Indian Institute of Technology Bombay

Pole Placement in Single Input Systems & Computation of Controllability Indices

B. Tech Project

Shubham Chandak
Final Year Undergraduate Student
Electrical Engineering

Supervisor
Prof. Harish Pillai

May 3, 2016

Contents

Part I
Pole Placement in Single Input Systems

Abstract

We look at applications of Gröbner basis in control theory, in particular to the pole placement problem through linear state feedback. A new relation between the feedback coefficients and the pole positions is presented and discussed. The proposed formula is compuatonally efficient if only a small number of poles need to be shifted. Also, we gain insight into relation between the feedback vector and (generalized) left eigenspaces of the system matrix. Section 1 contains background material, Section 2 contains the method and the main result. Section 3 looks at the implications of the result and Section 4 presents proofs for the result.

1 Background

1.1 Gröbner Basis

Consider a multivariate polynomial ring $K[x_1, x_2, \ldots, x_n]$ where K is a field. An ideal I is a subset of the ring which forms a additive subgroup and is closed under multiplication by any polynomial in the ring. For example, all polynomials with zero constant coefficient form an ideal. An ideal is said to be generated by polynomials $f_1, \ldots, f_m$ if every polynomials in the ideal can be written as $c_1 f_1 + \cdots + c_m f_m$ where c_i's are polynomials in the ring. The polynomial ring over a field is Noetherian ring, i.e. all the ideals are finitely generated. Contrast this with univariate polynomial ring where all the ideals are principal, i.e. generated by a single polynomial which is the GCD of the finitely many generators.

Before defining Gröbner basis, we need a notion of term orders and reductions. A monomial order is a total ordering defined on the set of monomials in $K[x_1, x_2, \ldots, x_n]$ such that (i) $m \geq 1, \forall m$ and (ii) $m_1 > m_2 \Rightarrow m_1 m_3 > m_2 m_3$. We have used the lexicographic term order in which we first define $x_1 > x_2 > \cdots > x_n$ and then any two monomials are compared first by their degree in x_1 and so on. Now given $f, g \in K[x_1, x_2, \ldots, x_n]$, we can generalize the notion of Euclidean division in the following way. f is said to be reducible by g if some monomial in f is a multiple of the leading monomial of g. Similarly f is said to be lead-reducible by g if the leading monomial in f is a multiple of the leading monomial of g. Thus f can be reduced by subtracting the appropriate multiple of g. Given a set of polynomials G we can reduce f w.r.t. these polynomials in some order until no further reductions are possible. Ideally we would want that f reduces to 0 iff it belongs to the ideal generated by G. But in general the result of the reduction can depend on the order in which the reductions are carried out and might not be 0 even if f is in the ideal.

Gröbner basis is a basis for the ideal which solves the above problems. Let $G = \{g_1, \ldots, g_t\}$ and I be the ideal generated by G. Then the following are equivalent definitions of G being a Gröbner basis[1]:

1. For all $f \in I$, there exists i such that leading monomial of g_i divides the leading monomial of f
2. The ideal formed by the leading terms in G is same as the ideal formed by the leading terms in I
3. $f \in I$ iff f reduces to 0 w.r.t. G
4. If f is reduced to r_1 and r_2 w.r.t. G then $r_1 = r_2$

For a given ideal and term ordering, the Gröbner basis is not unique since it is always possible to add more polynomials lying in the ideal while maintaining the properties. But if we put the additional constraint that each polynomial has leading coefficient 1 and that each of the polynomials in the Gröbner basis is completely reduced w.r.t. the rest then we get a reduced Gröbner basis which is unique.

There are various algorithms to compute the Gröbner basis such as Buchberger's algorithm and its improved versions. The computational complexity of finding the Gröbner basis is high and depends heavily on the particular ideal and term order in consideration. There are various interesting properties of Gröbner basis which lead to various applications. Some of these include[2]:

- Checking if $f \in I$ can be done by reducing f w.r.t. the Gröbner basis of the ideal.
- The standard monomials are the monomials not belonging to the ideal formed by the leading terms. These form a vector space basis for $K[x_1, x_2, \ldots, x_n]/I$ and thus give information about the dimension of the ideal.
- Gröbner basis can be used to find the intersection of ideals, quotient ideals and radical of ideals.
- Gröbner basis can be used to study Syzygy modules.

Here we've used one particular property: Elimination property.

1.1.1 Elimination Property

Given an ideal I we might be interested in studying the corresponding variety which is the subset of K^n where all the polynomials of I evaluate to 0. This means that we wish to solve a system of polynomial equations simultaneously by forming the ideal. For linear equations the standard procedure is Gaussian Elimination where we eliminate certain variables from certain equations. Gröbner basis generalizes this to polynomials if we use a lexicographic term ordering. For this we need to find the equations in I only involving certain variables.

We have the following theorem[1]:
THEOREM: Let I be a non-zero ideal of $K[x_1, x_2, \ldots, x_n]$ and let the Gröbner basis w.r.t. lexicographic ordering $x_1 < x_2 \cdots < x_n$ be $G = \{g_1, \ldots, g_t\}$. Then $G \cap K[x_1, \ldots, x_m]$ is a Gröbner basis for the ideal $I \cap K[x_1, \ldots, x_m]$ where $m < n$.

Thus if take those polynomials in G which involve only $x_1, \ldots, x_m$ then these form a Gröbner basis for the elimination ideal. Thus we can first eliminate all but one variable and solve the corresponding univariate system. Then we can substitute the solution in the rest of the equations and proceed similarly to get all solutions of the system of equations.

1.1.2 Readings for Gröbner basis

[2] has basic introduction to the abstract algebra prerequisites and an introduction to the Grobner basis theory and algorithms. [1] also covers the basic Gröbner basis theory and its applications including elimination ideals, Hilbert Nullstellensatz, Syzygy computation and improved Buchberger algorithm.

I read some chapters of [3] which introduces ideals and varieties, covers Gröbner bases and some applications. Chapter 4 deals with Hilbert's Nullstellensatz, radical ideals, Zariski closure and the general relationships between ideals and their varieties. Chapter 5 introduces functions on a variety and their relation with $k[\mathbf{x}]/I$ where I is the radical ideal associated with the variety. This is followed up in chapter 9 where dimension of a variety is considered. First monomial ideals are considered and the relation between the dimension of variety and the standard monomials is introduced. In case the number of standard monomials are infinite the Hilbert function and its properties are shown. Chapter 8 is on projective spaces, homogenous ideals and projective varieties.

I also read chapters 1 to 9 of [4]. This book talks about toric ideals using Gröbner bases and the corresponding polytopes and their applications to integer programming. It also has chapters on find primitive partitions and regular triangulations and their relations to toric ideals. It introduces Universal Gröbner basis (which is a Gröbner basis w.r.t. all term orders) and State polytopes and then specializes these notions for toric ideals which ideals that occur as kernels of certain homomorphisms.

1.2 Controllability and Pole Placement [5], [6]

Consider the single input system

$$\dot{\mathbf{x}} = A\mathbf{x} + Bu$$

$$\mathbf{y} = C\mathbf{x} + Du$$

Here $\mathbf{x}$ is the state vector, u is the input and $\mathbf{y}$ is the output vector. The eigenvalues of A correspond to the poles of the system. To change the system dynamics we can provide state feedback i.e. the new input to the system is $u + K\mathbf{x}$ (where K is a row vector). Then the equations become

$$\dot{\mathbf{x}} = (A + BK)\mathbf{x} + Bu$$

$$\mathbf{y} = C\mathbf{x} + Du$$

The question which arises is the following: Is it possible to place the eigenvalues of $A + BK$ at arbitrary positions?

Definition (Controllablity): The pair (A, B) is said to be controllable if for any initial state $\mathbf{x}(0) = \mathbf{x}_0$ and any final state $\mathbf{x}_1$, there exists an input that transfers $\mathbf{x}_0$ to $\mathbf{x}_1$ in finite time.

Thus controllability means that the states can be completely controlled using the input. It can be shown that the following are equivalent (Here the state is n-dimensional and input is 1-dimensional):

1. The system (A, B) is controllable.

2. The n x n controllability matrix $C = [B \ AB \ A^2B \ \dots \ A^{n-1}B]$ has rank n (it is invertible).

3. K can be chosen to place eigenvalues of $A + BK$ at any specified location.

4. The matrix $[A - \lambda I \ B]$ has full rank for all values of λ.

Now consider a controllable single-input system. Pole placement is the problem of finding the feedback matrix K such that the new system has some specified poles/eigenvalues. There are several standard methods for doing so:

1.2.1 Controller Canonical Form (CCF)

In the controller canonical form the matrices A and B are of the form

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}$$

where $a_0, \dots, a_{n-1}$ are the coefficients of the characteristic polynomial of A, $p(\lambda) = \lambda^n + a_{n-1}\lambda^{n-1} + \cdots + a_0$.

In such a case

$$A + BK = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -a_0 + k_1 & -a_1 + k_2 & -a_2 + k_3 & \cdots & -a_{n-1} + k_n \end{bmatrix}$$

where the matrix K is

$$K = \begin{bmatrix} k_1 & k_2 & \cdots & k_n \end{bmatrix}$$

Thus the pole placement problem reduces to finally the desired characteristic polynomial and then choosing k_i's so that the coefficients match the entries in $A + BK$.

In case A is not in the controller canonical form, we can do a change of basis and obtain the CCF (if the system is controllable). This is done as follows. First we find the CCF using the characteristic equation of A. Then we compute the comtrollabililty matrices C for the original system and the controllability matrix C_{CCF} in the controller canonical form. The change of basis matrix P is given by CC_{CCF}^{-1} and then $(P^{-1}AP, P^{-1}B)$ is in controller canonical form. Once we find K_{CCF}, we can go back to the original basis by $K = K_{CCF}P^{-1}$.

1.2.2 Bass-Gura formula

This formula directly finds the feedback vector in the original representation by transforming the feedback vector in the CCF.

$$K = -\begin{bmatrix}(a'_0 - a_0) & (a'_1 - a_1) & \cdots & (a'_{n-1} - a_{n-1})\end{bmatrix}(CC_{CCF}^{-1})^{-1}$$

where $a_0, \cdots, a_{n-1}$ are the coefficients of the original characteristic equation, $a'_0, \cdots, a'_{n-1}$ are the coefficients of the desired characteristic equation, C and C_{CCF} are the controllability matrices in the original representation and the CCF respectively. It can be shown that C_{CCF} is symmetric and its inverse is

$$C_{CCF}^{-1} = \begin{bmatrix} a_1 & a_2 & \cdots & a_{n-1} & 1 \\ a_2 & a_3 & \cdots & 1 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n-1} & 1 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 \end{bmatrix}$$

The Bass-Gura formula is just a change of basis from the CCF back to the original representation.

1.2.3 Ackermann's formula

Ackermann's formula can also be used to directly compute the feedback matrix without going to the CCF.

$$K = -\begin{bmatrix}0 & 0 & \cdots & 0 & 1\end{bmatrix}C^{-1}p'(A)$$

where C is the controllability matrix and $p'(\lambda)$ is the desired characteristic equation. The minus sign is because we have $A+BK$ instead of $A-BK$. The derivation, which is detailed in [6], starts from the expression for K in the canonical form and involves the use of the matrix C_{CCF}^{-1} and the structure of A_{CCF}.

1.2.4 Mayne-Murdoch formula [7]

Consider a system with distinct eigenvalues. The system is diagonalizable and assume that we are working the diagonalized basis. Then

$$k_i b_i = -\frac{\prod_{j=1}^{n}(\lambda_i - \lambda'_j)}{\prod_{j=1, j\neq i}^{n}(\lambda_i - \lambda_j)} \tag{1}$$

where k_i and b_i are the components of the vectors K and B respectively in the basis where A is diagonalized. $\lambda_1, \cdots, \lambda_n$ and $\lambda'_1, \cdots, \lambda'_n$ are the original and desired eigenvalues respectively. Note that in case some eigenvalue is unchanged, the corresponding product $k_i b_i$ is zero.

For the derivation of this formula, we first obtain the characteristic equation of $A + BK$ in terms of $k_i b_i$ by using Sylvester's determinant identity (note that A is diagonal). Then

this is compared to the desired characteristic equation by putting in certain values of λ and using the fact that two monic polynomials of degree n are equal if and only if they agree at n distinct points.

2 Problem Formulation and Gröbner Basis

We looked at the the pole placement problem for single input linear systems using Gröbner basis. The notation is as follows: A is an n x n matrix with entries $\{a_{ij}\}$, $\mathbf{b}$ is an n x 1 column vector with entries b_i and $\mathbf{k}$ is an n x 1 column vector with entries k_i. The eigenvalues of the original matrix A are $\lambda_1, \lambda_2, \ldots, \lambda_n$. The eigenvalues of the matrix after feedback, i.e. $A + \mathbf{b}\mathbf{k}^T$ are $\lambda_1 + c_1, \lambda_2 + c_2, \ldots, \lambda_n + c_n$. Note the difference from the previous formulations. Here our aim to get particular shifts in the eigenvalues rather than just obtaining a new set of eigenvalues. This is helpful in case we wish to find feedback vector $\mathbf{k}$ which moves only one eigenvalue and keeps the other eigenvalues fixed.

The ideal for studying the problem from a Gröbner basis perspective is the ideal formed by the relations for the symmetric polynomials of the eigenvalues. The polynomial ring is over the variables $a_{11}, \ldots, a_{nn},\ b_1, \ldots, b_n,\ k_1, \ldots, k_n,\ \lambda_1, \ldots, \lambda_n,\ c_1, \ldots, c_n$. For sake of computational ease we worked mostly with $n = 3$ but the results can be easily generalized for higher values of n. For $n = 3$ the ideal is generated by six polynomials:
(i) $a_{11} + a_{22} + a_{33} - \lambda_1 - \lambda_2 - \lambda_3$
(ii) $p - \lambda_1\lambda_2 - \lambda_1\lambda_3 - \lambda_2\lambda_3$ where p is the coefficient of λ in the characteristic equation of A
(iii) $det(A) - \lambda_1\lambda_2\lambda_3$

The other three equations are the corresponding equations for the eigenvalues $\lambda_1 + c_1, \lambda_2 + c_2, \lambda_3 + c_3$ of $A + \mathbf{b}\mathbf{k}^T$. The ordering used for the Gröbner basis was lexicographic but different orderings of the variables were considered.

Throughout the project SAGE was used for the computations. Although I looked at some other packages like Mathematica and Maple, SAGE was the most convenient. It has functions for finding Gröbner basis w.r.t. different term orders, finding elimination ideals, basic matrix and polynomial functions, finding Primary decompostion of ideals etc. It also has functions for finding the State polytope and Universal Gröbner basis but that was not used in this project.

2.1 Known A and b

If the system $(A, \mathbf{b})$ is known, the Gröbner basis can reveal information about the controllability and other properties. For instance if we eliminate $\lambda_1, \lambda_2, \lambda_3, k_1, k_2, k_3$, then we can recover the conditions on c_1, c_2, c_3. If the system is controllable there are no such conditions and we get (0) as the elimination ideal. Otherwise the system is uncontrollable.

It is observed that even if only the first eigenvalue is uncontrollable the elimination ideal is not (c_1). This happens because of two reasons:

- Due to the symmetry of the ideal, a particular variable λ_i is not assigned to a particular eigenvalue of A. This symmetry can be broken to get more specific results by using Primary decomposition of ideals. This basically means that a set of equations (or its corresponding solution set) can be broken down into simpler pieces. For example, the

ideal (xz, yz) corresponds to the variety: $xy - plane$ and $z - axis$. Decomposing this into the two pieces corresponds to the primary decomposition of the ideal into (x, y) and (z).

In our case on doing primary decomposition we get ideals corresponding to different permutations of $\lambda_1, \lambda_2, \lambda_3$. If we take the prime ideal corresponding to the intended assignment of eigenvalues e.g. the ideal with $(\lambda_1 - 1, \lambda_2 - 2, \lambda_3 - 3)$ as opposed to the ideal with $(\lambda_1 - 2, \lambda_2 - 3, \lambda_3 - 1)$ (where 1, 2 and 3 are the eigenvalues of A). After this eliminating the variables tells us precisely which eigenvalues are uncontrollable. As a by product, the primary decomposition also gives out the eigenvalues of A and hence is not useful when A and $\mathbf{b}$ are variables.

- It is also possible that λ_2 becomes λ_1 and λ_1 changes to something else and hence c_1 need not be zero. This issue arises because we are considering the changes in eigenvalues rather than the final eigenvalue and the mapping between these is not unique. For example, to move $\{1, 2, 3\}$ to $\{4, 5, 6\}$ we can have $c_1 = c_2 = c_3 = 3$ or we can have $c_1 = 5, c_2 = 3, c_1 = 1$. A similar artifact comes up later when we discuss the main result.

2.2 Main Result

In general when A and $\mathbf{b}$ are variable, the following relation was obtained by using Gröbner basis and then reducing the LHS polynomials w.r.t. the Gröbner basis:

For $n = 3$,

$$C^T\mathbf{k} = \sum_{i=1}^{3} c_i \begin{bmatrix} 1 \\ \lambda_i \\ \lambda_i^2 \end{bmatrix} - \sum_{1 \le i < j \le 3} c_i c_j \begin{bmatrix} 0 \\ 1 \\ \lambda_i + \lambda_j \end{bmatrix} + c_1 c_2 c_3 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{2}$$

where C^T is the transpose of the controllability matrix.

The ordering used for Gröbner basis computation was $k_i > c_i > \lambda_i > a_{ij} > b_i$ and then k_3 and k_2 were eliminated. The resulting ideal had some equations with only the eigenvalues and matrix A (due to the ordering chosen) and had one equation which was linear in k_1. The coefficient of k_1 was identified as the determinant of the controllability matrix - giving a clue that the expression is of the form $C^T\mathbf{k} = RHS$ or $C\mathbf{k} = RHS$. Then on reducing $C^T\mathbf{k}$ w.r.t. the Gröbner basis, the right hand side was obtained.

Given this equation it is possible to find out the $\mathbf{k}$ to get the appropriate eigenvalues by using Cramer's rule or any method of solving linear equations. Though the result should be the same as that obtained by other methods, these equations can give us some more information about the problem as discussed in the next section. Before that we consider the

generalization of the equation for larger value of n. For $n = 4$,

$$C^T\mathbf{k} = \sum_{i=1}^{4} c_i \begin{bmatrix} 1 \\ \lambda_i \\ \lambda_i^2 \\ \lambda_i^3 \end{bmatrix} - \sum_{1\le i<j\le 4} c_i c_j \begin{bmatrix} 0 \\ 1 \\ \lambda_i+\lambda_j \\ \lambda_i^2+\lambda_i\lambda_j+\lambda_j^2 \end{bmatrix} + \sum_{1\le i<j<k\le 4} c_i c_j c_k \begin{bmatrix} 0 \\ 0 \\ 1 \\ \lambda_i+\lambda_j+\lambda_k \end{bmatrix} - c_1c_2c_3c_4 \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Note that

$$\begin{bmatrix} 0 \\ 1 \\ \lambda_i+\lambda_j \\ \lambda_i^2+\lambda_i\lambda_j+\lambda_j^2 \end{bmatrix} = \frac{1}{\lambda_i-\lambda_j}\left(\begin{bmatrix} 1 \\ \lambda_i \\ \lambda_i^2 \\ \lambda_i^3 \end{bmatrix} - \begin{bmatrix} 1 \\ \lambda_j \\ \lambda_j^2 \\ \lambda_j^3 \end{bmatrix}\right)$$

$$\begin{bmatrix} 0 \\ 0 \\ 1 \\ \lambda_i+\lambda_j+\lambda_k \end{bmatrix} = \frac{1}{\lambda_i-\lambda_j}\left(\begin{bmatrix} 0 \\ 1 \\ \lambda_i+\lambda_k \\ \lambda_i^2+\lambda_i\lambda_k+\lambda_k^2 \end{bmatrix} - \begin{bmatrix} 0 \\ 1 \\ \lambda_j+\lambda_j \\ \lambda_j^2+\lambda_j\lambda_k+\lambda_k^2 \end{bmatrix}\right)$$

and

$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \frac{1}{\lambda_i-\lambda_j}\left(\begin{bmatrix} 0 \\ 0 \\ 1 \\ \lambda_i+\lambda_k+\lambda_l \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ 1 \\ \lambda_j+\lambda_k+\lambda_l \end{bmatrix}\right)$$

In this way equations for higher values of n can be obtained. The relations above have implications which are discussed in the next section. Also, in case $\lambda_i = \lambda_j$ the final formula still holds but the interpretation is different.

2.3 Simplified expression for the main result

Note that if most of the c_i's are non-zero, the number of terms on the right hand side of 2 grows exponentially with n. However, since the cross terms are obtained as differences (as seen in for $n = 4$ above), the expression can be simplified to (for **distinct eigenvalues**):

$$C^T\mathbf{k} = -\sum_{i=1}^{n} \frac{\prod_{j=1}^{n}(\lambda_i - \lambda'_j)}{\prod_{j=1, j\neq i}^{n}(\lambda_i - \lambda_j)} \begin{bmatrix} 1 \\ \lambda_i \\ \lambda_i^2 \\ \vdots \\ \lambda_i^{n-1} \end{bmatrix} \tag{3}$$

where $\lambda_1, \cdots, \lambda_n$ and $\lambda'_1, \cdots, \lambda'_n$ are the original and desired eigenvalues respectively. Note the resemblance of the coefficients to the ones in the Mayne-Murdoch formula (1). This way of expressing is helpful while deriving the formula from the standard methods of pole placement.

2.4 Repeated eigenvalues

In case A has repeated eigenvalues, the simplified expression 3 is no longer valid though the original formula 2 still works. For $n = 3$ and $\lambda_1 = \lambda_2 = \lambda_3 = \lambda$ we get,

$$C^T\mathbf{k} = (c_1 + c_2 + c_3)\begin{bmatrix}1\\ \lambda\\ \lambda^2\end{bmatrix} - (c_1c_2 + c_2c_3 + c_3c_1)\begin{bmatrix}0\\ 1\\ 2\lambda\end{bmatrix} + c_1c_2c_3\begin{bmatrix}0\\ 0\\ 1\end{bmatrix} \tag{4}$$

Let $\mathbf{\Lambda} = \begin{bmatrix}1 & \lambda & \cdots & \lambda^{n-1}\end{bmatrix}^T$ and let $\mathbf{\Lambda}^k$ denote the kth derivative (componentwise) of $\mathbf{\Lambda}$ w.r.t. λ. Clearly if $\mathbf{\Lambda} = \begin{bmatrix}1 & \lambda & \lambda^2\end{bmatrix}^T$, then $\mathbf{\Lambda}^1 = \begin{bmatrix}0 & 1 & 2\lambda\end{bmatrix}^T$ and $\mathbf{\Lambda}^2/2! = \begin{bmatrix}0 & 0 & 1\end{bmatrix}^T$. These are precisely the terms in the above expression.

In general, let $\lambda_1 = \cdots = \lambda_k = \lambda$ and rest of the eigenvalues be distinct. Then the formula reduces to

$$C^T\mathbf{k} = \frac{\prod_{j=k+1}^{n}(\lambda - \lambda'_j)}{\prod_{j=k+1}^{n}(\lambda - \lambda_j)}\tilde{\mathbf{\Lambda}} - \sum_{i=1}^{n}\frac{\prod_{j=1}^{n}(\lambda_i - \lambda'_j)}{\prod_{j=1,j\neq i}^{n}(\lambda_i - \lambda_j)}\begin{bmatrix}1\\ \lambda_i\\ \lambda_i^2\\ \vdots\\ \lambda_i^{n-1}\end{bmatrix} \tag{5}$$

where $\tilde{\mathbf{\Lambda}}$ generalizes the RHS of equation 4 for $n > 3$.

The case of repeated eigenvalues is more complicated and it seems difficult to give a proof for general n.

3 Discussions

For simplicity, assume $n = 3$.

3.1 Subsets of eigenvalues and left eigenspaces

First we discuss the case where we wish to move only one eigenvalue i.e. $c_1 \neq 0, c_2 = 0, c_3 = 0$. Equation 2 reduces to

$$C^T\mathbf{k} = c_1\begin{bmatrix}1\\ \lambda_1\\ \lambda_1^2\end{bmatrix} \tag{6}$$

First of all, we observe that $\mathbf{k}$ is directly proportional to the change c_1. Now the RHS vector is actually the eigenvector of A_{CCF} (A in its controller canonical form) since

$$\begin{bmatrix}0 & 1 & 0\\ 0 & 0 & 1\\ -a_0 & -a_1 & -a_2\end{bmatrix}\begin{bmatrix}1\\ \lambda_1\\ \lambda_1^2\end{bmatrix} = \begin{bmatrix}\lambda_1\\ \lambda_1^2\\ -a_0 - a_1\lambda_1 - a_2\lambda_1^2\end{bmatrix} = \lambda_1\begin{bmatrix}1\\ \lambda_1\\ \lambda_1^2\end{bmatrix}$$

since λ_1 satisfies the characteristic equation $\lambda^3 + a_2\lambda^2 + a_1\lambda + a_0$.

Thus to further simplify things we left multiply equation 6 by A_{CCF}. Then LHS =

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 \end{bmatrix} \begin{bmatrix} \mathbf{b}^T \\ \mathbf{b}^T A^T \\ \mathbf{b}^T (A^2)^T \end{bmatrix} \mathbf{k} = \begin{bmatrix} \mathbf{b}^T A^T \\ \mathbf{b}^T (A^2)^T \\ \mathbf{b}^T (-I - a_1 A - a_2 A^2)^T \end{bmatrix} \mathbf{k}$$

$$= C^T A^T \mathbf{k} = c_1 \lambda_1 \begin{bmatrix} 1 \\ \lambda_1 \\ \lambda_1^2 \end{bmatrix}$$

where we have used the Cayley-Hamilton theorem. Now subtracting λ_1 times equation 6 from this we obtain $C^T(A^T - \lambda I)\mathbf{k} = 0$. If the system is controllable this reduces to the fact that $\mathbf{k}^T$ is the left eigenvector of A corresponding to the eigenvalue λ_1.

Thus we get the following:

- If the system is controllable then the feedback vector needed to move only one eigenvalue is in the same direction as the left eigenvector of A corresponding to that eigenvalue.
- Looking at the formula 3 (for distinct eigenvalues), it is immediate from linearity that the feedback required for changing $\lambda_1, \cdots, \lambda_k$ is a linear combination of the left eigenvectors corresponding to these eigenvalues.
- In general (for distinct eigenvalues), if eigenvalues $\lambda_1, \cdots, \lambda_k$ are changed, the right eigenvectors for the rest of the eigenvalues is unchanged.

  This is because the feedback vector is a linear combination of the left eigenvectors corresponding to these eigenvalues which is orthogonal to right eigenvectors corresponding to $\lambda_{k+1}, \cdots, \lambda_n$ by diagonalization arguments. Hence $(A + \mathbf{b}\mathbf{k}^T)\mathbf{v}_i = A\mathbf{v}_i = \lambda_i \mathbf{v}_i$ for $i = k+1, \cdots, n$ where $\mathbf{v}_i$ is the right eigenvector of A corresponding to λ_i. However, nothing can be said about the left eigenvectors when $k > 1$.

3.1.1 Repeated eigenvalues and generalized eigenvectors

We saw above that the feedback $\mathbf{k}$ needed to move a set of eigenvalues is a linear combination of the left eigenvectors corresponding to these eigenvalues. However when we have repeated eigenvalues, the situation is slightly different. Let λ be a repeated root with multiplicity 2 and let $\mathbf{\Lambda}$ be as defined in section 2.4. By using the fact that λ is a repeated root, we can easily show that

$$(A_{CCF} - \lambda I)\mathbf{\Lambda} = 0$$

and

$$(A_{CCF} - \lambda I)^2 \mathbf{\Lambda}^1 = 0$$

where $\mathbf{\Lambda}^1$ is the derivative as in section 2.4. Thus $\mathbf{\Lambda}^1$ is a generalized eigenvector of A_{CCF} and using similar arguments as before, the corresponding $\mathbf{k}$ is the generalized left eigenvector of A.

For a controllable system, we have

- The matrix A is cyclic/simple, i.e. for each repeating eigenvalue we have a single block in the Jordan form. The generalized right eigenvectors can be explicitly found out in the CCF.

- Let λ be a repeated eigenvalue with multiplicity k. To move only one of these, $\mathbf{k}$ is in direction of left eigenvector. To move $l < k$ of these eigenvalues, the feedback $\mathbf{k}$ is a linear combination of the generalized left eigenvectors of ranks 1 to l.

To summarize, the space in which $\mathbf{k}$ lies is closely related to the left generalized eigenspace of A.

3.1.2 The cross-terms

We noted in section 2.1 that our choice of variables as the shifts in eigenvalues (c_i) rather than the final desired eigenvalues (λ_i') leaves open a number of ways to achieve the same result. Here we study how this is captured by our formula. Let's say we want to move λ_1 to λ_1' while keeping the rest same. Note that the feedback $\mathbf{k}$ should not depend on the particular way this is achieved and so the RHS of the equation should also be invariant. We consider the following two ways:

- $c_1 = \lambda_1' - \lambda_1, c_2 = c_3 = 0$: Here the RHS is $(\lambda_1' - \lambda_1)[1, \lambda_1, \lambda_1^2]^T$

- $c_1 = \lambda_2 - \lambda_1, c_2 = (\lambda_1' - \lambda_1) + (\lambda_1 - \lambda_2), c_3 = 0$: Here λ_2 moves to λ_1' via λ_1 and λ_1 changes to λ_2. Then the RHS =

$$c_1 \begin{bmatrix} 1 \\ \lambda_1 \\ \lambda_1^2 \end{bmatrix} + c_2 \begin{bmatrix} 1 \\ \lambda_2 \\ \lambda_2^2 \end{bmatrix} - \frac{c_1 c_2}{\lambda_1 - \lambda_2} \left(\begin{bmatrix} 1 \\ \lambda_1 \\ \lambda_1^2 \end{bmatrix} - \begin{bmatrix} 1 \\ \lambda_2 \\ \lambda_2^2 \end{bmatrix} \right)$$

$$= (\lambda_2 - \lambda_1) \begin{bmatrix} 1 \\ \lambda_1 \\ \lambda_1^2 \end{bmatrix} + ((\lambda_1' - \lambda_1) + (\lambda_1 - \lambda_2)) \left(\begin{bmatrix} 1 \\ \lambda_2 \\ \lambda_2^2 \end{bmatrix} + \begin{bmatrix} 1 \\ \lambda_1 \\ \lambda_1^2 \end{bmatrix} - \begin{bmatrix} 1 \\ \lambda_2 \\ \lambda_2^2 \end{bmatrix} \right)$$

$$= (\lambda_1' - \lambda_1) \begin{bmatrix} 1 \\ \lambda_1 \\ \lambda_1^2 \end{bmatrix}$$

Similarly we can move all three eigenvalues to get the desired final result but the thing to note is the way the cross-terms cancel out so that the final result remains invariant. Thus the case when we are moving more than one eigenvalue can be seen as an extension where we don't add any new directions (for $\mathbf{k}$) but add cross terms to make sure that expression is invariant w.r.t. the above cases (the vector with $c_1 c_2$ is a linear combination of the vectors with c_1 and c_2 and so on).

3.2 Computational efficiency

We compare the computational efficiency of our formula to the methods mentioned in section 1.2.

In our formula, once the right hand side is calculated we just need to solve a linear system which can be done by a variety of means. However the time required to calculate the right hand side is heavily dependent on the number of eigenvalues we wish to change. In cases where only a few eigenvalues need to be changed, this time is negligible.

Both the Bass-Gura and Ackermann's formula need one inversion as well other operations. For instance the Bass-Gura formula involves a matrix product and the computation of the coefficients of the characterisitc equations. Ackermann's formula requires the computation of $p'(A)$ and some matrix multiplications. These expressions do not leverage the fact that only a few eigenvalues need to be changed. Thus our formula is more efficient in such cases.

The Mayne-Murdoch formula does simplify when only few eigenvalues need to be moved. However it requires the computation of all eigenvalues in order to diagonalize A which makes it much slower.

4 Proof for the Result

Although the Gröbner basis constitutes a proof for the formula for $n = 3$ and $n = 4$, we present the derivations from the existing formulae which work for larger values of n. Throughout this section, assume for simplicity that A has **distinct eigenvalues**. Also assume that the system is **controllable**.

It should be noted that the LHS of 3 is invariant to the basis since $C_1 = C_2 P^{-1}$ and $\mathbf{k}_2^T = \mathbf{k}_1^T P^{-1}$ where P is the change of basis matrix between bases 1 and 2. Thus it is sufficient to obtain the result in some convenient basis.

4.1 From Mayne-Murdoch formula

Let us assume that we are working in the diagonalized basis. Then according to the Mayne-Murdoch formula (1),

$$k_i b_i = -\frac{\prod_{j=1}^{n}(\lambda_i - \lambda'_j)}{\prod_{j=1, j\neq i}^{n}(\lambda_i - \lambda_j)}$$

Also if A is diagonal with entries $\lambda_1, \cdots, \lambda_n$ then,

$$C^T\mathbf{k} = \begin{bmatrix} \sum_{i=1}^n b_i k_i \\ \sum_{i=1}^n b_i \lambda_i k_i \\ \vdots \\ \sum_{i=1}^n b_i \lambda_i^{n-1} k_i \end{bmatrix} = \sum_{i=1}^n b_i k_i \begin{bmatrix} 1 \\ \lambda_i \\ \vdots \\ \lambda_1^{n-1} \end{bmatrix}$$

which directly gives our result (3).

To get the Mayne-Murdoch formula from our result, consider the equation

$$\sum_{i=1}^{n} b_i k_i \begin{bmatrix} 1 \\ \lambda_i \\ \vdots \\ \lambda_1^{n-1} \end{bmatrix} = -\sum_{i=1}^{n} \frac{\prod_{j=1}^{n}(\lambda_i - \lambda'_j)}{\prod_{j=1, j\neq i}^{n}(\lambda_i - \lambda_j)} \begin{bmatrix} 1 \\ \lambda_i \\ \lambda_i^2 \\ \vdots \\ \lambda_i^{n-1} \end{bmatrix}$$

Think of the above equations as n equations in n variables $b_1k_1, \cdots, b_nk_n$ which have a unique solution because the coefficient matrix is Vandermonde.

4.2 From Controller Canonical Form

The Bass-Gura formula and the Ackermann's formula follow directly from the expression for $\mathbf{k}$ in the CCF. To derive our result in this basis, assume that we are working in the CCF. Let $p(\lambda) = \lambda^n + a_{n-1}\lambda^{n-1} + \cdots + a_0$ and $p'(\lambda) = \lambda^n + \alpha_{n-1}\lambda^{n-1} + \cdots + \alpha_0$ represent the original and desired characteristic equations.

We will use a result from [6] which says that C_{CCF} is symmetric and its inverse is

$$C_{CCF}^{-1} = \begin{bmatrix} a_1 & a_2 & \cdots & a_{n-1} & 1 \\ a_2 & a_3 & \cdots & 1 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n-1} & 1 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 \end{bmatrix}$$

We will start will our formula (3) and show that it reduces to the standard $\mathbf{k}$ in the CCF as in section 1.2.1. First consider

$$(C_{CCF}^T)^{-1} \begin{bmatrix} 1 \\ \lambda_i \\ \lambda_i^2 \\ \vdots \\ \lambda_i^{n-1} \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & \cdots & a_{n-1} & 1 \\ a_2 & a_3 & \cdots & 1 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n-1} & 1 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} 1 \\ \lambda_i \\ \lambda_i^2 \\ \vdots \\ \lambda_i^{n-1} \end{bmatrix}$$

It can be shown by induction that this vector is the vector of coefficients of the monic polynomial of degree $n-1$ with roots $\lambda_1, \cdots, \lambda_{i-1}, \lambda_{i+1}, \cdots, \lambda_n$. For example, for $n = 3$,

$$\begin{bmatrix} a_1 & a_2 & 1 \\ a_2 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ \lambda_1 \\ \lambda_1^2 \end{bmatrix} = \begin{bmatrix} \lambda_2\lambda_3 \\ -(\lambda_2 + \lambda_3) \\ 1 \end{bmatrix}$$

Multiplying our formula by C_{CCF}^{-1} and thinking of vectors as representing coefficients of polynomials of degree $n-1$, the left hand side (from CCF representation) is

$$\mathbf{k} = -\begin{bmatrix} a_0 - \alpha_0 \\ \vdots \\ a_{n-1} - \alpha_{n-1} \end{bmatrix} \equiv p(\lambda) - p'(\lambda) \tag{7}$$

According to the action of C_{CCF}^{-1} on $\begin{bmatrix}1 & \lambda & \cdots & \lambda^{n-1}\end{bmatrix}^T$ just described, the right hand side's polynomial representation is

$$-\sum_{i=1}^{n} \frac{\prod_{j=1}^{n}(\lambda_i - \lambda'_j)}{\prod_{j=1,j\neq i}^{n}(\lambda_i - \lambda_j)} \prod_{j=1,j\neq i}^{n}(\lambda - \lambda_j) \tag{8}$$

To prove our result we just need to show that the polynomials in equations 7 and 8 are the same. This can be done by showing that they agree on the n points $\lambda_1, \cdots, \lambda_n$ thus completing the proof.

5 Conclusion

We have presented a new formula for computation of feedback $\mathbf{k}$ using the Gröbner basis of the ideal. This formula is computationally efficient when only a few eigenvalues need to be changed. We also obtained relations between the feedback vector and the left generalized eigenspace of A. Finally we derived the result in two ways from the existing methods.

The case when the system is not controllable is more complicated since some eigenvalues cannot be moved at all while the feedback is non-unique for other eigenvalues. We also plan to work on extending the formula for the multi-input case. However the multi-input case is much more complicated since the equations are not linear in K and the feedback is not unique for a given pole assignment. Thus we do not expect the simple interpretation in terms of left eigenvectors in that case.

Part II
Controllability Indices using Exterior Algebra

Abstract

We look at applications of exterior algebra in control theory. The problem of computing the controllability indices, which are important invariants of a multi-input linear system, is studied using these tools. We present two algorithms for the case of two-input systems which are based on computing ranks of multivectors. Section 6 contains background material on exterior algebra and on controllability indices. Section 7 introduces the problem and the motivation leading up to the proposed algorithms. Section 8 presents the proposed algorithms along with their proofs of correctness.

6 Background

6.1 Exterior Algebra

Exterior algebra or Grassmann algebra provides tools to manipulate multivectors and subspaces in vector spaces. The following introduction is sufficient for the puposes here and is based on [8], [9], [10], [11] and [12].

Notation: v_i denotes vectors in the vector space V and e_i is the basis. $\langle v_1, \dots v_r \rangle$ represents the subspace spanned by $v_1, \dots, v_n$. x_i, y_i will denote elements of the exterior algebra $\Lambda(V)$. X_{ij} will denote the Plücker coordinates of the multivectors.

Let V be a real vector space of dimension n. The **exterior algebra** over V is a graded algebra

$$\Lambda(V) = \Lambda^0(V) \oplus \Lambda^1(V) \oplus \cdots \oplus \Lambda^n(V)$$

where $\Lambda^0(V)$ is $\mathbb{R}$, the underlying field, $\Lambda^1(V)$ is same as V. Elements of the exterior algebra are called **multivectors** and elements of $\Lambda^r(V)$ are called r-vectors.

For $2 \leq r \leq n$, $\Lambda^r(V)$ consists of linear combinations of elements of the form $v_1 \wedge v_2 \wedge \cdots \wedge v_r$ where $v_1, \dots, v_r \in V$. The wedge operator $\wedge$ can be thought of a placeholder satisfying certain properties:

1. **Anticommutativity**: In $\Lambda^2(V)$, $v_1 \wedge v_2 = -v_2 \wedge v_1$. In general for $\Lambda^r(V)$, $v_1 \wedge v_2 \wedge \cdots \wedge v_r = \operatorname{sgn}(\sigma) v_{\sigma(1)} \wedge v_{\sigma(2)} \wedge \cdots \wedge v_{\sigma(r)}$ where σ is some permutation and $\operatorname{sgn}(\sigma)$ is its sign. For example, in $\Lambda^3(V)$, $v_1 \wedge v_2 \wedge v_3 = -v_1 \wedge v_3 \wedge v_2 = v_3 \wedge v_1 \wedge v_2$, etc.

2. **Multilinearity**: In $\Lambda^r(V)$,

$$v_1 \wedge \cdots \wedge v_i \wedge \cdots \wedge v_r \; + \; v_1 \wedge \cdots \wedge v_i' \wedge \cdots \wedge v_r = v_1 \wedge \cdots \wedge (v_i + v_i') \wedge \cdots \wedge v_r$$

These properties are similar to the properties of determinants and we'll see that exterior products are closely related to determinants.

Any $x \in \Lambda(V)$ can be uniquely written as $x = x_0 + x_1 + \cdots + x_n$ where $x_r \in \Lambda^r(V)$. The multiplication $\wedge$ in $\Lambda(V)$ is associative, distributive, but not commutative. We have

$$x \wedge y = (-1)^{pq} y \wedge x, \quad x \in \Lambda^p(V), y \in \Lambda^q(V)$$

Note that this implies that $v \wedge v = 0$ for any vector $v \in V$. Thus any multiplication operation can be done using distributive property and anticommutativity.

The above conditions make sure that the exterior algebra over V is finite dimesional. The next two theorems highlight the basic structure of $\Lambda(V)$ after which some some examples are presented.

Theorem 6.1 [9]: If V is a n-dimensional vector then $\Lambda(V)$ is 2^n-dimensional. Given a basis $e_1, \ldots, e_n$ is a basis for V, then

$$e_{i_1} \wedge e_{i_2} \wedge \cdots \wedge e_{i_r}, \quad 1 \leq i_1 < \cdots < i_r \leq n, \ r = 1, 2, \ldots, n$$

From the basis of $\Lambda(V)$, it is clear that the dimension of $\Lambda^r(V)$ is $\binom{n}{r}$.

Theorem 6.2 [8]: Vectors $v_1, v_2, \ldots, v_r \in V$ are linearly dependent if and only if $v_1 \wedge v_2 \wedge \cdots \wedge v_r = 0$.

Definition 6.3: A multivector x is said to be **decomposable** if it can be written in the form $x = v_1 \wedge v_2 \wedge \cdots \wedge v_r$ for some vectors $v_1, \ldots, v_r \in V$.

Decomposability of multivectors is important for our applications and will be covered in more detail later.

Example 6.4: Let V be 4 dimensional with basis e_1, e_2, e_3, e_4. Then $\Lambda^0(V)$ is the same as $\mathbb{R}$ (dimension 1), $\Lambda^1(V)$ is the same as V (dimension 4). $\Lambda^2(V)$ is 6 dimensional with basis $\{e_1 \wedge e_2, e_1 \wedge e_3, e_1 \wedge e_4, e_2 \wedge e_3, e_2 \wedge e_4, e_3 \wedge e_4\}$. $\Lambda^3(V)$ is 4 dimensional with basis $\{e_1 \wedge e_2 \wedge e_3, e_1 \wedge e_2 \wedge e_4, e_1 \wedge e_3 \wedge e_4, e_2 \wedge e_3 \wedge e_4\}$. Finally $\Lambda^4(V)$ is 1 dimensional with basis $e_1 \wedge e_2 \wedge e_3 \wedge e_4$.

- $e_1 \wedge e_2 \wedge (e_1 + e_2) = 0$ by Theorem 6.2 or by expanding as
  $e_1 \wedge e_2 \wedge e_1 + e_1 \wedge e_2 \wedge e_2$
  $= -e_1 \wedge e_1 \wedge e_2 + e_1 \wedge e_2 \wedge e_2$
  $= -0 \wedge e_2 + e_1 \wedge 0$
  $= 0$.
- $e_1 \wedge e_2 + e_3 \wedge e_1$ is decomposable since it is the same as $e_1 \wedge (e_2 - e_3)$.
- $e_1 \wedge e_2 + e_3 \wedge e_4$ is not decomposable. This will be proved later.

6.1.1 Grassmannian and Plücker Coordinates[10], [11]

If a decomposable multivector (say $x = v_1 \wedge \cdots \wedge v_r$) is not 0, then it corresponds to the r-dimensional subspace of V spanned by $v_1, \ldots, v_r$. The space can also be described as

$$W = \{v \in V \mid x \wedge v = 0\}$$

This is because for any linear combination of $v_1, \ldots, v_r$, $x \wedge v$ is 0 by Theorem 6.2. If we take some other basis $v'_1, \ldots v'_r$ of the same subspace, then $x' = v'_1 \wedge \cdots \wedge v'_r$ is a non-zero scalar multiple of x (this will be shown later using Plücker coordinates). Thus r-dimensional subspaces of V can be associated with decomposable multivectors in $\Lambda^r(V)$, unique upto a scalar multiple.

Definition 6.5 The **Grassmannian** $\mathbf{Gr}(r, V)$ is the space of all r-dimensional subspaces of V.

As per the preceding discussion, the Grassmannian can be embedded in the projective space corresponding to $\Lambda^r(V)$. The subspaces correspond to the decomposable multivectors. Formally we have,

Definition 6.6 The **Plücker embedding** is the map ι

$$\iota : \mathbf{Gr}(r, V) \to \mathbf{P}(\Lambda^r(V))$$

where $\iota(\langle v_1, \ldots, v_r \rangle) = v_1 \wedge \cdots \wedge v_r$ and $\mathbf{P}(\Lambda^r(V))$ is the projective space.

Using the standard basis for $\Lambda^r(V)$ from Theorem 6.1, the $\binom{n}{r}$ coordinates of $v_1 \wedge \cdots \wedge v_r$ are called **Plücker coordinates** and can be computed by building the $n \times r$ matrix $\begin{bmatrix} v_1 & \ldots & v_n \end{bmatrix}$ and then the coordinate correspoding to $e_{i_1} \wedge \cdots \wedge e_{i_r}$ is the determinant of the $r \times r$ submatrix formed by the rows $e_{i_1}, \ldots, e_{i_r}$. These coordinates satisfy a set of quadratic conditions called the **Plücker relations**. Thus the Grassmannian forms an algebraic subvariety of $\mathbf{P}(\Lambda^r(V))$.

It is now easy to see that if we look at two different bases for a subspace, they can be related via a invertible matrix and that the Plücker coordinates will differ by the determinant of the transformation. Also the Plücker relations give a necessary and sufficient condition for a multivector to be decomposable.

Example 6.7: Let V be 4-dimensional and $r = 2$, then a multivector $x \in \Lambda^2(V)$ can be represented as $\begin{bmatrix} X_{12} & X_{13} & X_{14} & X_{23} & X_{24} & X_{34} \end{bmatrix}^T$ where X_{ij} is the coefficient of $e_i \wedge e_j$. Note that every multivector can be represented in this form but only the decomposable ones correspond to 2-dimensional subspaces and satisfy the Plücker relation ($X_{12}X_{34} - X_{13}X_{24} + X_{23}X_{14} = 0$ in this case).

If $x = (e_1 + 2e_2) \wedge (e_2 + e_3 + 3e_4)$ then the Plücker coordinates can be obtained by first constructing the matrix

$$\begin{bmatrix} 1 & 0 \\ 2 & 1 \\ 0 & 1 \\ 0 & 3 \end{bmatrix}$$

and then $X_{12} = \begin{vmatrix} 1 & 0 \\ 2 & 1 \end{vmatrix} = 1, X_{24} = \begin{vmatrix} 2 & 1 \\ 0 & 3 \end{vmatrix} = 6$ and so on.

6.1.2 Rank of a multivector

Any multivector in $\Lambda^r(V)$ can be written as a linear combination of decomposable multivectors, for instance, by writing it as the sum of the standard basis elements. The rank of a r-vector is the minimal number of decomposable r-vectors in its decomposition. Computation of rank of a multivector is a tough problem in general. In fact, as mentioned in [13], if $V = \mathbb{R}^n$, the set of r-vectors of rank $\leq k$ when $k > 1$ and $r > 2$ is not algebraic (i.e. it cannot be expressed as a variety - a set of polynomial equations) or even closed.

This leaves out a few special cases. $k = 1$ corresponds to rank 1, i.e. decomposable vectors - in this case the Plücker relations give a method to characterize such r-vectors. $r = 1, 0$ are trivial cases. The remaining case of $r = 2$ is of great interest in this work and is discussed next.

6.1.3 Rank of 2-vectors

We start with a few theorems characterizing ranks of 2-vectors (assume V is n-dimensional).

Theorem 6.8 [**9**]: For any 2-vector $x \in \Lambda^2(V)$, there exist linearly independent vectors $v_1, \ldots, v_{2k} \in V$ such that

$$x = v_1 \wedge v_2 + \cdots + v_{2k-1} \wedge v_{2k}$$

The integer k is the rank of the 2-vector x and depends only on x.

Theorem 6.9 [**8**]: A 2-vector $x \in \Lambda^2(V)$ has rank $k > 1$ if and only if $x^k \neq 0$ and $x^{k+1} = 0$.

where $x^k = x \wedge \cdots \wedge x$ (k times) and $x^k \in \Lambda^{2k}(V)$.

Note that the Theorem 6.9 does not provide an efficient way of finding rank since the dimension of $\Lambda^r(V)$ is $\binom{n}{r}$ which increases exponentially with n when r is proportional ot n. To obtain an efficient procedure we represent 2-vectors as skew-symmetric matrices.

Let $x \in \Lambda^2(V)$ with Plücker coordinates $X_{ij}, 1 \leq i < j \leq n$. From this we construct a skew symmetric matrix S with entries $S_{ij} = X_{ij}$ for $i < j$, $S_{ij} = -X_{ij}$ for $i > j$ and $S_{ij} = 0$ for $i = j$. For example, if $n = 4$,

$$S = \begin{bmatrix} 0 & X_{12} & X_{13} & X_{14} \\ -X_{12} & 0 & X_{23} & X_{24} \\ -X_{13} & -X_{23} & 0 & X_{34} \\ -X_{14} & -X_{24} & -X_{34} & 0 \end{bmatrix}$$

According to theorem on linear algebra on the normal form of skew-symmetric matrices (see [8]) and Theorem 6.8, a 2-vector x has rank k if and only if the corresponding skew symmmetric matrix has rank $2k$. Since the Plücker coordinates of decomposable vectors can be obtained as determinants, it can be seen that if $x = v_1 \wedge v_2$ then the matrix $S = v_1 v_2^T - v_2 v_1^T$. Therefore we obtain the following facts which are very important for understanding the results.

1. If $x = v_1 \wedge v_2 + \cdots + v_{2k-1} \wedge v_{2k}$ where $v_1, \ldots, v_{2k} \in V$ are linearly independent, then the corresponding skew-symmetrix matrix $S = v_1 v_2^T - v_2 v_1^T + \cdots + v_{2k-1} v_{2k}^T - v_{2k} v_{2k-1}^T$ has rank $2k$. The range-space (column space) of S is $\langle v_1, \ldots, v_{2k} \rangle$.

2. If $v_1, \ldots, v_{2k} \in V$ are linearly dependent then we can just say that the rank of the matrix S must be strictly less than $2k$ and hence the rank of x is strictly less than k. In this case the range-space of S may or may not be same as $\langle v_1, \ldots, v_{2k} \rangle$. To obtain the range-space one must write x using the minimal number of decomposable multivectors.

Example 6.10: Let $V = \mathbb{R}^4$. Then $x = e_1 \wedge e_2 + e_3 \wedge e_4$ has rank 2 since e_1, e_2, e_3, e_4 are linearly independent. The corresponding matrix S has rank 4 and range-space $\langle e_1, \ldots, e_4 \rangle = \mathbb{R}^4$.

$$S = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

If $x = e_1 \wedge e_2 + e_3 \wedge e_1$, then the rank of x is 1 since $x = e_1 \wedge (e_2 - e_3)$ and the corresponding matrix will have rank 2 and range-space $\langle e_1, e_2 - e_3 \rangle$ which is a proper subspace of $\langle e_1, e_2, e_3 \rangle$.

6.2 Controllability Indices [14]

Consider the linear system

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}$$

$$\mathbf{y} = C\mathbf{x} + D\mathbf{u}$$

Here $\mathbf{x}$ is the $n \times 1$ state vector, $\mathbf{u}$ is the $m \times 1$ input vector and $\mathbf{y}$ is the output vector. The eigenvalues of A correspond to the poles of the system. To change the system dynamics we can provide state feedback i.e. the new input to the system is $\mathbf{u} + K\mathbf{x}$ (where K is a $m \times n$ matrix). Then the equations become

$$\dot{\mathbf{x}} = (A + BK)\mathbf{x} + B\mathbf{u}$$

$$\mathbf{y} = C\mathbf{x} + D\mathbf{u}$$

The question which arises is the following: Is it possible to place the eigenvalues of $A + BK$ at arbitrary positions?

Definition (Controllablity): The pair (A, B) is said to be controllable if for any initial state $\mathbf{x}(0) = \mathbf{x}_0$ and any final state $\mathbf{x}_1$, there exists an input that transfers $\mathbf{x}_0$ to $\mathbf{x}_1$ in finite time.

Thus controllability means that the states can be completely controlled using the input. It can be shown that the following are equivalent:

1. The system (A, B) is controllable.

2. The n x nm controllability matrix $C = [B\ AB\ A^2B\ \dots\ A^{n-1}B]$ has rank n.

3. K can be chosen to place eigenvalues of $A + BK$ at any specified location.

4. The matrix $[A - \lambda I\ B]$ has full rank for all values of λ.

Now let $m > 2$, i.e. a multi-input system and let the system be controllable. We assume that the columns of the matrix B are linearly independent - if not, we are effectively working with a system with fewer inputs. Let $B = \begin{bmatrix} b_1 & \dots & b_m \end{bmatrix}$. Then $A^kB = \begin{bmatrix} A^kb_1 & \dots & A^kb_m \end{bmatrix}$. Since the system is controllable, the matrix $C = [B\ AB\ A^2B\ \dots\ A^{n-1}B]$
$= \begin{bmatrix} b_1 & \dots & b_m & \dots & A^{n-1}b_1 & \dots & A^{n-1}b_m \end{bmatrix}$ has rank n. Suppose we start forming a basis from this matrix by searching from left to right. It should be noted that if A^kb_i is linearly dependent of the vectors on its left, then so is A^lb_i for $l > k$.

The collection of the linearly independent columns obtained in this way will look like $\{b_1, \dots, A^{\mu_1-1}b_1, b_2, \dots, A^{\mu_2-1}b_2, \dots, b_m, \dots, A^{\mu_m-1}b_m\}$ where $1 \leq \mu_i \leq n$ (since B has full column rank). Also $\mu_1 + \dots + \mu_m = n$. The **controllability indices** $(\rho_1, \rho_2, \dots, \rho_m)$ are obtained by arranging the μ_i's in a decreasing order. The controllability indices satisfy (i) $\rho_1 \geq \rho_2 \geq \dots \geq \rho_m \geq 1$ and (ii) $\rho_1 + \dots + \rho_m = n$. The **controllability index** ρ is the maximum μ_i, i.e. $\rho = \rho_1$. Thus ρ is the smallest integer such that the rank of $\begin{bmatrix} B & AB & \dots & A^{\rho-1}B \end{bmatrix}$ is n.

Theorem 6.11 [14]: The set of controllability indices of a controllable system are invariant under a chage of basis or reordering of columns of B.

Therefore controllability indices are an invariant property of a controllable system and have certain applications (see [14]). For a 2-input system with n states, the controllability indices (ρ_1, ρ_2) can take $\lfloor \frac{n}{2} \rfloor$ possible values. For $n = 5$, the possibilities are (3,2), (4,1). For $n = 6$, we have three possibilities: (3,3), (4,2) and (5,1).

6.2.1 Determining Controllability Indices

For motivating some of our algorithms, we relate the controllability indices to the partial rank of the controllability matrix.

Definition 6.12 A **partial controllability matrix** is $C_k = \begin{bmatrix} B & AB & \dots & A^{k-1}B \end{bmatrix}$ for $k = 1, 2, \dots n$ (where $C_1 = B$). The rank of C_k is denoted by r_k.

Given the controllability indices, one can obtain the rank of C_k as $\sum_{i=1}^{n} min(\rho_i, k)$. Now suppose we are given the partial ranks $r_1, r_2, \dots$. We first obtain the following decreasing sequence $\{r_1, r_2 - r_1, r_3 - r_2, \dots\}$. This denotes the rank added at each step. Then the controllability indices are obtained as $\rho_i =$ no. of elements in the sequence which are greater than or equal to i. In particular $\rho_m = min\{k - 1 : r_k < mk\}$ (this is the first time some A^kb_i is a linear combination of the previous vectors).

The obvious method for obtaining controllalibility indices is to find the ranks of the matrices C_k for $k = 1, 2, 3, \ldots$. Then the controllability indices can be obtained as shown in the previous paragraph.

Example 6.13: Let $n = 10, m = 3$ and the partial ranks of C, r_i's be $3, 6, 8, 9, 10$ for $i = 1, \ldots, 5$. Then the difference sequence is $3, 3, 2, 1, 1, 0, \ldots$ and the controllability indices are $(5, 3, 2)$.

Another method for determining controllability indices by transforming A into a block-Hessenberg matrix is introduced in [15], [16]. Given a linear system (A, B), we find orthogonal transformation P such that

$$F = PAP^T = \begin{bmatrix} F_{11} & F_{12} & F_{13} & \ldots & F_{1,k} & F_{1,k+1} \\ F_{21} & F_{22} & F_{23} & \ldots & F_{2,k} & F_{2,k+1} \\ 0 & F_{32} & F_{33} & \ldots & F_{3,k} & F_{3,k+1} \\ 0 & 0 & F_{43} & \ldots & F_{k,1} & F_{k+1,1} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \ldots & F_{k,k} & F_{k,k+1} \\ 0 & 0 & 0 & \ldots & F_{k+1,k} & F_{k+1,k+1} \end{bmatrix}$$

and

$$G = PB = \begin{bmatrix} G_1^T & 0 & 0 & \ldots & 0 & 0 \end{bmatrix}^T$$

where F_{ij} is $l_{i-1} \times l_{j-1}$ matrix and G_1 is $l_0 \times m$ matrix. The integers l_i for $i = 0, \ldots, k$ are defined as follows: $l_0 = \text{rank}(B) = m$ and $l_i = \text{rank}(F_{i+1,i})$. We skip the details of the algorithm to obtain the transformation P and some further details of the $F_{i,j}$ and G_1. We just note that the integers $l_0, l_1, l_2, \ldots$ represent the rank increases in the controllability matrices ($r_i - r_{i-1}$ in our previous notation). Thus it very easy to obtain the controllability indices from this representation. Also, since this method uses only orthogonal transformations it is numerically very stable. It is also computationally efficient.

7 Problem Formulation and Preliminary Results

Our aim to find efficient algorithms for determining controllability indices using tools from exterior algebra. We will focus here on the case of 2-input systems which are relatively easy to work with. Thus from now on, assume $m = 2$. Also we always assume that the system is controllable. The matrix A is assumed to be invertible. A 2-vector $v_1 \wedge v_2$ becomes 0 whenever the subspace $\langle v_1, v_2 \rangle$ is not 2-dimensional and we lose the information about the subspace. Having B full-rank and A invertible ensures that $A^k b_1 \wedge A^k b_2 \neq 0$ for any k.

The matrix $B = \begin{bmatrix} b_1 & b_2 \end{bmatrix}$ can be equivalently represented using the 2-vector $b_1 \wedge b_2$ or the skew-symmetric matrix $S = b_1 b_2^T - b_2 b_1^T$. We'll work with both these representations throughout this section. The matrix S_1 corresponding to $Ab_1 \wedge Ab_2$ is $S_1 = Ab_1(Ab_2)^T - Ab_2(Ab_1)^T$ which is the same as ASA^T. Also recall that a 2-vector x has rank k iff the corresponding skew-symmetric matrix S has rank $2k$.

For a 2-input system the controllability indices are (ρ_1, ρ_2) where $\rho_1 \geq \rho_2$ and $\rho_1 + \rho_2 = n$. Thus it is sufficient to find $\rho_2 = min\{k - 1 : r_k < 2k\}$ where r_k denotes the partial rank of the controllability matrix (see Section 6.2.1). Thus the rank of the partial controllability

matrix C_k increases by two till $k = \rho_2 + 1$ and then it increases by one till the rank becomes n. That is to say

$$r_k = \begin{cases} 2k, & k \leq \rho_2 \\ 2\rho_2 + 1 \times (k - \rho_2) = k + \rho_2, & \rho_2 < k \leq \rho_1 \\ n, & k > \rho_1 \end{cases} \tag{9}$$

Note that the second case does not occur if $\rho_1 = \rho_2$.

Thus our aim is to find the first k for which the matrix C_k is not full rank.

One possible method would be to obtain

$$x_k = b_1 \wedge b_2 \wedge Ab_1 \wedge Ab_2 \wedge \cdots \wedge A^{k-1}b_1 \wedge A^{k-1}b_2, \quad k = 1, 2, \ldots$$

By Theorem 6.2, x_k is 0 iff C_k is not full rank. Thus by finding x_k for increasing values of k, we can detect the smallest k where $x_k = 0$ and then $\rho_2 = k - 1$. However since this method involves k-vectors for $k > 2$, it is not computationally efficient. For instance the dimension of $\Lambda^{\lfloor n/2 \rfloor}(\mathbb{R}^n)$ is of the order of 2^n. But using these ideas we can obtain our first algorithm.

Algorithm 7.1:

1. Define $S := b_1 b_2^T - b_2 b_1^T$, $S_1 := S$
2. Define $k := 2$
3. Repeat steps 4 to 8
4. $S_1 \leftarrow AS_1A^T$
5. $S \leftarrow S + S_1$
6. Compute rank of S
7. If $rank(S) < 2k$, declare $\rho_2 = k - 1$ and STOP.
8. $k \leftarrow k + 1$

After step 4, S_1 is $A^{k-1}b_1(A^{k-1}b_2)^T - A^{k-1}b_2(A^{k-1}b_1)^T$ and thus after step 5, $S = \sum_{i=1}^{k}(A^{i-1}b_1(A^{i-1}b_2)^T - A^{i-1}b_2(A^{i-1}b_1)^T)$. The correctness of the algorithm follows immediately from the preceding discussion and the results in Section 6.1.3.

7.1 Rank Evolution

The previous algorithm computes the rank of a number of matrices with increasing sizes until it gives the indices. However we show that to determine the controllability indices of a 2-input system, it is sufficient to know the rank of $C_{\lfloor n/2 \rfloor}$ which we denote by $r_{\lfloor n/2 \rfloor}$. From equation 9 and noting that $\rho_2 \leq \lfloor n/2 \rfloor \leq \rho_1$, we obtain,

$$r_{\lfloor n/2 \rfloor} = \lfloor n/2 \rfloor + \rho_2 \tag{10}$$

Thus given $r_{\lfloor n/2 \rfloor}$, we can obtain the controllability indices $(r_{\lfloor n/2 \rfloor} + \lceil n/2 \rceil, r_{\lfloor n/2 \rfloor} - \lfloor n/2 \rfloor)$.

Definition 7.2: Define the matrix $S_k = \sum_{i=1}^{k} (A^{i-1}b_1(A^{i-1}b_2)^T - A^{i-1}b_2(A^{i-1}b_1)^T)$ and let s_k be the rank of S_k.

Algorithm 7.1 found the ranks of S_k for values of $k = 1, 2, \ldots$ till it reached a point where $s_k < 2k$. In view of the preceding discussion and Equation 10, it would be interesting to study whether $s_{\lfloor n/2 \rfloor}$ is related to $r_{\lfloor n/2 \rfloor}$. However as mentioned in Section 6.1.3, if $v_1, \ldots, v_{2k} \in V$ are linearly dependent then we can only say that the rank of the matrix $S = v_1 v_2^T - v_2 v_1^T + \cdots + v_{2k-1} v_{2k}^T - v_{2k} v_{2k-1}^T$ must be strictly less than $2k$. In this case the range-space of S may or may not be same as $\langle v_1, \ldots, v_{2k} \rangle$. The best we can say is

Theorem 7.3: The rank $s_{\lfloor n/2 \rfloor}$ satisfies $2\rho_2 \leq s_{\lfloor n/2 \rfloor} \leq \lfloor n/2 \rfloor + \rho_2$.

Proof: $s_{\lfloor n/2 \rfloor} \leq \lfloor n/2 \rfloor + \rho_2$ holds because the rank of $S_{\lfloor n/2 \rfloor}$ can be at most equal to the rank of $C_{\lfloor n/2 \rfloor}$ (see Section 6.1.3) which is $r_{\lfloor n/2 \rfloor} = \lfloor n/2 \rfloor + \rho_2$. To show that $s_{\lfloor n/2 \rfloor} \geq 2\rho_2$, we show by induction that $s_k \geq 2\rho_2$ for $\rho_2 \leq k \leq \lfloor n/2 \rfloor$. For $k = \rho_2$, $s_k = 2\rho_2$ since $\{b_1, b_2, \ldots, A^{\rho_2 - 1}b_1, A^{\rho_2 - 1}b_2\}$ are linearly independent.

For the induction part we will work with the 2-vector corresponding to S_k, say $x_k = b_1 \wedge b_2 + Ab_1 \wedge Ab_2 + \cdots + A^{k-1}b_1 \wedge A^{k-1}b_2$. Now suppose S_k has rank $2l \geq 2\rho_2$, then x_k can be expressed as $x_k = v_1 \wedge v_2 + \cdots + v_{2l-1} \wedge v_{2l}$ where $\{v_1, v_2, \ldots, v_{2l}\}$ are linearly independent. Since $2\rho_2 < k \leq \lfloor n/2 \rfloor \leq \rho_1$, either $A^k b_1$ or $A^k b_2$ (suppose $A^k b_1$) is linearly dependent on $b_1, b_2, \ldots, A^{k-1}b_1, A^{k-1}b_2$. Since $\langle v_1, \ldots, v_{2l} \rangle$ is a subspace of $\langle b_1, b_2, \ldots A^{k-1}b_1, A^{k-1}b_2 \rangle$, there can be two possibilities:

1. $\langle v_1, \ldots, v_{2l}, A^k b_1, A^k b_2 \rangle$ are linearly independent. In this case S_{k+1} has rank $2l+2 \geq 2\rho_2$.
2. $A^k b_1 = \alpha_0 A^k b_2 + \sum_{i=1}^{2l} \alpha_i v_i$ and $A^k b_2$ is independent. In this case, x_{k+1} can be written as

$$x_{k+1} = x_k + A^k b_1 \wedge A^k b_2$$
$$= (v_1 - \alpha_2 A^k b_2) \wedge (v_1 - \alpha_2 A^k b_2) + \cdots + (v_{2l-1} - \alpha_{2l} A^k b_2) \wedge (v_{2l} - \alpha_{2l-1} A^k b_2)$$

where the vectors $\{v_1 - \alpha_2 A^k b_2), (v_1 - \alpha_2 A^k b_2), \ldots, (v_{2l-1} - \alpha_{2l} A^k b_2), (v_{2l} - \alpha_{2l-1} A^k b_2)\}$ are linearly independent and hence $s_{k+1} = 2l \geq 2\rho_2$.

Hence Proved.

In the worst case, $\rho_2 = \lfloor n/4 \rfloor$, the rank $s_{\lfloor n/2 \rfloor}$ can range from around $n/2$ to around $3n/4$. Thus finding the rank $s_{\lfloor n/2 \rfloor}$ still leaves a large number of possibilities. We illustrate this with a small example:

Example 7.4: Let $n = 6$, we will look at two systems with different controllability indices but same value for s_3.

1. Let $b_1 = e_1$, $b_2 = e_2$ and
$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$
The controllability indices are $(5, 1)$. Then $x_3 = b_1 \wedge b_2 + Ab_1 \wedge Ab_2 + A^2b_1 \wedge A^2b_2 = e_1 \wedge e_2 + (e_1 + e_2) \wedge e_3 + (e_1 + e_2 + e_3) \wedge e_4 = (e_1 - e_3) \wedge (e_2 + e_1) + (e_1 + e_2 + e_3) \wedge e_4$ and $s_3 = 4$.

2. Let $b_1 = e_1$, $b_2 = e_2$ and
$$A = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$
The controllability indices are $(4, 2)$. Then $x_3 = b_1 \wedge b_2 + Ab_1 \wedge Ab_2 + A^2b_1 \wedge A^2b_2 = e_1 \wedge e_2 + e_3 \wedge e_4 + (e_1 + e_2 + e_3) \wedge e_5 = (e_1 - e_5) \wedge (e_2 + e_5) + e_3 \wedge (e_4 + e_5)$ and $s_3 = 4$.

8 Main Results and Discussions

As seen in the previous section, the rank of $S_{\lfloor n/2 \rfloor}$ cannot determine the controllability indices when $\{b_1, b_2, \ldots, A^{\lfloor n/2 \rfloor - 1}b_1, A^{\lfloor n/2 \rfloor - 1}b_2\}$ are not linearly independent.To mitigate this issue, we embed the systtem in a higher dimension as follow: Let $n_* = 2n$,
$$B_* = \begin{bmatrix} b_1 & b_2 \\ 1 & 0 \\ 0 & 1 \\ \vdots & \vdots \\ 0 & 0 \end{bmatrix}$$
where B_* is a $n_* \times 2$ matrix.
$$A_* = \begin{bmatrix} A & 0 \\ 0 & M \end{bmatrix}$$
where A_* is $n_* \times n_*$ and M $(n \times n)$ is a circulant matrix with first row $\begin{bmatrix} 0 & 0 & \ldots & 1 & 0 \end{bmatrix}$. It transforms a vector $(a_1, a_2, \ldots, a_{n-1}, a_n)$ to the vector $(a_{n-1}, a_n, a_1, \ldots, a_{n-2})$. Recall that for a system (A, B) the matrix S_k was the skew-symmetric matrix defined as $S_k = \sum_{i=1}^{k} (A^{i-1}b_1(A^{i-1}b_2)^T - A^{i-1}b_2(A^{i-1}b_1)^T)$.

Algorithm 8.1:

1. For the system (A_*, B_*), compute the matrix $S_{\lfloor n/2 \rfloor}$ and call it S_*

2. Compute rank s_* of the submatrix formed by first n rows of S_*

3. Declare $\rho_2 = s_* - \lfloor n/2 \rfloor$ and STOP.

Proof of Correctness: Note that the 2-vector x_* corresponding to S_* is

$$x_* = \begin{bmatrix} b_1 \\ 1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \wedge \begin{bmatrix} b_2 \\ 0 \\ 1 \\ \vdots \\ 0 \\ 0 \end{bmatrix} + \cdots + \begin{bmatrix} A^{\lfloor n/2 \rfloor - 1} b_1 \\ 0 \\ 0 \\ \vdots \\ 1 \\ 0 \end{bmatrix} \wedge \begin{bmatrix} A^{\lfloor n/2 \rfloor - 1} b_2 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$

since A_* has a block diagonal structure and M just rotates the last n coordinates by two places. It is easy to see that these vectors are linearly independent, even if there is a linear dependence among $\{b_1, b_2, \ldots, A^{\lfloor n/2 \rfloor - 1} b_1, A^{\lfloor n/2 \rfloor - 1} b_2\}$, the last n coordinates cannot cancel out.

Thus, S_* has rank $2n$ and range space is the span of

$$\begin{bmatrix} b_1 \\ 1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} b_2 \\ 1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}, \ldots, \begin{bmatrix} A^{\lfloor n/2 \rfloor - 1} b_1 \\ 0 \\ 0 \\ \vdots \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} A^{\lfloor n/2 \rfloor - 1} b_2 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$

and hence the range of the first n rows is precisely the column space of $C_{\lfloor n/2 \rfloor}$ and its rank is $s_* = r_{\lfloor n/2 \rfloor}$ and thus $\rho_2 = s_* - \lfloor n/2 \rfloor$ (using Equation 10). Hence Proved.

This algorithm needs only one rank determination. Note that S_* is a sum of $\lfloor n/2 \rfloor$ matrices. The computation of S_* can be done using $O(\log(n))$ matrix multiplications by using a double and add technique - first obtain sum of 2 terms and find sum of first 4 and then first 8 terms (use the fact that $S_4 = S_2 + A_*^2 S_2 (A_*^2)^T$ etc.). We need $A_*, A_*^2, A_*^4, A_*^8, \ldots$ but these can also be obtained in $O(\log(n))$ multiplications. Overall the complexity of this method is $O(n^3 \log(n))$ if we use $O(n^3)$ algorithms for rank computation and matrix multiplication. This does not compare favourably against the existing methods which work in $O(n^3)$ time. The main reason behind this seems to be the fact that the Plücker coordinates themselves are of size $O(n^2)$. This problem is likely to exacerbate further for higher values of m.

If we allow randomized algorithms, we need to increase the dimension only by 1, let $n_+ = n + 1$ and

$$B_+ = \begin{bmatrix} b_1 & b_2 \\ c_1 & c_2 \end{bmatrix}, \quad A_+ = \begin{bmatrix} A & 0 \\ 0 & d \end{bmatrix}$$

where c_1, c_2 and d are independently chosen random variables from some suitable continuous distribution, say Gaussian.

Algorithm 8.2:

1. Generate random numbers c_1, c_2 and d independently distributed as the unit normal distribution.

2. For the system (A_+, B_+), compute the matrix $S_{\lfloor n/2 \rfloor}$ and call it S_+

3. Compute rank s_+ of the submatrix formed by first n rows of S_+

4. Declare $\rho_2 = s_+ - \lfloor n/2 \rfloor$ and STOP.

Claim 8.3: For any controllable system (A, B) with A invertible and B of rank 2, Algorithm 8.2 gives the correct controllability indices with probability 1.

Proof of Claim 8.3: We just need to show that the vectors

$$\begin{bmatrix} b_1 \\ c_1 \end{bmatrix}, \begin{bmatrix} b_2 \\ c_2 \end{bmatrix}, \begin{bmatrix} Ab_1 \\ c_1 d \end{bmatrix}, \begin{bmatrix} Ab_2 \\ c_2 d \end{bmatrix}, \cdots \begin{bmatrix} A^{\lfloor n/2 \rfloor - 1} b_1 \\ c_1 d^{\lfloor n/2 \rfloor - 1} \end{bmatrix}, \begin{bmatrix} A^{\lfloor n/2 \rfloor - 1} b_2 \\ c_2 d^{\lfloor n/2 \rfloor - 1} \end{bmatrix}$$

are linearly independent with probability 1 and then the rest of the proof is same as that for Algorithm 8.1. If there is linear dependence among $\{b_1, b_2, \ldots, A^{\lfloor n/2 \rfloor - 1} b_1, A^{\lfloor n/2 \rfloor - 1} b_2\}$, the extended vectors are linearly dependent iff the last row of random vectors lies in some proper subspace because the random row must satisfy the same linear dependence relations as the first n rows. This event has probability of zero for the Gaussian distribution. Hence Proved.

The complexity analysis of Algorithm 8.2 is very similar to that of Algorithm 8.3. Only the sizes of the matrices reduce from $2n$ to $n+1$ which leads to some computational saving.

9 Conclusion and Future Work

This project aimed to use ideas from exterior algebra to find algorithms for determining controllability indices of linear systems. For the special case of 2-input systems we have presented three algorithms. As of now, these do not provide computational advantages over the existing methods. Thus, there is great scope for further work to find new methods.

None of the suggested algorithms extend to systems with more than 3 inputs. This is because these are based on computing ranks of multivectors and no known characterization exists for $m > 2$. Thus to design algorithms that work for a general control system, we can't rely on rank-based methods and need to look beyond.

References

[1] W.W.A.P. Loustaunau. *An Introduction to Grobner Bases*. American Mathematical Soc., 1994.

[2] T. Becker, H. Kredel, and V. Weispfenning. *Gröbner bases: a computational approach to commutative algebra*. Springer-Verlag, London, UK, 0 edition, 4 1993.

[3] David A. Cox, John Little, and Donal O'Shea. *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra, 3/e (Undergraduate Texts in Mathematics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2007.

[4] B. Sturmfels. *Gröbner Bases and Convex Polytopes*. Memoirs of the American Mathematical Society. American Mathematical Soc.

[5] N.S. Nise. *Control Systems Engineering*. Number v. 1 in Control Systems Engineering. Wiley, 2004.

[6] R.L. Williams and D.A. Lawrence. *Linear State-Space Control Systems*. John Wiley & Sons, 2007.

[7] D. Q. MAYNE and P. MURDOCH. Modal control of linear time invariant systems. *International Journal of Control*, 11(2):223–227, 1970.

[8] Robert L. Bryant, S. S. Chern, Robert B. Gardner, Hubert L. Goldschmidt, and P. A. Griffiths. *Exterior Differential Systems*. Springer New York, 1991.

[9] R. Hermann. Review: S. sternberg, lectures on differential geometry. *Bull. Amer. Math. Soc.*, 71(2):332–337, 03 1965.

[10] P. Griffiths and J. Harris. *Principles of Algebraic Geometry*. Wiley Classics Library. Wiley, 2011.

[11] Wikipedia. Plcker embedding — wikipedia, the free encyclopedia, 2016. [Online; accessed 28-April-2016].

[12] Wikipedia. Exterior algebra — wikipedia, the free encyclopedia, 2016. [Online; accessed 27-April-2016].

[13] G B Kleiner. Multivectors of rank 2 over fields and commutative rings. *Sbornik: Mathematics*, 193(5):709, 2002.

[14] Chi-Tsong Chen. *Linear System Theory and Design*. Oxford University Press, Inc., New York, NY, USA, 3rd edition, 1998.

[15] R. Patel. Computation of matrix fraction descriptions of linear time-invariant systems. *IEEE Transactions on Automatic Control*, 26(1):148–161, Feb 1981.

[16] P. Van Dooren. The generalized eigenstructure problem in linear system theory. *IEEE Transactions on Automatic Control*, 26(1):111–129, Feb 1981.