γραμμές και δύο στήλες.



Εικόνα 3.5

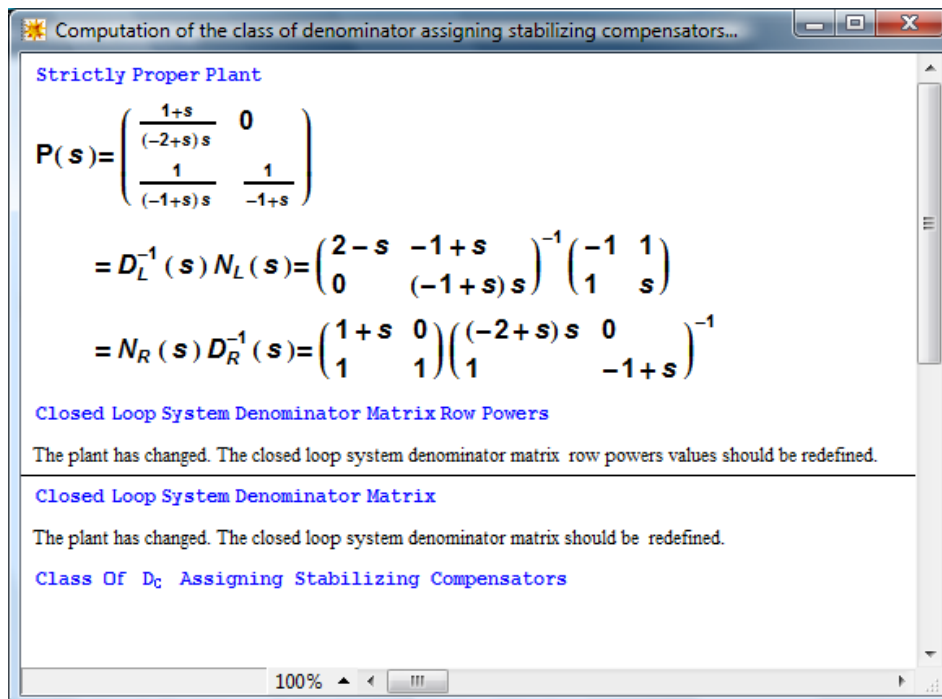
Πατώντας σε κάθε ένα από τα στοιχεία του και δίνοντας τις αντίστοιχες τιμές στα παράθυρα διαλόγου που εμφανίζονται, ορίζουμε τον πίνακα (εικόνα 3.6). Τελειώνοντας

τον ορισμό όλων των στοιχείων, πατάμε πάνω στο πλήκτρο “Finish” που εμφανίζεται κάτω από τον πίνακα (αν χρειάζεται μεγεθύνουμε ελαφρώς το παράθυρο).



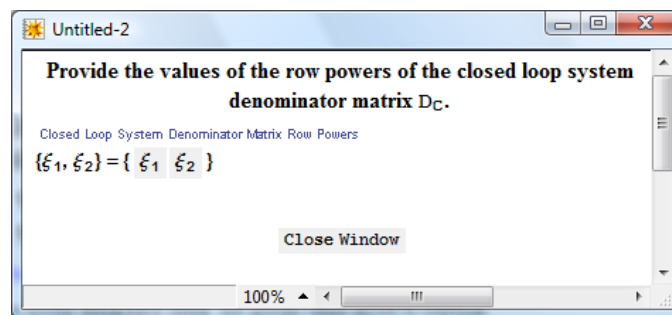
Εικόνα 3.6

Εφόσον η εισαγωγή του πίνακα γίνει επιτυχώς, στο πεδίο “Strictly Proper Plant” του παραθύρου “Computation of the denominator assigning stabilizing compensators” θα εμφανιστεί ο πίνακας που μόλις έχουμε ορίσει μαζί με μία αριστερά πρώτη και μία δεξιά πρώτη κλασματική περιγραφή του, όπως φαίνεται στην εικόνα 3.7 ■

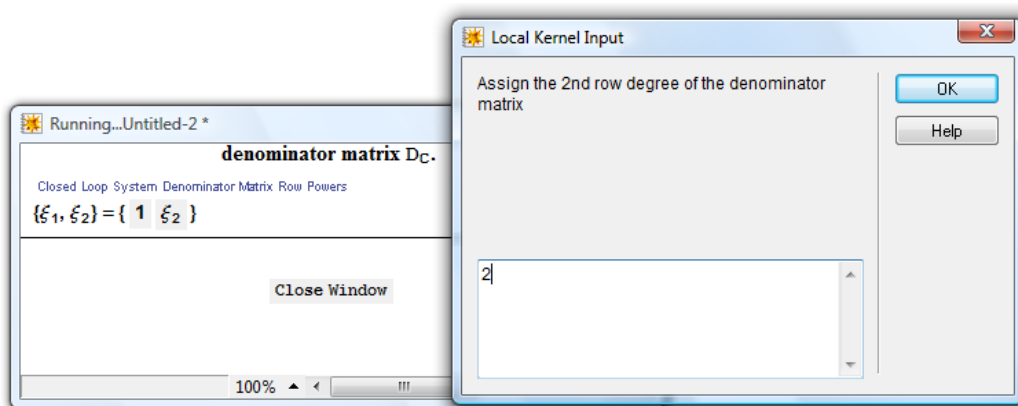


Εικόνα 3.7

Στη συνέχεια της διαδικασίας, απαιτείται η εισαγωγή των επιθυμητών δυνάμεων των γραμμών του πίνακα παρονομαστή, δηλαδή των  $\xi_i, i \in \mathbf{m}$  του αλγορίθμου της παραγράφου 2.3. Αυτό επιτυγχάνεται με το πάτημα του πλήκτρου “Define Denominator Matrix Row Powers”. Το αποτέλεσμα αυτής της ενέργειας είναι η δημιουργία του παραθύρου της εικόνας 3.8. Με ακριβώς ανάλογο τρόπο όπως και στα προηγούμενα, πατώντας πάνω σε κάποιο  $\xi_i$  εμφανίζεται παράθυρο διαλόγου για τον ορισμό της δύναμη της αντίστοιχης γραμμής του πίνακα παρονομαστή (εικόνα 3.9). Οι επιλεγόμενες τιμές πρέπει να είναι φυσικοί αριθμοί μεγαλύτεροι ή ίσοι από τον δείκτη παρατηρησιμότητας του αρχικού συστήματος μειωμένου κατά μία μονάδα, διαφορετικά δεν γίνονται δεκτοί και απαιτείται εισαγωγή εκ νέου.



Εικόνα 3.8

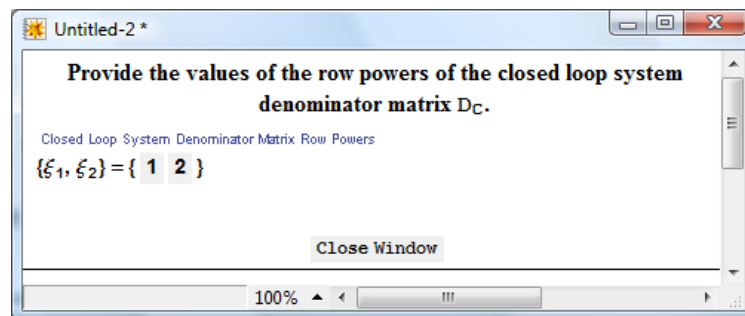


Εικόνα 3.9

Αφού οριστούν όλες οι δυνάμεις γραμμών του πίνακα παρονομαστή, πατάμε στο ‘Close Window’.

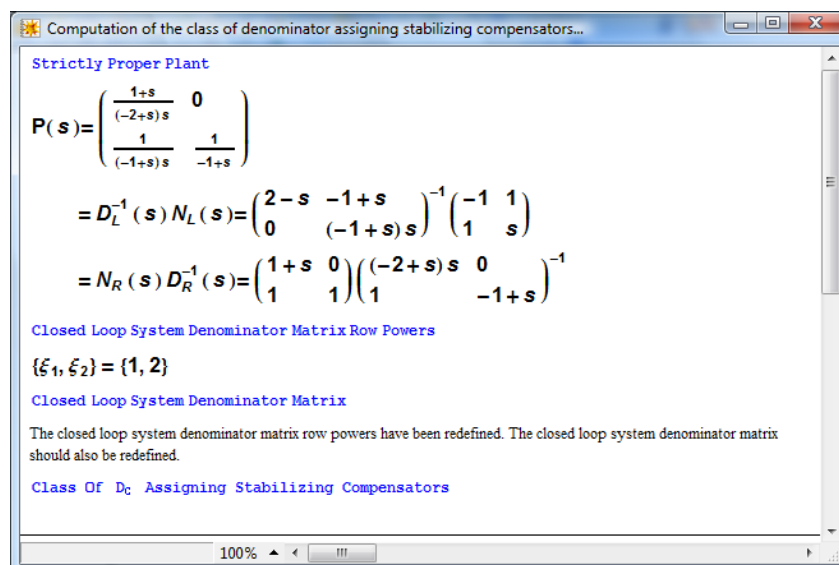
**Παράδειγμα (συνέχεια προηγούμενου) ορισμού των δυνάμεων των γραμμών του πίνακα παρονομαστή**

Συνεχίζοντας το προηγούμενο παράδειγμα, χρειάζεται να ορίσουμε τις δυνάμεις των γραμμών του πίνακα παρονομαστή,  $\xi_i \geq \mu - 1$ ,  $i = 1, 2$ . Όπως και στο παράδειγμα της παραγράφου 2.4 θα ορίσουμε  $\xi_1 = 1 \geq \mu - 1 = 1$ ,  $\xi_2 = 2 \geq \mu - 1 = 1$ . Με πάτημα του πλήκτρου “Define Denominator Matrix Row Degrees” δημιουργείται το παράθυρο της εικόνας 3.8. Πατάμε το “ $\xi_1$ ” και στο παράθυρο διαλόγου που εμφανίζεται, εισάγουμε την τιμή ‘1’. Πιέζοντας το πλήκτρο ‘Enter’ ή κάνοντας κλικ στο “OK” εισάγεται η τιμή ‘1’ στη θέση του “ $\xi_1$ ”. Με ανάλογο τρόπο εισάγουμε την τιμή ‘2’ στη θέση του “ $\xi_2$ ”. Στο παράθυρο με τις ορισθείσες, πλέον, τιμές για τις δυνάμεις γραμμών του παρονομαστή πίνακα (εικόνα 3.10) πατάμε “Close Window”.



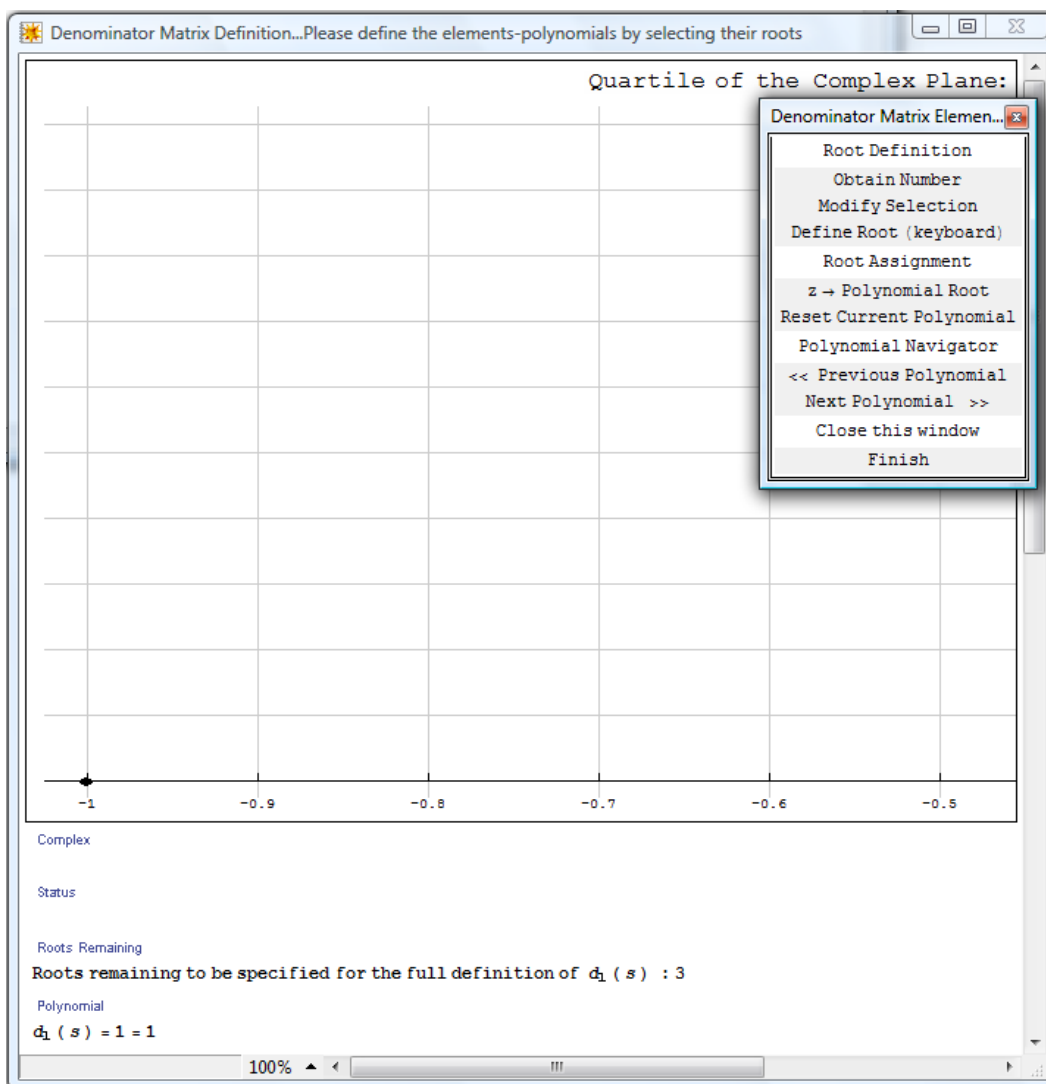
Εικόνα 3.10

Επιστρέφοντας στο παράθυρο “Computation of the denominator assigning stabilizing compensators”, στο πεδίο “Closed Loop System Denominator Matrix Row Powers” θα αναγράφονται και οι ορισθείσες τιμές για τα  $\xi_1, \xi_2$  (εικόνα 3.11) ■



Εικόνα 3.11

Μετά τον ορισμό των δυνάμεων των γραμμών του πίνακα παρονομαστή, η διαδικασία συνεχίζεται με τον ακριβή ορισμό των διαγώνιων στοιχείων του. Ο πίνακας παρονομαστής επιλέγεται διαγώνιος γιατί η μορφή αυτή είναι απλή και ταυτόχρονα πληρούνται οι συνθήκες της ταυτόχρονης κανονικότητας ως προς τις γραμμές και τις στήλες καθώς και της ισότητας των βαθμών γραμμών – στηλών σύμφωνα με τη συνθήκη  $\deg_{ci} D_C(s) = \deg_{ri} D_C(s) = k_i + \xi_i, i \in \mathbf{m}$  που τίθεται στην ανάλυση του κεφαλαίου 2. Τα πολυωνυμικά στοιχεία της διαγωνίου ορίζονται μέσω του ορισμού όλων των μιγαδικών ριζών τους. Πατώντας λοιπόν πάνω στο πλήκτρο “*Define Denominator Matrix*” εμφανίζονται τα παράθυρα της εικόνας 3.12.



Εικόνα 3.12

Το παράθυρο με τίτλο “*Denominator Matrix Definition...Please define the elements-polynomials by selecting their roots*” παρέχει πληροφορίες για την κατάσταση της διαδικασίας ορισμού των στοιχείων του πίνακα παρονομαστή του κλειστού συστήματος.

Συγκεκριμένα, στο πεδίο “Complex” εμφανίζεται ο αριθμός που έχει επιλεγεί για να ανατεθεί ως ρίζα στο τρέχον πολυωνυμικό στοιχείο της διαγωνίου του πίνακα παρονομαστή. Στο πεδίο “Status”, όταν θα έχει ολοκληρωθεί ο ορισμός του πίνακα παρονομαστή, εμφανίζεται το μήνυμα ‘The denominator matrix is defined. Press the “Finish” button to exit.’. Στο πεδίο “Roots Remaining” εμφανίζεται ο υπολειπόμενος αριθμός ριζών που πρέπει να ανατεθούν στο τρέχον πολυώνυμο ώστε αυτό να έχει προσδιοριστεί πλήρως και τέλος στο πεδίο “Polynomial” εμφανίζεται το τρέχον στοιχείο - πολυώνυμο, σε ανεπτυγμένη και παραγοντοποιημένη μορφή, ως γινόμενο των ριζών οι οποίες του έχουν ανατεθεί έως το σημείο εκείνο της διαδικασίας.

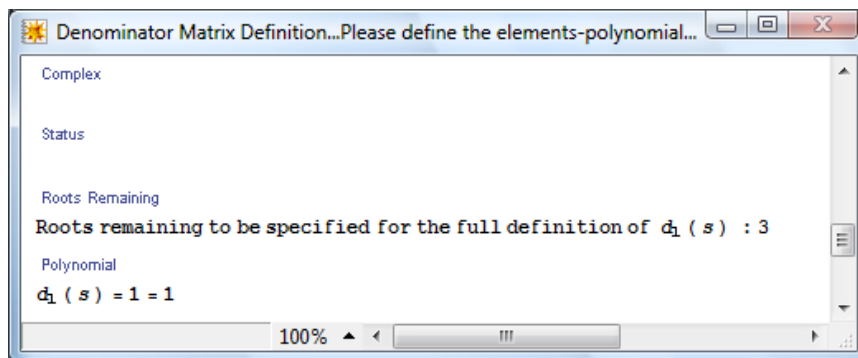
Το παράθυρο με τίτλο “Denominator Matrix Elements Definition” περιέχει όλα τα πλήκτρα εκείνα που αντιστοιχούν στις ενέργειες επιλογής και ανάθεσης ριζών σε πολυώνυμο καθώς και επανάθεσης των ριζών σε κάποιο πολυώνυμο (σε περίπτωση σφάλματος ανάθεσης ρίζας). Συγκεκριμένα, η επιλογή ενός αριθμού ως ρίζας επιτυγχάνεται με το “Define Root (keyboard)”. Η ανάθεση του επιλεγθέντος αριθμού σε πολυώνυμο ως ρίζα του, πραγματοποιείται με το πλήκτρο “z --> Polynomial Root”. Η ανάθεση αφορά στην τρέχουσα επιλογή πολυωνύμου και πραγματοποιείται μόνο εφόσον δεν έχουν ήδη ορισθεί οι απαιτούμενοι πλήθους ρίζες, διαφορετικά το πρόγραμμα αγνοεί την ενέργεια μας. Αυτό συμβαίνει για να εξασφαλισθεί ότι οι εκθέτες των πολυωνυμικών στοιχείων του πίνακα δεν θα υπερβούν, από λάθος, τις προκαθορισμένες τιμές τους. Επίσης, κάθε ανάθεση γνήσιου μιγαδικού αριθμού ως ρίζας πραγματοποιείται μόνο στην περίπτωση που το υπολειπόμενο πλήθος ριζών για το τρέχον στοιχείο είναι τουλάχιστον δύο. Αυτό συμβαίνει γιατί μαζί με κάθε γνήσια μιγαδική ρίζα, γίνεται, αυτομάτως, ανάθεση και της συζυγούς της. Κατά τον προσδιορισμό μίας ρίζας, απαιτείται το πραγματικό μέρος της να είναι αρνητικό, ώστε να μην δημιουργηθεί αστάθεια στο κλειστό σύστημα. Η διαδικασία περιλαμβάνει έλεγχο και δεν συνεχίζεται παρά μόνο όταν εισαχθεί αριθμός με αρνητικό πραγματικό μέρος. Σε κάθε ανάθεση ρίζας ενημερώνονται αυτόματα τα πεδία “Roots Remaining” και “Polynomial” και εφόσον μία ανάθεση θεωρείται η τελευταία απαιτούμενη, εμφανίζεται το σχετικό μήνυμα στο πεδίο “Status”. Αν για οποιοδήποτε λόγο, επιθυμείται ο εκ νέου ορισμός των ριζών του τρέχοντος στοιχείου του πίνακα παρονομαστή, το “Reset Current Polynomial” θα επαναθέσει το στοιχείο στην τιμή ‘1’. Τέλος, για την επιλογή του στοιχείου στο οποίο επιθυμείται η ανάθεση ριζών, χρησιμοποιούνται τα πλήκτρα “ << Previous Polynomial” και “ Next Polynomial >>” τα οποία μεταφέρουν την επιλογή για το τρέχον πολυώνυμο στο προηγούμενο ή επόμενο

στοιχείο της διαγωνίου του πίνακα παρονομαστή, αντιστοίχως. Η διαδικασία ορισμού του πίνακα παρονομαστή θεωρείται ολοκληρωμένη όταν έχουν προσδιοριστεί όλες οι απαιτούμενες ρίζες, όλων των στοιχείων-πολυωνύμων της διαγωνίου του πίνακα. Το πλήκτρο “Finish” κλείνει τα παράθυρα ορισμού του πίνακα παρονομαστή.

### Παράδειγμα (συνέχεια προηγούμενου) ορισμού του πίνακα παρονομαστή

Μετά τον ορισμό των δυνάμεων των γραμμών του πίνακα παρονομαστή, συνεχίζουμε το παράδειγμα με τον ορισμό των στοιχείων του πίνακα, ο οποίος επιλέγεται να είναι διαγώνιος, επομένως αρκεί ο ορισμός της κύριας διαγωνίου του. Σύμφωνα με τα προηγούμενα, το (1,1) στοιχείο του πίνακα θα έχει βαθμό  $\xi_1 + k_1 = 3$ . Ομοίως, το (2,2) στοιχείο του πίνακα θα έχει βαθμό  $\xi_2 + k_2 = 3$ . Επιλέγουμε τις 3 ρίζες του στοιχείου (1,1) να είναι  $-2 \pm 2i$ ,  $-4$  και τις 3 ρίζες του στοιχείου (2,2) να είναι  $-1$ ,  $-6$ ,  $-8$ .

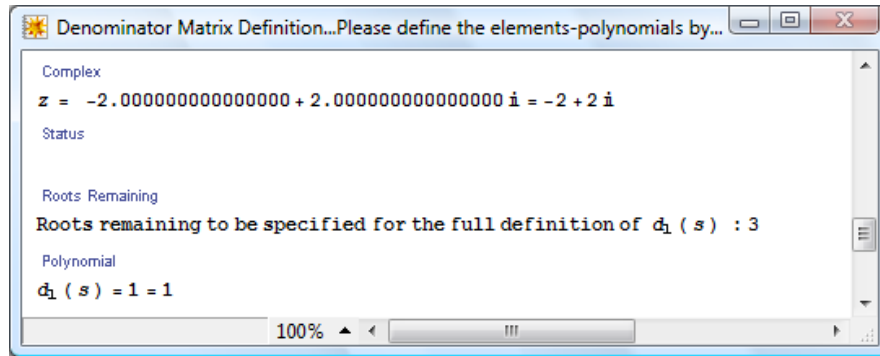
Επομένως, πατώντας το πλήκτρο “Define Denominator Matrix” εμφανίζονται τα παράθυρα της εικόνας 3.12. Τα πεδία “Polynomial” και “Roots remaining” περιέχουν τις πληροφορίες ότι το τρέχον πολώνυμο είναι το πρώτο στοιχείο της διαγωνίου του πίνακα παρονομαστή και ότι υπολείπονται οι ορισμοί 3 ριζών για τον πλήρη προσδιορισμό του στοιχείου (εικόνα 3.13).



Εικόνα 3.13

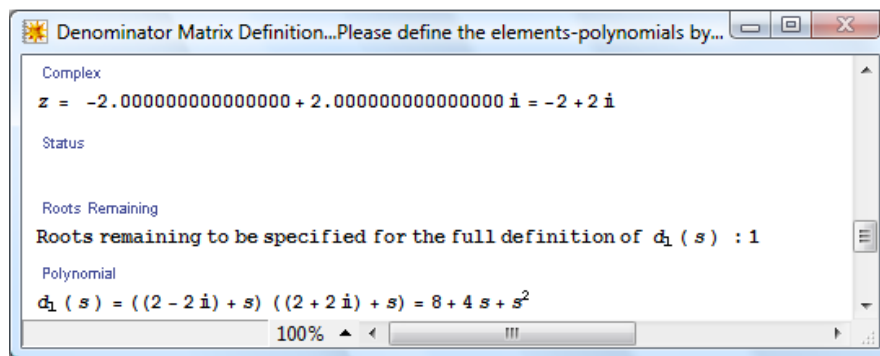
Πατώντας το πλήκτρο “Define Root (keyboard)” εμφανίζεται παράθυρο διαλόγου μέσα στο οποίο πληκτρολογούμε  $-2+2i$ . Πατώντας “OK” ο αριθμός επιλέγεται και εμφανίζεται στο πεδίο “Complex” (εικόνα 3.14).





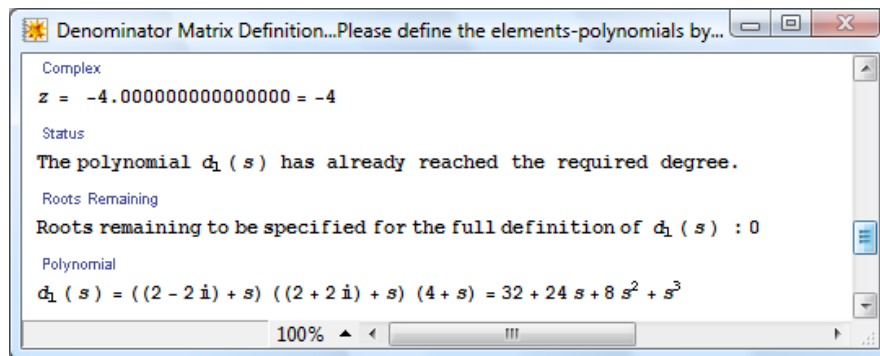
Εικόνα 3.14

Για να αναθέσουμε τον επιλεγμένο αριθμό ως ρίζα στο πολυώνυμο (1,1) πατάμε στο πλήκτρο “z --> Polynomial Root”. Αυτομάτως, γίνεται ταυτόχρονη ανάθεση της ρίζας και της συζυγούς της και ανανεώνονται τα πεδία “Polynomial” και “Roots remaining”, όπως στην εικόνα 3.15



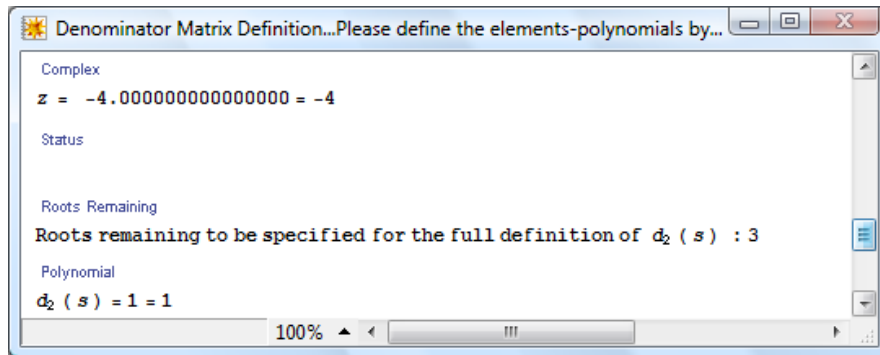
Εικόνα 3.15

Συνεχίζουμε με την εισαγωγή του -4 ως επόμενης ρίζας του πολυωνύμου (1,1). Ακολουθώντας τα ίδια βήματα όπως και για την προηγούμενη (“Define Root (keyboard)” / “z --> Polynomial Root”), το πρώτο πολυώνυμο θα έχει οριστεί πλήρως (εικόνα 3.16)



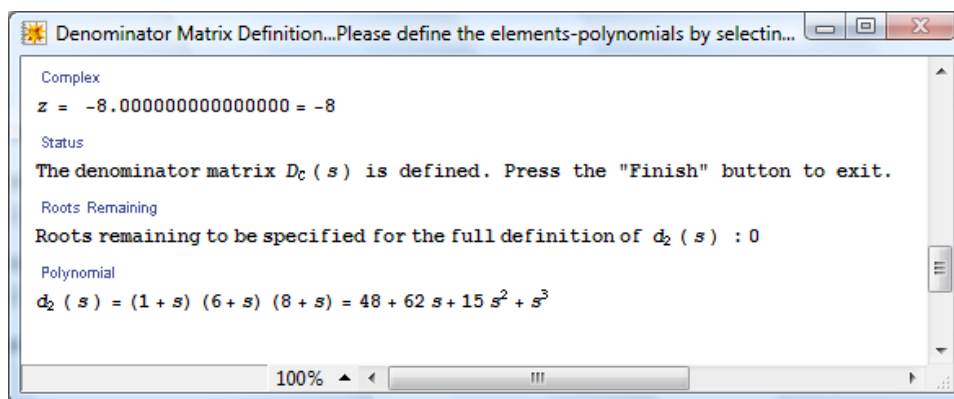
Εικόνα 3.16

Προχωρούμε στον ορισμό του πολυωνύμου (2,2) του πίνακα παρονομαστή. Επιλέγουμε το δεύτερο πολυώνυμο πατώντας στο “*Next Polynomial >>*”. Αμέσως εμφανίζονται στα αντίστοιχα πεδία οι σχετικές πληροφορίες για το δεύτερο πολυώνυμο. Σύμφωνα με αυτές, για τον υπολογισμό του δεύτερου πολυωνύμου, υπολείπονται 3 ρίζες (εικόνα 3.17).



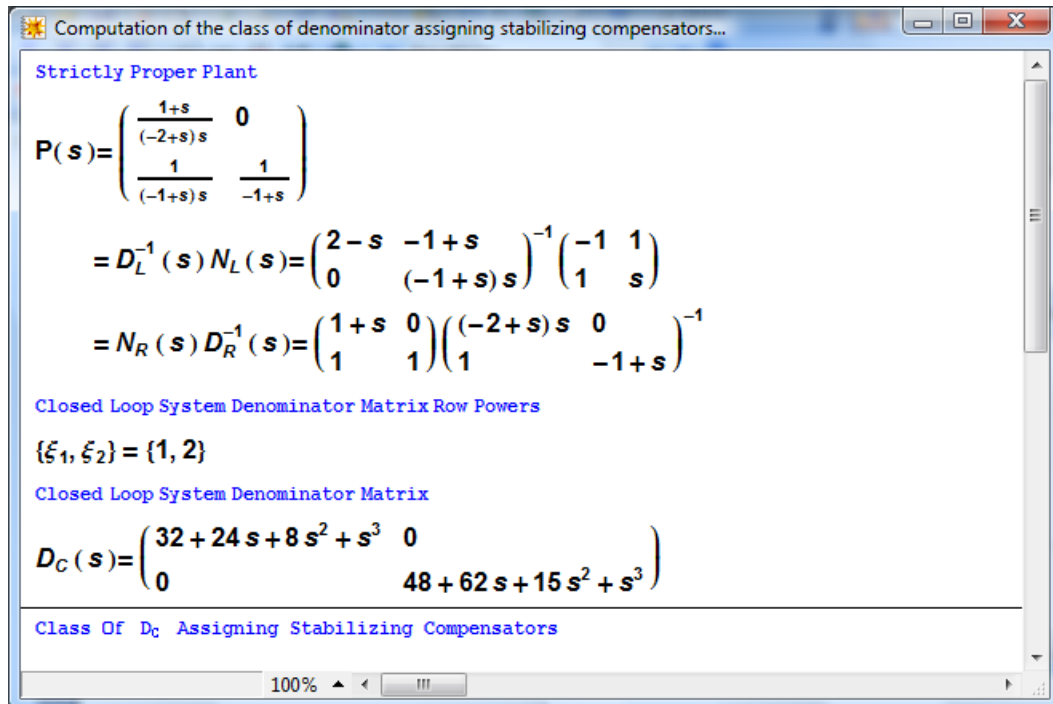
Εικόνα 3.17

Επαναλαμβάνοντας τα βήματα της διαδικασίας εισαγωγής – ανάθεσης των ριζών (“*Define Root (keyboard)*” / “*z --> Polynomial Root*”) για τις επιθυμητές τιμές -1, -6, -8, στο πολυωνυμικό στοιχείο (2,2) του πίνακα παρονομαστή, ολοκληρώνεται η διαδικασία του ορισμού του πίνακα (εικόνα 3.18).



Εικόνα 3.18

Πατώντας το πλήκτρο “*Finish*” επιστρέφουμε στα δύο κύρια παράθυρα της εφαρμογής. Στο πεδίο “*Closed Loop System Denominator Matrix*” εμφανίζεται ο πίνακας παρονομαστής με τις επιθυμητές ρίζες (εικόνα 3.19) ■



Εικόνα 3.19

Το τελευταίο βήμα της διαδικασίας, ο υπολογισμός, δηλαδή, της οικογένειας των αντισταθμιστών με κανονικές συναρτήσεις μεταφοράς, πραγματοποιείται με το πλήκτρο “Compute Compensators”.

**Παράδειγμα (συνέχεια προηγούμενου) υπολογισμού οικογένειας αντισταθμιστών**

Πατώντας “Compute Compensators” υπολογίζονται οι αντισταθμιστές υπό τη μορφή παραμετρικής οικογένειας και το αποτέλεσμα εμφανίζεται στο πεδίο “Class of  $D_c$  Assigning Stabilizing Compensators” (εικόνα 3.20) ■

Computation of the class of denominator assigning stabilizing compensators...

Strictly Proper Plant

$$P(s) = \begin{pmatrix} \frac{1+s}{(-2+s)s} & 0 \\ \frac{1}{(-1+s)s} & \frac{1}{-1+s} \end{pmatrix}$$

$$= D_L^{-1}(s) N_L(s) = \begin{pmatrix} 2-s & -1+s \\ 0 & (-1+s)s \end{pmatrix}^{-1} \begin{pmatrix} -1 & 1 \\ 1 & s \end{pmatrix}$$

$$= N_R(s) D_R^{-1}(s) = \begin{pmatrix} 1+s & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} (-2+s)s & 0 \\ 1 & -1+s \end{pmatrix}^{-1}$$

Closed Loop System Denominator Matrix Row Powers

$$\{\xi_1, \xi_2\} = \{1, 2\}$$

Closed Loop System Denominator Matrix

$$D_C(s) = \begin{pmatrix} 32 + 24s + 8s^2 + s^3 & 0 \\ 0 & 48 + 62s + 15s^2 + s^3 \end{pmatrix}$$

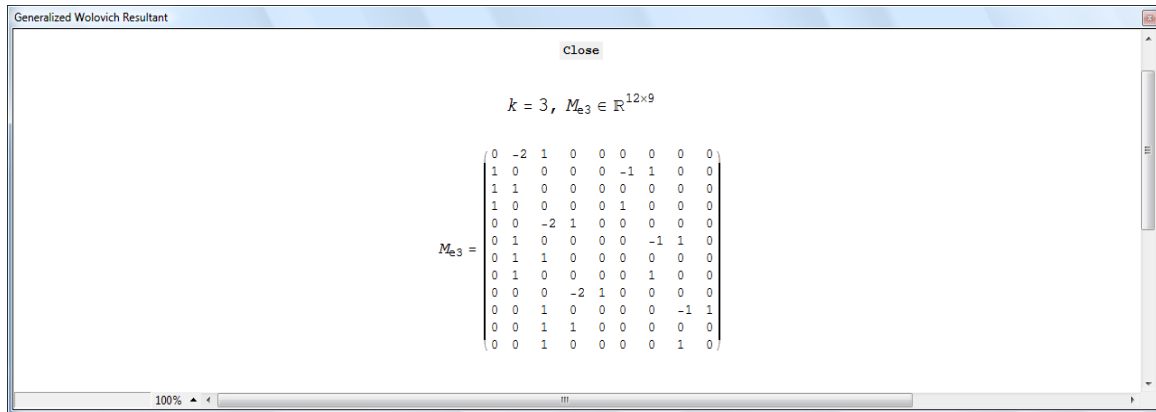
Class Of  $D_C$  Assigning Stabilizing Compensators

$$C(s) = \begin{pmatrix} 6 + s + t_1 & -t_1 \\ -63 - t_3 - s t_3 + t_4 & 78 + s^2 + s(16 - t_2) - t_2 - t_4 \end{pmatrix}^{-1}$$

$$\begin{pmatrix} 32 + s(4 - t_1) + 2 t_1 & -t_1 + s t_1 \\ -204 + 2 t_2 + s^2 t_3 + s(62 - t_2 - 2 t_3 - t_4) + 2 t_4 & 126 - t_2 + s^2 t_2 - t_4 + s t_4 \end{pmatrix}$$

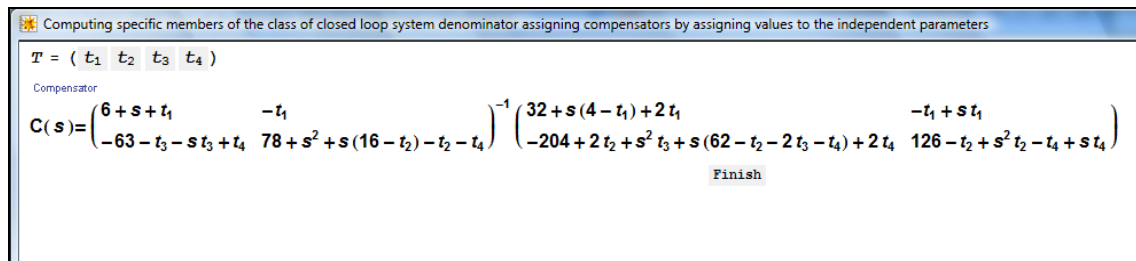
Εικόνα 3.20

Στο παράθυρο με τίτλο “Basic Buttons” συμπεριλαμβάνεται και το πλήκτρο “Display Resultant”. Η χρήση του έχει πληροφοριακό χαρακτήρα και μπορεί να χρησιμοποιηθεί σε οποιαδήποτε φάση μετά τον ορισμό του των δυνάμεων των γραμμών  $(\xi_i, i \in \mathbf{m})$  του πίνακα παρονομαστή. Εμφανίζει την γενικευμένη Wolovich resultant με την μέγιστη απαιτούμενη τάξη (δηλαδή τάξης  $\xi + 1$ ,  $\xi := \max_{i \in \mathbf{m}} \{\xi_i\}$ ) για την επίλυση των εξισώσεων (2.12). Στο παράδειγμά μας, πατώντας το πλήκτρο “Display Resultant” θα εμφανιζόταν το παράθυρο της εικόνας 3.21. Στη συνέχεια πατώντας “Close”, το παράθυρο θα έκλεινε και θα επιστρέφαμε στα δύο κύρια παράθυρα της εφαρμογής.



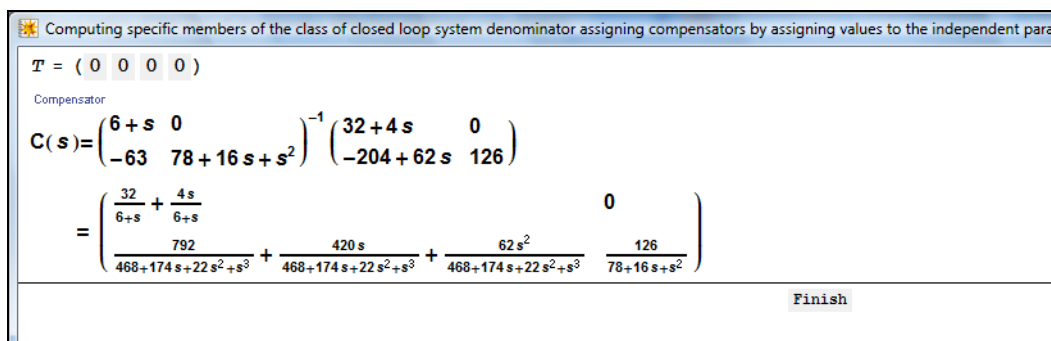
Εικόνα 3.21

Επίσης, μετά τον υπολογισμό της οικογένειας των αντισταθμιστών, πατώντας το πλήκτρο “Parameters Values”, μπορούμε να υπολογίσουμε συγκεκριμένους αντισταθμιστές, δίνοντας τιμές στις ελεύθερες παραμέτρους που ενδεχομένως περιέχονται στην έκφραση της οικογένειας. Στο παράδειγμα μας, πατώντας το “Parameters Values” εμφανίζεται το παράθυρο της εικόνας 3.22.



Εικόνα 3.22

Δίνοντας τις τιμές ‘0’, ‘0’, ‘0’, ‘0’ στα  $t_1, t_2, t_3, t_4$  με τρόπο ανάλογο όπως στα προηγούμενα, λαμβάνουμε τον αντισταθμιστή της εικόνας 3.23.



Εικόνα 3.23

Πατώντας “Finish” επιστρέφουμε στα δύο βασικά παράθυρα της εφαρμογής.

Η έξοδος από την εφαρμογή πραγματοποιείται με το πλήκτρο “Exit”, μέσα στο παράθυρο με τίτλο “Basic Buttons”.

Κλείνοντας την παράγραφο, σημειώνεται ότι όλα τα παραπάνω λειτουργούν και στην περίπτωση συστημάτων μίας εισόδου – μίας εξόδου, με την προϋπόθεση ότι κατά την εισαγωγή του αρχικού συστήματος η συνάρτηση μεταφοράς θα εισαχθεί με μορφή πίνακα μίας γραμμής – μίας στήλης και όχι ως βαθμωτό μέγεθος.

**ΒΙΒΛΙΟΓΡΑΦΙΑ**

- Antoniou, E. N. & Vardulakis, A. I. G. (2005). On the computation and parametrization of proper denominator assigning compensators for strictly proper plants, *IMA J. Math. Control Inform.* **22**, no. 1, 12-25
- Callier, F.M. & Desoer, C. A. (1982). *Multivariable Feedback Systems*. New York: Springer
- Kailath T. (1980). *Linear Systems*. Englewood Cliffs, NJ: Prentice Hall
- Kucera, V. & Zagalak, P. (1999), Proper solutions of polynomial equations, *Proc. 14<sup>th</sup> World Congress of IFAC* (London), pp. 357-362.
- Vardulakis, A. I. G. (1991) *Linear Multivariable Control – Algebraic Analysis and Synthesis Methods*. New York: Wiley





## ΠΑΡΑΡΤΗΜΑ

## Πλήρης Κώδικας Εφαρμογής

```
(*INITIALIZATION CELL*)

BeginPackage["Compensators`"]

Off[General::"spell1"]
Off [General::"spell"]
InitializationNB=SelectedNotebook[];

(*GENERAL MATRIX MANIPULATION FUNCTIONS*)

MatElLCM[x_]:=
(
  Apply[PolynomialLCM,Flatten[x]]
)

MatElGCD[x_]:=
(
  Apply[PolynomialGCD,Flatten[x]]
)

Rank[x_] :=Last[Dimensions[x]]-Length[NullSpace[x]]

AddZeroBlocks[x_,y_,z_]:=Module[{d,X,TX,TPX,AX},
(
  d=Dimensions[x];
  X=x;
  Do[X[[i]]=PadRight[x[[i]],d[[2]]+z,{i,1,d[[1]]}];
  TX=Transpose[X];
  TPX=TX;
  Do[TPX[[i]]=PadRight[TPX[[i]],d[[1]]+y,{i,1,d[[2]]+z}];
  AX=Transpose[TPX]
)]

SmithForm[x_]:=Module[{r,dim,m,mn,Sm0,Sm},
(
  r=Rank[x];

dim=Dimensions[x];m=Range[r];Do[m[[1]]=MatElGCD[Minors[x,1]],{1,1,r}];
mn=m;Do[mn[[b]]=mn[[b]]/m[[b-1]],{b,2,r}];
Sm0=DiagonalMatrix[mn];
Sm=AddZeroBlocks[Sm0,dim[[1]]-r,dim[[2]]-r];
Factor[Sm]
)]

McMillanForm[x_]:=Module[{DenMat,denGCD,NS,SN,MX1,MX2,MX},
(
  DenMat=Denominator[x];
  denGCD=MatElGCD[DenMat];
  NS=denGCD x;
  SN=SmithForm[NS];
  MX1=SN/denGCD;
  MX2=Cancel[MX1];
  MX=Factor[MX2]
)
```

```

    )]

DeleteLastBlocks[x_,p_,m_]:=Module[{DimX,R1,R2,X1,X2},
(
  DimX=Dimensions[x];
  R1=Partition[Range[DimX[[2]]+1-m,DimX[[2]]],1];
  R2=Partition[Range[DimX[[1]]+1-p,DimX[[1]]],1];
  X1=Delete[Transpose[x],R1];
  X2=Delete[Transpose[X1],R2]
)]

VerticalBlockMatrix[x_List]:=Module[{nb,AllMatricesQ,Columns},
(
  nb=Length[x];
  If[AllMatricesQ=
    Apply[And,Map[MatrixQ,x]],(If[
      Apply[Equal,
Map[Dimensions,x][[All,2]],(Columns=Dimensions[x[[1]]][[2]];
      Partition[Flatten[x],Columns]),
      Message[VerticalBlockMatrix::dims,x]],
      Message[VerticalBlockMatrix::arg,x]
    )]
  VerticalBlockMatrix::dims:="Elements of `1` consist of different number
of columns"

  VerticalBlockMatrix::arg:="`1` is not a list of matrices"

HorizontalBlockMatrix[x_List]:=Module[{AllMatricesQ},
(
  If[AllMatricesQ=
    Apply[And,Map[MatrixQ,x]],(If[
Apply[Equal,Map[Dimensions,x][[All,1]],Map[Flatten,Transpose[x]],
      Message[HorizontalBlockMatrix::dims,x]],
      Message[HorizontalBlockMatrix::arg,x]
    )]
  HorizontalBlockMatrix::dims:="Elements of `1` consist of different number
of rows"

  HorizontalBlockMatrix::arg:="`1` is not a list of matrices"

DegDifference[x_,var_]:=
  Exponent[Numerator[x],var]-Exponent[Denominator[x],var]

MinFiniteDegreeDiff[x_List,var1_]:=Module[{MDD},
(
  MDD=\[Infinity];
  Do[If[DegDifference[Flatten[x][[k]],var1]!=\[Infinity],
    MDD=Min[MDD,DegDifference[Flatten[x][[k]],var1]],{k,1,
    Length[Flatten[x]]}];
  MDD
)]

ColumnPermutation[x_,y_,z_]:=Module[{IPerm},
(
  IPerm=IdentityMatrix[Dimensions[x][[-1]]];
  IPerm[[y,y]]=0;
  IPerm[[z,z]]=0;
  IPerm[[y,z]]=1;

```

```

    IPerm[[z,y]]=1;
    {x.IPerm,IPerm}
  )]

PolynomialReduceQuotients[n_List,d_,var_]:=Module[{qlist0},
  (
    qlist0=n;

Do[qlist0[[i]]=PolynomialQuotient[n[[i]],d,var],{i,1,Length[n]}];qlist0
  )]

PRQuotient[pr1_,pr2_,var_]:=
  Module[{n1,d1,n2,d2},(n1=Numerator[pr1];d1=Denominator[pr1];
    n2=Numerator[pr2];d2=Denominator[pr2];
    PolynomialQuotient[n1*d2,n2*d1,var])]

PRRemainder[pr1_,pr2_,var_]:=
  Module[{n1,d1,n2,d2},(n1=Numerator[pr1];d1=Denominator[pr1];
    n2=Numerator[pr2];d2=Denominator[pr2];
    FullSimplify[PolynomialRemainder[n1*d2,n2*d1,var]/(d1*d2)])]

PRReduceQuotients[n_List,d_,var_]:=Map[PRQuotient[#,d,var]&,n]

RowDegreeReduction[x_,r1_,c1_,v_]:=
  Module[{rn, TM, med, cpos, PM, TX, RP, RPQ, IDX, ITX},
  (
    rn=x[[r1,Range[c1,Dimensions[x][[-1]]]];
    TM=IdentityMatrix[Dimensions[x][[-1]]];
    med=MinFiniteDegreeDiff[rn,v];
    cpos=Flatten[Position[DegDifference[rn,v],med][[1]]]+c1-1;
    PM=ColumnPermutation[x,c1,cpos];
    TX=PM[[1]];
    TM=TM.PM[[2]];
    RP=TX[[r1,Range[c1+1,Dimensions[TX][[-1]]]];
    RPQ=PRReduceQuotients[RP,TX[[r1,c1]],v];
    IDX=IdentityMatrix[Dimensions[TX][[-1]]];
    ITX=IDX;
    ITX[[r1,Range[c1+1,Dimensions[TX][[-1]]]]]=-RPQ;
    TM=TM.ITX;
    Factor[Simplify[{x.TM, TM}]]
  )]

ZeroVector[n_]:=0*Range[1,n]

PolynomialMatrixColumnRealCoefficientSimplify[m_,vr1_]:=Module[{cc,IT0},
  (
    cc=CoefficientList[Transpose[m],vr1]/.{}->0;
    IT0=IdentityMatrix[Dimensions[m][[-1]]];
    Do[If[MatElGCD[cc[[g]]]!=0,IT0[[g,g]]=1/MatElGCD[cc[[g]]],
      IT0[[g,g]]=1],{g,1,Dimensions[m][[-1]]}];
    Simplify[{m.IT0,IT0}]
  )]

RationalMatrixColumnRealCoefficientSimplify[m_List,var_]:=
  Module[{NumeratorReductionMatrix,DenominatorReductionMatrix,IT0,
    RightTransformationMatrix,
    cc},(NumeratorReductionMatrix=

PolynomialMatrixColumnRealCoefficientSimplify[Numerator[m],var][[2]];
  DenominatorReductionMatrix=(cc=

```

```

Map[Part[#, -1] &,
  CoefficientList[Transpose[Denominator[m]],
    var]/.(0->1), {2}];
IT0=IdentityMatrix[Dimensions[m][[-1]]];
Do[IT0[[g,g]]=1/MatElLCM[cc[[g]], {g,1,Dimensions[m][[-1]]}];
IT0/.0->1];
RightTransformationMatrix=
  DiagonalMatrix[
    Table[NumeratorReductionMatrix[[i,i]]/
      DenominatorReductionMatrix[[i,i]], {i,Length[IT0]}]];
Factor[
Simplify[{m.RightTransformationMatrix,RightTransformationMatrix}]]]

LeftStructureMatrix[RationalMatrix_List, var_]:=
Module[{StructureMatrixLeft, ColumnTransformationMatrix,
  ElementaryTransformationMatrix},
(
  StructureMatrixLeft=RationalMatrix;
  ColumnTransformationMatrix=
    IdentityMatrix[Dimensions[RationalMatrix][[2]]];
  Do[While[
    StructureMatrixLeft[[i,
      Range[i+1,
        Dimensions[
StructureMatrixLeft][[2]]]]]/(StructureMatrixLeft[[i,
      Range[i+1,Dimensions[StructureMatrixLeft][[2]]]])/.
      0->1) !=
    ZeroVector[
      Dimensions[StructureMatrixLeft][[2]]-
i], {StructureMatrixLeft,
      ElementaryTransformationMatrix}=
      RowDegreeReduction[StructureMatrixLeft, i, i, var];
      ColumnTransformationMatrix=
ColumnTransformationMatrix.ElementaryTransformationMatrix}, {i,1,
      Length[RationalMatrix]}];
{StructureMatrixLeft,ElementaryTransformationMatrix}=
RationalMatrixColumnRealCoefficientSimplify[StructureMatrixLeft, var];
ColumnTransformationMatrix=
  ColumnTransformationMatrix.ElementaryTransformationMatrix;
Factor[Simplify[{StructureMatrixLeft,ColumnTransformationMatrix}]]
)]

RightStructureMatrix[X_, v0_]:=Module[{TSX, TXLSM, RSM, LTRMAT},
(
  TSX=Transpose[X];
  TXLSM=LeftStructureMatrix[TSX, v0];
  RSM=Transpose[TXLSM[[1]]];
  LTRMAT=Transpose[TXLSM[[2]]];
  {RSM, LTRMAT}
)]

PMLeftGCD[x_List, var_]:=Module[{T0, T1, T2, LeftGCD},
(
  T0=HorizontalBlockMatrix[x];
  T1=LeftStructureMatrix[T0, var][[1]];
  T2=Transpose[T1];

```

```

LeftGCD=Transpose[T2[[Range[1,Dimensions[T1][[1]]]]]]
)]

PMRightGCD[x_List,var_]:=Module[{T0,T1,RightGCD},
(
T0=VerticalBlockMatrix[x];
T1=RightStructureMatrix[T0,var][[1]];
RightGCD=T1[[Range[1,Dimensions[T1][[-1]]]]]]
)]

SimplifyRightMFDCoefficients[nom_List,den_List,var_]:=
Module[{MFD,SimpleMFD,MFDTransformationMatrix,SimpleNR,SimpleDR,M,TRANSMAT,
DenominatorElementsHiCoef,NegC,PosC},
(
MFD=Join[nom,den];
{SimpleMFD,MFDTransformationMatrix}=
RationalMatrixColumnRealCoefficientSimplify[MFD,var];
SimpleNR=nom;
SimpleDR=den;
Do[If[i<=Length[nom],
SimpleNR=ReplacePart[SimpleNR,SimpleMFD[[i]],i],
SimpleDR=ReplacePart[SimpleDR,SimpleMFD[[i]],i-
Length[nom]]],{i,
Length[MFD]}];
M=HighestRowDegreeCoefficientMatrix[SimpleDR,var];
If[Det[M]!=0,
TRANSMAT=LinearSolve[M,IdentityMatrix[Length[M]]];
SimpleDR=Simplify[SimpleDR.TRANSMAT];
SimpleNR=Factor[Simplify[SimpleNR.TRANSMAT]];
{SimpleDR,TRANSMAT}=LeftStructureMatrix[SimpleDR,var];
SimpleNR=Factor[Simplify[SimpleNR.TRANSMAT]];
DenominatorElementsHiCoef=
Partition[
MapThread[
Coefficient,{Flatten[var*SimpleDR],
var^Flatten[Exponent[SimpleDR*var,var]/.-Infinity->1]}],
Dimensions[SimpleDR][[2]]];
NegC=Map[Count[#,x_/;x<0]&,Transpose[DenominatorElementsHiCoef]];
PosC=Map[Count[#,x_/;x>0]&,Transpose[DenominatorElementsHiCoef]];
TRANSMAT=DiagonalMatrix[Map[If[#<0,-1,1]&,PosC-NegC]];
{Factor[Simplify[SimpleNR.TRANSMAT]],
Factor[Simplify[SimpleDR.TRANSMAT]]}
)]

SimplifyLeftMFDCoefficients[den_List,nom_List,var_]:=
Module[{MFD,SimpleMFD,MFDTransformationMatrix,SimpleNL,SimpleDL,M,
TrTRANSMAT,TRANSMAT,DenominatorElementsHiCoef,NegC,PosC},
(
MFD=Join[Transpose[den],Transpose[nom]];
{SimpleMFD,MFDTransformationMatrix}=
RationalMatrixColumnRealCoefficientSimplify[MFD,var];
SimpleDL=Transpose[den];
SimpleNL=Transpose[nom];
Do[If[i<=Length[SimpleDL],
SimpleDL=ReplacePart[SimpleDL,SimpleMFD[[i]],i],
SimpleNL=
ReplacePart[SimpleNL,SimpleMFD[[i]],i-Length[SimpleDL]],{i,
Length[MFD]}];

```

```

{SimpleDL, SimpleNL}={Transpose[SimpleDL], Transpose[SimpleNL]};
M=HighestColumnDegreeCoefficientMatrix[SimpleDL];
If [Det[M]!=0,
  TrTRANSMAT=LinearSolve[Transpose[M], IdentityMatrix[Length[M]]],
  TRANSMAT=Transpose[TrTRANSMAT];
  SimpleDL=Factor[Simplify[TRANSMAT.SimpleDL]];
  SimpleNL=Factor[Simplify[TRANSMAT.SimpleNL]];
{SimpleDL, TRANSMAT}=RightStructureMatrix[SimpleDL, var];
SimpleNL=Factor[Simplify[TRANSMAT.SimpleNL]];
DenominatorElementsHiCoef=
  Partition[
    MapThread[
      Coefficient, {Flatten[var*SimpleDL],
        var^Flatten[Exponent[SimpleDL*var, var]/.-Infinity->1]}],
      Dimensions[SimpleDL][[2]]];
NegC=Map[Count[#, x_/;x<0]&, DenominatorElementsHiCoef];
PosC=Map[Count[#, x_/;x>0]&, DenominatorElementsHiCoef];
TRANSMAT=DiagonalMatrix[Map[If[#<0, -1, 1]&, PosC-NegC]];
{Factor[Simplify[TRANSMAT.SimpleDL]],
  Factor[Simplify[TRANSMAT.SimpleNL]]}
  ]
]

RightMFD[x_]:=Module[{dims, DX, TDX, V, B0},
  (
    dims=Dimensions[x];
    DX=Denominator[x];
    TDX=Transpose[DX];
    V=Range[1, dims[[-1]]];
    Do[V[[i]]=MatElLCM[TDX[[i]], {i, 1, dims[[-1]]}];
    B0=DiagonalMatrix[V];
    Factor[Simplify[{x.B0, B0}]]
  )
]

RightCoprimeMFD[x_, var_]:=Module[{PM1, PM2, rightgcd},
  (
    {PM1, PM2}=RightMFD[x];
    rightgcd=PMRightGCD[{PM1, PM2}, var];
    If[NumberQ[
      Det[rightgcd]/.ConstantsList->
        Range[1, Length[ConstantsList]], {PM1, PM2},
      PM1=PM1.Inverse[rightgcd]; PM2=PM2.Inverse[rightgcd]];
    {PM1, PM2}=
      SimplifyRightMFDCoefficients[Factor[Simplify[PM1]],
        Factor[Simplify[PM2]], var]
  )
]

LeftMFD[x_]:=Module[{dims, DX, LeftDenMat},
  (
    dims=Dimensions[x];
    DX=Denominator[x];
    LeftDenMat=Range[1, dims[[1]]];
    Do[LeftDenMat[[i]]=MatElLCM[DX[[i]], {i, 1, dims[[1]]}];
    LeftDenMat=DiagonalMatrix[LeftDenMat];
    Factor[Simplify[{LeftDenMat, LeftDenMat.x}]]
  )
]

LeftCoprimeMFD[x_, var_]:=Module[{PM1, PM2, leftgcd},
  (
    {PM1, PM2}=LeftMFD[x];

```

```

leftgcd=PMLeftGCD[{PM1 , PM2}, var];
If[NumberQ[
  Det[leftgcd].ConstantsList->
  Range[1, Length[ConstantsList]], {PM1, PM2},
  PM1=Inverse[leftgcd].PM1; PM2=Inverse[leftgcd].PM2];
{PM1, PM2}=
  SimplifyLeftMFDCoefficients[Factor[Simplify[PM1]],
  Factor[Simplify[PM2]], var]
)]

HighestRowDegreeCoefficientMatrix[mat_List, var_]:=Module[{MatHR},
(
  MatHR=mat;
  Do[MatHR[[i]]=
    PolynomialReduceQuotients[mat[[i]],
    var^(Max[Exponent[mat[[i]], var]]//.-Infinity->0), var], {i, 1,
    Dimensions[mat][[1]]}];
  MatHR
)]

HighestColumnDegreeCoefficientMatrix[mat_List, var_]:=Module[{TMatHC},
(
  TMatHC=HighestRowDegreeCoefficientMatrix[Transpose[mat], var];
  Transpose[TMatHC]
)]

ColumnProperQ[mat_List, var_]:=
Module[{numrows, numcolumns, MatrixDegree, columndegrees, ColumnComplexity},
({numrows, numcolumns}=Dimensions[mat];
  If[numrows>numcolumns,
    MatrixDegree=Max[Exponent[Flatten[Minors[mat, Rank[mat]]], var]],
    If[Rank[mat]==numcolumns,
      MatrixDegree=Max[Exponent[Det[mat], var]],
      If[Rank[HighestColumnDegreeCoefficientMatrix[mat, var]]==
        Min[{numrows, numcolumns}], True, False]]];
  columndegrees=MatrixColumnDegrees[mat, var];
  ColumnComplexity=Apply[Plus, columndegrees];
  If[ColumnComplexity==MatrixDegree, True, False]
)]

RowDependenceCoefficients[MTRX_List]:=
Module[{MTRXRows, A, MProd, GeneralSol, SA, a},
(
  MTRXRows=Dimensions[MTRX][[1]];
  A=Array[a, MTRXRows];
  MProd=A.MTRX;
  Off[Solve::"svars"];
  GeneralSol=Solve[MProd==0, A];
  On[Solve::"svars"];
  SA=A/.GeneralSol;
  Do[SA=SA/.a[i]->1, {i, MTRXRows}];
  Flatten[SA]
)]

ColumnDependenceCoefficients[MTRX_List]:=
Module[{MTRXCols, A, MProd, GeneralSol, SA, a},
(
  MTRXCols=Dimensions[MTRX][[2]];
  A=Array[a, MTRXCols];

```

```

MProd=MTRX.Partition[A,1];
Off[Solve::"svars"];
GeneralSol=Solve[MProd==0,A];
On[Solve::"svars"];
SA=A/.GeneralSol;
Do[SA=SA/.a[i]->1,{i,MTRXCols}];
Flatten[SA]
)]

```

MatrixReductionToRowProper[M\_List,var\_]:=

```

Module[{MP,MNumRows,MNumCols,MLTrans,Mhr,AR,r0,rowdeg,rd,ARS,INR,RowPosr0,
,
MLr},
(
MP=M;
MNumRows=Dimensions[M][[1]];
MNumCols=Dimensions[M][[2]];
MLTrans=IdentityMatrix[MNumRows];
While[
Rank[Mhr=HighestRowDegreeCoefficientMatrix[MP,var]]!=
Min[MNumRows,MNumCols],
AR=
NullSpace[
Transpose[Mhr=HighestRowDegreeCoefficientMatrix[MP,var]][[1]];
r0=Max[Exponent[MP,var]];rowdeg=Array[rd,MNumRows];
Do[rowdeg[[i]]=Max[Exponent[MP[[i]],var]],{i,MNumRows}];
ARS=AR*var^(r0-rowdeg);INR=IdentityMatrix[MNumRows];
RowPosr0=Position[Expand[MP],var^r0][[1,1]];
MLr=ReplacePart[INR,ARS,RowPosr0];MLTrans=MLr.MLTrans;
MP=Factor[Simplify[MLr.MP]];
Factor[Simplify[{MP,MLTrans}]]
)]

```

MatrixReductionToColumnProper[M\_List,var\_]:=

```

Module[{MP,MNumRows,MNumCols,MRTrans,Mhc,AC,c0,coldeg,cd,ACS,INC,ColPosc0,
,
MRc},
(
MP=M;
MNumRows=Dimensions[M][[1]];
MNumCols=Dimensions[M][[2]];
MRTrans=IdentityMatrix[MNumCols];
While[
Rank[Mhc=HighestColumnDegreeCoefficientMatrix[MP,var]]!=
Min[MNumRows,MNumCols],
AC=NullSpace[Mhc=HighestColumnDegreeCoefficientMatrix[MP,var]][[1]];
c0=Max[Exponent[MP,var]];coldeg=Array[cd,MNumCols];
Do[coldeg[[i]]=Max[Exponent[Transpose[MP][[i]],var]],{i,MNumCols}];
ACS=AC*var^(c0-coldeg);INC=IdentityMatrix[MNumCols];
ColPosc0=Position[Expand[Transpose[MP]],var^c0][[1,1]];
MRc=Transpose[ReplacePart[INC,ACS,ColPosc0]];MRTrans=MRTrans.MRc;
MP=Factor[Simplify[MP.MRc]];
Factor[Simplify[{MP,MRTrans}]]
)]

```



```

ColumnProperDenominatorRightCoprimeMFD[Mat_List,var_]:=
Module[{NR,DR,DR1,DRTrans},
(
{NR,DR}=RightCoprimeMFD[Mat,var];
{DR1,DRTrans}=MatrixReductionToColumnProper[DR,var];
Factor[Simplify[{NR,DRTrans,DR1}]]
)]

RowProperDenominatorLeftCoprimeMFD[Mat_List,var_]:=
Module[{DL,NL,DL1,DLTrans},
(
{DL,NL}=LeftCoprimeMFD[Mat,var];
{DL1,DLTrans}=MatrixReductionToRowProper[DL,var];
Factor[Simplify[{DL1,DLTrans,NL}]]
)]

MatrixColumnDegrees[Mat_List,var_]:=Module[{NumCols,ColumnDegs,cdeg},
(
NumCols=Dimensions[Mat][[-1]];
ColumnDegs=Array[cdeg,NumCols];

Do[ColumnDegs[[i]]=Max[Exponent[Transpose[Expand[Mat]][[i]],var]],{i,
NumCols}];
ColumnDegs
)]

MatrixRowDegrees[Mat_List,var_]:=Module[{NumRows,RowDegs,rdeg},
(
NumRows=Dimensions[Mat][[1]];
RowDegs=Array[rdeg,NumRows];
Do[RowDegs[[i]]=Max[Exponent[Expand[Mat]][[i]],var]],{i,NumRows}];
RowDegs
)]

PowerVector[x_,p_]:=Module[{pv},
(
pv={1};
Do[pv=Append[pv,x^expi],{expi,1,p}];
pv
)]

(*ADDITIONAL FUNCTIONS APPLYING TO RATIONAL MATRICES*)

StrictlyProperRationalMatrixQ[mat_,var_]:=Module[{DD},
(
If[(TensorRank[mat]!=0&&Length[mat]!=0),
DD=DegDifference[mat,var];
TrueQ[Cases[Flatten[DD],n_/;n<0]==Flatten[DD]],
TrueQ[0==1]]
)]

(*PIECES NEEDED FOR THE COMPUTATION OF THE CLASS OF PROPER POLE PLACING \
STABILIZING COMPENSATORS FOR THE STRICTLY PROPER PLANT*)

(*UPDATED ALGORITHM FUNCTIONS*)

DenominatorElementsCoefficients[PolMat_List,var_,
ColumnUpToDegree_List:Map[Max,
Exponent[Transpose[PolMat],var];/VectorQ[ColumnUpToDegree]==True&&

```

```

Length[ColumnUpToDegree]==Dimensions[PolMat][[2]]]:=Module[{DK},
(
  DK=CoefficientList[PolMat,var];
  Do[DK[[All,ci]]=
    Map[Append[#,ZeroVector[1+ColumnUpToDegree[[ci]]-Length[#]]&,
      DK[[All,ci]]],{ci,Dimensions[PolMat][[2]]}];
  DK=Partition[Flatten[DK],
    Apply[Plus,ColumnUpToDegree]+Dimensions[PolMat][[2]]]
  )]

ComputeClassOfCompensators[Plant_,DenominatorMatrix_,var_]:=
Module[{Dleft,Nleft,Dright,Nright,PlantRows,PlantColumns,

DrightColumnDegrees,mi,DenominatorMatrixDegrees,Ksi,KsiMax,gwres,Dk,
MekDkM,ColReducedMekDkM,ji1,ji2,ws,Ws,
w},{Dleft,Nleft}=
  RowProperDenominatorLeftCoprimeMFD[Plant,var];{Nright,Dright}=
  ColumnProperDenominatorRightCoprimeMFD[Plant,var];{PlantRows,
  PlantColumns}=Dimensions[Plant];
  DrightColumnDegrees=Map[Max,Exponent[Transpose[Dright],var]];
  mi=Max[Flatten[Exponent[Dleft,var]]];
  DenominatorMatrixDegrees=Map[Max,Exponent[DenominatorMatrix,var]];
  Map[Set[Ksi[#],(DenominatorMatrixDegrees-
DrightColumnDegrees)[[#]]&,
  Range[PlantColumns]];KsiMax=Max[Array[Ksi,PlantColumns]];
  gwres=GWResultant[Dright,Nright,var,KsiMax+1];
  Dk=DenominatorElementsCoefficients[DenominatorMatrix,var,

DrightColumnDegrees+KsiMax];MekDkM=VerticalBlockMatrix[{gwres,Dk}];
  ColReducedMekDkM=Transpose[RowReduce[Transpose[MekDkM]]];
  Do[parsysmat[ji1]=
    Transpose[
      ColReducedMekDkM[[Range[(Length[Dleft]+
      Length[Dright]) (Ksi[ji1]+1)],
      Range[Dimensions[
      ColReducedMekDkM][[2]]-(Max[
      Array[Ksi,Length[Dright]]]-Ksi[ji1])Length[
      Dright]]]];nsp[ji1]=NullSpace[parsysmat[ji1]];
      nspd[ji1]=Length[nsp[ji1]];
      cumnumpar[ji1]=Apply[Plus,Map[nspd,Range[ji1]],{ji1,1,
      Length[Dright]}];cumnumpar[0]=0;Ws=Array[ws,Length[Dright]];
  Do[w[ji2]=
    LinearSolve[parsysmat[ji2],
      ColReducedMekDkM[[-Length[Dright]+ji2-1,
      Range[Dimensions[
      ColReducedMekDkM][[2]]-(Max[
      Array[Ksi,Length[Dright]]]-Ksi[ji2])Length[
      Dright]]]]]+
      Array[t,nspd[ji2],cumnumpar[ji2-1]+1].nsp[ji2],{ji2,1,
      Length[Dright]}];
  Do[ws[ji3]=
    Apply[Plus,
      Partition[w[ji3],Length[w[ji3]]/(Ksi[ji3]+1)]*
      PowerVector[var,Ksi[ji3]],{ji3,1,PlantColumns}];{XL,YL,

numofparameters}={Transpose[Transpose[Ws][[Range[PlantColumns]]],
  Transpose[

Transpose[Ws][[Range[PlantColumns+1,PlantColumns+PlantRows]]]],

```

```

cumnumpar [Length [Dright] ] } ) ]

PolynomialToList [poly_, pow_Integer /; pow >= 0, var_] :=
  Flatten [Append [CoefficientList [poly, var],
    ZeroVector [pow - Length [CoefficientList [poly, var]] + 1]]]

GWResultant [D_, N_, var_, k_Integer /; k >= 1] :=
  Module [ {F, degCF, MkfirstBlock, Mk},
    (
      F = VerticalBlockMatrix [ {D, N} ];
      degCF = Map [Max, Exponent [Transpose [F], var]] /. -Infinity -> 0;
      MkfirstBlock =
        Table [PolynomialToList [F [ [i, j] ], degCF [ [j] ] + k - 1, var], {i, 1,
          Dimensions [F] [ [1] ] }, {j, 1, Dimensions [F] [ [2] ] }];
      Mk = Partition [
        Flatten [NestList [Map [RotateRight [# , 1] & , #, {2}] & , MkfirstBlock, k -
1]],
        Dimensions [F] [ [2] ] * k + Apply [Plus, degCF] ]
    ) ]

GeneralizedRe [z_] := ComplexExpand [Re [z]]
GeneralizedIm [z_] := ComplexExpand [Im [z]]
GeneralizedConjugate [w_] := ComplexExpand [Conjugate [w]]

(* DEFINITION OF ALL PROCEDURES THAT LAUNCH DIALOG WINDOWS ASKING FOR DATA *)

(* The following is the definition of the process that asks for the \
dimensions of the plant matrix *)
GetPlantDimensions :=
  Module [ {dims}, (Off [Syntax :: "com"];
    dims = InputString [
      "Please define strictly proper plant dimensions. Input should be
of the form                                '{number of
rows, number of columns}' ." ] \
; If [SyntaxQ [dims], dims = ToExpression [dims]; On [Syntax :: "com"];
    If [ (VectorQ [dims] && Length [dims] == 2 &&

Apply [And, Map [IntegerQ, dims]] && Apply [And, Map [Positive, dims]]),
dims, GetPlantDimensions], On [Syntax :: "com"]; GetPlantDimensions]]

(*The dialog asking for the value of a specific row degree of the
denominator \
matrix*)
KsiDefinitionDialog [i_Integer, mi_Integer] :=
  Module [ {ksivalue}, (Off [Syntax :: "com"];
    ksivalue = InputString [
      StringForm ["Assign the `1` `2` row degree of the denominator
matrix",
        i, Switch [i, 1, "st", 2, "nd", 3, "rd", _Integer /; i > 3, "th"] ]];

If [SyntaxQ [ksivalue], ksivalue = ToExpression [ksivalue]; On [Syntax :: "com"];
  If [IntegerQ [ksivalue] && ksivalue >= mi - 1, ksivalue,
    KsiDefinitionDialog [i, mi]], On [Syntax :: "com"];
  KsiDefinitionDialog [i, mi]] ]

(* The following is the definition for the dialog box that asks for a
complex \
root*)

```

```

GetComplexRoot:=
Module[{comroot}, (Off[Syntax::"com"];
  comroot=InputString[
    "Please enter root. Input should be of the form ' a+b I ' (a<0
and b real number). NOTE: Input should not contain the quotation marks"]\
;If[SyntaxQ[comroot],comroot=ToExpression[comroot];On[Syntax::"com"];
  If[NumericQ[
    comroot//.Map[#->1&,Cases[comroot,_Symbol,Infinity]]],
  If[NumericQ[GeneralizedRe[comroot]]&&
    GeneralizedRe[comroot]>=0,GetComplexRoot,comroot],
  comroot],On[Syntax::"com"];GetComplexRoot)])

(* This defines the process that asks for the variable specification*)

GetVariableSymbol:=
Module[{varsym}, (Off[Syntax::"com"];
  varsym=
  InputString[
    "Specify the variable name (The symbol which should be
considered as the variable of the matrix...):"]\
;
  If[SyntaxQ[varsym],varsym=ToExpression[varsym];On[Syntax::"com"];
  If[Head[varsym]==Symbol,varsym,GetVariableSymbol],
  On[Syntax::"com"];GetVariableSymbol)])

(*This defines the process that asks for the definition of the plant*)

GetSPPlant:=
Module[{plant}, (Off[Syntax::"com"];
  plant=InputString["Please define the strictly proper plant:"];
  If[SyntaxQ[plant],plant=ToExpression[plant];On[Syntax::"com"];
  Rationalize[plant],GetSPPlant)])

(*MainDisplayNBObject INTERFACE FUNCTIONS AND OBJECTS*)
DisplayPlant[P_List,DL_List,NL_List,DR_List,
  NR_List]:= (NotebookLocate[{MainDisplayNBObject,"Strictly Proper
Plant"}]);
SelectionMove[MainDisplayNBObject,All,Cell];
NotebookWrite[MainDisplayNBObject,
  Cell[BoxData[
    RowBox[{ToBoxes["P ("],var,ToBoxes[")"]=\[NoBreak]"},
      ToBoxes[MatrixForm[P]], "\n          =",
    RowBox[{SubsuperscriptBox["D", "L", -1], ToBoxes["("], var,
      ToBoxes[")"], SubscriptBox["N", "L"], ToBoxes["("], var,
      ToBoxes[")"]=\[NoBreak]"}]],
    SuperscriptBox[ToBoxes[MatrixForm[DL]], -1],
    ToBoxes[MatrixForm[NL]], "\n          =",
  RowBox[{SubscriptBox["N", "R"], ToBoxes["("], var, ToBoxes[")"],
    SubsuperscriptBox["D", "R", -1], ToBoxes["("], var,
    ToBoxes[")"]}], ToBoxes[MatrixForm[NR]],
    SuperscriptBox[ToBoxes[MatrixForm[DR]], -1],
  1]]], "TraditionalForm",
  FontSize->16, FontFamily->"Arial", FontWeight->"Bold",
  CellTags->"Strictly Proper Plant", ShowCellTags->False,
  Selectable->True, CellHorizontalScrolling->True], After])

DisplayCompensators[xl_List,

```

```

    yl_List]:= (NotebookLocate[{MainDisplayNBObject,"Class Of
Compensators"}]];
    SelectionMove[MainDisplayNBObject,All,Cell];
    NotebookWrite[MainDisplayNBObject,
    Cell[BoxData[
        RowBox[{"C(",var,ToBoxes["]="],
            SuperscriptBox[ToBoxes[MatrixForm[xl]],-1],
            ToBoxes[MatrixForm[yl]]}], "TraditionalForm",
        FontSize -> 16, FontFamily -> "Arial", FontWeight -> "Bold",
        CellTags->"Class Of Compensators", ShowCellTags->False,
        Selectable->True, CellHorizontalScrolling->True], After]
(*This Function Constructs the pxm Plant Array*)

CreatePlantArray[p_Integer,m_Integer]:= (PlantB=Array[pl,{p,m}] );

(*This Function Creates the Button Representation of the {i,
j} element of the Plant Array*)
ButtonElement[i_Integer,j_Integer]:=
    ButtonBox[
        pl[i,j]/.pl[i,
            j]:>StyleForm[Subscript[p,StringForm["`1`,`2`",i,j]],
                FontColor->RGBColor[1,0,0]]/;Head[pl[i,j]]==pl,
        ButtonEvaluator->Automatic,Active->True,
        ButtonFunction:>(pl[i,j]=
            Input[StringForm["Please define the (`1`,`2`) element of the
plant.",
                i,j]]);NotebookLocate[
                "Define Plant"];
        NotebookWrite[SelectedNotebook[],
            Cell[BoxData[

RowBox[{"(",ToBoxes[ButtonElementsArray[dim1,dim2]],")"}]], "Outline2",
        CellTags-> "Define Plant", ShowCellTags->True,
        ShowCellBracket->False, CellMargins->{{1,1},{1,1}},
        CellHorizontalScrolling->True]]]

(*This Function Creates And Displays The Array Of the Buttons
Representing \
The Elements of The Plant Array*)

ButtonElementsArray[dim1_Integer,dim2_Integer]:=
    GridBox[Array[ButtonElement,{dim1,dim2}],ColumnSpacings->0,
        RowSpacings->0]//DisplayForm

(*This is the expression of the notebook for the definition of the plant
\
using buttons*)

DefinePlantUsingBoxesNB:=
    Notebook[{Cell[

BoxData[RowBox[{"(",ToBoxes[ButtonElementsArray[dim1,dim2]],")"}]], "Outli
ne2",
        CellTags-> "Define Plant", ShowCellTags->True,
        ShowCellBracket->False, CellMargins->{{1,1},{1,1}},
        CellHorizontalScrolling->True],
        Cell[BoxData[CloseDefinePlantUsingBoxesNBButton], "Outline2",
            ShowCellBracket->False, CellMargins->{{1,1},{1,1}},

```

```

CellFrame->True,CellFrameMargins->2,
TextAlignment->Center,
Background->Options[ButtonBox,Background][[1,2]]},
WindowFrame->"Palette",WindowTitle->"Please Define The Strictly
Proper Plant",
WindowSize->FitAll,WindowFloating->True,
WindowFrameElements->{"ResizeArea"},Visible->False,
ScrollingOptions->{PagewiseScrolling->Inherited,
PagewiseDisplay->Inherited,RealTimeScrolling->Inherited,
JumpScrolling->Inherited,HorizontalScrollRange-> 12000,
SpeedParameters->Inherited}};

```

(\*This is the definition of the button used to initiate the process of \ defining the plant using boxes\*)

```

DefinePlantB:=
ButtonBox["Define Plant (Boxes)",ButtonEvaluator->Automatic,
Active->True,
ButtonFunction:>(If[
Count[Flatten[Attributes[{dim1,dim2}]],Protected]==0,
SetOptions[MainButtonsPaletteObject,Editable->True,
Visible->False];
NotebookLocate[{MainDisplayNBObject,"Class Of Compensators"}];
SelectionMove[MainDisplayNBObject,All,Cell];
NotebookWrite[MainDisplayNBObject,
Cell["","Input",CellTags->"Class Of Compensators",
ShowCellTags->False,Selectable->True],After];
ComputedCompensator=False;{dim1,dim2}=GetPlantDimensions;
SetAttributes[{dim1,dim2},Protected];
PlantDefinitionNBObject=NotebookPut[DefinePlantUsingBoxesNB]];
NotebookCenter[PlantDefinitionNBObject];
SetOptions[PlantDefinitionNBObject,Visible->True]])

```

(\*This is the expression of the button that closes the "Definition of the plant using boxes"-

notebook and assigns the defined matrix as a strictly proper plant, if the former has been defined properly\*)

```

CloseDefinePlantUsingBoxesNBButton:=
ButtonBox["Finish",ButtonEvaluator->Automatic,Active->True,
ButtonFunction:>(CreatePlantArray[dim1,dim2];
SetOptions[ButtonNotebook[],WindowFloating->False];
If[MatrixQ[StrictlyProperPlant]==True,
LastValidPlant=StrictlyProperPlant;LastValidVar=var];
If[Length[Variables[PlantB]]==1,
PlantBvar=Variables[PlantB][[1]],PlantBvar=GetVariableSymbol];
StrictlyProperPlant=Rationalize[PlantB];var=PlantBvar;
StrictlyProperPlant=Factor[StrictlyProperPlant];
If[Equal[StrictlyProperRationalMatrixQ[StrictlyProperPlant,var],
True],(Set[AppropriatePlantData,True];KsiRowDegreesDefined=False;
If[Equal[Length[Dimensions[StrictlyProperPlant]],1],
Set[StrictlyProperPlant,Partition[StrictlyProperPlant,1]]];
Set[ConstantsList,
DeleteCases[Variables[StrictlyProperPlant],var]];
Set[{PlantRows,PlantColumns},Dimensions[StrictlyProperPlant]];
Set[{Dleft,Nleft},
RowProperDenominatorLeftCoprimeMFD[StrictlyProperPlant,var]]);

```

```

Set[{Nright, Dright},
  ColumnProperDenominatorRightCoprimeMFD[StrictlyProperPlant,
var]]; DisplayPlant[StrictlyProperPlant, Dleft, Nleft, Dright,
  Nright]; Set[DrightColumnDegrees,
  MatrixColumnDegrees[Dright, var]];
Set[DleftRowDegrees, MatrixRowDegrees[Dleft, var]];

Set[n, Apply[Plus, DleftRowDegrees]]; Set[mi, Max[DleftRowDegrees]];
AppropriateDenominatorMatrix=False;
NotebookLocate[{MainDisplayNBOBJECT,
  "Closed Loop System Denominator Matrix"}];
SelectionMove[MainDisplayNBOBJECT, All, Cell];
NotebookWrite[MainDisplayNBOBJECT,
  Cell[TextData[{"The plant has changed. The closed loop
system denominator matrix should be redefined."}\
], CellTags->"Closed Loop System Denominator Matrix", ShowCellTags->False,
  Selectable->True], After];
NotebookLocate[{MainDisplayNBOBJECT,
  "Closed Loop System Denominator Matrix Row Powers"}];
SelectionMove[MainDisplayNBOBJECT, All, Cell];
NotebookWrite[MainDisplayNBOBJECT,
  Cell[TextData[{"The plant has changed. The closed loop
system denominator matrix row powers should be redefined."}\
], CellTags->"Closed Loop System Denominator Matrix Row Powers",
  ShowCellTags->False, Selectable->True], After]],
If[StrictlyProperRationalMatrixQ[LastValidPlant,
  LastValidVar]==True, StrictlyProperPlant=LastValidPlant;
var=LastValidVar,
NotebookLocate[{MainDisplayNBOBJECT, "Strictly Proper
Plant"}]];
SelectionMove[MainDisplayNBOBJECT, All, Cell];
NotebookWrite[MainDisplayNBOBJECT,
  Cell[TextData[
  "The Plant Definition is Invalid. Please Redefine the
system's transfer matrix."]\
], CellTags->"Strictly Proper Plant", ShowCellTags->False, Selectable->True],
  After]]]; ClearAttributes[{dim1, dim2}, Protected];
NotebookClose[ButtonNotebook[]];
SetOptions[MainButtonsPaletteObject, Visible->True,
  WindowFloating->True, Editable->False]]

(*This is the definition of the button used to textually define the
plant*)

DefinePlantT:=
  ButtonBox[RowBox[{"Define", " ", "Plant", "(Standard)"}],
  ButtonFunction:>

(Off[MessageName[Syntax, "sntxi"]; Off[MessageName[Flatten, "normal"]];
  NotebookLocate[{MainDisplayNBOBJECT, "Class Of Compensators"}];
  SelectionMove[MainDisplayNBOBJECT, All, Cell];
  NotebookWrite[MainDisplayNBOBJECT,
    Cell["", "Input", CellTags->"Class Of Compensators",
      ShowCellTags->False, Selectable->True], After];
  ComputedCompensator=False;
  If[MatrixQ[StrictlyProperPlant]==True,
    LastValidPlant=StrictlyProperPlant; LastValidVar=var];
  Set[StrictlyProperPlant, GetSPPlant];
  If[Equal[Length[Variables[StrictlyProperPlant]], 1],

```

```

Set[var, Part[Variables[StrictlyProperPlant], 1]],
Set[var, GetVariableSymbol]];
Set[StrictlyProperPlant, Factor[StrictlyProperPlant]];
If[Equal[StrictlyProperRationalMatrixQ[StrictlyProperPlant, var],
True], (Set[AppropriatePlantData, True];
If[Equal[Length[Dimensions[StrictlyProperPlant]], 1],
Set[StrictlyProperPlant, Partition[StrictlyProperPlant, 1]]];
Set[ConstantsList,
DeleteCases[Variables[StrictlyProperPlant], var]];

Set[{PlantRows, PlantColumns}, Dimensions[StrictlyProperPlant]];
Set[{Dleft, Nleft},

RowProperDenominatorLeftCoprimeMFD[StrictlyProperPlant, var]];
Set[{Nright, Dright},
ColumnProperDenominatorRightCoprimeMFD[StrictlyProperPlant,

var]]; DisplayPlant[StrictlyProperPlant, Dleft, Nleft, Dright,
Nright]; Set[DrightColumnDegrees,
MatrixColumnDegrees[Dright, var]];
Set[DleftRowDegrees, MatrixRowDegrees[Dleft, var]];
Set[n, Apply[Plus, DleftRowDegrees]];
Set[mi, Max[DleftRowDegrees]] (*If[
MatrixQ[DenominatorMatrix]&&
Exponent[(Range[1, Dimensions[DenominatorMatrix][[1]])/
Range[1,
Dimensions[
DenominatorMatrix][[1]])].DenominatorMatrix,
var]==DrightColumnDegrees+mi-1,
AppropriateDenominatorMatrix=True, *]);

AppropriateDenominatorMatrix=False; KsiRowDegreesDefined=False;
NotebookLocate[{MainDisplayNBOobject,
"Closed Loop System Denominator Matrix"}];
SelectionMove[MainDisplayNBOobject, All, Cell];
NotebookWrite[MainDisplayNBOobject,
Cell[TextData[{"The plant has changed. The closed loop
system denominator matrix should be redefined."}\
], CellTags->"Closed Loop System Denominator Matrix", ShowCellTags->False,
Selectable->True], After] (* *)];
NotebookLocate[{MainDisplayNBOobject,
"Closed Loop System Denominator Matrix Row Powers"}];
SelectionMove[MainDisplayNBOobject, All, Cell];
NotebookWrite[MainDisplayNBOobject,
Cell[TextData[{"The plant has changed. The closed loop
system denominator matrix row powers values should be redefined."}\
], CellTags->"Closed Loop System Denominator Matrix Row Powers",
ShowCellTags->False, Selectable->True], After]],
If[StrictlyProperRationalMatrixQ[LastValidPlant,
LastValidVar]==True, StrictlyProperPlant=LastValidPlant;
var=LastValidVar,
NotebookLocate[{MainDisplayNBOobject, "Strictly Proper
Plant"}]];
SelectionMove[MainDisplayNBOobject, All, Cell];
NotebookWrite[MainDisplayNBOobject,
Cell[TextData[
"The Plant Definition is Invalid. Please Redefine the
system's transfer matrix."]\
], CellTags->"Strictly Proper Plant", ShowCellTags->False, Selectable->True],

```



```

        After]]];
    SetOptions[MainButtonsPaletteObject, Visible->True,
        WindowFloating->True, Editable->False]),
    ButtonEvaluator->Automatic, Active->True]

(*This is the definition of the notebook that is launched in order to
define \
the denominator matrix row powers*)

KsiDefinitionNB:=
    Notebook[{Cell[
        TextData[{"Provide the values of the row powers of the closed
loop system denominator matrix ",
            Cell[RowBox[SubscriptBox["D", "C"]], "(, var, )", ".", "\n Row
powers MUST NOT be less than  $\mu-1 =$ ",
            mi-1, "."}], TextAlignment->Center, ShowCellBracket->False,
        FontSize->16, FontWeight->"Bold"],
    Cell[
        BoxData[RowBox[{ToBoxes[
            Array["ξ", {PlantColumns}]/.
            "ξ"[i_]->Subscript["ξ", i]], ToBoxes["="],
            ToBoxes["{"]},
        ToBoxes[RowBox[Array[KsiDefinitionButton[#]&, {PlantColumns}]]//
            DisplayForm], ToBoxes["}"}]]], "TraditionalForm",
        FontSize->14, FontWeight->"Bold", FontFamily->"Arial",
        CellHorizontalScrolling->True, CellTags->"Closed Loop System
Denominator Matrix Row Powers",
        ShowCellTags->True, ShowCellBracket->False,
        Editable->False], Cell[TextData[""], ShowCellBracket->False],
        Cell[BoxData[ToBoxes[CloseKsiDefinitionNBObjectButton]],
            TextAlignment->Center, ShowCellBracket->False,
            Editable->False, FontWeight->"Bold",
            CellHorizontalScrolling->False]]]

(*Definition of the button unit that launches the dialog asking for the
value \
of a specific row power of the denominator matrix*)

KsiDefinitionButton[ri_Integer]:=
    ButtonBox[
        Ksi[ri]/.Ksi[ri]:>
            Subscript["ξ", ri]/;Head[Ksi[ri]]==Ksi, Active->True,
        ButtonEvaluator->Automatic,
        ButtonFunction:>(Ksi[ri]=KsiDefinitionDialog[ri, mi] (*;
            refresh notebook cell with array*);
            NotebookLocate[{KsiDefinitionNBObject,
                "Closed Loop System Denominator Matrix Row Powers"}];
            SelectionMove[SelectedNotebook[], All, Cell];
            NotebookWrite[SelectedNotebook[],
                Cell[BoxData[
                    RowBox[{ToBoxes[
                        Array["ξ", {PlantColumns}]/.
                        "ξ"[i_]->Subscript["ξ", i]], ToBoxes["="],
                        ToBoxes["{"]},
                    ToBoxes[
        RowBox[Array[KsiDefinitionButton[#]&, {PlantColumns}]]//
            DisplayForm], ToBoxes["}"}]]], "TraditionalForm",

```

```

CellTags->"Closed Loop System Denominator Matrix Row
Powers",
ShowCellTags->True, ShowCellBracket->False,
FontSize->14, FontFamily->"Arial", FontWeight->"Bold",
CellHorizontalScrolling->True]]]//DisplayForm

(*Definition of the button that closes the denominator Row Powers
definition \
notebook*)

CloseKsiDefinitionNBOBJECTButton:=
  ButtonBox["Close Window", Active->True, ButtonEvaluator->Automatic,
    ButtonFunction:>(If[
      Count[Map[IntegerQ[#]&, Map[Ksi, Range[PlantColumns]]], True]==
        PlantColumns, KsiRowDegreesDefined=True;
      AppropriateDenominatorMatrix=False; ComputedCompensator=False;
      NotebookLocate[{MainDisplayNBOBJECT,
        "Closed Loop System Denominator Matrix Row Powers"}];
      SelectionMove[MainDisplayNBOBJECT, All, Cell];
      NotebookWrite[MainDisplayNBOBJECT,
        Cell[BoxData[
          RowBox[{ToBoxes[
            Array["ξ", PlantColumns]/.
              "ξ"[i_]->Subscript["ξ", i]], "="},
            ToBoxes[Map[Ksi, Range[PlantColumns]]]}], "TraditionalForm",
            FontSize->16, FontFamily->"Arial", FontWeight->"Bold",
            CellHorizontalScrolling->True, ShowCellTags->False,
            CellTags->"Closed Loop System Denominator Matrix Row
Powers",
            Selectable->False, ShowCellBracket->False]];
      NotebookLocate[{MainDisplayNBOBJECT,
        "Closed Loop System Denominator Matrix"}];
      SelectionMove[MainDisplayNBOBJECT, All, Cell];
      NotebookWrite[MainDisplayNBOBJECT,
        Cell[TextData[{"The closed loop system denominator matrix
row powers have been redefined. The closed loop system denominator matrix
should also be redefined."}]
], CellTags->"Closed Loop System Denominator Matrix", ShowCellTags->False,
      Selectable->True], After];
      NotebookLocate[{MainDisplayNBOBJECT, "Class Of
Compensators"}];
      SelectionMove[MainDisplayNBOBJECT, All, Cell];
      NotebookWrite[MainDisplayNBOBJECT,
        Cell[TextData[{""}], CellTags->"Class Of Compensators",
          ShowCellTags->False, Selectable->True], After],

KsiRowDegreesDefined=False; AppropriateDenominatorMatrix=False;
ComputedCompensator=False;
NotebookLocate[{MainDisplayNBOBJECT,
  "Closed Loop System Denominator Matrix Row Powers"}];
SelectionMove[MainDisplayNBOBJECT, All, Cell];
NotebookWrite[MainDisplayNBOBJECT,
  Cell[TextData[
    "The values of the row powers of the closed loop system
denominator matrix have not been provided."]]
, ShowCellTags->False, CellTags->"Closed Loop System Denominator Matrix Row
Powers",
  Selectable->False, ShowCellBracket->False]];
NotebookLocate[{MainDisplayNBOBJECT,

```

```

        "Closed Loop System Denominator Matrix"}];
    SelectionMove[MainDisplayNBOobject, All, Cell];
    NotebookWrite[MainDisplayNBOobject,
        Cell[TextData[{""}], CellTags->"Closed Loop System
Denominator Matrix",
        ShowCellTags->False, Selectable->True], After];
    NotebookLocate[{MainDisplayNBOobject, "Class Of
Compensators"}];
    SelectionMove[MainDisplayNBOobject, All, Cell];
    NotebookWrite[MainDisplayNBOobject,
        Cell[TextData[{""}], CellTags->"Class Of Compensators",
        ShowCellTags->False, Selectable->True], After];
    NotebookClose[KsiDefinitionNBOobject];
    SetOptions[MainButtonsPaletteObject, Visible->True,
        WindowFloating->True, Editable->False]]//DisplayForm

(*This is the definition of the button in the main button palette
launching \
the notebook for the definition of denominator's matrix Row Powers*)

DefineKsiRowDegreesButton:=
    ButtonBox["Define Denominator Matrix Row Powers", Active->True,
        ButtonEvaluator->Automatic,
        ButtonFunction:>(If[AppropriatePlantData===True,
            SetOptions[MainButtonsPaletteObject, Editable->True,
                WindowFloating->False, Visible->False]; ClearAll[Ksi];
            KsiRowDegreesDefined=False;
            KsiDefinitionNBOobject=NotebookPut[KsiDefinitionNB] ,
            NotebookPut[
                NotebookPut[{Cell[
                    TextData[
                        "There is no valid definition for the plant. Please
define the plant first."]\
], Cell[BoxData[ButtonBox[
                    ToBoxes[
                        StyleForm[StringForm["Close"],
                            FontWeight->"Bold"]],
                        ButtonEvaluator->Automatic, Active->True,
                        ButtonFunction:>(NotebookClose[
                            ButtonNotebook[]];
                        SetOptions[MainButtonsPaletteObject,
                            Visible->True, WindowFloating->True,
                            Editable->False]]]]],
                    TextAlignment->0, ShowCellBracket->False,
                    WindowFrame->"Palette", WindowFrameElements->{},
                    WindowElements->{}, Editable->False,
                    Background->Options[ButtonBox, Background][[1,2]],
                    ScrollingOptions->{PagewiseScrolling->True,
                        PagewiseDisplay->Inherited,
                        RealTimeScrolling->Inherited,
                        JumpScrolling->Inherited,
                        HorizontalScrollRange->Inherited,
                        SpeedParameters->Inherited},
                    WindowSize->{Automatic, FitAll},
                    WindowTitle->"Closed loop system denominator row
powers"}];]])]

(* This is the definition of the button that displays the generalized \
Wolovich Resultant Notebook*)

```

```

DisplayGWResultant:=
  ButtonBox["Display Resultant",Active->True,
    ButtonEvaluator->Automatic,
    ButtonFunction:>
      If[KsiRowDegreesDefined===True,
        SetOptions[MainButtonsPaletteObject,Editable->True,
          WindowFloating->False,Visible->False];
        DisplayResultantNBObject=
          NotebookCenter[NotebookPut[DisplayResultantNB]],
        NotebookPut[
          Notebook[{Cell[
            TextData[
              "There are no valid definitions for the closed loop
system denominator matrix row powers. Please define the closed loop
system denominator matrix row powers first."]\
],Cell[BoxData[ButtonBox[ToBoxes[StyleForm["Close",FontWeight->"Bold"]],
          ButtonEvaluator->Automatic,Active->True,
          ButtonFunction:>(NotebookClose[
            ButtonNotebook[]];
          SetOptions[MainButtonsPaletteObject,
            Visible->True,WindowFloating->True,
            Editable->False]]]],TextAlignment->0,
          ShowCellBracket->False,WindowFrame->"Palette",
          WindowFrameElements->{},WindowElements->{},
          Editable->False,
          Background->
            Options[ButtonBox,Background][[1,
              2]],
          ScrollingOptions->{PagewiseScrolling->True,
            PagewiseDisplay->Inherited,
            RealTimeScrolling->Inherited,
            JumpScrolling->Inherited,
            HorizontalScrollRange->Inherited,
            SpeedParameters->Inherited},
          WindowSize->{Automatic,FitAll},
          WindowTitle->"Generalized Wolovich Resultant"];]]

(*This is the definition of the generalized wolovich resultant Notebook*)

DisplayResultantNB:=
  Notebook[{Cell[
    TextData[{"The generalized Wolovich Resultant "},
    Cell[BoxData[FormBox[SubscriptBox["M","ek"],TraditionalForm]]]],
    TextAlignment->Center,ShowCellBracket->False,
    FontSize->18,FontWeight->"Bold",
    Cell["",ShowCellBracket->False],
    Cell[BoxData[
      ButtonBox[
        ToBoxes[StyleForm["Close","StandardForm",FontWeight->"Bold",
          TextAlignment->Center,FontSize->14]],
        ButtonEvaluator->Automatic,Active->True,
        ButtonFunction:>(NotebookClose[ButtonNotebook[]];
          SetOptions[MainButtonsPaletteObject,Visible->True,
            WindowFloating->True,Editable->False]]],
        TextAlignment->Center,ShowCellBracket->False],
    Cell["",ShowCellBracket->False],
    Cell[BoxData[
      RowBox[{FormBox["k=",TraditionalForm],
        Max[Ksi/@Range[PlantColumns]]+1,"",

```

```

FormBox[SubscriptBox["M",
  ToBoxes[
    StringForm["e`1`",Max[Ksi/@Range[PlantColumns]]+1]]],
  "TraditionalForm"]
,FormBox["\[Element]",TraditionalForm],
FormBox[SuperscriptBox["\[DoubleStruckCapitalR]",
  ToBoxes[
    Dimensions[
      GWResultant[Dright,Nright,var,
        Max[Ksi/@Range[PlantColumns]]+1]]/.{r_,c_}->
        StringForm["`1`\[Times]^2`",r,c]]],
    TraditionalForm]]],FontSize->20,FontWeight->"Normal",
ShowCellBracket->False,TextAlignment->Center],
Cell["",ShowCellBracket->False],
Cell[BoxData[
  RowBox[{Cell[
    BoxData[
      FormBox[
        SubscriptBox["M",
          ToBoxes[
StringForm["e`1`",Max[Ksi/@Range[PlantColumns]]+1]]],
          "TraditionalForm"]],FontSize->18],
      Cell[BoxData[FormBox["=", "TraditionalForm"]],FontSize->18],
      FormBox[ToBoxes[
        MatrixForm[
          GWResultant[Dright,Nright,var,
            Max[Ksi/@Range[PlantColumns]]+1]]],
          "TraditionalForm"]]]],TextAlignment->Center,
ShowCellBracket->False,FontSize->14}},
WindowFrameElements->{},WindowSize->{2500,FitAll},
WindowTitle->"Generalized Wolovich Resultant",WindowFrame->"Palette"]

(*This is the definition of the button that launches the process of
selecting \
the polynomial elements of the
closed loop denominator matrix*)

DefineDenominatorMatrix:=
  ButtonBox[RowBox[{"Define"," ","Denominator"," ","Matrix"}],
  ButtonFunction:>
  If[KsiRowDegreesDefined==False,
  NotebookPut[
    Notebook[{Cell[
      TextData[
        "The closed loop system denominator matrix row powers
need to be defined first"]\
],Cell[BoxData[ButtonBox[
  ToBoxes[
    StyleForm[StringForm["Close"],FontWeight->"Bold"]],
    ButtonEvaluator->Automatic,Active->True,
    ButtonFunction:>(NotebookClose[
      ButtonNotebook[]];
    SetOptions[MainButtonsPaletteObject,
      Visible->True,WindowFloating->True,
      Editable->False]]]]],TextAlignment->0,
ShowCellBracket->False,WindowFrame->"Palette",
WindowFrameElements->{},WindowElements->{},
Editable->False,
Background->Options[ButtonBox,Background][[1,2]],

```

```

ScrollingOptions->{PagewiseScrolling->True,
  PagewiseDisplay->Inherited,
  RealTimeScrolling->Inherited,
  JumpScrolling->Inherited,
  HorizontalScrollRange->Inherited,
  SpeedParameters->Inherited},
WindowSize->{Automatic,FitAll},
WindowTitle->"Closed Loop System Denominator Definition"]],
If[Not[AppropriatePlantData],
CompoundExpression[
  NotebookLocate[{MainDisplayNBOobject,
    "Closed Loop System Denominator Matrix"}],
  SelectionMove[MainDisplayNBOobject,All,Cell],
  NotebookWrite[MainDisplayNBOobject,
    Cell[TextData[
      "The Plant has not been properly defined.Please define
the strictly proper plant first."]\
,CellTags->"Closed Loop System Denominator Matrix",ShowCellTags->False,
  Selectable->True],After]],
  NotebookLocate[{MainDisplayNBOobject,"Class Of Compensators"}];
  SelectionMove[MainDisplayNBOobject,All,Cell];
  NotebookWrite[MainDisplayNBOobject,
    Cell["","Input",CellTags->"Class Of Compensators",
      ShowCellTags->False,Selectable->True],After];
  ComputedCompensator=False;
  CompoundExpression[Set[PolynomialsIndex,1],
    Set[PolynomialsNumber,PlantColumns],
    Set[PolynomialsList,
      Times[Range[1,PolynomialsNumber],
        Power[Range[1,PolynomialsNumber],-1]]],
    Set[NumberOfRoots,ZeroVector[PolynomialsNumber]],
    Set[DkPolynomialsDegrees,
      Map[Plus[DrightColumnDegrees[[#]],Ksi[#]]&,
        Range[PlantColumns]]],
    SetOptions[MainButtonsPaletteObject,Editable->True,
      Visible->False],
    ComplexPlaneNBOobject=NotebookPut[ComplexPlaneNB],
    NotebookCenter[ComplexPlaneNBOobject],
    RootsAssignmentNBOobject=NotebookPut[RootsAssignmentNB],
    DisplayPolynomial[PolynomialsIndex,var],
    DisplayRootsRemaining[PolynomialsIndex]]],
  ButtonEvaluator->Automatic,Active->True]

(*This is the definition of the button that exits*)
ExitApp:=ButtonBox["Exit",ButtonEvaluator->Automatic,Active->True,
  ButtonFunction:>(Map[
NotebookClose[#]&,{MainDisplayNBOobject,MainButtonsPaletteObject,
  ComplexPlaneNBOobject,RootsAssignmentNBOobject,
RootModificationNBOobject,PlantDefinitionNBOobject,HelpNBOobject}];
  Clear["Compensators`*"];Remove["Compensators`*"])]

(*This is the definition of the button that computes the
CPPPSCompensators*)

ComputeCompensators:=
  ButtonBox[RowBox[{"Compute"," ","Compensators"}],
  ButtonFunction:>(If[
    And[AppropriatePlantData,AppropriateDenominatorMatrix],
    CompoundExpression[{XL,YL}=

```

```

        ComputeClassOfCompensators[StrictlyProperPlant,
DenominatorMatrix, var][[Range[2]]], ComputedCompensator=True,
        Set[TDXL, ReplaceAll[XL, t[Pattern[x, Blank[]]]]-
>Subscript[t, x]],
        Set[TDYL, ReplaceAll[YL, t[Pattern[x, Blank[]]]]-
>Subscript[t, x]],
        DisplayCompensators[TDXL, TDYL]],
    CompoundExpression[
        NotebookLocate[{MainDisplayNBOBJECT, "Class Of
Compensators"}],
        SelectionMove[MainDisplayNBOBJECT, All, Cell],
        NotebookWrite[MainDisplayNBOBJECT,
            Cell[TextData[
                "The Plant or/and the Denominator Matrix definitions
are invalid, or the Plant has changed and the Denominator should be
redefined ."]\
, CellTags->"Class Of Compensators", ShowCellTags->False, Selectable->True],
            After]]];
    SetOptions[MainButtonsPaletteObject, Editable->False,
WindowFloating->False];
    SetOptions[MainButtonsPaletteObject, Visible->True,
WindowFloating->True, Editable->False]),
    ButtonEvaluator->Automatic, Active->True]

ParametersValuesButton:=
    ButtonBox["Parameters Values", ButtonEvaluator->Automatic,
Active->True,
    ButtonFunction:>(SetOptions[MainButtonsPaletteObject,
        Editable->True, Visible->False];
        If[VectorQ[ParametersValuesArray], ClearAll[parval]];
        If[ComputedCompensator==True&&

ComputeClassOfCompensators[StrictlyProperPlant, DenominatorMatrix,
var][[3]]!=0,
ParametersValuesArray=
    Array[parval,
        ComputeClassOfCompensators[StrictlyProperPlant,
DenominatorMatrix, var][[3]]];
    NotebookCenter[
        ParametersValuesNBOBJECT=NotebookPut[ParametersValuesNB]],
    NotebookPut[
        Notebook[{Cell[
            TextData[
                "There are no independent parameters in the
expression of the class of compensators or the class has not been
computed yet."]\
], Cell[BoxData[ButtonBox[
                ToBoxes[
                    StyleForm[StringForm["Close"],
                        FontWeight->"Bold"]],
                    ButtonEvaluator->Automatic, Active->True,
                    ButtonFunction:>(NotebookClose[
                        ButtonNotebook[]];
                    SetOptions[MainButtonsPaletteObject,
                        Visible->True, WindowFloating->True,
                        Editable->False] ]]]], TextAlignment->0,
                    ShowCellBracket->False, WindowFrame->"Palette",
                    WindowFrameElements->{}, WindowElements->{},
                    Editable->False,

```

```

Background->Options[ButtonBox,Background][[1,2]],
ScrollingOptions->{PagewiseScrolling->True,
  PagewiseDisplay->Inherited,
  RealTimeScrolling->Inherited,
  JumpScrolling->Inherited,
  HorizontalScrollRange->Inherited,
  SpeedParameters->Inherited},WindowSize->FitAll,
WindowTitle->"Independent Parameters Values"]]]])

ParameterBox[i_]:=
  ButtonBox[

If[NumericQ[parval[i]],parval[i],SubscriptBox["t",i]]//TraditionalForm,
  ButtonEvaluator->Automatic,Active->True,
  ButtonFunction:>(parval[i]=GetParameterValue[i];
  RefreshParametersValuesNB)]

GetParameterValue[i_Integer]:=
  Module[{pv},(Off[Syntax::"com"];
  pv=InputString[
  StringForm["Please define the `1`2` independent parameter
(The value should be any real number) .",
  i,Switch[i,1,"st",2,"nd",3,"rd",x_Integer;/x>3,"th"]]];
  If[SyntaxQ[pv],pv=ToExpression[pv];On[Syntax::"com"];
  If[NumericQ[pv]&&Im[pv]==0,pv,GetParameterValue[i]],
  On[Syntax::"com"];GetParameterValue[i]])]

ParametersValuesNB:=
  Notebook[{Cell[
  BoxData[RowBox[{"T","=","(",
  ToBoxes[RowBox[
  Array[ParameterBox[#]&,

Dimensions[ParametersValuesArray]]//DisplayForm]," "],
  " "]]],"TraditionalForm",FontSize->16,FontWeight->"Bold",
  CellTags->"Parameters Values",CellHorizontalScrolling->True],
  Cell[BoxData[
  RowBox[{"C(",var,ToBoxes["]="),
  SuperscriptBox[ToBoxes[MatrixForm[TDXL]],-1],
  ToBoxes[MatrixForm[TDYL]],""}]],"TraditionalForm",
  FontFamily->"Arial",FontSize->16,FontWeight->"Bold",
  CellTags->"Compensator",ShowCellTags->True,
  CellHorizontalScrolling->True],
  Cell[BoxData[
  ButtonBox["Finish",ButtonEvaluator->Local,Active->"True",
  ButtonFunction:>(NotebookClose[ButtonNotebook[]];
  SetOptions[MainButtonsPaletteObject,Visible->True,
  WindowFloating->True,Editable->False]]],
  FontSize->12,FontWeight->"Bold",
  CellHorizontalScrolling->False,TextAlignment->0,
  Background->Options[ButtonBox,Background][[1,2]],
  CellMargins->{{0,0},{1,Inherited}}],WindowTitle->"Computing
specific members of the class of closed loop system denominator assigning
compensators by assigning values to the independent parameters",
  ShowCellBracket->False ,WindowSize->{2500,2500},
  WindowFrameElements->{},
  ScrollingOptions->{PagewiseScrolling->Inherited,
  PagewiseDisplay->Inherited,RealTimeScrolling->Inherited,
  JumpScrolling->Inherited,HorizontalScrollRange->29500,
  SpeedParameters->Inherited}]

```



```

RefreshParametersValuesNB:= (NotebookLocate[{ParametersValuesNBObject,
  "Parameters Values"}];SelectionMove[SelectedNotebook[],All,Cell];
NotebookWrite[SelectedNotebook[],
  Cell[BoxData[
    RowBox[{"T", "=", "(" ,
      ToBoxes[RowBox[
        Array[ParameterBox[#]&,
          Dimensions[ParametersValuesArray]]//DisplayForm],")",
        " "]]], "TraditionalForm",FontSize->16,FontWeight->"Bold",
    CellTags->"Parameters Values",CellHorizontalScrolling->True],
  After];If[
  Apply[And,Map[NumericQ,Flatten[ParametersValuesArray]]]==
  True,{CXL,CYL}=
  Collect[Expand[
    Simplify[{XL,YL} //. Flatten[
      Table[t[i]->parval[i],{i,
        Length[ParametersValuesArray]}]]],var];
NotebookLocate[{ParametersValuesNBObject,"Compensator"}];
SelectionMove[SelectedNotebook[],All,Cell];
NotebookWrite[SelectedNotebook[],
  Cell[BoxData[
    RowBox[{"C(",var,
      ToBoxes[""]="\ [NoBreak]",SuperscriptBox[
        ToBoxes[MatrixForm[CXL]],-1],
      ToBoxes[MatrixForm[CYL]],
      "\n" =\ [NoBreak]",ToBoxes[
        MatrixForm[
          Rationalize[
            Collect[Expand[Simplify[Inverse[CXL].CYL]],var]]]]}], "TraditionalForm",
    FontFamily->"Arial",FontSize->16,FontWeight->"Bold",
    CellTags->"Compensator",ShowCellTags->True,
    CellHorizontalScrolling->True],After]]
(*This is the button that displays the help notebook*)
HelpNBButton:=
  ButtonBox["Help",ButtonEvaluator->Automatic,Active->True,
    ButtonFunction:>(HelpNBObject=NotebookOpen["Help2.nb"])]

(*COMPLEX PLANE NOTEBOOK INTERFACE FUNCTIONS AND OBJECTS*)

ComplexForm[x_List]:=
  (If[Length[x]==2&&VectorQ[x],x[[1]]+x[[2]]*I,
    "Error:Input should be a vector of the form '{Rez,Imz}'"])

RoundAtDigit[x_,y_Integer]:=Round[x*10^(-y)]*10^y

(*DIORTHOSI*)
ObtainComplexFromSelectedCoordinates:=(GetCoordinates=
  NotebookRead[ComplexPlaneNBObject];
  If[StringQ[GetCoordinates],
    CoordinatesAsList=N[RoundAtDigit[ToExpression[GetCoordinates],-3]],
    CoordinatesAsList=N[RoundAtDigit[GetCoordinates,-3]])]

DisplayComplex[x_List]:=(ComplexRoot=ComplexForm[x];
  NotebookLocate[{ComplexPlaneNBObject,"Complex"}];
  SelectionMove[ComplexPlaneNBObject,All,Cell,AutoScroll->False];
  NotebookWrite[ComplexPlaneNBObject,
    Cell[BoxData[

```

```

RowBox[{"z = ",
  ToBoxes[If[NumericQ[ComplexRoot],
NumberForm[N[ComplexRoot,17],16],Rationalize[ComplexRoot]]],
  "=",ToBoxes[Rationalize[ComplexRoot]]}], "TraditionalForm",
  FontWeight->"Bold", CellTags->"Complex",
  ShowCellTags->True, Selectable->False,
  ShowCellBracket->False], After];
NotebookLocate[{ComplexPlaneNBOobject, "Cplot"}];
SelectionMove[ComplexPlaneNBOobject, Previous, Cell])

DisplayPolynomial[Pol_, var_] :=
  (NotebookLocate[{ComplexPlaneNBOobject, "Polynomial"}];
  SelectionMove[ComplexPlaneNBOobject, All, Cell, AutoScroll->False];
  NotebookWrite[ComplexPlaneNBOobject,
  Cell[BoxData[
    RowBox[{SubscriptBox["d", PolynomialsIndex], ToBoxes["(", var,
ToBoxes[")"], "=", ToBoxes[PolynomialsList[[PolynomialsIndex]]], "=",
ToBoxes[Collect[Expand[PolynomialsList[[PolynomialsIndex]],
  var]]}], "TraditionalForm", FontWeight->"Bold",
  CellTags->"Polynomial", ShowCellTags->True,
  Selectable->False, ShowCellBracket->False], After];
  NotebookLocate[{ComplexPlaneNBOobject, "Cplot"}];
  SelectionMove[ComplexPlaneNBOobject, Previous, Cell])

DisplayRootsRemaining[PolIndx_Integer] :=
  (RootsRemaining=DkPolynomialsDegrees[[PolIndx]]-
  NumberOfRoots[[PolIndx]];
  NotebookLocate[{ComplexPlaneNBOobject, "Roots Remaining"}];
  SelectionMove[ComplexPlaneNBOobject, All, Cell, AutoScroll->False];
  NotebookWrite[ComplexPlaneNBOobject,
  Cell[BoxData[
    RowBox[{"Roots remaining to be specified for the full
definition of ",
  SubscriptBox["d", PolIndx], ToBoxes["(", var, ToBoxes[") :"],
  RootsRemaining}], "TraditionalForm", FontWeight->Bold,
  CellTags->"Roots Remaining", ShowCellTags->True,
  ShowCellBracket->False], After];
  NotebookLocate[{ComplexPlaneNBOobject, "Cplot"}];
  SelectionMove[ComplexPlaneNBOobject, Previous, Cell]
  )

StatusMessage[msg_Cell] := (NotebookLocate[{ComplexPlaneNBOobject, "Status"}]
;
  SelectionMove[ComplexPlaneNBOobject, All, Cell, AutoScroll->False];
  NotebookWrite[ComplexPlaneNBOobject, msg, After];
  NotebookLocate[{ComplexPlaneNBOobject, "Cplot"}];
  SelectionMove[ComplexPlaneNBOobject, Previous, Cell])

(*Here are all the "selection manipulation" notebook related definitions
*)

CreateRootMultiplesArray[
  x_List?VectorQ] := (Array[(ComplexForm[{10^(#1-3), 10^(#2-
3)}*x]) &, {5, 5}])

RootMultiplesGridBox[rootmultiples_?MatrixQ] :=
  GridBox[Array[

```

```

        ButtonBox[rootmultiples[[#1,#2]],Active->True,
            ButtonEvaluator->Automatic,
            ButtonFunction:>(LastRootSelection=CoordinatesAsList;
                CoordinatesAsList={GeneralizedRe[rootmultiples[[#1,#2]]],
                    GeneralizedIm[rootmultiples[[#1,#2]]]};
                RefreshRootModificationNB)&,{5,5}],ColumnSpacings->0,
            RowSpacings->0]//DisplayForm

RefreshRootModificationNB:=(NotebookLocate[{RootModificationNBObject,
    "Current Root Selection"}];
    MoveSelection[RootModificationNBObject,All,Cell];
    NotebookWrite[RootModificationNBObject,
        Cell[BoxData[
            RowBox[{"z=",ToBoxes[ComplexForm[CoordinatesAsList]],"
",
                GridBox[{{UndoButton},{ResetButton}}]}}],CellTags->"Current
Root Selection",
            TextAlignment->0],After];
    NotebookLocate[{RootModificationNBObject,"Coordinates Scaling"}];
    MoveSelection[RootModificationNBObject,All,Cell];
    NotebookWrite[RootModificationNBObject,
        Cell[BoxData[
            RowBox[{{ToBoxes[
                RootMultiplesGridBox[
                    CreateRootMultiplesArray[CoordinatesAsList]]}}],CellTags->"Coordinates
Scaling",
            TextAlignment->-1,CellHorizontalScrolling -> True]]]

KeepRealPartButton:=
    ButtonBox["Im(z)\[Rule]0",ButtonEvaluator->Automatic,Active->True,
        ButtonFunction:>(LastRootSelection=CoordinatesAsList;
            CoordinatesAsList={CoordinatesAsList[[1]],0};
            RefreshRootModificationNB)]//DisplayForm

KeepImaginaryPartButton:=
    ButtonBox["Re(z)\[Rule]0",ButtonEvaluator->Automatic,Active->True,
        ButtonFunction:>(LastRootSelection=CoordinatesAsList;
            CoordinatesAsList={0,CoordinatesAsList[[2]]};
            RefreshRootModificationNB)]//DisplayForm

ReImPartPrecisionGridBox:=
    GridBox[Prepend[
        Array[{ButtonBox[
            StyleForm[SuperscriptBox[10,(#1-3)],FontSize->13],
            ButtonEvaluator->Automatic,Active->True,
            ButtonFunction:>(LastRootSelection=
                CoordinatesAsList;
                CoordinatesAsList={RoundAtDigit[
                    CoordinatesAsList[[1]],#1-3],
                    CoordinatesAsList[[2]]};RefreshRootModificationNB)],
            ButtonBox[
            StyleForm[SuperscriptBox[10,(#1-3)],FontSize->13],
            ButtonEvaluator->Automatic,Active->True,
            ButtonFunction:>(LastRootSelection=
                CoordinatesAsList;
                CoordinatesAsList={CoordinatesAsList[[1]],
                    RoundAtDigit[CoordinatesAsList[[2]],#1-3]};

```

```

RefreshRootModificationNB)]]&, {3}], {StyleForm["Re (z)
Precision",
    FontSize->12],
    StyleForm["Im(z) Precision", FontSize->12]} ],
    ColumnSpacings->1, RowSpacings->0]//DisplayForm

UndoButton:=
    ButtonBox["Undo", ButtonEvaluator->Automatic, Active->True,
        ButtonFunction:>(CoordinatesAsList=LastRootSelection;
            RefreshRootModificationNB)]

CancelButton:=
    ButtonBox["Cancel", ButtonEvaluator->Automatic, Active->True,
        ButtonFunction:>(CoordinatesAsList=InitialRootValue;
            InitialRootValue=.; LastRootSelection=.;
            NotebookClose[ButtonNotebook[]];
            SetOptions[RootsAssignmentNBOobject, Visible->True,
                WindowFloating->True, Editable->False])]//DisplayForm

ResetButton:=
    ButtonBox["Reset", ButtonEvaluator->Automatic, Active->True,
        ButtonFunction:>(LastRootSelection=CoordinatesAsList;
            CoordinatesAsList=InitialRootValue; RefreshRootModificationNB)]

CloseRootModificationNBButton:=
    ButtonBox["Finish", ButtonEvaluator->Automatic, Active->True,
        ButtonFunction:>(InitialRootValue=.; LastRootSelection=.;
            ComplexRoot=ComplexForm[CoordinatesAsList]);

DisplayComplex[CoordinatesAsList]; NotebookClose[ButtonNotebook[]];
    SetOptions[RootsAssignmentNBOobject, Visible->True,
        WindowFloating->True, Editable->False])]//DisplayForm

(*This is the definition of the notebook that contains all the root
selection \
variations*)

RootModificationNB:=
    Notebook[{Cell[
        BoxData[RowBox[{"z=", ToBoxes[ComplexForm[CoordinatesAsList]], "
",
            Cell[BoxData[
                RowBox[{{GridBox[{{UndoButton}, {ResetButton}}]}]}],
            CellTags->"Current Root Selection", TextAlignment->0],
        Cell[BoxData[
            RowBox[{{ToBoxes[
                RootMultiplesGridBox[
CreateRootMultiplesArray[CoordinatesAsList]]}]}], CellTags->"Coordinates
Scaling",
            TextAlignment->-1, CellHorizontalScrolling -> True],
        Cell[BoxData[RowBox[{{ToBoxes[ReImPartPrecisionGridBox]}]}],
            TextAlignment->0],
        Cell[BoxData[RowBox[{{ToBoxes[KeepRealPartButton]}]}],
            TextAlignment->0],
        Cell[BoxData[
            RowBox[{{ToBoxes[CancelButton],
                ToBoxes[CloseRootModificationNBButton]}]}], TextAlignment->1,
            FontSize->14}], ShowCellTags->True,

```

```

ShowCellBracket->False,WindowSize->{FitAll,400},
ScrollingOptions->{PagewiseScrolling->True,
  PagewiseDisplay->Inherited,RealTimeScrolling->Inherited,
  JumpScrolling->Inherited,HorizontalScrollRange->3500,
  SpeedParameters->Inherited},WindowFloating->True,
Background->Options[ButtonBox,Background][[1,2]],
ShowSelection->False,WindowFrame->"Palette",
WindowElements->{"HorizontalScrollBar"},
WindowFrameElements->{""},WindowTitle->"Selection Manipulation";
(*This is the button definition for the "obtain number" button *)
ObtainNumberB:=
  ButtonBox[RowBox[{"Obtain", " ", "Number"}],
    ButtonFunction:>
      CompoundExpression[ObtainComplexFromSelectedCoordinates,
        DisplayComplex[CoordinatesAsList]],ButtonEvaluator->Automatic,
        Active->True]
(*This is the definition of the button that launches the "Selection
Modification" \
notebook*)
SelectionModificationButton:=
  ButtonBox["Modify Selection",ButtonEvaluator->Automatic,
    Active->True,
    ButtonFunction:>(If[VectorQ[CoordinatesAsList],
      InitialRootValue=CoordinatesAsList;
      RootModificationNBObject=NotebookPut[RootModificationNB]])]

(*This is the definition of "multiply" button.*)

MultiplyB:=
  ButtonBox["Multiply",
    ButtonFunction:>
      CompoundExpression[
        Set[MultiplySelection,
          Input["Multiply Selection by....(if input has the form {x,y}
then the real part and the imaginary will be multiplied by x and y
respectively")\
],If[LessEqual[Length[MultiplySelection],2],
CompoundExpression[TimesBy[CoordinatesAsList,MultiplySelection],
  Set[ComplexRoot,ComplexForm[CoordinatesAsList]]],
  DisplayComplex[CoordinatesAsList]],ButtonEvaluator->Automatic,
Active->True]

(*This is the definition of the "round coordinates button" *)

RoundCoordinatesB:=
  ButtonBox[RowBox[{"Round", " ", "Coordinates"}],
    ButtonFunction:>
      CompoundExpression[
        Set[DigitPos,
          Input["Select position of digit to be rounded...(Negatives
corespond to decimal digits)"]\
],Set[CoordinatesAsList,RoundAtDigit[CoordinatesAsList,DigitPos]],
        Set[ComplexRoot,ComplexForm[CoordinatesAsList]],
        DisplayComplex[CoordinatesAsList]],ButtonEvaluator->Automatic,
        Active->True]

(*This is the definition of the button that allows to define roots from \
keyboard*)

```

```

EnterRootKeyboard:=
  ButtonBox["Define Root (keyboard)", ButtonEvaluator->Automatic,
    Active->True,
    ButtonFunction:>(ComplexRoot=GetComplexRoot;
      CoordinatesAsList={GeneralizedRe[ComplexRoot],
        GeneralizedIm[ComplexRoot]};DisplayComplex[CoordinatesAsList])]
(*This is the definition of the "z as polynomial root" button *)

ToPolynomialAsRootB:=
  ButtonBox[RowBox[{"z", "\[Rule]", RowBox[{"Polynomial", " ", "Root"}]}],
    ButtonFunction:>
      CompoundExpression[
        Set[Dif, Plus[Part[DkPolynomialsDegrees, PolynomialsIndex],
          Times[-1, Part[NumberOfRoots, PolynomialsIndex]]],
          If[And[GreaterEqual[Dif, 2],
            Unequal[ToString[ComplexExpand[Times[2, ComplexRoot]]],
              ToString[
                ComplexExpand[
          Plus[ComplexRoot, GeneralizedConjugate[ComplexRoot]]]]],
            CompoundExpression[
              Set[Part[PolynomialsList, PolynomialsIndex],
                Rationalize[
                  Times[Part[PolynomialsList, PolynomialsIndex],
                    Plus[var, Times[-1, ComplexRoot]],
                    Plus[var,
                      Times[-1,
            ComplexExpand[GeneralizedConjugate[ComplexRoot]]]]],
                AddTo[Part[NumberOfRoots, PolynomialsIndex], 2]],
              If[And[GreaterEqual[Dif, 1],
                Equal[ToString[Times[2, ComplexRoot]],
                  ToString[
                    ComplexExpand[
          Plus[ComplexRoot, GeneralizedConjugate[ComplexRoot]]]]],
                CompoundExpression[
                  Set[Part[PolynomialsList, PolynomialsIndex],
                    Rationalize[
                      Times[Part[PolynomialsList, PolynomialsIndex],
                        Plus[var, Times[-1, ComplexRoot]]]]],
                    AddTo[Part[NumberOfRoots, PolynomialsIndex], 1]],
                  DisplayPolynomial[Part[PolynomialsList, PolynomialsIndex], var],
                  DisplayRootsRemaining[PolynomialsIndex],
                  If[Equal[Part[NumberOfRoots, PolynomialsIndex],
                    Part[DkPolynomialsDegrees, PolynomialsIndex]],
                    StatusMessage[
                      Cell[BoxData[
                        RowBox[{"The polynomial
", SubscriptBox["d", PolynomialsIndex],
                          ToBoxes[" (", var,
                            ToBoxes[") has already reached the required
degree."}]]], "TraditionalForm",
                          FontWeight->"Bold", CellTags->"Status",
                          ShowCellTags->True, ShowCellBracket->False]],
                    StatusMessage[
                      Cell[TextData[{" "}], "Input", CellTags->"Status",
                        ShowCellTags->True, ShowCellBracket->False]]],

```

```

If[Equal[NumberOfRoots,DkPolynomialsDegrees],
CompoundExpression[Set[AppropriateDenominatorMatrix,True],
StatusMessage[
Cell[BoxData[
RowBox[{"The denominator matrix
",SubscriptBox["D","C"],
ToBoxes["("],var,
ToBoxes[
") is defined. Press the \"Finish\" button to
exit."}}\
]],"TraditionalForm",FontWeight->"Bold",CellTags->"Status",
ShowCellTags->True,ShowCellBracket->False]],
Set[AppropriateDenominatorMatrix,False]],
ButtonEvaluator->Automatic,Active->True]

(*This is the definition of "Reset Current Polynomial" button *)

ResetPolynomialB:=
ButtonBox[RowBox[{"Reset"," ","Current"," ","Polynomial"}],
ButtonFunction:>
CompoundExpression[Set[Part[PolynomialsList,PolynomialsIndex],1],
Set[Part[NumberOfRoots,PolynomialsIndex],0],
DisplayPolynomial[Part[PolynomialsList,PolynomialsIndex],var],
DisplayRootsRemaining[PolynomialsIndex],
StatusMessage[
Cell[TextData[{""}],"Input",CellTags->"Status",
ShowCellTags->True,ShowCellBracket->False]],
ButtonEvaluator->Automatic,Active->True]

(*This is the definition of the "PreviousPolynomial" button*)

PreviousPolyB:=
ButtonBox[RowBox[RowBox[{"<<"," ","Previous"}]," ","Polynomial"}],
ButtonFunction:>
CompoundExpression[
If[Greater[PolynomialsIndex,1],AddTo[PolynomialsIndex,-1]],
If[Unequal[DkPolynomialsDegrees,NumberOfRoots],
If[Equal[Part[DkPolynomialsDegrees,PolynomialsIndex],
Part[NumberOfRoots,PolynomialsIndex]],
StatusMessage[
Cell[TextData[{"The polynomial ",
Cell[BoxData[SubscriptBox["d",PolynomialsIndex]]],
ToBoxes["("],var,
ToBoxes[") has already acquired its required
degree."}}],"Input",
CellTags->"Status",ShowCellTags->True,
ShowCellBracket->False]],
StatusMessage[
Cell[TextData[{""}],"Input",CellTags->"Status",
ShowCellTags->True,ShowCellBracket->False]],
StatusMessageCell[
BoxData[RowBox[{"The denominator matrix
",SubscriptBox["D","C"],
ToBoxes["("],var,
ToBoxes[
") is defined. Press the \"Finish\" button to
exit."}}]],"TraditionalForm",
FontWeight->"Bold",CellTags->"Status",
ShowCellTags->True]],
DisplayPolynomial[Part[PolynomialsList,PolynomialsIndex],var],

```

```

    DisplayRootsRemaining[PolynomialsIndex]],
    ButtonEvaluator->Automatic,Active->True]

(*This is the definition of the "NextPolynomial" button*)

NextPolyB:=
  ButtonBox[RowBox[{{RowBox[{"Next", " ", "Polynomial"}], " ", ">>"}],
    ButtonFunction:>
      CompoundExpression[
        If[Less[PolynomialsIndex,PolynomialsNumber],
          AddTo[PolynomialsIndex,1]],
        If[Unequal[DkPolynomialsDegrees,NumberOfRoots],
          If[Equal[Part[DkPolynomialsDegrees,PolynomialsIndex],
            Part[NumberOfRoots,PolynomialsIndex]],
            StatusMessage[
              Cell[TextData[{"The polynomial ",
                Cell[BoxData[SubscriptBox["d",PolynomialsIndex]],
                ToBoxes["("],var,
                ToBoxes[") has already acquired its required
degree."}], "Input",
              CellTags->"Status",ShowCellTags->True,
              ShowCellBracket->False]],
            StatusMessage[
              Cell[TextData[{""}], "Input", CellTags->"Status",
              ShowCellTags->True, ShowCellBracket->False]]],
            StatusMessage[
              Cell[BoxData[
                RowBox[{"The denominator matrix ",SubscriptBox["D","C"],
                ToBoxes["("],var,
                ToBoxes[
                  ") is defined. Press the \"Finish\" button to
exit."}], "Input",
              CellTags->"Status",
              ShowCellTags->True]]],
            DisplayPolynomial[Part[PolynomialsList,PolynomialsIndex],var],
            DisplayRootsRemaining[PolynomialsIndex]],
            ButtonEvaluator->Automatic,Active->True]

(*This is the definition of the "Close Complex Plane Window" button *)

CloseComplexWindowsB:=
  ButtonBox[RowBox[{"Finish"}],
    ButtonFunction:>
      CompoundExpression[Unset[z],
        If[Equal[DkPolynomialsDegrees,NumberOfRoots],
          CompoundExpression[Set[AppropriateDenominatorMatrix,True],
            Set[DenominatorMatrix,

Collect[Expand[Rationalize[DiagonalMatrix[PolynomialsList]],
  var]],NotebookLocate[{MainDisplayNBObject,
  "Closed Loop System Denominator Matrix"}],
  SelectionMove[MainDisplayNBObject,All,Cell],
  NotebookWrite[MainDisplayNBObject,
  Cell[BoxData[
    RowBox[{{SubscriptBox["D","C"],ToBoxes["("],var,
ToBoxes[")"}],ToBoxes[MatrixForm[DenominatorMatrix]]}],
  "TraditionalForm",FontSize->16,FontFamily->"Arial",
  FontWeight->"Bold",CellTags->"Closed Loop System
Denominator Matrix",

```



```

        CellHorizontalScrolling->True,ShowCellTags->False],
        After]],Set[AppropriateDenominatorMatrix,False];
    NotebookLocate[{MainDisplayNBObject,
        "Closed Loop System Denominator Matrix"}];
    SelectionMove[MainDisplayNBObject,All,Cell];
    NotebookWrite[MainDisplayNBObject,
        Cell[TextData[
            "The Denominator Matrix has not been defined yet.Please
            define the Denominator Matix."]\
        ,CellTags->"Closed Loop System Denominator Matrix",ShowCellTags->False,
        Selectable-
    >True],After]],NotebookClose[ComplexPlaneNBObject],
    NotebookClose[RootsAssignmentNBObject],
    SetOptions[MainButtonsPaletteObject,Visible->True,
        Editable->False]],ButtonEvaluator->Automatic,
    Active->True]

(*GENERAL NOTEBOOK MANIPULATION TOOLS*)
NotebookCenter[x_NotebookObject]:=
(
    VisibleDef=Options[x,Visible][[1,2]];
    SetOptions[x,Visible->False];
    WindowDimensionsSmallerBy=
        AbsoluteOptions[x,ScreenRectangle][[1,2,All,2]]-{0,25}-
        AbsoluteOptions[x,WindowSize][[1,2]];
    Margins:=Round[
        Transpose[{(WindowDimensionsSmallerBy/
            2)//.(val_/;
                NumberQ[val]&&
                val<0):>0,(WindowDimensionsSmallerBy/
            2)//.(val_/;NumberQ[val]&&val<0):>0}]];
    SetOptions[x,WindowMargins->Margins,Visible->VisibleDef,
        WindowSize->{Inherited,Automatic}]
)

ComplexPlaneWidth:=Options[SelectedNotebook[],ScreenRectangle][[1,2,1,2]]
-60;

(*Essential Notebooks*)
MainDisplayNB=
    Notebook[{Cell[BoxData[RowBox[{"Strictly Proper Plant"}]],FontWeight-
    >"Bold",
        FontColor->RGBColor[0,0,1]],
        Cell["",Selectable->True,ShowCellTags->False,
        CellTags->"Strictly Proper Plant",
        CellHorizontalScrolling->True],
        Cell[BoxData[
            RowBox[{"Closed Loop System Denominator Matrix Row
            Powers"}]],
            FontWeight->"Bold",FontColor->RGBColor[0,0,1]],
        Cell["",Selectable->True,ShowCellTags->False,
        CellTags->"Closed Loop System Denominator Matrix Row Powers",
        CellHorizontalScrolling->True],
        Cell[BoxData[RowBox[{"Closed Loop System Denominator Matrix"}]],
        FontWeight->"Bold",FontColor->RGBColor[0,0,1]],
        Cell["",Selectable->True,ShowCellTags->False,
        CellTags->"Closed Loop System Denominator Matrix"],
        Cell[BoxData[
            RowBox[{"Class Of ",SubscriptBox["D","C"],
            ToBoxes[" Assigning Stabilizing Compensators"]}]]],

```

```

    FontWeight->"Bold", FontColor->RGBColor[0,0,1]],
    Cell["", Selectable->True, ShowCellTags->False,
    CellTags->"Class Of Compensators"]],
    WindowSize->{2500,2500}, Selectable->False,
    ShowCellBracket->False, WindowTitle->"Computation of the class of
denominator assigning stabilizing compensators...",
    ScrollingOptions->{PagewiseScrolling->False,
    PagewiseDisplay->Inherited, RealTimeScrolling->Inherited,
    JumpScrolling->Inherited, HorizontalScrollRange->29500,
    SpeedParameters->Inherited}];

MainButtonsPalette:=
    Notebook[{Cell[
        BoxData[GridBox[{{"Strictly Proper Plant"},{DefinePlantT},{\
DefinePlantB},{"Closed Loop System"},{"Denominator Matrix"},{\
DefineKsiRowDegreesButton},{DisplayGWResultant},{DefineDenominatorMatrix}
,{"Class Of Denominator Assigning"}\
,{" Stabilizing
Compensators"},{ComputeCompensators},{ParametersValuesButton},\
{"Time
Response"},{ResponsePlotsButton},{"Program"},{ExitApp},{HelpNBBButton}},\
        RowSpacings->{1,0,1,1,1,0,0,1,1,1,0,1,1,0}]]]],
    WindowFrame->Palette, WindowSize->FitAll,
    WindowElements->{}, ShowCellBracket->False,
    Background->Options[ButtonBox,Background][[1,2]],
    CellMargins->{{0,0},{0,0}}, WindowFrameElements->{},
    CellFrame->1, CellFrameMargins->{{0,0},{0,4}},
    Editable->False, WindowTitle->"Basic Buttons",
    WindowFloating->True,
    WindowMargins->{{Automatic,0},{Automatic,0}},
    ScrollingOptions->{PagewiseScrolling->True,
    PagewiseDisplay->Inherited, RealTimeScrolling->Inherited,
    JumpScrolling->Inherited, HorizontalScrollRange->Inherited,
    SpeedParameters->Inherited}];

RootsAssignmentNB:=
    Notebook[{Cell[
        BoxData[GridBox[{{"Root Definition"},{ObtainNumberB},{\
SelectionModificationButton},{EnterRootKeyboard},{"Root Assignment"},{\
ToPolynomialAsRootB},{ResetPolynomialB},{"Polynomial Navigator"},{\
PreviousPolyB},{NextPolyB},{"Close this window"},{CloseComplexWindowsB}},\
        RowSpacings->{1,0,0,1,1,0,1,1,0,1,1,0}]]]],
    WindowFrame->Palette, WindowSize->FitAll,
    WindowElements->{}, WindowFrameElements->{"CloseBox"},
    ShowCellBracket->False,
    Background->Options[ButtonBox,Background][[1,2]],
    CellMargins->{{0,0},{0,0}}, CellFrame->1,
    CellFrameMargins->{{0,0},{0,4}}, Editable->False, WindowTitle-
>"Denominator Matrix Elements Definition",
    WindowFloating->True,
    WindowMargins->{{Automatic,0},{Automatic,0}}];

ComplexPlaneNB=Notebook[{Cell[GraphicsData["PostScript", "\<\
%!
%%Creator: Mathematica
%%AspectRatio: .415
MathPictureStart
/Mabs {
Mgmatrix idtransform
Mtmatrix dtransform

```

```

} bind def
/Mabsadd { Mabs
3 -1 roll add
3 1 roll add
exch } bind def
%% Graphics
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10 scalefont setfont
% Scaling calculations
0.97619 0.952381 0.0147151 0.588604 [
[.02381 .00222 -6 -9 ]
[.02381 .00222 6 0 ]
[.11905 .00222 -12 -9 ]
[.11905 .00222 12 0 ]
[.21429 .00222 -12 -9 ]
[.21429 .00222 12 0 ]
[.30952 .00222 -12 -9 ]
[.30952 .00222 12 0 ]
[.40476 .00222 -12 -9 ]
[.40476 .00222 12 0 ]
[.5 .00222 -12 -9 ]
[.5 .00222 12 0 ]
[.59524 .00222 -12 -9 ]
[.59524 .00222 12 0 ]
[.69048 .00222 -12 -9 ]
[.69048 .00222 12 0 ]
[.78571 .00222 -12 -9 ]
[.78571 .00222 12 0 ]
[.88095 .00222 -12 -9 ]
[.88095 .00222 12 0 ]
[.96369 .07358 -30.625 -6.28125 ]
[.96369 .07358 0 6.28125 ]
[.96369 .13244 -30.625 -6.28125 ]
[.96369 .13244 0 6.28125 ]
[.96369 .1913 -30.625 -6.28125 ]
[.96369 .1913 0 6.28125 ]
[.96369 .25016 -30.625 -6.28125 ]
[.96369 .25016 0 6.28125 ]
[.96369 .30902 -30.625 -6.28125 ]
[.96369 .30902 0 6.28125 ]
[.96369 .36788 -30.625 -6.28125 ]
[.96369 .36788 0 6.28125 ]
[.96369 .42674 -30.625 -6.28125 ]
[.96369 .42674 0 6.28125 ]
[.96369 .4856 -30.625 -6.28125 ]
[.96369 .4856 0 6.28125 ]
[.96369 .54446 -30.625 -6.28125 ]
[.96369 .54446 0 6.28125 ]
[.96369 .60332 -10 -6.28125 ]
[.96369 .60332 0 6.28125 ]
[.5 .63053 -241.25 0 ]
[.5 .63053 241.25 15.1875 ]
[ 0 0 0 0 ]
[ 1 .61803 0 0 ]
] MathScale
% Start of Graphics
1 setlinecap
1 setlinejoin
newpath

```

.8 g  
 .25 Mabswid  
 [ ] 0 setdash  
 .88095 0 m  
 .88095 .61803 L  
 s  
 .78571 0 m  
 .78571 .61803 L  
 s  
 .69048 0 m  
 .69048 .61803 L  
 s  
 .59524 0 m  
 .59524 .61803 L  
 s  
 .5 0 m  
 .5 .61803 L  
 s  
 .40476 0 m  
 .40476 .61803 L  
 s  
 .30952 0 m  
 .30952 .61803 L  
 s  
 .21429 0 m  
 .21429 .61803 L  
 s  
 .11905 0 m  
 .11905 .61803 L  
 s  
 .02381 0 m  
 .02381 .61803 L  
 s  
 0 .07358 m  
 1 .07358 L  
 s  
 0 .13244 m  
 1 .13244 L  
 s  
 0 .1913 m  
 1 .1913 L  
 s  
 0 .25016 m  
 1 .25016 L  
 s  
 0 .30902 m  
 1 .30902 L  
 s  
 0 .36788 m  
 1 .36788 L  
 s  
 0 .42674 m  
 1 .42674 L  
 s  
 0 .4856 m  
 1 .4856 L  
 s  
 0 .54446 m  
 1 .54446 L  
 s

```

0 .60332 m
1 .60332 L
s
0 g
.02381 .01472 m
.02381 .02097 L
s
[(-1)] .02381 .00222 0 1 Mshowa
.11905 .01472 m
.11905 .02097 L
s
[(-0.9)] .11905 .00222 0 1 Mshowa
.21429 .01472 m
.21429 .02097 L
s
[(-0.8)] .21429 .00222 0 1 Mshowa
.30952 .01472 m
.30952 .02097 L
s
[(-0.7)] .30952 .00222 0 1 Mshowa
.40476 .01472 m
.40476 .02097 L
s
[(-0.6)] .40476 .00222 0 1 Mshowa
.5 .01472 m
.5 .02097 L
s
[(-0.5)] .5 .00222 0 1 Mshowa
.59524 .01472 m
.59524 .02097 L
s
[(-0.4)] .59524 .00222 0 1 Mshowa
.69048 .01472 m
.69048 .02097 L
s
[(-0.3)] .69048 .00222 0 1 Mshowa
.78571 .01472 m
.78571 .02097 L
s
[(-0.2)] .78571 .00222 0 1 Mshowa
.88095 .01472 m
.88095 .02097 L
s
[(-0.1)] .88095 .00222 0 1 Mshowa
0 .01472 m
1 .01472 L
s
.97619 .07358 m
.98244 .07358 L
s
gsave
.96369 .07358 -91.625 -10.2813 Mabsadd m
1 1 Mabs scale
currentpoint translate
0 20.5625 translate 1 -1 scale
0.996 0.996 0.996 setrgbcolor
newpath
61.000 16.563 moveto
100000.000 16.563 lineto
100000.000 4.000 lineto

```

```

61.000 4.000 lineto
61.000 16.563 lineto
closepath
fill
0.000 0.000 0.000 setrgbcolor
63.000 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(0.1) show
83.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
83.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(i) show
89.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
1.000 setlinewidth
grestore
.97619 .13244 m
.98244 .13244 L
s
gsave
.96369 .13244 -91.625 -10.2813 Mabsadd m
1 1 Mabs scale
currentpoint translate
0 20.5625 translate 1 -1 scale
0.996 0.996 0.996 setrgbcolor
newpath
61.000 16.563 moveto
100000.000 16.563 lineto
100000.000 4.000 lineto
61.000 4.000 lineto
61.000 16.563 lineto
closepath
fill
0.000 0.000 0.000 setrgbcolor
63.000 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(0.2) show

```

```

83.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
83.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(i) show
89.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
1.000 setlinewidth
grestore
.97619 .1913 m
.98244 .1913 L
s
gsave
.96369 .1913 -91.625 -10.2813 Mabsadd m
1 1 Mabs scale
currentpoint translate
0 20.5625 translate 1 -1 scale
0.996 0.996 0.996 setrgbcolor
newpath
  61.000 16.563 moveto
  100000.000 16.563 lineto
  100000.000 4.000 lineto
  61.000 4.000 lineto
  61.000 16.563 lineto
closepath
fill
0.000 0.000 0.000 setrgbcolor
63.000 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(0.3) show
83.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
83.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont

```

```

0.000 0.000 0.000 setrgbcolor
(i) show
89.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
1.000 setlinewidth
grestore
.97619 .25016 m
.98244 .25016 L
s
gsave
.96369 .25016 -91.625 -10.2813 Mabsadd m
1 1 Mabs scale
currentpoint translate
0 20.5625 translate 1 -1 scale
0.996 0.996 0.996 setrgbcolor
newpath
  61.000 16.563 moveto
  100000.000 16.563 lineto
  100000.000 4.000 lineto
  61.000 4.000 lineto
  61.000 16.563 lineto
closepath
fill
0.000 0.000 0.000 setrgbcolor
63.000 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(0.4) show
83.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
83.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(i) show
89.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
1.000 setlinewidth
grestore
.97619 .30902 m

```



```

.98244 .30902 L
s
gsave
.96369 .30902 -91.625 -10.2813 Mabsadd m
1 1 Mabs scale
currentpoint translate
0 20.5625 translate 1 -1 scale
0.996 0.996 0.996 setrgbcolor
newpath
  61.000 16.563 moveto
  100000.000 16.563 lineto
  100000.000 4.000 lineto
  61.000 4.000 lineto
  61.000 16.563 lineto
closepath
fill
0.000 0.000 0.000 setrgbcolor
63.000 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(0.5) show
83.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
83.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(i) show
89.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
1.000 setlinewidth
grestore
.97619 .36788 m
.98244 .36788 L
s
gsave
.96369 .36788 -91.625 -10.2813 Mabsadd m
1 1 Mabs scale
currentpoint translate
0 20.5625 translate 1 -1 scale
0.996 0.996 0.996 setrgbcolor
newpath
  61.000 16.563 moveto
  100000.000 16.563 lineto
  100000.000 4.000 lineto

```

```

61.000 4.000 lineto
61.000 16.563 lineto
closepath
fill
0.000 0.000 0.000 setrgbcolor
63.000 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(0.6) show
83.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
83.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(i) show
89.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
1.000 setlinewidth
grestore
.97619 .42674 m
.98244 .42674 L
s
gsave
.96369 .42674 -91.625 -10.2813 Mabsadd m
1 1 Mabs scale
currentpoint translate
0 20.5625 translate 1 -1 scale
0.996 0.996 0.996 setrgbcolor
newpath
61.000 16.563 moveto
100000.000 16.563 lineto
100000.000 4.000 lineto
61.000 4.000 lineto
61.000 16.563 lineto
closepath
fill
0.000 0.000 0.000 setrgbcolor
63.000 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(0.7) show

```

```

83.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
83.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(i) show
89.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
1.000 setlinewidth
grestore
.97619 .4856 m
.98244 .4856 L
s
gsave
.96369 .4856 -91.625 -10.2813 Mabsadd m
1 1 Mabs scale
currentpoint translate
0 20.5625 translate 1 -1 scale
0.996 0.996 0.996 setrgbcolor
newpath
  61.000 16.563 moveto
  100000.000 16.563 lineto
  100000.000 4.000 lineto
  61.000 4.000 lineto
  61.000 16.563 lineto
closepath
fill
0.000 0.000 0.000 setrgbcolor
63.000 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(0.8) show
83.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
83.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont

```

```

0.000 0.000 0.000 setrgbcolor
(i) show
89.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
1.000 setlinewidth
grestore
.97619 .54446 m
.98244 .54446 L
s
gsave
.96369 .54446 -91.625 -10.2813 Mabsadd m
1 1 Mabs scale
currentpoint translate
0 20.5625 translate 1 -1 scale
0.996 0.996 0.996 setrgbcolor
newpath
  61.000 16.563 moveto
  100000.000 16.563 lineto
  100000.000 4.000 lineto
  61.000 4.000 lineto
  61.000 16.563 lineto
closepath
fill
0.000 0.000 0.000 setrgbcolor
63.000 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(0.9) show
83.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
83.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(i) show
89.625 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
1.000 setlinewidth
grestore
.97619 .60332 m

```

```

.98244 .60332 L
s
gsave
.96369 .60332 -71 -10.2813 Mabsadd m
1 1 Mabs scale
currentpoint translate
0 20.5625 translate 1 -1 scale
0.996 0.996 0.996 setrgbcolor
newpath
 61.000 16.563 moveto
100000.000 16.563 lineto
100000.000 4.000 lineto
 61.000 4.000 lineto
 61.000 16.563 lineto
closepath
fill
0.000 0.000 0.000 setrgbcolor
63.000 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
63.000 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(i) show
69.000 12.813 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 10.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
1.000 setlinewidth
grestore
.97619 0 m
.97619 .61803 L
s
gsave
.5 .63053 -302.25 -4 Mabsadd m
1 1 Mabs scale
currentpoint translate
0 23.1875 translate 1 -1 scale
/MISOfy
{
  /newfontname exch def
  /oldfontname exch def

  oldfontname findfont
  dup length dict begin
    {1 index /FID ne {def} {pop pop} ifelse} forall
    /Encoding WindowsANSIEncoding def
  currentdict
end

```

```

        newfontname exch definefont pop
    } def
    0.996 0.996 0.996 setrgbcolor
    newpath
        61.000 19.188 moveto
        100000.000 19.188 lineto
        100000.000 4.000 lineto
        61.000 4.000 lineto
        61.000 19.188 lineto
    closepath
    fill
    0.000 0.000 0.000 setrgbcolor
    63.000 15.188 moveto
    %%IncludeResource: font Courier
    %%IncludeFont: Courier
    /Courier findfont 16.000 scalefont
    [1 0 0 -1 0 0 ] makefont setfont
    0.000 0.000 0.000 setrgbcolor
    0.000 0.000 rmoveto
    63.000 15.188 moveto
    %%IncludeResource: font Courier
    %%IncludeFont: Courier
    /Courier findfont 16.000 scalefont
    [1 0 0 -1 0 0 ] makefont setfont
    0.000 0.000 0.000 setrgbcolor
    (Quartile) show
    149.063 15.188 moveto
    (of) show
    177.750 15.188 moveto
    (the) show
    216.000 15.188 moveto
    (Complex) show
    292.500 15.188 moveto
    (Plane) show
    340.313 15.188 moveto
    %%IncludeResource: font Courier
    %%IncludeFont: Courier
    /Courier findfont 16.000 scalefont
    [1 0 0 -1 0 0 ] makefont setfont
    0.000 0.000 0.000 setrgbcolor
    (: ) show
    %%IncludeResource: font Math2Mono-Bold
    %%IncludeFont: Math2Mono-Bold
    /Math2Mono-Bold findfont 16.000 scalefont
    [1 0 0 -1 0 0 ] makefont setfont
    0.000 0.000 0.000 setrgbcolor
    359.438 15.188 moveto
    (8) show
    369.000 15.188 moveto
    %%IncludeResource: font Courier
    %%IncludeFont: Courier
    /Courier findfont 16.000 scalefont
    [1 0 0 -1 0 0 ] makefont setfont
    0.000 0.000 0.000 setrgbcolor
    (z) show
    %%IncludeResource: font Math1Mono-Bold
    %%IncludeFont: Math1Mono-Bold
    /Math1Mono-Bold findfont 16.000 scalefont
    [1 0 0 -1 0 0 ] makefont setfont
    0.000 0.000 0.000 setrgbcolor

```

```

(e) show
%%IncludeResource: font Math5Mono-Bold
%%IncludeFont: Math5Mono-Bold
/Math5Mono-Bold findfont 16.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(\\302) show
%%IncludeResource: font Math2Mono-Bold
%%IncludeFont: Math2Mono-Bold
/Math2Mono-Bold findfont 16.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
407.625 15.188 moveto
(\\220) show
426.750 15.188 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 16.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(Rez) show
455.438 15.188 moveto
%%IncludeResource: font Math1Mono-Bold
%%IncludeFont: Math1Mono-Bold
/Math1Mono-Bold findfont 16.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(<) show
465.000 15.188 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 16.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(0) show
474.563 15.188 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 16.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(,) show
484.125 15.188 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 16.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(Imz) show
512.813 15.188 moveto
%%IncludeResource: font Math1Mono-Bold
%%IncludeFont: Math1Mono-Bold
/Math1Mono-Bold findfont 16.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(\\263) show
522.375 15.188 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 16.000 scalefont

```

```

[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
(0) show
%%IncludeResource: font Math2Mono-Bold
%%IncludeFont: Math2Mono-Bold
/Math2Mono-Bold findfont 16.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
531.938 15.188 moveto
(<) show
541.500 15.188 moveto
%%IncludeResource: font Courier
%%IncludeFont: Courier
/Courier findfont 16.000 scalefont
[1 0 0 -1 0 0 ] makefont setfont
0.000 0.000 0.000 setrgbcolor
0.000 0.000 rmoveto
1.000 setlinewidth
grestore
0 0 m
1 0 L
1 .61803 L
0 .61803 L
closepath
clip
newpath
.008 w
.97619 .60332 Mdot
.02381 .01472 Mdot
% End of Graphics
MathPictureEnd
\
\>"], "Graphics",CellFrame->True,CellTags->"CPlot",
ShowCellTags->False,ShowCellBracket->False,
DefaultNaturalLanguage->Language,AspectRatioFixed->False,
ImageSize->{ComplexPlaneWidth,Automatic},
ImageMargins->{{5,0},{0,0}},ImageRegion->{{0,1},{0,1}},
FontFamily->"Courier New CE"],
Cell[BoxData[""],"Input",Selectable->False,
ShowCellBracket->False,
CellMargins->{{Inherited,Inherited},{2.` ,2.` }},
CellTags->"Complex"],
Cell["","Input",Selectable->False,CellFrame->False,
ShowCellBracket->False,
CellMargins->{{Inherited,Inherited},{2.` ,2.` }},
ShowCellTags->True,CellTags->"Status"],
Cell["","Input",Selectable->False,CellFrame->False,
ShowCellBracket->False,
CellMargins->{{Inherited,Inherited},{2.` ,2.` }},
ShowCellTags->True,CellTags->"Roots Remaining"],
Cell["","Input",Selectable->False,ShowCellBracket->False,
CellMargins->{{Inherited,Inherited},{2.` ,2.` }},
ShowCellTags->True,CellTags->"Polynomial"]},
ShowCellTags->True,WindowSize->{2500,2500},WindowTitle-
>"Denominator Matrix Definition...Please define the elements-polynomials
by selecting their roots",
ShowSelection->False,
ScrollingOptions->{PagewiseScrolling->False,
PagewiseDisplay->Inherited,RealTimeScrolling->Inherited,
JumpScrolling->Inherited,HorizontalScrollRange->Inherited,

```



```

        SpeedParameters->Inherited},
        WindowFrameElements->{"ZoomBox", "ResizeArea"}];

(*PLOT WINDOW*)

ResponsePlotsNB:=
  Notebook[{Cell["Feedback System Time Response"],
    Cell["", CellTags->"Response Plot", ShowCellTags->True]}]

(*PLOT FUNCTIONS*)

MyDisplayFunction[
  gr:(_Graphics|_Graphics3D|_SurfaceGraphics|_DensityGraphics|_
  ContourGraphics)]:=
  Module[{psstring, cellexpr},
    NotebookLocate[{ResponsePlotsNBObject, "Response Plot"}];

    SelectionMove[ResponsePlotsNBObject, All, Cell]; psstring=DisplayString[gr];

    cellexpr=Cell[TextData[{Cell[GraphicsData["PostScript", psstring]}], "Grap
    hics",
      TextAlignment->Center, CellTags->"Response Plot",
      ShowCellTags->True];
    NotebookWrite[ResponsePlotsNBObject, cellexpr]]

DisplayResponse[OutputIndex_, RespVec_, tmaximum_]:=
  Plot[Evaluate[ComplexExpand[RespVec[[OutputIndex, 1]]], {t, 0, tmaximum},
    DisplayFunction->MyDisplayFunction,
    PlotLabel->
      StyleForm[
        StringForm["`1` System Output Response when `2` Input is the `3`
        function",
          Subscript["Y", OutputSelection], Subscript["U", InputSelection],
          InputType]
        , FontSize->20], PlotRange->All, ImageSize->600]

(* RESPONSE COMPUTATION FUNCTIONS *)

FSTransferFunction[Plant_, XL_, YL_]:=
  Inverse[IdentityMatrix[Dimensions[Plant][[1]]]+
    Plant.Inverse[XL].YL].Plant.Inverse[XL].YL

ResponseVector[H_, InputVec_, var1_, var2_]:=
  Simplify[ComplexExpand[InverseLaplaceTransform[H.InputVec, var1, var2]]]

(* PARAMETER VALUES CONTROL WINDOW*)

IncreaseIncrementButton:=
  ButtonBox["\[Times]10", Active->True, ButtonEvaluator->Local,
    ButtonFrame->"DialogBox",
    ButtonFunction:>(ParIncrement=ParIncrement*10;
    RefreshParametersIncrement)]

DecreaseIncrementButton:=
  ButtonBox[SuperscriptBox["\[Times]10", "-1"], Active->True,
    ButtonEvaluator->Local, ButtonFrame->"DialogBox",
    ButtonFunction:>(ParIncrement=ParIncrement/10;
    RefreshParametersIncrement)]

EnterIncrementButton:=

```

```

    ButtonBox[ToBoxes[ParIncrement],Active->True,ButtonFrame->"Palette",
      ButtonEvaluator->Local]

RefreshParametersIncrement:=(NotebookLocate["Increment"];
  NotebookWrite[SelectedNotebook[],
    Cell[BoxData[
      GridBox[{"Increment: i = ",EnterIncrementButton,
        IncreaseIncrementButton,DecreaseIncrementButton}],
      ColumnSpacings->0.1]],CellTags->"Increment"]);
  SetOptions[SelectedNotebook[],WindowSize->FitAll])

ParameterIncreaseButton[i_Integer]:=
  ButtonBox["+",Active->True,ButtonFrame->"DialogBox",
    ButtonEvaluator->Local,
    ButtonFunction:>(tparval[[i]]=tparval[[i]]+ParIncrement;
      RefreshParameterValue;ParameterChangeAction )])

ParameterDecreaseButton[i_Integer]:=
  ButtonBox["-",Active->True,ButtonEvaluator->Local,ButtonFrame-
    >"DialogBox",
    ButtonFunction:>(tparval[[i]]=tparval[[i]]-ParIncrement;
      RefreshParameterValue; ParameterChangeAction )])

EnterParameterValue[i_Integer]:=
  ButtonBox[ToBoxes[N[tparval[[i]],Infinity]],ButtonEvaluator->Local,
    Active->True,ButtonFrame->"Palette"]

ParameterChangeAction:=
  Module[{TM},(TM=
    ClosedLoopSystemTransferFunction//.Table[
      t[i]->tparval[[i]],{i,1,numofparameters}];
    ResVec=ResponseVector[TM,InputVector,s,t];
    DisplayResponse[OutputSelection,ResVec,Maxt]))

RefreshParameterValue:=(NotebookLocate["Parameters"];
  NotebookWrite[SelectedNotebook[],
    Cell[BoxData[
      GridBox[Table[{SubscriptBox["t",j],"=",EnterParameterValue[j],
        ParameterIncreaseButton[j],ParameterDecreaseButton[j]},{j,
          numofparameters}],ColumnSpacings->0.1]],
      CellTags->"Parameters"]];
    SetOptions[SelectedNotebook[],WindowSize->FitAll])

CloseParametersGraphControlNotebook:=
  ButtonBox["Close",Active->True,ButtonEvaluator->Local,
    ButtonFrame->"DialogBox",
    ButtonFunction:>
      NotebookClose[ParametersGraphControlNotebookObject]]

ParametersGraphControlNB[i_Integer]:=
  Notebook[{Cell[
    BoxData[GridBox[{"Increment: i = ",EnterIncrementButton,
      IncreaseIncrementButton,DecreaseIncrementButton}],
      ColumnSpacings->0.1]],CellTags->"Increment"],
    Cell[BoxData[
      GridBox[Table[{SubscriptBox["t",j],"=",EnterParameterValue[j],
        ParameterIncreaseButton[j],ParameterDecreaseButton[j]},{j,i}],
      ColumnSpacings->0.1]],CellTags->"Parameters"],

```

```

Cell[BoxData[CloseParametersGraphControlNotebook]]}, ShowCellTags-
>True,
WindowFrame->"Palette", WindowSize->FitAll,
ShowCellBracket->False,
WindowMargins->{{Automatic, 0}, {Automatic, 0}},
TextAlignment->Center, WindowElements->{}]

(* INPUT OUTPUT CONTROL WINDOW *)

OutputSelectorUpButton:=
  ButtonBox["\[UpArrow]", Active->True, ButtonFrame->"DialogBox",
    ButtonEvaluator->Local,
    ButtonFunction:>
      If[OutputSelection<NumOut, (OutputSelection++;
        NotebookLocate[{SelectedNotebook[], "Output"}];
        NotebookWrite[SelectedNotebook[],
          Cell[BoxData[
            RowBox[{"Display Output Signal: ", OutputSelection, " ",
              OutputSelectorUpButton, OutputSelectorDownButton}],
            CellTags->"Output", ShowCellTags->True,
            ShowCellBracket->False], All];
          DisplayResponse[OutputSelection, ResVec, Maxt]])]

OutputSelectorDownButton:=
  ButtonBox["\[DownArrow]", Active->True, ButtonFrame->"DialogBox",
    ButtonEvaluator->Local,
    ButtonFunction:>
      If[OutputSelection>1, (OutputSelection--;
        NotebookLocate[{SelectedNotebook[], "Output"}];
        NotebookWrite[SelectedNotebook[],
          Cell[BoxData[
            RowBox[{"Display Output Signal: ", OutputSelection, " ",
              OutputSelectorUpButton, OutputSelectorDownButton}],
            CellTags->"Output", ShowCellTags->True,
            ShowCellBracket->False], All];
          DisplayResponse[OutputSelection, ResVec, Maxt]])]

TypeOfInputButton[type_String]:=
  ButtonBox[type, Active->True, ButtonEvaluator->Local, ButtonFrame-
  >"DialogBox",
    ButtonFunction:> (If[type=="Step", InputType="Impulse";
      InputVector=ZeroVector[NumIn]; InputVector[[InputSelection]]=1;
      InputVector=Partition[InputVector, 1], InputType="Step";

InputVector=ZeroVector[NumIn]; InputVector[[InputSelection]]=1/s;
InputVector=Partition[InputVector, 1]];
NotebookLocate[{SelectedNotebook[], "Type Of Input"}];
NotebookWrite[SelectedNotebook[],
  Cell[BoxData[
    RowBox[{"Type Of Input Function: ",
      TypeOfInputButton[InputType]}]], CellTags->"Type Of
Input",
    ShowCellTags->True, ShowCellBracket->False], All];
Module[{TM}, {TM=
  ClosedLoopSystemTransferFunction//.Table[
    t[i]->tparval[[i]], {i, 1, numofparameters}];
ResVec=ResponseVector[TM, InputVector, s, t];
DisplayResponse[OutputSelection, ResVec, Maxt]})]

```

```

InputSelectorUpButton:=
  ButtonBox["\[UpArrow]",Active->True,ButtonFrame->"DialogBox",
    ButtonEvaluator->Local,
    ButtonFunction:>
      If[InputSelection<NumIn,(InputSelection++;
        NotebookLocate[{SelectedNotebook[],"Input"}];
        NotebookWrite[SelectedNotebook[],
          Cell[BoxData[
            RowBox[{"Apply Input Signal: ",InputSelection," ",
              InputSelectorUpButton,InputSelectorDownButton}]],
            CellTags->"Input",ShowCellTags->True,
            ShowCellBracket->False],All];
        If[InputType=="Impulse",InputVector=ZeroVector[NumIn];
          InputVector[[InputSelection]]=1;
          InputVector=Partition[InputVector,1],

InputVector=ZeroVector[NumIn];InputVector[[InputSelection]]=1/s;
  InputVector=Partition[InputVector,1]];
  Module[{TM},(TM=
    ClosedLoopSystemTransferFunction//.Table[
      t[i]->tparval[[i]},{i,1,numofparameters}]];
    ResVec=ResponseVector[TM,InputVector,s,t];
    DisplayResponse[OutputSelection,ResVec,Maxt])]]]

InputSelectorDownButton:=
  ButtonBox["\[DownArrow]",Active->True,ButtonFrame->"DialogBox",
    ButtonEvaluator->Local,
    ButtonFunction:>
      If[InputSelection>1,(InputSelection--;
        NotebookLocate[{SelectedNotebook[],"Input"}];
        NotebookWrite[SelectedNotebook[],
          Cell[BoxData[
            RowBox[{"Apply Input Signal: ",InputSelection," ",
              InputSelectorUpButton,InputSelectorDownButton}]],
            CellTags->"Input",ShowCellTags->True,
            ShowCellBracket->False],All];
        If[InputType=="Impulse",InputVector=ZeroVector[NumIn];
          InputVector[[InputSelection]]=1;
          InputVector=Partition[InputVector,1],

InputVector=ZeroVector[NumIn];InputVector[[InputSelection]]=1/s;
  InputVector=Partition[InputVector,1]];
  Module[{TM},(TM=
    ClosedLoopSystemTransferFunction//.Table[
      t[i]->tparval[[i]},{i,1,numofparameters}]];
    ResVec=ResponseVector[TM,InputVector,s,t];
    DisplayResponse[OutputSelection,ResVec,Maxt])]]]

MaxTimeButton:=
  ButtonBox[Maxt,ButtonFrame->"DialogBox",ButtonEvaluator->Local,
    Active->True,
    ButtonFunction:>(Maxt=GetMaxt;
      NotebookLocate[{SelectedNotebook[],"Time t Range"}];
      NotebookWrite[SelectedNotebook[],
        Cell[BoxData[RowBox[{"Plot Response up to t=",MaxTimeButton}]],
          CellTags->"Time t Range",ShowCellTags->True,
          ShowCellBracket->False],All];
      DisplayResponse[OutputSelection,ResVec,Maxt])

GetMaxt:=Module[{tMax},(tMax=Input["Set the maximum value for t:"]);

```

```

If[NumericQ[tMax]&&Positive[tMax],tMax,GetMaxt]])

InputOutputNB:=
  Notebook[{Cell[
    BoxData[RowBox[{"Apply Input Signal: ",InputSelection," ",
      InputSelectorUpButton,InputSelectorDownButton}]],CellTags-
>"Input",
    ShowCellTags->True,ShowCellBracket->False],
    Cell[BoxData[
      RowBox[{"Type Of Input Function:
",TypeOfInputButton[InputType]}]],
      CellTags->"Type Of Input",ShowCellTags->True,
      ShowCellBracket->False],
      Cell[BoxData[
        RowBox[{"Display Output Signal: ",OutputSelection," ",
OutputSelectorUpButton,OutputSelectorDownButton}]],CellTags->"Output",
        ShowCellTags->True,ShowCellBracket->False],
        Cell[BoxData[RowBox[{"Plot Response up to t=",MaxTimeButton}]],
        CellTags->"Time t Range",ShowCellTags->True,
        ShowCellBracket->False}],WindowFrame->"Palette",
        WindowSize->FitAll,WindowTitle->"Input/Output Control",
        WindowMargins->{{Automatic,0},{0,Automatic}},
        TextAlignment->Center,WindowElements->{}]

(* BASIC BUTTON THAT TRIGGERS FUNCTIONALITY *)

ResponsePlotsButton:=
  ButtonBox["Feedback System Response Plots",Active->True,
    ButtonEvaluator->Automatic,
    ButtonFunction:>(If[ComputedCompensator,
      SetOptions[MainButtonsPaletteObject,Editable->True,
        Visible->False];({NumOut,
        NumIn}={Dimensions[StrictlyProperPlant][[1]],
        Dimensions[StrictlyProperPlant][[1]]};Maxt=10;
        tparval=ZeroVector[numofparameters];ParIncrement=1;
        InputSelection=1;OutputSelection=1;InputType="Impulse";

InputVector=ZeroVector[NumIn];InputVector[[InputSelection]]=1;
InputVector=Partition[InputVector,1];
ClosedLoopSystemTransferFunction=
  FSTransferFunction[StrictlyProperPlant,XL,YL];
ResVec=ResponseVector[
  ClosedLoopSystemTransferFunction//.Table[
    t[i]->tparval[[i]},{i,1,numofparameters}],
    InputVector,s,t];
ResponsePlotsNBObject=NotebookPut[ResponsePlotsNB];
ParametersGraphControlNBObject=
  NotebookPut[ParametersGraphControlNB[numofparameters]];
InputOutputNBObject=NotebookPut[InputOutputNB];
DisplayResponse[OutputSelection,ResVec,10]),(SelectionMove[
  MainDisplayNBObject,All,Cell];
NotebookWrite[MainDisplayNBObject,
  Cell[TextData[
    "Cannot compute the closed loop system transfer
function. Compute class of Compensators first."]\
,CellTags->"Class Of Compensators",ShowCellTags->False,Selectable->True],
  After]]])

(*Initialization and display of MainDisplayNotebook*)

```

```
MainDisplayNBObject=  
  NotebookPut [  
    MainDisplayNB];NotebookCenter[MainDisplayNBObject];\  
MainButtonsPaletteObject=  
  NotebookPut[MainButtonsPalette];ComputedCompensator=False;  
  
On [General::"spell1"]  
On [General::"spell"]  
  
EndPackage[]
```