Proceedings of the 2006 IEEE/RSJ
International Conference on Intelligent Robots and Systems
October 9 - 15, 2006, Beijing, China

Cayley-Hamilton for roboticists

Martijn Visser
Dutch Space BV
m.visser@dutchspace.nl

Stefano Stramigioli
University of Twente
s.stramigioli@utwente.nl

Cock Heemskerk
Dutch Space BV
c.heemskerk@dutchspace.nl

***Abstract*— The *Cayley-Hamilton theorem* is an important theorem of linear algebra which is well known and used in system theory. Unfortunatelly, this powerful result is practically never used in robotics even though it is of extreme relevance. This article is a review of the use of this result for the calculation of general matrix functions which are very common in robotics. It will be shown how any analytic matrix function like exponential, logarithm and more complicated expressions in robotics, can be easily and analytically calculated in an explicit form. Examples are given for the exponential map, inverse of the exponential map, and the derivative of the exponential map. For the first two examples there exist well known expressions in the literature, but the last one is not as easy to compute without the presented methods.**

I. Introduction

The Cayley-Hamilton theorem is a well known theorem in control theory which is used among others to study the Jordan form of a linear system. This theorem states that a matrix A satisfies its own characteristic equation. If the characteristic equation does have roots with multiplicity higher than one, there may also exist a smaller polynomial, called minimal polynomial, with the same roots but with lower multiplicity, which is also satistied by the matrix itself.

The Cayley-Hamilton theorem was discovered by the famous mathematicians Arthur Cayley and William Hamilton. Arthur Cayley (1821–1895, see figure 1) was a British scholar, who helped in founding the British school of pure mathematics. Sir William Hamilton (1805–1865, see figure 2) was an Irish mathematician, as well as a physicist and astronomer. The Cayley-Hamilton theorem allows us to compute functions of matrices like the matrix exponent or matrix logarithm, in a straightforward and uniform way, by making use of the fact that a matrix taken to a certain power, can be expressed as a sum of lower powers of the original matrix.

The Cayley-Hamilton theorem and its associated corollaries have been known for a long time, and are commonly used in the fields of linear algebra and system theory [1]. On the other hand, to the author's knowledge they are not commonly used in the field of robotics, despite several powerful applications described in this article. Robotics makes elaborate use of matrix functions. For example 3 by 3 rotation matrices are used to describe rigid body orientations, and 4 by 4 homogeneous transformations matrices are used to describe rigid body motions.

The techniques described in this article are related to the spectral decomposition of a matrix, see [2], [3]. The spectral decomposition decomposes a matrix along a basis of mutually annihalating idempotent and nilpotent matrices.

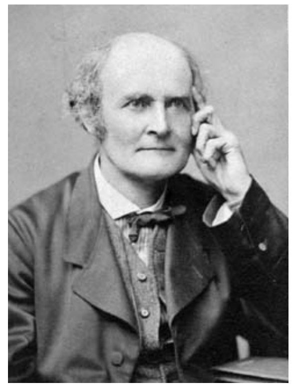

Fig. 1: Arthur Cayley

Fig. 2: William Hamilton

The rest of this article is organized as follows. First in section II the theorem with its associated corollaries are reviewed. In the subsections their use is illustrated with important examples from the field of robotics. Finally in section III the conclusions are drawn. Appendix contains a small Maple library that automatically performs the procedures explained in this article.

II. Matrix functions

We start by stating the Cayley-Hamilton theorem.

Theorem 1 (Cayley-Hamilton): Let A be a square n by n matrix. Its characteristic equation is

$$\det(A-\lambda I)=0 \tag{1}$$

This is a polynomial equation of degree n

$$\lambda^n + a_{n-1}\lambda^{n-1} + \ldots + a_1\lambda + a_0 = 0 \tag{2}$$

The roots λ_i of the characteristic equation are the eigenvalues of A. Then the matrix satisfies its own characteristic equation

$$A^n + a_{n-1}A^{n-1} + \ldots + a_1 A + a_0 I = 0 \tag{3}$$

The proof follows from the definition of eigenvalue $Av=\lambda v$ where v is an eigenvector of A. Eigenvectors are characterized by a direction only, and are independent on their length. If v is an eigenvector then also λv is an eigenvector and therefore $A^2 v = \lambda^2 v$ and so on. This theorem has two important corollaries.

Corollary 1 (Matrix polynomial): Any matrix polynomial in A of degree $\geq n$ can be written as a matrix polynomial of degree $< n$. This is because the higher powers of A can be written as sums of the lower powers with

$$A^n = -a_{n-1}A^{n-1} - \ldots - a_1 A - a_0 I \tag{4}$$

This procedure is illustrated by Maple function *reducePolynomial( )* in the appendix.

Corollary 2 (Matrix inverse): The inverse of a non-singular matrix A can be written as a matrix polynomial of degree $< n$

$$A^{-1} = -\frac{1}{a_0}A^{n-1} - \frac{a_{n-1}}{a_0}A^{n-2} - \ldots - \frac{a_1}{a_0}I \tag{5}$$

The Cayley-Hamilton theorem can be used to extend scalar functions of scalar arguments to matrix functions of matrix arguments. The following theorem states that any scalar function that can be written as a polynomial, can be extended to a matrix function. Note that every analytic function can be written as a polynomial with a Taylor expansion. [1]

Theorem 2 (Matrix functions): Any analytic, scalar function $f(x)$ of a scalar argument x, can be extended to a square matrix function $F(A)$ of a square matrix argument A. With corollary 1 the matrix function $F(A)$ can be written as a polynomial $P(A)$ of degree $< n$

$$F(A) = P(A) = p_{n-1}A^{n-1} + \ldots + p_1 A + p_0 I \tag{6}$$

The coefficients of $P(A)$ can be found by evaluating $f(x)$ at the eigenvalues of A.
Proof: let λ_i be an eigenvalue of A with corresponding eigenvector v_i. Then

$$f(\lambda_i)v_i = F(A)v_i = P(A)v_i = p(\lambda_i)v_i \tag{7}$$

The first equality follows because $f(x)$ can be written as a Taylor expansion, and from the definition of eigenvalue $Av_i = \lambda_i v_i$. This implies that $f(\lambda_i)$ is an eigenvalue of $F(A)$ with corresponding eigenvector v_i. The second equality follows from corollary 1. The third equality follows again from the definition of eigenvalue. The coefficients of $P(A)$ are the same as the coefficients of $p(x)$ and can be found by solving

$$f(\lambda_i) = p(\lambda_i) \quad \text{for} \quad i = 1 \ldots n \tag{8}$$

If all eigenvalues have multiplicity $m = 1$ this yields n independent equations. For eigenvalues with multiplicity $m > 1$ also the first $(m-1)$ derivatives of $f(x)$ must be evaluated

$$\frac{\partial^q f}{\partial x^q}(\lambda_i) = \frac{\partial^q p}{\partial x^q}(\lambda_i) \quad \text{with} \quad q = 1 \ldots (m-1) \tag{9}$$

This yields n equations which can be solved for the n unknown coefficients. This procedure is illustrated by the Maple function *matrixFunction( )* in the appendix. Now it will be proven that these equations are indeed independent.

Equation (8) can be written in matrix form as

$$\begin{pmatrix} f(\lambda_1) \\ f(\lambda_2) \\ \vdots \\ f(\lambda_n) \end{pmatrix} = \begin{pmatrix} 1 & \lambda_1 & \lambda_1^2 & \ldots & \lambda_1^{n-1} \\ 1 & \lambda_2 & \lambda_2^2 & \ldots & \lambda_2^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \lambda_n & \lambda_n^2 & \ldots & \lambda_n^{n-1} \end{pmatrix} \begin{pmatrix} p_0 \\ p_1 \\ \vdots \\ p_{n-1} \end{pmatrix} \tag{10}$$

or $F = \Lambda P$. Here F is a vector of function evaluations, P is a vector of unknown polynomial coefficients, and Λ is a matrix of powers of eigenvalues. Each of the rows correspond to one eigenvalue. If Λ is invertable, then P can be solved with $P = \Lambda^{-1}F$. Now Λ has a special form which is called a *Vandermonde matrix*. It can be proven that if all eigenvalues are different with multiplicity $m = 1$, than all rows are independent, and the matrix is invertable. For a proof see for example [4].

If some eigenvalues are the same with multiplicity $m > 1$, then the corresponding rows would become dependent, and the matrix would become singular. Then equation (9) can be used to replace those dependent rows with

$$\begin{pmatrix} \frac{\partial f}{\partial x}(\lambda_i) \\ \frac{\partial^2 f}{\partial x^2}(\lambda_i) \\ \vdots \\ \frac{\partial^{m-1} f}{\partial x^{m-1}}(\lambda_i) \end{pmatrix} = \begin{pmatrix} 0 & 1 & 2\lambda_i & \ldots & (n-1)\lambda_i^{n-2} \\ 0 & 0 & 1 & \ldots & (n-2)\lambda_i^{n-3} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \ldots & (n-m+1)\lambda_i^{n-m} \end{pmatrix} \begin{pmatrix} p_0 \\ p_1 \\ \vdots \\ p_{n-1} \end{pmatrix}$$

These rows are called *confluent rows*, and are the first $(m-1)$ derivatives of the original *Vandermonde* row. The resulting matrix is called a *confluent Vadermonde matrix*. It can be proven that a confluent Vandermonde matrix is always invertable. For a proof see [4].

Two matrix functions that are very important for robotics are the exponential map or matrix exponential, together with its inverse or matrix logarithm. In Lie theory the matrix exponential maps an element of a Lie algebra onto an element of the corresponding Lie group. The matrix logarithm maps an element of a Lie group back onto an element of the corresponding Lie algebra. Consider for example the Lie algebra of angular velocities $so(3)$ with corresponding Lie group of rotations $SO(3)$, or the Lie algebra of twists $se(3)$ with corresponding Lie group of rigid body motions $SE(3)$. 3 by 3 rotation matrices are elements of $SO(3)$. 4 by 4 homogeneous transformation matrices are elements of $SE(3)$.

A. Eigenvalues of rotations and motions

In this section the eigenvalues of rotations and motions are computed. These eigenvalues are used in the computation of the matrix functions in the following sections. It is easiest to first compute the eigenvalues of elements of the Lie algebras. The eigenvalues of the elements of the Lie groups then follow from the following theorem.

Theorem 3 (Eigenvalues): Let A and B be square matrices with $B = \exp(A)$. Then if λ is an eigenvalue of A, then $\mu = \exp(\lambda)$ is an eigenvalue of B. The proof follows from the

[1] An analytic function is more than a infinitely differentiable function. A typical example of an infinite differentiable function which is not analytic is a function which is constant on a finite interval. In that interval all derivatives vanish.

eigendecomposition $A = V\Lambda V^{-1}$ and the property of the matrix exponential that $\exp(V\Lambda V^{-1}) = V\exp(\Lambda)V^{-1}$.

A rotation can be represented as an element of the Lie group $SO(3)$ with a 3 by 3 orthonormal matrix R, or as an element of the corresponding Lie algebra $so(3)$ with a 3 vector $\omega = (\omega_1, \omega_2, \omega_3)$. The 3 vector ω can be written as a 3 by 3 skew-symmetric matrix Ω as

$$\Omega = \begin{pmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{pmatrix} \tag{11}$$

R is commonly known as a rotation matrix, while ω is sometimes referred to as the equivalent angle and axis representation, see for example [5]: the length of ω is the equivalent angle, and the direction of ω the equivalent axis. In the rest of this article the symbol w will be used for the length of the vector ω. R and Ω are related by the matrix functions $R = \exp(\Omega)$ and $\Omega = \ln(R)$.

The characteristic equation of Ω is

$$\det(\Omega - \lambda I) = -\lambda^3 - \lambda\omega_1^2 - \lambda\omega_2^2 - \lambda\omega_3^2 \tag{12}$$
$$= \lambda(\lambda - iw)(\lambda + iw) = 0 \quad \text{with} \quad w = \sqrt{\omega_1^2 + \omega_2^2 + \omega_3^2}$$

so its eigenvalues are $\lambda_1 = 0$, $\lambda_2 = iw$ and $\lambda_3 = -iw$. The eigenvalues of $R = \exp(\Omega)$ follow from theorem 3. They are $\lambda_1 = \exp(0) = 1$, $\lambda_2 = \exp(iw)$ and $\lambda_3 = \exp(-iw)$.

A motion (rotation and translation), can be represented as an element of the Lie group $SE(3)$ with a 4 by 4 homogeneous transformation matrix

$$H = \begin{pmatrix} R & p \\ 0 & 1 \end{pmatrix} \tag{13}$$

where rotational part R is a 3 by 3 orthonormal rotation matrix and the translational part p is a position 3 vector. Alternatively it can be represented as an element of the corresponding Lie algebra $se(3)$ with a 4 by 4 matrix

$$T = \begin{pmatrix} \Omega & v \\ 0 & 0 \end{pmatrix} \tag{14}$$

where the rotational part Ω is a 3 by 3 anti-symmetric matrix and the translational part v is a 3 vector. T is commonly known as the twist. H and T are related by the matrix functions $H = \exp(T)$ and $T = \ln(H)$.

The characteristic equation of T is

$$\det(T - \lambda I) = -\lambda^4 - \lambda^2 w_1^2 - \lambda^2 w_2^2 - \lambda^2 w_3^2 \tag{15}$$
$$= \lambda^2(\lambda - iw)(\lambda + iw) = 0$$

so the its eigenvalues are $\lambda_1 = \lambda_2 = 0$, $\lambda_3 = iw$ and $\lambda_4 = -iw$. Note that they do not depend on the elements of v. The eigenvalues of $H = \exp(T)$ follow from theorem 3. They are $\lambda_1 = \lambda_2 = \exp(0) = 1$, $\lambda_3 = \exp(iw)$ and $\lambda_4 = \exp(-iw)$.

B. *Exponential map*

In this section the matrix exponentials for rotations and motions are computed, using the eigenvalues from the previous section. First the exponential map for rotations $R = \exp(\Omega)$ is computed. Ω has the eigenvalues $\lambda_1 = 0$, $\lambda_2 = iw$ and $\lambda_3 = -iw$. Using theorem 2

$$\begin{pmatrix} 1 \\ \exp(iw) \\ \exp(-iw) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & iw & -w^2 \\ 1 & -iw & -w^2 \end{pmatrix} \begin{pmatrix} p_0 \\ p_1 \\ p_2 \end{pmatrix} \tag{16}$$

This yields 3 independent equations which can be solved for the 3 coefficients

$$p_0 = 1 \tag{17}$$
$$p_1 = \frac{\sin w}{w}$$
$$p_2 = \frac{1 - \cos w}{w^2}$$

The resulting polynomial is

$$R = \exp(\Omega) = \frac{1 - \cos w}{w^2}\Omega^2 + \frac{\sin w}{w}\Omega + I \tag{18}$$

This result is the same as the well known Rodriguez formula, see again [5]. Because the matrix exponential for rotations over an angle of 2π more or less yields the same rotation matrix, $R = \exp(\Omega)$ is not injective (it is not one-to-one).

Now the exponential map for motions $H = \exp(T)$ is computed. T has the eigenvalues $\lambda_1 = \lambda_2 = 0$, $\lambda_3 = iw$ and $\lambda_4 = -iw$. Evaluating $f(\lambda_i) = p(\lambda_i)$. Because eigenvalue $\lambda_1 = \lambda_2 = 0$ has multiplicity $m = 2$ here also the first derivative $\partial f/\partial x(\lambda_1) = \partial p/\partial x(\lambda_1)$ must be evaluated

$$\begin{pmatrix} 1 \\ 1 \\ \exp(iw) \\ \exp(-iw) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & iw & -w^2 & -iw^3 \\ 1 & -iw & -w^2 & iw^3 \end{pmatrix} \begin{pmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{pmatrix}$$

Since the original matrix is real, and the image matrix is real, the coefficients must be real. Here that fact is used.

$$p_0 = 1 \tag{19}$$
$$p_1 = 1$$
$$p_2 = \frac{1 - \cos w}{w^2}$$
$$p_3 = \frac{w - \sin w}{w^3}$$

The resulting polynomial is

$$H = \exp(T) = \frac{w - \sin w}{w^3}T^3 + \frac{1 - \cos w}{w^2}T^2 + T + I$$

This implementation of the exponential map for motions can be checked against other implementations from the literature, see for example [5].

C. *Inverse of the exponential map*

In this section the inverses of the exponential maps or matrix logarithms for rotations and motions are computed. First the inverse of the exponential map for rotations $\Omega = \ln(R)$ is computed. R has the eigenvalues $\lambda_1 = 1$, $\lambda_2 = \exp(iw)$ and $\lambda_3 = \exp(-iw)$. Evaluating $f(\lambda_i) = p(\lambda_i)$

$$\begin{pmatrix} 0 \\ iw \\ -iw \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & \exp(iw) & \exp(2iw) \\ 1 & \exp(-iw) & \exp(-2iw) \end{pmatrix} \begin{pmatrix} p_0 \\ p_1 \\ p_2 \end{pmatrix}$$

Solving for the coefficients

$$p_0 = \frac{-w(2\cos w + 1)}{2\sin w} \tag{20}$$
$$p_1 = \frac{w \sin w}{\cos w - 1}$$
$$p_2 = \frac{-w}{2\sin w}$$

The resulting polynomial is

$$\Omega = \ln(R) = \frac{-w}{2\sin w} R^2 + \frac{w \sin w}{\cos w - 1} R + \frac{-w(2\cos w + 1)}{2\sin w} I$$

This implementation of the inverse of the exponential map for rotations, can be checked against other implementations from the literature, see for example [5]. Because the matrix logarithm for rotations always yields rotations with an angle $w < \pi$, $\Omega = \ln(R)$ is not surjective (it is not onto). It is singular in $w = 0$ and $w = \pi$. For $w = 0$ this is because the rotation axis is not well defined for a rotation angle of 0. Close to 0, an infinitely small rotation can point the axis into any arbitrary direction. For $w = \pi$ this is because there are two possibilities for the sign of the rotation (clockwise or counterclockwise).

Now the inverse of the exponential map for motions $T = \ln(H)$ is computed. H has eigenvalues $\lambda_1 = \lambda_2 = 1$, $\lambda_3 = \exp(iw)$ and $\lambda 4 = \exp(-iw)$. Evaluating $f(\lambda_i) = p(\lambda_i)$. Because eigenvalue $\lambda_1 = \lambda_2 = 1$ has multiplicity $m = 2$ here also the first derivative must be evaluated

$$\begin{pmatrix} 0 \\ 1 \\ iw \\ -iw \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 3 \\ 1 & \exp(iw) & \exp(2iw) & \exp(3iw) \\ 1 & \exp(-iw) & \exp(-2iw) & \exp(-3iw) \end{pmatrix} \begin{pmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{pmatrix}$$

Solving for the coefficients. Here the fact is used that the coefficients must be real.

$$p_0 = \frac{2w\cos^2 w - \sin w - w}{2\sin w(1 - \cos w)} \tag{21}$$
$$p_1 = \frac{-4w\cos^2 w + 2\sin w \cos w - w\cos w + 2w + \sin w}{2\sin w(1 - \cos w)}$$
$$p_2 = \frac{2w\cos^2 w - 2\sin w \cos w + 2w\cos w - \sin w - w}{2\sin w(1 - \cos w)}$$
$$p_3 = \frac{-w\cos w + \sin w}{2\sin w(1 - \cos w)}$$

This formula for the computation of the matrix logarithm for motions can be checked against other implementations from the literature, see for example Appendix A of [5].

D. Derivative of the exponential map

Let $S = dH/dtH^{-1}$ and $T = \ln(H)$. Note that both S and T are both twists, but T has the dimension of meters and radians and is a "position", while S has the dimension of meters and radians per second and is a "velocity". The relation between dT/dt and S is given by the derivative of the exponential map. It can be computed by taking the Taylor expansion of $\ln(H)$, and is given in [6]. This relation is very useful in the modelling and simulation of spatial springs in exponential coordinates, see [7]. Another application is in describing the dynamics of a free floating robot with the Boltzmann-Hamel equations expressed in exponential coordinates.

$$\frac{dT}{dt} = \frac{d}{dt}\ln(H) = \frac{\mathrm{ad}(T)}{\exp(\mathrm{ad}(T)) - I} S = J(T)S \tag{22}$$

Here $\mathrm{ad}(T)$ is the adjoint representation of the Lie algebra of T, and $\exp(\mathrm{ad}(T)) = \mathrm{Ad}(H)$ is the Adjoint representation of the Lie group of H. This is a matrix division and can be written in this way because the "numerator" and the "denominator" commute, so left and right division are the same. The only problem is that the denominator is singular: $\mathrm{ad}(T)$ has an eigenvalue of 0 with multiplicity 2. Therefore $\mathrm{Ad}(H) = \exp(\mathrm{ad}(T))$ has an eigenvalue of 1 with multiplicity 2. Therefore $(\mathrm{Ad}(H) - I))$ has an eigenvalue 0 with multiplicity 2. Nevertheless the limit of J exists, and is continuous in $\mathrm{ad}(T)$, so it can be computed with the Cayley-Hamilton theorem. This is a good example of a matrix function that can be computed with Cayley-Hamilton, that could otherwise not be easily computed.

For rotations

$$\begin{pmatrix} 1 \\ \frac{iw}{\exp(iw) - 1} \\ \frac{-iw}{\exp(-iw) - 1} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & iw & -w^2 \\ 1 & -iw & -w^2 \end{pmatrix} \begin{pmatrix} p_0 \\ p_1 \\ p_2 \end{pmatrix}$$

Solving for the coefficients. Here the fact is used that the coefficients must be real.

$$p_0 = 1 \tag{23}$$
$$p_1 = \frac{-1}{2}$$
$$p_2 = \frac{-w\cos w + 2\sin w - w}{2w^2 \sin w}$$

For motions

$$\begin{pmatrix} 1 \\ \frac{-1}{2} \\ f_3 \\ f_4 \\ f_5 \\ f_6 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & iw & -w^2 & -iw^3 & w^4 & iw^5 \\ 0 & 1 & 2iw & -3w^2 & -4iw^3 & 5w^4 \\ 1 & -iw & -w^2 & iw^3 & w^4 & -iw^5 \\ 0 & 1 & -2iw & -3w^2 & 4iw^3 & 5w^4 \end{pmatrix} \begin{pmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \end{pmatrix}$$

with

$$f_3 = \frac{iw}{\exp(iw) - 1} \tag{24}$$
$$f_4 = \frac{-iw}{\exp(-iw) - 1}$$
$$f_5 = \frac{1}{\exp(iw) - 1} - \frac{w\exp(iw)}{(\exp(iw) - 1)^2}$$
$$f_6 = \frac{1}{\exp(-iw) - 1} - \frac{w\exp(-iw)}{(\exp(-iw) - 1)^2}$$

Solving for the coefficients. Here the fact is used that the coefficients must be real.

$$
\begin{aligned}
p_0 &= 1 \\
p_1 &= \frac{-1}{2} \\
p_2 &= \frac{3w\sin w + 8\cos w - 8 + w^2}{4(\cos w - 1)w^2} \\
p_3 &= 0 \\
p_4 &= \frac{4\cos w + w\sin w - 4 + w^2}{4w^4(\cos w - 1)} \\
p_5 &= 0
\end{aligned}
\tag{25}
$$

III. Conclusions

In this article the use of the Cayley-Hamilton theorem for roboticists has been explained, and illustrated with some examples. The Cayley-Hamilton theorem allows to any analytic scalar functions to a matrix function, and to compute these matrix functions in a straightforward way. Roboticists make extensive use of the exponential map or matrix exponent, and the inverse of the exponential map or matrix logarithm. Examples have been given of how to implement these matrix functions with the Cayley-Hamilton theorem. The appendix contains a small Maple library that does this automatically.

For most of these functions there exist other implementations from the literature. These other implementations of course yield the same results, but were computed in a less straightforward and uniform way. For other functions there are no known other implementations. A good example is the Jacobian of the exponential map. This function is useful for the modelling and simulation of spatial springs, which will be demonstrated in a separate article [7]. This is an expression that is singular but whose limit nevertheless exists, and can be computed nicely and efficiently with the Cayley-Hamilton theorem.

In this article examples were given for matrices from the group of rotations $SO(3)$ and motions $SO(3)$, but the procedures work for all matrices. The procedures also work for complex matrices from the group of quaternions $SU(2)$, and even dual matrices from the groups of dual rotations and dual quaternions.

Appendix

This section presents a small library of Maple functions that automatically perform the procedures explained in this article. The arguments of the functions are lists, list of coefficients of polynomials, and lists of eigenvalues of matrices.

pout = reducePolynomial(pin, eigs) This function takes a polynomial *pin* in a matrix with eigenvalues *eigs*, and returns the minimal polynomial *pout*.

pout = matrixFunction(fun, eigs) This function takes a matrix function *fun* of a matrix with eigenvalues *eigs*, and returns the coefficients of the corresponding polynomial *pout*.

pout = changeVariable(pin, fun, eigs) This function takes a polynomial *pin* in a matrix variable with eigenvalues *eigs*, then performs a change of variable *fun*, and returns the corresponding polynomial *pout* in the other variable.

```
reducePolynomial := proc( pin, eigs )
  local n1, p1, n2, t1, t2, pout;
  n1 := nops( pin );
  p1 := sum( pin[ i1 ] * x ^ ( i1 - 1 ), i1 = 1..n1 );
  n2 := nops( eigs );
  t1 := product( x - eigs[ i1 ], i1 = 1..n2 );
  t1 := simplify( t1 );
  n1 := degree( p1, x );
  while ( n1 >= n2 ) do
    t2 := coeff( p1, x, n1 );
    # substitute characteristic polynomial
    p1 := p1 - t2 * t1 * ( x ^ ( n1 - n2 ) );
    p1 := simplify( p1 );
    p1 := collect( p1, x );
    n1 := degree( p1, x );
  end;
  pout := PolynomialTools:-CoefficientList( p1, x );
  simplify( pout );
end:

matrixFunction := proc( fun, eigs )
  local n1, pout, s1, e2, n2, t1, f2, t2, s2, i2, i3, s3, l3;
  n1 := nops( eigs );
  pout := [ seq( p||i1, i1 = 0..( n1 - 1 ) ) ];
  s1 := convert( pout, set );
  # list of eigenvalues and their multiplicities
  e2 := convert( eigs, multiset );
  n2 := nops( e2 );
  t1 := sum( 'p||i1 * x ^ i1', i1 = 0..( n1 - 1 ) );
  f2 := unapply( t1, x );
  s2 := { };
  for i2 to n2 do
    # evaluate function and derivatives
    for i3 from 0 to ( e2[ i2, 2 ] - 1 ) do
      try
        t2 := ( D@@i3 )( fun )( e2[ i2, 1 ] );
      catch "numeric exception: division by zero":
        t2 := limit( ( D@@i3 )( fun )( x ), x = e2[ i2,
          1 ] );
      end;
      s2 := { op( s2 ), t2 = ( D@@i3 )( f2 )( e2[ i2, 1 ]
        ) };
    end;
  end;
  s3 := solve( s2, s1 );
  s3 := simplify( s3 );
  pout := subs( s3, pout );
  pout := map( Re, pout );
  simplify( pout );
end:

changeVariable := proc( pin, fun, eigs )
  local n1, p1, l2, n2, p2, p3, pout;
```

```
    n1 := nops( pin );
    p1 := sum( pin[ i1 ] * x ^ ( i1 − 1 ), i1 = 1..n1 );
    l2 := matrixFunction( fun, eigs );
    n2 := nops( l2 );
    p2 := sum( l2[ i1 ] * y ^ ( i1 − 1 ), i1 = 1..n2 );
    p3 := subs( x = p2, p1 );
    p3 := simplify( p3 );
    p3 := collect( p3, y );
    pout := PolynomialTools:−CoefficientList( p3, y );
    reducePolynomial( pout, eigs );
end:
```

References

[1] T. Kailath, *Linear Systems*. New Jersey, USA: Prentice Hall, 1980.

[2] J. M. Selig, *Geometric Fundamentals of Robotics*, 2nd ed. New York, USA: Springer, 2005.

[3] G. Sobczyk, "The generalized spectral decomposition of a linear operator," *College Mathematics Journal*, vol. 28, no. 1, pp. 27–38, Jan. 1997.

[4] G. H. Golub and C. F. van Loan, *Matrix Computations*, 3rd ed. Baltimore, USA: John Hopkins University Press, 1996.

[5] R. M. Murray, Z. Li, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. New York, USA: CRC Press, 1994.

[6] W. Rossman, *Lie Groups: An Introduction through Linear Groups*. Oxford, UK: Oxford University Press, 2002.

[7] M. Visser, S. Stramigioli, and C. Heemskerk, "Spatial springs in generalized coordinates," July 2005, submitted to ....