Università di Pisa

Dipartimento di Informatica
Corso di Laurea in Informatica

Computation of Kronecker's Canonical Form in a Computer Algebra System

Relatore
Prof. Federico Poloni

Candidato
Giacomo Trapani

Anno Accademico 2021/2022

Contents

Introduction

Tempor amet nostrud ex aliquip adipisicing aliqua. Laboris in laboris cillum anim. Mollit ea ut ad sit ut exercitation culpa. Enim reprehenderit Lorem id quis anim. Est commodo pariatur pariatur consequat labore.

Ea incididunt laboris labore aliquip ipsum sunt labore velit labore minim aliquip elit esse. Ad anim et voluptate ea veniam ipsum ea laborum. Quis laboris reprehenderit duis proident culpa Lorem. Ea commodo ea et proident incididunt laborum voluptate. Magna sunt id occaecat sit laboris commodo sint quis occaecat cupidatat qui duis.

Sit fugiat ea dolor est ut ullamco sint dolore irure aliqua. Minim sunt nostrud quis ipsum duis aute consequat aliquip voluptate voluptate ullamco adipisicing. Tempor Lorem commodo sunt nisi proident irure veniam.

Veniam reprehenderit nulla aliquip laboris non culpa velit sint. Aute qui dolore consequat commodo ullamco. Eu elit eiusmod qui enim reprehenderit tempor occaecat ut officia do occaecat dolor anim. Labore occaecat sit do labore minim ut. Magna proident labore magna id ullamco irure aliqua id. In incididunt occaecat ullamco ut eu dolor mollit ad dolore anim.

Ut nostrud qui pariatur nostrud excepteur veniam elit Lorem ut do eiusmod quis duis exercitation. Deserunt occaecat tempor exercitation in minim eu fugiat. Aliqua ex sit qui deserunt dolore consequat esse amet anim occaecat nisi. Officia anim pariatur veniam officia.

Ea ut esse sit non. Ipsum sit cillum cillum cillum ex adipisicing eu sint laboris dolore id. Sit ipsum et in nulla ut eiusmod cillum pariatur quis labore. Et reprehenderit enim quis Lorem exercitation consequat. Aute irure officia anim est elit aute veniam ut nostrud ut quis minim reprehenderit. Duis Lorem esse nostrud duis.

Id proident ad commodo fugiat qui proident dolore dolor adipisicing dolore duis velit. Cillum proident Lorem fugiat ut laborum ipsum officia ex sit. Est nulla velit adipisicing exercitation exercitation aliqua veniam eu do excepteur qui incididunt adipisicing. Ea amet id reprehenderit duis tempor duis irure dolor et labore consequat veniam. Adipisicing id aliqua dolor excepteur elit commodo adipisicing.

Mollit est anim adipisicing excepteur nisi aliquip quis aliquip mollit. Cillum excepteur eu aliqua cupidatat est laborum consectetur. Anim nisi veniam sint nulla magna ad sit nulla velit eu eu culpa nostrud. Ipsum veniam duis eiusmod et aute reprehenderit sunt commodo voluptate do.

Reprehenderit deserunt ex qui sit reprehenderit fugiat pariatur nisi deserunt ut ad do do. Nisi pariatur amet ipsum nisi labore dolor anim enim veniam. Excepteur ullamco voluptate nisi dolore tempor tempor.

Consequat enim nisi irure non exercitation dolore officia. Est ex sint irure dolor qui nisi. Nostrud sint ex et enim aute consectetur deserunt nisi eu. Velit commodo deserunt occaecat nostrud ullamco voluptate amet consectetur duis dolore sunt culpa sunt aliquip. Lorem sit deserunt commodo sunt.

Eiusmod deserunt cillum fugiat incididunt sit laborum anim ea. Aute sit proident enim sint elit ex. Aute enim amet ad nisi aliqua exercitation dolor reprehenderit duis elit ad. Ullamco qui do est voluptate incididunt nisi qui mollit cupidatat excepteur. Duis do aliqua occaecat ex culpa proident nostrud. Reprehenderit incididunt id adipisicing do dolor. Irure sit irure laboris non ea mollit laborum.

Duis reprehenderit irure ipsum voluptate sint velit. Do nisi laboris aliqua amet voluptate do cupidatat nulla elit amet deserunt ad sint. Laboris proident aute deserunt amet amet veniam eiusmod laboris.

In do do quis in irure commodo ut cillum ad. Ullamco aute do cillum sint in consectetur tempor sit laborum. Aute eiusmod cupidatat est nulla anim deserunt nostrud ullamco occaecat exercitation sit magna anim. Dolor veniam tempor dolore commodo tempor sit quis. Consequat adipisicing aliquip minim labore aliquip ea eiusmod dolor sint veniam sit consectetur reprehenderit. Officia labore Lorem sint velit ipsum laboris id ut dolore sint.

Veniam laborum deserunt in magna non. Eiusmod ex nulla nostrud ut sint ullamco commodo nostrud ea qui. Sunt ad do incididunt culpa fugiat dolor nostrud consectetur non veniam elit officia sunt occaecat. Ullamco ullamco ut ea elit labore. Ullamco irure officia velit pariatur aliquip in enim.

Esse cupidatat est esse nulla ea est pariatur ad velit voluptate dolor sit. Cupidatat eu tempor ad sunt est. Commodo ipsum do enim duis sunt ea voluptate Lorem fugiat deserunt tempor consectetur commodo. Ea minim officia amet sint. Magna exercitation proident officia magna veniam cillum Lorem occaecat esse velit. Aliqua Lorem Lorem eiusmod minim commodo nostrud minim dolor et dolore duis qui aute. Minim nisi consequat eiusmod consequat Lorem laboris consequat non ut laboris quis id dolor dolore.

Eiusmod cillum pariatur est culpa nulla mollit magna voluptate amet in quis id mollit. Ad cupidatat id dolor ad ex adipisicing laborum pariatur nostrud nisi incididunt. Mollit sunt aliqua adipisicing enim ipsum sint eiusmod dolore magna qui reprehenderit proident. Et veniam id nisi dolor occaecat est labore qui sit non enim. Esse nisi voluptate et mollit dolor ad.

Cupidatat mollit do veniam quis quis minim eiusmod sint labore duis commodo sunt esse nostrud. Culpa excepteur culpa sint ipsum irure cillum deserunt pariatur enim labore. Id velit in irure ex nisi eu ea commodo irure ad ullamco et excepteur. Eu pariatur deserunt duis ullamco qui Lorem dolore reprehenderit velit id excepteur. Non labore eiusmod amet aute consequat magna ut occaecat. Laboris nisi excepteur cillum sit sunt non. Cillum amet voluptate nostrud magna commodo cillum cillum.

Proident veniam enim eiusmod et mollit consectetur incididunt anim. Cillum incididunt quis ad duis excepteur amet est aute commodo velit est incididunt. Ea irure labore anim consectetur officia consequat officia culpa sit minim id commodo. Deserunt qui tempor laborum excepteur nisi magna et pariatur.

Anim adipisicing laboris commodo in qui labore laboris exercitation exercitation exercitation et dolore amet. Laborum qui laboris officia Lorem eu est do. Consequat tempor dolor cillum qui dolore. Sint labore dolor eu ipsum mollit duis excepteur veniam aute anim labore aliqua. Nisi cupidatat ex officia ad.

Laborum reprehenderit sint cupidatat fugiat minim. Consectetur adipisicing elit proident exercitation consectetur exercitation consectetur voluptate veniam sit velit consectetur cupidatat. Velit do cillum consectetur cupidatat labore esse ad adipisicing sunt minim quis elit ad. Id et fugiat enim deserunt.

Commodo velit irure non irure. Fugiat duis nulla enim enim ut consectetur id. Esse proident anim laboris labore amet.

Consectetur nulla ullamco tempor amet nisi sit ad deserunt fugiat mollit. Dolor exercitation et nulla aliqua pariatur occaecat fugiat velit consequat. Aliquip minim enim ut deserunt nostrud. Cupidatat laboris pariatur enim Lorem dolore. Sint eu esse fugiat deserunt nisi cupidatat proident irure consequat esse aliquip culpa adipisicing proident. Duis et aliquip magna ipsum sint pariatur dolore ut aliquip. In enim enim ex est ut ipsum ea nostrud tempor qui.

Cillum quis irure Lorem esse commodo sint velit et eu ea culpa. Eu adipisicing quis laborum ex minim commodo cillum incididunt exercitation excepteur eiusmod anim. Occaecat amet magna irure ea minim nostrud. Qui mollit deserunt irure tempor anim et sit ut fugiat ullamco pariatur non. Irure eiusmod elit commodo aliquip id commodo. Non voluptate culpa laborum Lorem elit tempor consequat quis.

Eiusmod quis eu pariatur eu. Sunt fugiat adipisicing cillum laborum magna fugiat excepteur sit commodo et enim. Cupidatat reprehenderit exercitation aliqua proident eiusmod in sunt esse minim elit non Lorem commodo enim. In eu aliqua quis officia Lorem do ut sunt commodo aliqua velit eiusmod culpa. Fugiat est excepteur consequat adipisicing voluptate et reprehenderit. Adipisicing deserunt exercitation tempor mollit qui aliqua est dolore aute.

Exercitation aliqua dolore dolor voluptate pariatur esse ex qui non. Excepteur qui qui nisi do deserunt anim laboris pariatur quis in dolor dolor eu. Consequat aliqua culpa adipisicing occaecat adipisicing velit tempor eiusmod enim officia nulla. Lorem enim et irure esse culpa sunt. In qui adipisicing ad laborum enim nulla ex aliquip consectetur est mollit amet et nisi.

Quis cupidatat labore qui et aliqua ullamco veniam enim qui voluptate. Dolore laboris in fugiat non sint. Aliqua sunt quis dolore nostrud proident ex.

In dolore sint proident duis consequat eu nisi eu labore occaecat culpa minim sint. Aute nulla irure irure deserunt laborum veniam aliquip ad veniam anim pariatur ex elit ipsum. Cupidatat qui aute do mollit. Enim duis nostrud anim ipsum ad consectetur do ea magna ea deserunt sunt. Do irure occaecat amet non ut dolore eu ea pariatur est proident et aliqua pariatur. Cillum laboris dolor aliqua fugiat culpa do. Velit ad proident velit minim quis ipsum dolore nostrud voluptate tempor.

Commodo sunt officia eu in anim culpa. Eiusmod commodo proident et aliqua in ex culpa laborum qui amet. Laboris cupidatat incididunt elit ipsum irure velit quis tempor irure duis. Adipisicing veniam duis proident cupidatat minim. Non ullamco sint incididunt duis qui eu. Sunt eu id sint pariatur incididunt anim enim officia adipisicing in do anim.

Nostrud qui excepteur laboris veniam labore eu sunt deserunt magna ea fugiat aute. Ut ex et labore culpa ea mollit incididunt ullamco Lorem fugiat. Nulla proident fugiat sunt et pariatur ipsum anim nostrud quis nisi qui pariatur. Tempor dolore ipsum tempor aliquip nostrud. Do nostrud esse enim amet qui anim consequat aute. Nisi officia ad sint ut proident proident. Minim labore ut elit id consequat sit ea.

Background

This chapter will serve as prerequisite knowledge throughout the rest of this thesis.

We shall briefly present SageMath, the software system used to implement the algorithm discussed in the following chapters, by introducing computer algebra systems and comparing numerical computations against computer algebra showcasing an example; then, the reader shall familiarize with the concept of condition number as an emphasis on it will be put in every part.

Subsequently, definitions and properties of eigenvalues and eigenvectors shall be concisely introduced.

Lastly, we shall describe the Jordan canonical form of a matrix.

Computer algebra.

Computers have fundamentally two ways to reason about a mathematical expression: **numerical computations**, which are performed using *only numbers* to represent values and **computer algebra** (or **symbolic computations**), which - by contrast - use *both numbers and symbols.*

First, we shall introduce the concept of **floating point number system**, which is the system used to handle numerical computations.

Definition 2.1 (Normalized-floating point number system)**.** A normalized-floating point number system F is characterized by the 4-tuple of integers β, p, L, U:

- β is called base or radix,
- p precision,
- $[L, U]$ exponent range (with L, U denoting lower and upper bound respectively).

Given a number $x \in \mathbb{R}$, $x \neq 0$ its representation in a floating point number system shall be written out as $fl(x)$ and has the form

$$x = sign(x)\beta^E \sum_{i=0}^{p-1} d_i \beta^{-i}$$

with $L \leq E \leq U$ and the sequence $\{d_i\}$ (which is called mantissa) made up of natural numbers such that $d_0 \neq 0$, $0 \leq d_i \leq \beta - 1$ and d_i eventually different from $\beta - 1$.

The notation δx shall be used to denote the difference between a symbol x and its floating point approximation $fl(x)$

$$\delta x = x - fl(x).$$

It is important to notice that a floating point number system F is discrete and finite: it approximates real numbers with finite numbers; in other words, a floating point number system may introduce errors when representing a real number.

A de facto standard for computers to work with floating point approximations is IEEE 754 [7], the details of which shall not be discussed.

Definition 2.2 (Machine epsilon)**.** Machine epsilon is the maximum possible absolute relative error in representing a nonzero real number x in a floating point number system

$$\epsilon_{mach} = \max_{x} \frac{|x - fl(x)|}{|x|}.$$

Example 2.1. Let us define the matrix (made up of both symbols and numbers) M

$$\begin{bmatrix} \sqrt{2} & 1 \\ 2 & \sqrt{2} \end{bmatrix}.$$

Consider the matrix $\tilde{M}$, having as entries the floating point approximation of those of M

$$\begin{bmatrix} fl(\sqrt{2}) & 1 \\ 2 & fl(\sqrt{2}) \end{bmatrix}.$$

Computing its determinant gives out $2 + 2\epsilon\sqrt{2} + \epsilon^2 - 2 \doteq 2 + 2\epsilon\sqrt{2} - 2 \neq 0.$

Introducing a small change (i.e. an "error") in the input argument may either cause a large or a small change in the result. Now, we shall define what condition numbers are.

Definition 2.3 (Condition number)**.** A condition number of a problem measures the sensitivity of the solution to small perturbations in the input data. Given a function f, we define

$$cond(f, x) = \lim_{\epsilon \to 0} \sup_{\|\Delta x\| \leq \epsilon \|x\|} \frac{\big\|f(x + \Delta x) - f(x)\big\|}{\epsilon \big\|f(x)\big\|}.$$

Given a problem, if its condition number is low it is said to be **well-conditioned** (typically $cond(f, x) \sim 1$), while a problem with a high condition number is (said to be) **ill-conditioned** ($cond(f, x) \gg 1$).

Let us now consider the problem of solving a linear equation subjected to a perturbation.

Let A be a non-singular matrix and assume we introduce a perturbation in the constant term $\tilde{\mathbf{b}} = \mathbf{b} + \delta\mathbf{b}$. The equation can be written as

$$A\tilde{\mathbf{x}} = \tilde{\mathbf{b}}$$

with $\tilde{\mathbf{x}} = \mathbf{x} + \delta\mathbf{x}$. We can obtain

$$\tilde{\mathbf{x}} - \mathbf{x} = A^{-1}\tilde{\mathbf{b}} - A^{-1}\mathbf{b} = A^{-1}\delta\mathbf{b}$$

and, by using matrix norms, we can write

$$\|\tilde{\mathbf{x}} - \mathbf{x}\| = \left\|A^{-1}\delta\mathbf{b}\right\| \leq \left\|A^{-1}\right\|\|\delta\mathbf{b}\|.$$

It is also known that

$$\|\mathbf{b}\| = \|A\mathbf{x}\| \leq \|A\|\|\mathbf{x}\|$$

which implies

$$\|\mathbf{x}\| \geq \frac{\|\mathbf{b}\|}{\|A\|}.$$

Tying all this together we can conclude

$$\frac{\|\tilde{\mathbf{x}} - \mathbf{x}\|}{\|\mathbf{x}\|} \leq \|A\|\left\|A^{-1}\right\|\frac{\|\delta\mathbf{b}\|}{\|\mathbf{b}\|}.$$

Definition 2.4 (Condition number of a matrix)**.** The condition number of a non-singular matrix A is defined as:

$$\kappa(A) = \left\|A^{-1}\right\|\|A\|.$$

Now, let us refocus on the topic of math expressions. Let us investigate what would happen if symbols are allowed in computations by introducing a framework that allows us to work with computer algebra.

Definition 2.5 (Computer algebra system)**.** A computer algebra system (CAS) is a mathematics software package that can perform *both symbolic and numerical mathematical computations.*

A CAS is usually a **REPL** expected to support a few functionalities [8]:

- **Arithmetic**: arithmetic over different fields with arbitrary precision.
- **Linear algebra**: matrix algebra and knowledge of different operations and properties of matrices (i.e. determinants, eigenvalues and eigenvectors).
- **Polynomial manipulation**: factorization over different fields, simplification and partial fraction decomposition of rational functions.
- **Transcendental functions**: support for transcendental functions and their properties.
- **Calculus**: limits, derivatives, integration and expansions of functions.
- **Solving equations**: solving systems of linear equations, computing with radicals solutions of polynomials of degree less than five.
- **Programming language**: users may implement their own algorithms using a programming language.

The CAS chosen for this work is **SageMath** [11], the features and functionalities of which shall not be discussed here.

SageMath is an open source CAS distributed under the terms of the GNU GPLv3 [6].

Hereafter, an example in which symbolic computations are put against numerical (computations) shall be made.

Example 2.2. Take matrix M from Example 2.1:

$$\begin{bmatrix} \sqrt{2} & 1 \\ 2 & \sqrt{2} \end{bmatrix}.$$

Compare the different results given out when computing its determinant by defining M over the *symbolic ring SR* and the *finite-precision ring CDF*:

```
sage: matrix(SR, [[sqrt(2), 1], [2, sqrt(2)]]).det()
0
sage: matrix(CDF, [[sqrt(2), 1], [2, sqrt(2)]]).det()
-3.14018491736755e-16
```

We can observe that in SR $(\sqrt{2})^2 = 2$ since no approximations are made.

Now, take the polynomial $p(x)$:

$$p(x) = x^6 + 5\,x^5 - 3\,x^4 - 42\,x^3 + 12\,x^2 - x + 1.$$

If an attempt to calculate its roots over SR is made an exception will be thrown (here, a reader may refer to Abel-Ruffini theorem for further explanations); however, doing this over a finite-precision ring (such as CDF) will work:

```
sage: p = x^6 + 5*x^5 - 3*x^4 -42*x^3 + 12*x^2 - x + 1
sage: p.roots(ring=SR)
    RuntimeError: no explicit roots found
sage: p.roots(ring=CDF)
[(-3.865705050148171 - 1.5654017866113432*I, 1),
(-3.8657050501481702 + 1.5654017866113419*I, 1),
(-0.04843174828928114 - 0.2430512799158686*I, 1),
(-0.048431748289281144 + 0.24305127991586856*I, 1),
(0.38275295887213723 + 7.286537374692244e-17*I, 1),
(2.4455206380027437 - 1.995314986816126e-16*I, 1)]
```

What we may conclude from such an example is that numerical analysis is certainly a powerful tool as it allows for computations which could not happen with computer algebra, **but** computer algebra being able to compute an exact answer without any approximation will prove to be useful in our use case.

For deeper reasoning about the limits of computer algebra systems, one may refer to Mitic [10].

Eigenvalues, eigenvectors

In the following section, we shall define **eigenvalues** and **eigenvectors** and discuss the numerical stability of their computation; a reader may also consult Axler [1] or Strang [12] for further explanations.

Definition 2.6 (Eigenvalue, eigenvector)**.** Given a linear transformation T in a finite-dimensional vector space V over a field F into itself and a nonzero vector $\mathbf{v}$, $\mathbf{v}$ is an eigenvector of T if and only if

$$A\mathbf{u} = \lambda\mathbf{u}$$

with A the matrix representation of T, $\mathbf{u}$ the coordinate vector of $\mathbf{u}$ and λ a scalar in F known as eigenvalue associated with $\mathbf{v}$.

Similarly, we can define a row vector $\mathbf{x}_L$, and a scalar λ_L such that

$$\mathbf{x}_L A = \lambda_L \mathbf{x}_L,$$

which are called **left eigenvector** and **left eigenvalue** respectively.

Remark. Note that writing $A\mathbf{u} = \lambda\mathbf{u}$ is equivalent to $(A - \lambda I)\mathbf{u} = 0$.
It follows that the eigenvalues of A are the roots of

$$det(A - \lambda I)$$

which is a polynomial in λ known as the **characteristic polynomial** $ch_A(\lambda)$.

Definition 2.7 (Eigenspace)**.** Given a square matrix A and its eigenvalue λ, we define the eigenspace of A associated with λ the subspace E_A of all vectors satisfying the equation

$$E_A = \{\mathbf{u} : (A - \lambda I)\mathbf{u} = 0\} = ker(A - \lambda I).$$

Definition 2.8 (Algebraic, geometric multiplicities of eigenvalues)**.** Given a square matrix A and a scalar $\lambda \in \mathbb{C}$: we define the algebraic multiplicity of λ as

$$m_A(\lambda) = max\{k : (\exists s(x) : s(x)(x - \lambda)^k = ch_A(x))\}.$$

The geometric multiplicity of λ is defined as

$$\nu_A(\lambda) = dim(ker(A - \lambda I)).$$

Remark. Suppose A is a real square matrix, then the following statements are true:

- the eigenvalues of the left and right eigenvectors of A are the same,
- the left eigenvectors simplify into the transpose of the right eigenvectors of A^T.

Now, let us investigate how introducing perturbations in the representation of a matrix may influence the numerical stability of its eigenvalues.

Let A be a square matrix, $\lambda \in \mathbb{C}$ its eigenvalue, $\mathbf{x}$, $\mathbf{y}$ the right and left eigenvectors associated with λ. Consider the perturbed problem

$$\tilde{A}\tilde{\mathbf{x}} = \tilde{\lambda}\tilde{\mathbf{x}}$$

with ϵ the machine epsilon, $\tilde{A} = A + \epsilon\delta A$, $\tilde{\mathbf{x}} = \mathbf{x} + \epsilon\delta\mathbf{x}$, $\tilde{\lambda} = \lambda + \epsilon\delta\lambda$.

Differentiating w.r.t. ϵ and multiplying by $\mathbf{y}^T$ on the left side gives

$$\mathbf{y}^T \delta A \mathbf{x} + \mathbf{y}^T A fl(\mathbf{x}) = fl(\lambda)\mathbf{y}^T \mathbf{x} + \mathbf{y}^T \lambda fl(\mathbf{x})$$

and, since $\mathbf{y}$ is the left eigenvector we can rewrite it as

$$\frac{\delta\lambda}{\delta\epsilon} = \frac{\mathbf{y}^T \delta A \mathbf{x}}{\mathbf{y}^T \mathbf{x}}.$$

Assuming $\|\delta A\| = 1$ and using the definition of dot product for $\mathbf{y}^T\mathbf{x}$ we get

$$|\delta\lambda| \leq \frac{1}{|\cos(\theta_\lambda)|}|\delta\epsilon|.$$

Definition 2.9 (Condition number of an eigenvalue)**.** Given a square matrix A, the eigenvalue $\lambda \in \mathbb{C}$ and θ_λ the angle between the left and right eigenvectors associated with λ, the quantity

$$k_A(\lambda) = \frac{1}{\cos(\theta_\lambda)}$$

is called the condition number of the eigenvalue λ.

Jordan canonical form

In the following section, we shall define **Jordan matrices** and discuss the stability of a transformation of a matrix into its Jordan canonical form.

Definition 2.10 (Jordan matrix)**.** A diagonal block matrix M is called a Jordan matrix if and only if each block along the diagonal is of the form

$$\begin{bmatrix} \lambda & 1 & 0 & \cdots & 0 \\ 0 & \lambda & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \lambda & 1 \\ 0 & 0 & 0 & 0 & \lambda \end{bmatrix},$$

and we can write such a matrix M as $M = diag(J_{\lambda_1,n_1}, ..., J_{\lambda_k,n_k})$ with k the number of diagonal blocks it is made up of.

Each $n \times n$ block can be completely characterized by the tuple (λ, n) as it can fully describe both the structure and the dimension of a block.

Remark. Let V be a vector space defined over a field F and A a matrix defined in V. If the characteristic polynomial of A $ch_A(t)$ can be factorized into its linear factors over K, then A is similar to a Jordan matrix J. We define J the **Jordan canonical form** (**JCF**) of A.

Definition 2.11 (Defective matrix, defective eigenvalue)**.** Given a square $n \times n$ matrix A, if it has less than n distinct eigenvalues then it is called a defective matrix.

Furthermore, we define an eigenvalue λ of such a matrix as a defective eigenvalue if and only if

$$m_A(\lambda) > \nu_A(\lambda).$$

Now, we shall give a result on the stability of such a transformation the proof of which can be found in other works, such as Datta [3].

Theorem 2.1 (Stability of the JCF transformation)**.** Given a matrix A and its JCF $A = P^{-1}JP$, the transforming matrix P is highly ill-conditioned whenever A has at least a defective or nearly defective eigenvalue.

Lastly, we shall give an example to show the implications of this theorem by showing the differences in the JCF of a matrix and its perturbed version.

Example 2.3. Consider the $n \times n$ matrices

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 1 \\ \alpha & 0 & 0 & 0 & 0 \end{bmatrix}$$

with $\alpha > 0$.

It is evident how A has a defective eigenvalue in $\lambda_A = 0$ and $m_A(0) = n$, $\nu_A(0) = 1$; furthermore, A is already in JCF.

Now, let us switch our focus to B. To compute its eigenvalues, take the characteristic polynomial $ch_B(t) = t^n - \alpha$: it has n distinct roots in

$$t_j = z_n^j \sqrt[n]{\alpha}$$

with $j = 1, ..., n$, $z_n = \cos\left(\frac{2\pi}{n}\right) + \mathrm{i}\sin\left(\frac{2\pi}{n}\right)$ and i imaginary unit such that $\mathrm{i}^2 = -1$.

To conclude, we shall show the JCF of A and B defined in SR computed by SageMath when $n = 4$.

```
sage: A = matrix(SR, [
            [1 if i == j-1
                else 0 for j in range(4)]
            for i in range(4)
        ])
sage: B = matrix(SR, [
            [1 if i == j-1
                else x if j == 0 and i == 3
                    else 0 for j in range(4)]
            for i in range(4)
        ])
sage: A
[0 1 0 0]
[0 0 1 0]
[0 0 0 1]
[0 0 0 0]
sage: A.jordan_form()
[0 1 0 0]
[0 0 1 0]
[0 0 0 1]
[0 0 0 0]
sage: B
[0 1 0 0]
[0 0 1 0]
[0 0 0 1]
[x 0 0 0]
sage: B.jordan_form()
[ I*x^(1/4)|         0|         0|         0]
[----------+----------+----------+----------]
[         0|  -x^(1/4)|         0|         0]
[----------+----------+----------+----------]
[         0|         0|-I*x^(1/4)|         0]
[----------+----------+----------+----------]
[         0|         0|         0|   x^(1/4)]
```

For the sake of clarity, we shall also show what the implications of B having such eigenvalues are.

Suppose $x = 10^{-10}$.

```
sage: B = matrix(SR, [
            [1 if i == j-1
                else 10**-10 if j == 0 and i == 3
                    else 0 for j in range(4)]
            for i in range(4)
        ])
sage: B
[           0            1            0            0]
[           0            0            1            0]
[           0            0            0            1]
[1/10000000000            0            0            0]
sage: P =  = B.jordan_form(transformation=True)[1]
sage: cond = norm(P.inverse()) * norm(P)
sage: cond
31622776.60168379
```

We can see that $\kappa(P) \gg 1$, as stated in Stability of the JCF transformation (theorem 2.1).

Theory and applications of pencils of matrices

This chapter will introduce the reader to the concept of a linear pencil of matrices and its properties. Throughout this chapter and the following one, a reader may refer to Gäntmacher [4], Kunkel, Mehrmann [9] and Beelen, Van Dooren [2].

Definition 3.1 (Linear matrix pencil)**.** A linear pencil of matrices is defined as a 1-degree polynomial with matrix coefficients

$$\Gamma(\lambda) = A + \lambda B$$

with $\lambda \in \mathbb{C}, A$ and B $m \times n$ matrices. A linear pencil of matrices may also be called a **pair of matrices** and, in this thesis, we shall use synonymously both terms.

Regular pencils.

Now, we consider the case where $(A + \lambda B)$ is a regular pencil of matrices.

Definition 3.2 (Regular pencil)**.** A matrix pair (A, B) is said to be regular if and only if A and B are square matrices of the same size and the determinant $det(A + \lambda B)$ is not identically zero.

Consider the regular pencil of matrices

$$\Gamma(\lambda) = A + \lambda B,$$

let F be the field the entries of A and B belong to and r the rank of the pencil.

Denote with $D_j(\lambda)$ the greatest common divisor of all minors of order j of $\Gamma(\lambda)$ (with $j = 1, ..., r$) and assume without any loss of generality $D_j(\lambda)$ is monic and $D_0(\lambda) = 1$. Given the sequence,

$$D_r(\lambda),\ D_{r-1}(\lambda),\ ...,\ D_1(\lambda), D_0(\lambda)$$

we define the **invariant polynomials** of the pencil of matrices $\Gamma(\lambda)$ as the fractions

$$i_1(\lambda) = \frac{D_r(\lambda)}{D_{r-1}(\lambda)},\ i_2(\lambda) = \frac{D_{r-1}(\lambda)}{D_{r-2}(\lambda)},\ ...,\ i_r(\lambda) = D_1(\lambda).$$

We can now write the expansion of the invariant polynomials into irreducible factors in F as

$$i_1(\lambda) = \prod_{i=1}^{k} p_i(\lambda)^{\alpha_{1,i}},\ i_2(\lambda) = \prod_{i=1}^{k} p_i(\lambda)^{\alpha_{2,i}},\ \dots$$
$$i_r(\lambda) = \prod_{i=1}^{k} p_i(\lambda)^{\alpha_{r,i}},$$

with p_i an irreducible polynomial appearing in the expansion.

We define the **elementary**(**finite**) **divisors** e_i of the pencil of matrices $\Gamma(\lambda)$ all the polynomials $p_i(\lambda)^{\alpha_{j,i}}$ (with $j = 1, ..., r$) that are not equal to one.

A similar procedure may be defined for the pencil of matrices

$$\Theta(\mu, \lambda) = \mu A + \lambda B$$

leading to polynomials in two variables (μ, λ). Clearly, having $\mu = 1$ would lead to obtaining the elementary finite divisors of $\Gamma(\lambda)$; however, for each elementary divisor of degree q we have

$$e_i(\mu, \lambda) = \mu^q e_i(\frac{\lambda}{\mu}),$$

and, with this technique, it is possible to generate all the elementary divisors of $\Theta(\mu, \lambda)$ except for those of the form μ^q, which are called **elementary infinite divisors** of the pencil of matrices $\Theta(\mu, \lambda)$.

Remark. A regular pencil of matrices $\Gamma(\lambda) = A + \lambda B$ has elementary infinite divisors if and only if $det(B) = 0$.

We can now give a result on the equivalence of regular pencils.

Theorem 3.1 (Equivalence of regular pencils of matrices)**.** Two regular matrix pairs

$$\Gamma_1(\lambda) = (A_1, B_1) \qquad\qquad \Gamma_2(\lambda) = (A_2, B_2)$$

are equivalent if and only if they have the same finite and infinite elementary divisors.

In other words, Γ_1 and Γ_2 are equivalent if and only if the invariant polynomials of the two pencils $\Theta_1(\mu,\lambda)$ and $\Theta_2(\mu,\lambda)$ are equivalent, with

$$\Theta_1(\mu,\lambda) = \mu A + \lambda B \qquad\qquad \Theta_2(\mu,\lambda) = \mu A + \lambda B$$

Lemma 3.2. For any two regular equivalent pencils of $n \times n$ matrices the entries of which lie in a field F

$$\Gamma_1(\lambda) = A_1 + \lambda B_1 \qquad\qquad \Gamma_2(\lambda) = A_2 + \lambda B_2$$

there exist two nonsingular matrices $P \in F^{n\times n}$, $Q \in F^{n\times n}$ such that

$$P\Gamma_1(\lambda)Q = \Gamma_2(\lambda),$$

and we call such a transformation an **equivalence transformation**.

Singular pencils.

Next, we shall investigate the most general case of $m \times n$ pencils of matrices in order to introduce the reader to the concept of minimal indices of a pencil of matrices.

Definition 3.3 (Singular pencil)**.** A matrix pair A, B is said to be singular if and only if it is not regular.

Consider the singular pencil of (rectangular) matrices $\Gamma(\lambda) = A + \lambda B$ and assume its rank r is smaller than its number of columns n.

This implies the equation

$$(A + \lambda B)\mathbf{x} = 0$$

has nontrivial solutions $\mathbf{x}_1(\lambda), ..., \mathbf{x}_k(\lambda)$ and let X be the polynomial matrix made up of such polynomials

$$X = \begin{bmatrix} x_{1,1} & x_{2,1} & \dots & x_{k,1} \\ x_{1,2} & x_{2,2} & \dots & x_{k,1} \\ \vdots & & & \vdots \\ x_{1,n} & x_{2,n} & \dots & x_{k,n} \end{bmatrix}$$

The columns $\mathbf{x}_i(\lambda)$ (with $i = 1, ..., k$) of X can be chosen to be linearly independent; as a matter of fact, the columns are linearly dependent if the rank of X is less than k and, (only) in this case, we can choose k nontrivial polynomials $p_i(\lambda)$ such that

$$\sum_{i=1}^{k} p_i(\lambda)\mathbf{x}_i(\lambda) \equiv 0.$$

We choose the nontrivial polynomial $\mathbf{x}_1(\lambda)$ of least degree ϵ_1 in λ; next, amongst the very same solutions, we choose $\mathbf{x}_2(\lambda)$ of least degree ϵ_2 etc.

Since the maximum number of linearly independent solutions of the aforementioned equation is no more than $n - r$, this process has a finite number of steps.

To summarize, from an equation $(A+\lambda B)\mathbf{x} = 0$ we can obtain a sequence of solutions of non-increasing degree

$$\epsilon_1 \leq \epsilon_2 \leq ... \leq \epsilon_p,$$

and we define ϵ_i a **minimal index for the columns** of the pencil of matrices $\Gamma(\lambda)$.

We can also introduce **minimal indices for the rows** $\eta_1, \eta_2, ..., \eta_q$ of the pencil $(A + \lambda B)$ which we can yield working with the transpose of the pencil $A^T + \lambda B^T$ or, in other words, with the equation

$$(A^T + \lambda B^T)\mathbf{y} = 0.$$

We can now give another result on the equivalence of pencils of matrices.

Theorem 3.3 (Necessary condition for the equivalence of pencils)**.** Two arbitrary equivalent pencils have the same minimal indices for rows and columns.

Kronecker canonical form.

At this point, we can put together the theorems on the equivalence of regular and singular pencil of matrices, namely Equivalence of regular pencils of matrices (theorem 3.1) and Necessary condition for the equivalence of pencils (theorem 3.3) and give a result on the **equivalence of arbitrary matrix pairs**.

Theorem 3.4 (Kronecker)**.** Two pencils $(A+\lambda B)$, $(A_1+\lambda B_1)$ of rectangular $m \times n$ matrices are equivalent if and only if they have the same minimal indices for rows and columns and the same elementary finite and infinite divisors.

Remark. Restating Kronecker (theorem 3.4) using different words, we can say that a matrix pair (A, B) is completely characterized by its minimal indices for rows and columns and (its) elementary finite and infinite divisors and does not depend on their order.

Coherently, it should be possible to define a canonical form for pencils of matrices completely determined by both the minimal indices for rows and columns and the elementary finite and infinite divisors, and it is. We can now introduce the Kronecker canonical form of a matrix pair.

Theorem 3.5 (Kronecker canonical form)**.** Let $\Gamma(\lambda) = A + \lambda B$ be an arbitrary pencil of matrices and h, g the maximal number of constant independent solutions of the two equations

$$(A + \lambda B)\mathbf{x} = 0 \qquad\qquad (A^T + \lambda B^T)\mathbf{y} = 0.$$

The pencil $\Gamma(\lambda)$ is strictly equivalent to a quasi-diagonal matrix

$$\{Z; L_{\epsilon_{g+1}}, L_{\epsilon_{g+2}}, ..., L_{\epsilon_p}; L^T_{\eta_{h+1}}, L^T_{\eta_{h+2}}, ..., L_{\eta^T_q}; N^{(u_1)}, N^{(u_2)}, ..., N^{(u_s)}; J + \lambda I\},$$

where Z is an $h \times g$ null matrix, L_ϵ a rectangular $\epsilon \times (\epsilon + 1)$ matrix of the form

$$L_\epsilon = \begin{bmatrix} \lambda & 1 & 0 & \dots & 0 & 0 & \\ 0 & \lambda & 1 & & \vdots & \vdots & \\ \vdots & \vdots & & \ddots & & & \\ 0 & 0 & & & \lambda & 1 & \end{bmatrix}$$

and

$$N^u = I^u + \lambda H^u$$

with I the $u \times u$ identity matrix and H the $u \times u$ upper shift matrix, and J a Jordan matrix.

We shall call such a matrix the Kronecker canonical form of $\Gamma(\lambda)$ and write it out as $K(\Gamma(\lambda))$.

For a proof of Kronecker (theorem 3.4) and Kronecker canonical form (theorem 3.5) the reader shall refer to the following chapter.

Corollary 3.5.1. Given a matrix pair (A, B) there exists a tuple of matrices P, Q such that

$$PAQ = K(A) \qquad\qquad PBQ = K(B),$$

with P an $m \times m$ and Q an $n \times n$ matrices respectively.

Example 3.1. Kronecker's canonical form of the pencil of matrices

$$\begin{bmatrix} 0 & & & & & \\ & \lambda & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & 1 & \lambda & \\ & & & 0 & 1 & \\ & & & & & \lambda+1 \end{bmatrix}.$$

is made up of three rectangular blocks $Z^{1\times 1}$, L_0, L_1^T, two nilpotent blocks $N^{(1)}$, $N^{(2)}$ and one Jordan block $J + \lambda I^{(1)}$.

Fundamental applications.

The following section shall introduce the reader to differential-algebraic equations (which we may shorten as DAE) in order to explain what is the use case of the KCF of a pencil of matrices while keeping a strong focus on examples.

Definition 3.4 (Differential-algebraic equation)**.** An equation

$$F(t, x, \dot{x}) = 0$$

with $F : I \times D_x \times D_x \to \mathbb{C}^m$, $I \subseteq \mathbb{R}$ is a compact interval, D_x, $D_x \subseteq \mathbb{C}^n$ are open, $x : I \to \mathbb{C}^n$ a differentiable function and m, $n \in \mathbb{N}$ is called differential-algebraic equation.

What we're attempting to determine is a function x such that

$$\forall x \in I.F(t, x(t), \dot{x(t)}) = 0.$$

Differential-algebraic equations are used to model physical systems with a dynamic behaviour the states of which are subjected to certain constraints. For our purposes, we'll narrow the definition given above and rewrite it as follows.

Definition 3.5 (Linear DAE with constant coefficients)**.** An equation

$$B\dot{x} + Ax = f(t)$$

with (A, $B \in \mathbb{C}^{m\times n}$), $f : I \to \mathbb{C}^m$ a continuous and twice-differentiable function, $I \subseteq \mathbb{R}$ a compact interval and $m \in \mathbb{N}$ is called a linear differential-algebraic equation with constant coefficients.

It may be paired with an initial condition

$$x(t_0) = x_0.$$

Remark. The equations $B\dot{x} + Ax = f(t)$ and $\tilde{B}\dot{\tilde{x}} + \tilde{A}\tilde{x} = \tilde{f}(t)$, with

$$\tilde{A} = PAQ, \qquad \tilde{B} = PBQ, \qquad \tilde{f} = Pf, \quad \tilde{x} = Q^{-1}x,$$
$$P \in \mathbb{C}^{m\times m}, \qquad Q \in \mathbb{C}^{n\times n}$$

and P, Q nonsingular matrices, are both linear differential-algebraic equations with constant coefficients and the relation $x = Q\tilde{x}$ provides a one-to-one mapping between their solution sets.

Next, we shall take as examples pencils of matrices made up of the blocks described before in order to show how the linear DAE with constant coefficients associated may be solved.

Example 3.2. Consider the pencil

$$\Gamma(\lambda) = \begin{bmatrix} \lambda & 1 & 0 \\ 0 & \lambda & 1 \end{bmatrix}.$$

We write the linear DAE with constant coefficients associated with $\Gamma(\lambda)$ as

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \end{bmatrix} + \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} f_1(t) \\ f_2(t) \end{bmatrix}.$$

To compute its solutions we must solve the linear system associated with the DAE specified above

$$\begin{cases} \dot{x_1} + x_2 - f_1(t) = 0 \\ \dot{x_2} + x_3 - f_2(t) = 0 \end{cases} \tag{1}$$

Denoting with g a generic C^2 function, the solution is

$$\begin{cases} x_1 = g \\ x_2 = -\dot{x_1} + f_1(t) \\ x_3 = \ddot{x_1} - \dot{f_1}(t) + f_2(t) \end{cases}$$

Example 3.3. For this example, we'll take the transpose of the pencil of matrices given in 3.2

$$\Gamma(\lambda) = \begin{bmatrix} \lambda & 0 \\ 1 & \lambda \\ 0 & 1 \end{bmatrix}.$$

The DAE associated with $\Gamma(\lambda)$ is

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{x_1} \\ \dot{x_2} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} f_1(t) \\ f_2(t) \\ f_3(t) \end{bmatrix}.$$

We write this as a linear system

$$\begin{cases} \dot{x_1} - f_1(t) = 0 \\ \dot{x_2} + x_1 - f_2(t) = 0 \\ x_2 - f_3(t) = 0 \end{cases}$$

The system can be solved if and only if

$$f_1(t) = \dot{f}_2(t) - \ddot{f}_3(t),$$

and its solution is

$$\begin{cases} x_1 = f_2(t) - \dot{f}_3(t) \\ x_2 = f_3(t) \end{cases}$$

Example 3.4. Let $\Gamma(\lambda)$ be the pencil of matrices

$$\Gamma(\lambda) = \begin{bmatrix} 1 & \lambda & 0 & 0 \\ 0 & 1 & \lambda & 0 \\ 0 & 0 & 1 & \lambda \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Rewriting this as a linear DAE with constant coefficients yields

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \\ \dot{x_4} \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} f_1(t) \\ f_2(t) \\ f_3(t) \\ f_4(t) \end{bmatrix}$$

We follow the same steps as in other examples

$$\begin{cases} \dot{x_2} + x_1 - f_1(t) = 0 \\ \dot{x_3} + x_2 - f_2(t) = 0 \\ \dot{x_4} + x_3 - f_3(t) = 0 \\ x_4 - f_4(t) = 0 \end{cases}$$

The solution is

$$\begin{cases} x_1 = f_1(t) - \dot{f}_2(t) + \ddot{f}_3(t) - \dddot{f}_4(t) \\ x_2 = f_2(t) - \dot{f}_3(t) + \ddot{f}_4(t) \\ x_3 = f_3(t) - \dot{f}_4(t) \\ x_4 = f_4(t) \end{cases}$$

Example 3.5. Let us take the pencil

$$\Gamma(\lambda) = \begin{bmatrix} \lambda & 1 & 0 & 0 \\ 0 & \lambda & 0 & 0 \\ 0 & 0 & \lambda+2 & 1 \\ 0 & 0 & 0 & \lambda+2 \end{bmatrix}$$

and write the DAE associated with it

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \\ \dot{x_4} \end{bmatrix} + \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 1 \\ 0 & 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} f_1(t) \\ f_2(t) \\ f_3(t) \\ f_4(t) \end{bmatrix}.$$

The linear system to solve is

$$\begin{cases} \dot{x_1} + x_2 - f_1(t) = 0 \\ \dot{x_2} - f_2(t) = 0 \\ \dot{x_3} + 2x_3 + x_4 - f_3(t) = 0 \\ \dot{x_4} + 2x_4 - f_4(t) = 0 \end{cases}$$

Denoting with g a C^2 function such that

$$\ddot{g} = \dot{f_1}(t) - f_2(t),$$

and (with) D, K arbitrary constants, the solution is

$$\begin{cases} x_1 = g \\ x_2 = f_1(t) - \dot{g} \\ x_3 = -\dfrac{1}{4}\left(4Kt - 2f_3(t)e^{2t} + f_4(t)e^{2t} - 4K\right)e^{-2t} \\ x_4 = \dfrac{1}{2}\left(f_4(t)e^{2t} + 2D\right)e^{-2t} \end{cases}$$

Computation of Kronecker's Canonical Form

The following chapter deals with the problem of computing Kronecker's canonical form for a matrix pair (A, B). The approach described will be shown to be correct, and an implementation of it using the CAS SageMath has been made publicly available under MIT License on GitHub [5].

We shall divide the procedure into two steps: the first one deals with regular pencils of matrices, the other with singular pencils.

At the end of each of the following sections, the steps described shall be summarised in pseudocode.

Regular pencils.

Let $\Gamma(\lambda) = A + \lambda B$ be a regular pencil of matrices defined in a vector space over a field F.

First, we shall find a value $c \in F$ such that $det(\Gamma(c)) \neq 0$ and define the matrix

$$A_1 = A + cB.$$

Now, we rewrite $\Gamma(\lambda)$ in terms of A_1,

$$\Gamma(\lambda) = A_1 + (\lambda - c)B,$$

and premultiply it by A_1^{-1}

$$A_1^{-1}\Gamma(\lambda) = I + (\lambda - c)A_1^{-1}B. \tag{2}$$

Let us denote with J the JCF of $A_1^{-1}B$ and P_1 the similarity matrix used to compute it

$$J = P_1^{-1}(A_1^{-1}B)P_1,$$

and let us assume it is of the form

$$J = \begin{bmatrix} J_1 & \\ & J_0 \end{bmatrix}$$

with J_0 a nilpotent Jordan matrix and J_1 such that $det(J_1) \neq 0$ (which is the form returned by SageMath when computing Jordan matrices); call j, k the number of rows in J_0 and J_1 respectively.

Define the permutation matrix

$$P_\pi = \begin{bmatrix} & I^{(k)} \\ I^{(j)} & \end{bmatrix}$$

so that we can write

$$P_\pi^T J P_\pi = J' = \begin{bmatrix} J_0 & \\ & J_1 \end{bmatrix}.$$

Now, we rewrite (2) substituting $A_1^{-1}B$ with its permuted Jordan form

$$A_1^{-1}\Gamma(\lambda) = I + (\lambda - c)P_1 P_\pi^{-T} J' P_\pi^{-1} P_1^{-1}. \tag{3}$$

We can write the identity matrix in terms of P_1 as $I = P_1 P_1^{-1}$ in (3)

$$\begin{aligned}
&A_1^{-1}\Gamma(\lambda) = \\
&= P_1 P_1^{-1} + (\lambda - c)P_1 P_\pi^{-T} J' P_\pi^{-1} P_1^{-1} = \\
&= P_1(P_1^{-1} + (\lambda - c)P_\pi^{-T} J' P_\pi^{-1} P_1^{-1}).
\end{aligned}$$

We premultiply it by P_1^{-1}

$$P_1^{-1} A_1^{-1}\Gamma(\lambda) = P_1^{-1} + (\lambda - c)P_\pi^{-T} J' P_\pi^{-1} P_1^{-1}$$

and then postmultiply by P_1

$$P_1^{-1} A_1^{-1}\Gamma(\lambda) P_1 = I + (\lambda - c)P_\pi^{-T} J' P_\pi^{-1}.$$

The very same steps can be followed for P_π^{-T} and P_π^{-1}.

The final result is

$$P_\pi^T P_1^{-1} A_1^{-1}\Gamma(\lambda) P_1 P_\pi = I + (\lambda - c)J'. \tag{4}$$

Now, we may work on the blocks J_0, J_1 of J'.

For the following steps, it is useful to rewrite the expression on the right side of (4) so that the form of its diagonal blocks is explicitly readable, meaning

$$I + (\lambda - c)J' = \begin{bmatrix} I^{(j)} - cJ_0 + \lambda J_0 & \\ & I^{(k)} - cJ_1 + \lambda J_1 \end{bmatrix}.$$

Let us start from the first diagonal block

$$I^{(j)} - cJ_0 + \lambda J_0.$$

First, we need to postmultiply it in (4) by $K = (I^{(j)} - cJ_0)^{-1}$

$$P_\pi^T P_1^{-1} A_1^{-1} \Gamma(\lambda) P_1 P_\pi \begin{bmatrix} K & \\ & I^{(k)} \end{bmatrix} = \begin{bmatrix} I^{(j)} + \lambda K J_0 & \\ & I^{(k)} - cJ_1 + \lambda J_1 \end{bmatrix}.$$

Let us denote with $H^{(j)}$ the JCF of KJ_0, P_2 the similarity matrix used to compute it and $N^{(j)} = I^{(j)} + \lambda H^{(j)}$

$$N^{(j)} = I^{(j)} + \lambda H^{(j)} = I^{(j)} + \lambda P_2^{-1}(KJ_0)P_2.$$

Remark. H is an upper shift matrix as it is the JCF of a nilpotent matrix.

Following the analogous steps to handle P_2 and its inverse yields us

$$\begin{bmatrix} P_2^{-1} & \\ & I \end{bmatrix} P_\pi^T P_1^{-1} A_1^{-1} \Gamma(\lambda) P_1 P_\pi \begin{bmatrix} KP_2 & \\ & I^{(k)} \end{bmatrix} = \begin{bmatrix} N^{(j)} & \\ & I^{(k)} - cJ_1 + \lambda J_1 \end{bmatrix}. \quad (5)$$

At this point, we can focus on the second diagonal block

$$I^{(k)} - cJ_1 + \lambda J_1.$$

We postmultiply it in (5) by J_1^{-1}

$$\begin{bmatrix} P_2^{-1} & \\ & I \end{bmatrix} P_\pi^T P_1^{-1} A_1^{-1} \Gamma(\lambda) P_1 P_\pi \begin{bmatrix} KP_2 & \\ & J_1^{-1} \end{bmatrix} = \begin{bmatrix} N^{(j)} & \\ & J_1^{-1} + (\lambda - c)I^{(k)} \end{bmatrix}. \quad (6)$$

Let us denote with G the JCF of the constant term $J_1^{-1} - cI^{(k)}$ and P_3 the similarity matrix used to compute it

$$J_1^{-1} - cI^{(k)} = P_3^{-1} G P_3.$$

Again, we follow the same procedure to handle P_3 and its inverse, thus obtaining our end result

$$\begin{bmatrix} P_2^{-1} & \\ & P_3^{-1} \end{bmatrix} P_\pi^T P_1^{-1} A_1^{-1} \Gamma(\lambda) P_1 P_\pi \begin{bmatrix} KP_2 & \\ & J_1^{-1} P_3 \end{bmatrix} = \begin{bmatrix} N^{(j)} & \\ & G + \lambda I^{(k)} \end{bmatrix}. \quad (7)$$

To conclude, we shall present the aforementioned steps used in order to compute Kronecker's canonical form $K(\Gamma(\lambda))$ in pseudocode.

Algorithm 1: Procedure to compute KCF of a regular pencil.

```
Data: Γ(λ) = A + λB: regular pencil
Result: K(Γ(λ)): KCF of the pencil of matrices Γ(λ)
while True do
    c ← random value;
    if det(Γ(c)) ≠ 0 then
        break
    end
end
A_1 ← A + c * B;
J ← jordan(A_1^{-1} * B);
{J_0, J_1} ← submatrices(J);
{N_i}_{i≥0} ← jordan((I − c * J_0)^{-1} * J_0);
G ← jordan(J_1^{-1} − c * I);
K(Γ(λ)) ← diag({N_i}, G + λI)
```

We have now proved the following theorem.

Theorem 4.1 (KCF of a regular pencil of matrices)**.** Every regular pencil of matrices $\Gamma(\lambda) = A + \lambda B$ can be reduced to a matrix of the form

$$\begin{bmatrix} N^{(u_1)} & & & & \\ & N^{(u_2)} & & & \\ & & \ddots & & \\ & & & N^{(u_s)} & \\ & & & & J + \lambda I \end{bmatrix},$$

where the first s diagonal blocks correspond to infinite elementary divisors $\mu^{u_1}, ..., \mu^{u_s}$ of $\Gamma(\lambda)$ and the last block is uniquely determined by the finite elementary divisors of the given pencil.

Theorem 4.2 (Stability of KCF transformation for a regular pencil)**.** The problem of computing Kronecker's canonical form for a regular pencil is ill-conditioned.

We can see Stability of KCF transformation for a regular pencil (theorem 4.2) is trivially true as it involves the computation of many JCFs.

Now, we'll give an example to show how this may influence the end result.

Example 4.1.

Singular pencils.

In order to handle singular pencils of matrices, we must first prove the following theorem.

Theorem 4.3. Given a singular pencil of matrices $\Gamma(\lambda) = A + \lambda B$ of dimensions $m \times n$ and rank $r < n$, the degree ϵ of the polynomial

$$\mathbf{x}(\lambda) = x_0 - \lambda x_1 + \lambda^2 x_2 - ... + (-1)^\epsilon \lambda^\epsilon x_\epsilon$$

with $x_\epsilon \neq 0$ is the least value for which the sign $<$ holds in the relation $\rho_k \leq (k+1)n$ where ρ_k is the rank of a $k \times k+1$ matrix of the form

$$\begin{bmatrix} A & 0 & \dots & 0 \\ B & A & & \vdots \\ 0 & B & \ddots & \\ \vdots & \vdots & \ddots & A \\ 0 & 0 & \dots & B \end{bmatrix}.$$

Proof. Since the columns of the pencil of matrices $\Gamma(\lambda)$ are linearly dependent, the equation

$$(A + \lambda B)\mathbf{x} = 0$$

has a non-zero solution. We choose the solution $\mathbf{x}(\lambda)$ of the least possible degree ϵ.

We substitute $\mathbf{x}(\lambda)$ in the equation and obtain

$$Ax_0 = 0, \quad Bx_0 - Ax_1 = 0, \quad ..., \quad Bx_{\epsilon-1} - Ax_\epsilon = 0, \quad Bx_\epsilon = 0,$$

which can, in turn, be considered as a system of linear homogenous equations for the elements of the columns $x_0 - x_1, x_2..., (-1)^\epsilon x_\epsilon$; we can write the coefficient matrix of the system as

$$M_\epsilon = \begin{bmatrix} A & 0 & \dots & 0 \\ B & A & & \vdots \\ 0 & B & \ddots & \\ \vdots & \vdots & \ddots & A \\ 0 & 0 & \dots & B \end{bmatrix}.$$

We know by construction that the rank of the matrix M_ϵ $\rho_\epsilon < (\epsilon+1)n$ and (that), since ϵ is minimal, the ranks $\rho_0, \rho_1, ..., \rho_{\epsilon-1}$ of the matrices

$$M_0 = \begin{bmatrix} A \\ B \end{bmatrix}, \qquad M_1 = \begin{bmatrix} A & 0 \\ B & A \\ 0 & B \end{bmatrix}, \qquad M_{\epsilon-1} = \begin{bmatrix} A & 0 & \dots & 0 \\ B & A & & \vdots \\ 0 & B & \ddots & \\ \vdots & \vdots & \ddots & A \\ 0 & 0 & \dots & B \end{bmatrix}$$

satisfy the equations $\rho_0 = n, \rho_1 = 2, \rho_{\epsilon-1} = \epsilon n$ thus proving the theorem. $\square$

At this point, we can state and prove a fundamental theorem which will certainly aid us in proving Kronecker canonical form (theorem 3.5).

Theorem 4.4 (Reduction theorem)**.** If the equation given by a matrix pair $\Gamma(\lambda) = (A, B)$ has a solution of minimal degree $\epsilon > 0$, then $\Gamma(\lambda)$ is strictly equivalent to a pencil of matrices of the form

$$\begin{bmatrix} L_\epsilon & 0 \\ 0 & \tilde{A} + \lambda \tilde{B} \end{bmatrix},$$

where the equation analogous for $(\tilde{A}, \tilde{B})$ has no solution of degree $\alpha < \epsilon$ and L_ϵ is an $\epsilon \times (\epsilon+1)$ matrix such that

$$\begin{bmatrix} \lambda & 1 & 0 & \dots & 0 & 0 \\ 0 & \lambda & 1 & & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & & \\ 0 & & & & \lambda & 1 \end{bmatrix}.$$

Proof. □

Conclusions

Non aliquip id esse qui consequat do in sit incididunt ipsum. Pariatur aute ut excepteur incididunt qui aute mollit ea. Dolor consequat nostrud nisi duis est esse. In labore pariatur excepteur incididunt. Minim et culpa consequat amet id excepteur amet esse. Cillum ut proident minim esse cillum ea laborum. Commodo ad adipisicing nulla velit irure sunt commodo sunt sunt.

Eiusmod adipisicing incididunt reprehenderit amet dolore veniam aute cupidatat tempor officia id adipisicing. Cupidatat non enim deserunt nisi exercitation fugiat. Exercitation exercitation magna ullamco id adipisicing deserunt irure cupidatat veniam sit reprehenderit non reprehenderit. Ipsum adipisicing anim sint ullamco incididunt pariatur amet consequat nulla dolore qui esse. Veniam ullamco aliquip voluptate est ea cupidatat occaecat id exercitation proident irure non.

Culpa sit ullamco ipsum eiusmod Lorem et. Consequat ex consectetur officia non sint id. Et culpa velit nulla Lorem Lorem adipisicing aute enim cillum officia commodo sint adipisicing. Sit veniam laboris esse magna ipsum aute tempor velit incididunt sint.

Consequat ex aute adipisicing sint pariatur mollit aute eu voluptate reprehenderit dolore laboris sunt ex. Ea nisi laborum nisi excepteur adipisicing cupidatat duis occaecat reprehenderit. Consequat consectetur qui incididunt voluptate culpa. Velit et magna laborum excepteur sint minim proident. Aliquip ipsum non minim qui cupidatat et quis. Non magna occaecat nostrud reprehenderit sint proident ad cupidatat eiusmod elit occaecat enim et nostrud.

Labore reprehenderit amet incididunt irure velit. Ipsum qui reprehenderit dolore adipisicing. Incididunt eiusmod ad do exercitation aute do fugiat elit mollit. Irure exercitation aliqua non minim consequat do adipisicing commodo enim id magna. Quis eu minim culpa eu.

Fugiat ea in pariatur nostrud esse id duis ipsum officia ut. Voluptate tempor est velit pariatur ipsum incididunt mollit consectetur laborum enim laborum dolore dolor. Nostrud culpa ad aliquip magna velit magna ipsum consectetur exercitation dolore dolor. Culpa ullamco aliquip aute deserunt. Proident voluptate cillum adipisicing culpa deserunt eiusmod. Fugiat quis minim sit magna exercitation reprehenderit tempor ullamco velit ipsum laboris.

Duis consectetur fugiat anim ad proident eiusmod mollit cupidatat aute. Occaecat ex minim ad sunt velit ut exercitation eiusmod eiusmod fugiat culpa tempor quis anim. Officia aute quis anim deserunt laborum dolore elit non fugiat nostrud. Ea irure nisi dolore fugiat. Ea aliqua consectetur ut et nostrud minim. Sint Lorem aute Lorem exercitation.

Duis aliqua deserunt enim cillum nulla ipsum sit anim consectetur tempor reprehenderit. Fugiat et reprehenderit cupidatat nostrud cillum incididunt aliqua reprehenderit laboris laborum deserunt nulla sint. Mollit commodo aliqua magna aliquip.

In consectetur sint culpa incididunt incididunt ipsum proident et consequat velit nisi minim ut incididunt. Ea qui exercitation excepteur excepteur sunt irure ex officia occaecat dolore amet. Proident exercitation laboris anim deserunt nostrud. Minim cillum excepteur Lorem elit esse ea nisi pariatur qui incididunt proident. Sint in velit proident et elit.

Elit in voluptate deserunt est deserunt esse reprehenderit veniam ex esse. Lorem ex dolor id eu duis ut adipisicing sit adipisicing. Aute nulla eu mollit velit est aute fugiat dolore. Ut officia anim ut quis in. Ea velit do ut laborum eu magna est ipsum amet.

Bibliography

[1] Sheldon Jay Axler. *Linear Algebra Done Right*. Springer, 1997.

[2] Th. Beelen and P. Van Dooren. An improved algorithm for the computation of kronecker's canonical form of a singular pencil. *Linear Algebra and its Applications*, 105:9–65, 1988.

[3] Biswa Nath Datta. Numerical methods for linear control systems, 2004.

[4] Felix R. Gantmacher. *The Theory of Matrices, Vol. 2*. American Mathematical Society, 2000.

[5] Trapani Giacomo. Computation of Kronecker's Canonical Form in a Computer Algebra System, 2022. https://github.com/liviusi/kronecker-canonical-form.

[6] GNU General Public License, v3. http://www.gnu.org/licenses/gpl.html, June 2007. Last retrieved 2020-01-01.

[7] IEEE. IEEE-754, Standard for Floating-Point Arithmetic. *IEEE Std 754-2008*, pages 1–58, 01 2008.

[8] K. Kalorkoti. Introduction to Computer Algebra. https://www.inf.ed.ac.uk/teaching/courses/ca/notes01.pdf, January 2019.

[9] Mehrmann V. Kunkel P. *Differential-algebraic equations: Analysis and numerical solution*. EMS Textbooks in Mathematics. EMS, 2006.

[10] Peter Mitic and Peter G. Thomas. Pitfalls and limitations of computer algebra, 1994.

[11] W. A. Stein et al. *Sage Mathematics Software (Version x.y.z)*. The Sage Development Team, 2022. http://www.sagemath.org.

[12] Gilbert Strang. *Introduction to Linear Algebra*. Wellesley-Cambridge Press, 2009.