Feedback control over constrained robotic systems through the Udvadia-Kalaba approach

May 4, 2024

Contents

Chapter 1

Introduction

The advancement of microelectronics and sensor technologies has catalyzed a widespread integration of robotic systems into various domains. Manipulators, delivery robots, and flying drones have become ubiquitous, presenting developers and researchers with novel challenges in terms of robustness, adaptiveness, and synchronization. Among these challenges, synchronization stands out as a critical issue, especially given the proliferation and increasing complexity of such systems. In real-world applications, synchronization of robotic systems is crucial across various industries. For instance, in manufacturing assembly lines, synchronized robots ensure smooth production flow and maximize throughput. Similarly, in warehouse logistics, coordinated robotic systems optimize order fulfillment times and enhance overall productivity. Furthermore, collaborative robotics scenarios in construction projects benefit from synchronized robotic systems for tasks like concrete pouring and steel beam placement, improving efficiency and minimizing delays. A common challenge in this realm involves effectively controlling robots constrained by a shared object. This paper proposes a straightforward methodology to address such problems through the Udvadia-Kalaba[1] approach.

This proposed methodology gains particular significance within the context of modern physics simulation, derivation, and optimization libraries. These tools offer substantial computational speed, enabling efficient problem-solving. In this article, we leverage MuJoCo[2], Pinocchio[3], and ProxSuite[4] for simulation, robotic dynamics computation, and optimization, respectively. Pinocchio, in particular, emerges as a key tool for computing the dynamics of robotic systems. However, its limitation to open-loop physics models poses challenges for controlling and synchronizing multiple robots.

Previous solutions have often involved constructing models with pre-existing constraints or employing the KKT (Karush-Kuhn-Tucker) approach. However, these methods suffer from computational complexity and issues with constraint prioritization. The proposed methodology combines the strengths of both approaches, integrating physical grounding from the former and simplicity of application from the latter. Notably, it allows for manual fine-tuning of constraint priorities and boasts superior computational speed through auto code generation.

The implementation of the proposed methodology demonstrates significant advancements in the control and synchronization of robotic systems constrained by shared objects. Through the integration of robust physics simulation, precise robotic dynamics computation, and efficient optimization techniques, the method achieves remarkable outcomes across various simulation experiments. By leveraging advanced simulation, computation, and optimization techniques, the method has improved levels of efficiency, accuracy, and adaptability in robotic operations, paving the way for further advancements in automation and robotics technology.

In subsequent chapters, we delve deeper into various aspects of the proposed method. Chapter 2 provides an exhaustive review of recent literature, highlighting the existing landscape of solutions and their limitations. Chapter 3 elucidates the methodology underlying the proposed approach, offering insights into its theoretical underpinnings. Implementation details and code snippets are presented in Chapter 4, demonstrating the practical application of the method. Chapter 5 undertakes a comparative analysis, pitting technique against established methods to gauge its efficacy. Finally, Chapter 6 summarizes findings, discusses implications, and outlines avenues for future research.

Chapter 2

Literature Review

The control of the interacting physical systems is a challenging task. Three problems should be solved to achieve high efficiency and precision in the aforementioned problem. They are the following: the right mathematical model selection, the unified methodology for defining the interaction, and the well-defined virtual constraint. The first point is necessary to cover a wide range of physical systems. The second one is crucial to work with different mechanical connections. Finally, the last one is needed for stability and robustness analysis.

This literature review covers all these subproblems and explores them from the point of view of the work of Firdaus Udwadia and Robert Kalaba [1]. The Section 2.1 considers a mathematical model that utilized in the recent studies. The 2.2 part reviews a rigorous methodology to define how physical systems can affect each other. The Section 2.3 explores the previous research in virtual constraint defining and analysis on different manifolds. Finally, the last subsection 2.4 contains concluding words about reviewed techniques that are utilized in this study.

The selection of the papers for this review aims to provide solid grounds for the choice of the methods in this study. Some of the papers were chosen as a reference for theoretical background. Other papers were reviewed to study the existing techniques.

2.1 The mathematical model

This section contains a review of existing mathematical models for physical systems. Moreover, this piece has a comparison of them in terms of numerical integration convenience and simplicity of defining the initial conditions. It is necessary to clarify that from now this study considers only systems of rigid bodies with only inner stiffness. The detailed explanation of such a choice will be conducted later in Chapter 3.

Firstly, let's consider the most popular model. The articles [1], [5], and [6] rely on it. Usually, it is called the canonical manipulator equation. The model can be formulated as

$$M(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + g(\mathbf{q}) = \boldsymbol{\tau} \tag{2.1}$$

where $M(\mathbf{q})$ is known as inertia matrix, $C(\mathbf{q}, \dot{\mathbf{q}})$ is Centrifugal-Coriolis matrix, $g(\mathbf{q})$ is a gradient of potential forces, and $\mathbf{q}$, $\boldsymbol{\tau}$ are generalized coordinates and torques respectively. This equation will be explored in detail in the next Chapter 3.

Conversely, Udwadia [7] utilized a Hamiltonian view of the problem. This equation also manipulates with generalized coordinates but additionally introduces generalized momentum and Hamiltonian. The choice of such variables makes this approach more convenient in terms of numerical integration.

The aforementioned equations are proven to be equivalent. Thus, both models can be utilized to achieve the main goal. However, the crucial difference lies in the other plane.

In the vast majority of cases, both differential equations can be solved analytically. Therefore, it is necessary to use the numerical methods. The equation (2.1) has a second-time derivative. It forces to construction of proper state variables for numerical integration. On the opposite, the Hamiltonian equations have only the first derivative. Hence, the usage of it is simpler. However, the definition of initial conditions is harder. Nevertheless, the mentioned problem is already solved. So, choice of model fully lies on the specific discussed task.

Most reviewed papers rely on the canonical model (2.1). Therefore, this research will utilize this equation too.

2.2 The methodology to defining coupling

The second crucial point in this paper is the definition of coupling between systems. This piece considers how it can be achieved, and which methodology is better in the context of the discussed question. The important remark of this section is that only interaction via rigid bodies will be considered below. The schematic representation of such coupling is shown in the Figure 2.1

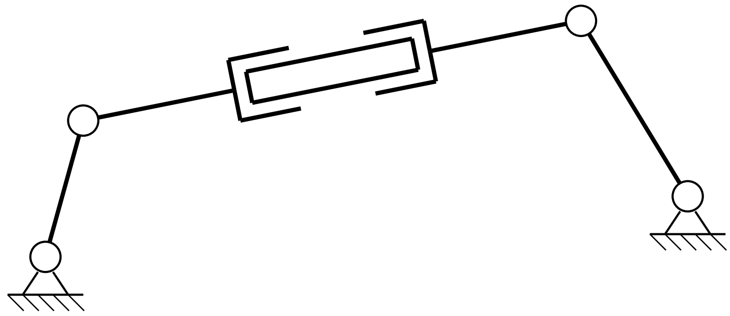

Fig. 2.1. Rigid body coupling

The first straightforward solution is to initially impose an interaction inside to the mathematical model. In the discussed case it can be achieved by writing the $M(\mathbf{q})$, $C(\mathbf{q},\dot{\mathbf{q}})$, and $g(\mathbf{q})$ in equation (2.1) as formulation of closed-loop system. It can be achieved, if it is possible to formulate and analytically solve the the following equation,

$$\varphi(\mathbf{q},t) = 0 \tag{2.2}$$

From it, the minimal set of generalized coordinates can be constructed which implies rewriting the aforementioned parts of the canonical manipulator equation. Thus, the advantage of this method is unnecessary of stabilization, because it is guaranteed by a model itself. Furthermore, this formulation can be used with a great range of control techniques. However, [8] demonstrates a lower computation speed in comparison to open-loop algorithms [3]. Moreover, proposed closed-loop version requires a predefined description of a whole system, and works well only with systems that have a small number of DoFs. Thus, it cannot be used in real time in a dynamic environment.

The second discussed approach is analyzed in [9] and [10]. These articles propose to use a force cone to define a contact between a physical system and a solid surface. The actuation force can be formulated as

$$\boldsymbol{\lambda} = \sum_{i=1}^{N_d} \beta_i(\mathbf{n} + \mu\,\mathbf{d}_i),\ \beta_i \geqslant 0 \tag{2.3}$$

where $\mathbf{n}$ is a normal force, $\mathbf{d}_i$ is a tangent to contact vector, μ is the Coulomb friction coefficient, and N_d is a amount of used tangent vectors. This contact force later can be translated to joint space via respective Jacobian - $J^T(\mathbf{q})$.

Using this approach it is possible to emulate an interaction between physical systems via a rigid body. It can be achieved by defining the motion of the connection body through force acting on it. However, this method cannot guarantee stability. Moreover, constructing a feedback loop in this case is not a trivial task.

Finally, the third approach is defining the right constraint. In this study the action of the rigid body on connected systems can be described by holonomic constraint. It is described by the equation (2.2). Further, this equation can be used in the KKT (Karush-Kuhn-Tucker) technique to define a system with imposed constraints. Moreover, using the equation (2.2) it is straightforward to construct a stabilization mechanism. For instance, Baumgarte's approach can be used with the KKT method to achieve this goal. Nevertheless, this study does not rely on the aforementioned technique (KKT), instead, it utilized the Udwadia-Kalaba approach, which rewrites generalized accelerations obtained from the equation (2.2) as affine transform. Details are shown in the Chapter 3.

2.3 Virtual constraint

The last crucial subproblem is the rigorous defining of virtual constraints to ensure stabilization. Especially, it is important in the context of the chosen type of interaction. Therefore, this section reviews how to calculate the discrepancy between the current system state and the desired one in the form of an equation (2.2) and how to guarantee its' convergence in the context of the form proposed by the Udwadia-Kalaba approach.

In the initial [1] study Ferdaus Udwadia and Robert Kalaba states that the straightforward differentiation of the holonomic constraint (2.2) produces instability during numerical integration (constraint drift). Thus the authors recommend to use Baumgarte's stabilization [11] via rewriting the linear second order differential equation with φ to proposed matrix form.

Nevertheless, below in this study will show that it is hard to use this technique in the case of attitude tracking. The problem mostly is caused by the fact that the linear differential equations cannot capture the topology of the SE(3). Therefore, I was forced to use non-linear ones by utilizing the right pose difference formulation. The studies [12]–[16] describe methods for achieving it. The majority [13]–[16] of source explores the problem through utilizing rotations matrices. On the other hand [12] uses quaternions for achieving the goal. Let's compare these solutions.

The [12] states the advantage of quaternions on the rotations matrix. This research shows that the proposed method achieves the *exponential asymptotic stability* against *almost global stability* of techniques from [13], [14]. Furthermore, the quaternion-based method in the discussed study presents the superiority of the analogous ones. The authors prove that their technique avoids the unwinding phenomenon, i.e. error converges to zero by shortest path in $\mathbb{S}^3$.

The solution via utilizing the rotations matrices is presented in [13]–[16]. However, in the aforementioned quaternion-based the approach shows that basing on such structure methods has disadvantages. The naive error computation via matrices is a calculation of dot product between base vectors of current and desired orientation. As mentioned above this approach can achieve only *almost global stability*. Nevertheless, studies [15], [16] relies on deep topology analysis of $SE(3)$ group. These research use a Lie theory to construct a right discrepancy between attitudes. In Chapter 3 the *exponential convergence* of the method based on the articles above is proven.

To conclude, in this study the rotation matrices are chosen to define an attitude error. As starting point of investigations articles [15], [16] are chosen. The main reason for such a choice is the convenience of work in the context of framework [3] that is utilized for numerical experiments.

2.4 Conclusion

In this chapter, the solutions to the necessary aforementioned subproblems were reviewed. The canonical manipulator equation (**??**) was chosen as the base for investigations because it is most widely spread among other studies. To define a coupling between systems I decided to utilize an Udwadia-Kalaba approach and proposed by it form of coupling. Finally, for building a stable virtual constraint the discrepancy based on Lie theory was taken.

Chapter 3

Methodology

This chapter will present a detailed explanation of the principles underlying the proposed methodology. The section 3.1 offers a concise overview of rigid-body systems and constraints that can be imposed on them. The next section 3.2 introduces the Udwadia-Kalaba approach and physical principles that stays behind it.

3.1 Rigid-body systems and constraints

The most common class of physical systems that can be observed in the real life cases is the rigid-body one. Some examples of such systems are demonstrated in Figure 3.1. The key feature of rigid-body systems is non-deformable parts that forms it. With these limitations only an inner stiffness can be described.

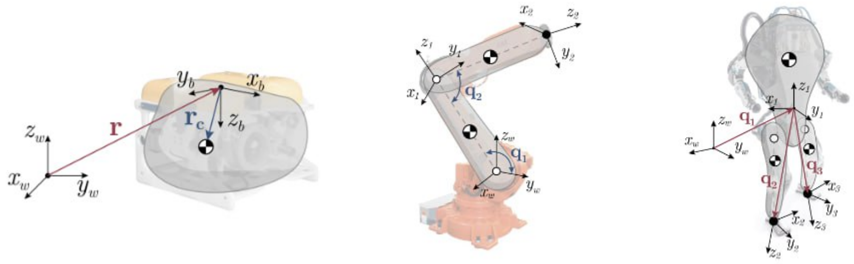

Fig. 3.1. Examples of rigid-body systems

Nevertheless, the mathematical tools to work with rigid-body systems was introduced in XVIII-th century by Newton and Euler works. Using this approach it is easy to formulate a set f deferential equations that describes the system behavior. However, in this study the another (more convenient) technique is utilized. It is called the least action principle. The formulation is the following equation

$$\min_{\mathbf{q}} \quad \int_{t_1}^{t_2} L(\mathbf{q}(t), \mathbf{v}(t), t) dt \tag{3.1}$$

where L is Lagrangian of the system, and $\mathbf{q}$, $\mathbf{v}$ are functions of the generalized coordinates and velocities respectively. The solution of the variational problem (3.1) is the Euler-Lagrange differential equations. The solution through utilizing the Lagrangian can be easily generalized to all rigid-body systems. This generalization is aforementioned the canonical manipulator equation (2.1).

In this chapter the convenient representation of this equation is used. The main idea of rewriting is combination of the inertial forces with non-inertial ones.

$$M\dot{\mathbf{v}} = \mathbf{Q} \tag{3.2}$$

In the above equation $\mathbf{Q} = -C(\mathbf{q}, \mathbf{v})\dot{\mathbf{q}} - g(\mathbf{q}) + \boldsymbol{\tau}$, $M \succcurlyeq 0$ is inertia matrix, C is Centrifugal-Coriolis matrix, g is gradient of conservative forces. The dependency from coordinates and velocities is amended for convenience. In the most generalized case $\dim \mathbf{q} = n_q \neq n_v = \dim \mathbf{v}$.

The aforementioned description is applicable for open and closed loop systems. The closed one is described in Figure 2.1. However, as mentioned above utilizing only equation 3.2 to handle such system is not efficient. Thus, the approach via constraints described by equation (2.2) is more preferable. It can be inserted in (3.1), which leads to the following variational problem,

$$\min_{\mathbf{q}} \quad \int_{t_1}^{t_2} [L(\mathbf{q}(t), \mathbf{v}(t), t) - \boldsymbol{\lambda}^T \varphi(\mathbf{q}(t), t)] dt \tag{3.3}$$

The solution the above equation can be expressed in terms of the KKT matrix in the following way,

$$\begin{bmatrix} M & J^T \\ J & 0 \end{bmatrix} \begin{bmatrix} \dot{\mathbf{v}} \\ -\boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{Q} \\ -\dot{J}\mathbf{v} \end{bmatrix}, \; J = \frac{\partial \varphi}{\partial \mathbf{q}} \tag{3.4}$$

The downside of the equation (3.4) is that it does not support non-holonomic constraints generally. This type of constraints can be defined as

$$\varphi(\mathbf{q}, \mathbf{v}, t) = 0 \tag{3.5}$$

Therefore, the dependency from generalized velocity makes the KKT approach not applicable. The handle of such constraints is crucial for defining a control. Hence, it is necessary to use another technique.

3.2 The Udwadia-Kalaba approach

The main differences of the Udwadia-Kalaba approach from the method mentioned above are the another view on constraints, and utilizing different physical principle.

Let's start from reviewing the constraints form in the discussed technique. It can be derived from holonomic and non-holonomic type by differentiating over time, and transforming to an affine form. The general equation is the following

$$A(\mathbf{q}, \mathbf{v}, t)\dot{\mathbf{v}} = \mathbf{b}(\mathbf{q}, \mathbf{v}, t) \tag{3.6}$$

The above equation can be used in the Gauss least constraint principle. For further convenience the arguments of matrix functions is amended. It leads to the following optimization problem:

$$\begin{aligned} \min_{\dot{\mathbf{v}}} \quad & [\dot{\mathbf{v}} - \mathbf{a}]^T M [\dot{\mathbf{v}} - \mathbf{a}] \\ \text{s.t.} \quad & A\dot{\mathbf{v}} = \mathbf{b} \\ & \mathbf{a} = M^{-1}\mathbf{Q} \end{aligned} \tag{3.7}$$

This optimization can be solved analytically as Ferdaus Udwadia and Robert Kalaba demonstrated [1]. This solution is shown below.

$$\dot{\mathbf{v}} = \mathbf{a} + M^{-1/2}(AM^{-1/2})^{+}(\mathbf{b} - AM^{-1}\mathbf{Q}) \tag{3.8}$$

where $[\cdot]^{+}$ is the Moore-Penrose inverse, and $M^{\pm 1/2} = W\Lambda^{\pm 1/2}W^T$, W is the orthogonal matrix of eigen vectors.

In case of positive semi-definiteness of M the equation (3.8) is not applicable because M^{-1}, $M^{-1/2}$ cannot exist. The Firdaus Udwadia and Aaron Schutte demonstrates [17] a methodology to bypass this problem by replacing M, and $\mathbf{a}$ by the following quantities respectively

$$\begin{aligned} M_A &= M + A^+A \succ 0 \\ \mathbf{a}_A &= M_A^{-1}\mathbf{Q} \end{aligned} \tag{3.9}$$

Utilizing the mentioned quantities it is possible to define constraints force over the wide range of rigid-body systems. These forces can be expressed in the following manner

$$\mathbf{Q}_C = M^{1/2}(AM^{-1/2})^+(\mathbf{b} - AM^{-1}\mathbf{Q}) \tag{3.10}$$

Hence, the Udwadia-Kalaba approach is applicable for emulation a of rigid body constraint in computation easily. Only the problem with rigorous defining A and $\mathbf{b}$ remains.

3.3 Defying constraints over multiple systems

This section introduces a common holonomic constraint, which serves as a foundation for further investigations. This constraint delineates a rigid body's behavior, applicable to multiple manipulators.

Let $M_1, M_2, \ldots, M_p$ are p iinertia matrices for p independent systems. The i-th system has n_q^i generalized coordinates, n_v^i generalized velocity components and Q_i bias force. Thus, the common unconstrained dynamics is

$$\begin{bmatrix} M_1 & 0 & \dots & 0 \\ 0 & M_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & M_p \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}}_1 \\ \ddot{\mathbf{q}}_2 \\ \vdots \\ \ddot{\mathbf{q}}_p \end{bmatrix} = \begin{bmatrix} Q_1 \\ Q_2 \\ \vdots \\ Q_p \end{bmatrix} \tag{3.11}$$

The equation 3.11 can be rewriten in the manner of **??**. In such compact form it is convinient for futher analysis. Hense, let

$$M_s = \begin{bmatrix} M_1 & 0 & \dots & 0 \\ 0 & M_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & M_p \end{bmatrix} \tag{3.12}$$

$$q_s = \begin{bmatrix} \mathbf{q}_1 & \mathbf{q}_2 & \dots & \mathbf{q}_p \end{bmatrix}^T \tag{3.13}$$

$$Q_s = \begin{bmatrix} Q_1 & Q_2 & \dots & Q_p \end{bmatrix}^T \tag{3.14}$$

It implies to the following dynamics equation that describes motion of p independent systems

$$M_s \ddot{\mathbf{q}}_s = \mathbf{Q}_s \tag{3.15}$$

Let R_{ij} and $\mathbf{p}_{ij}$ are rotation and position of j-th frame attached to some link of the i-th system respectevly. The rigid body connection between two frames now can be defined as

$$\begin{cases} R_{ij}R_d = R_{\alpha\beta} & (3.16) \\ (\mathbf{p}_{ij} - \mathbf{p}_{\alpha\beta})^T(\mathbf{p}_{ij} - \mathbf{p}_{\alpha\beta}) = l & (3.17) \end{cases}$$

where R_d is a fixed rotation matrix and l a distence between connection points inside a rigid body.

Substituting ${}^{\alpha\beta}_{ij}\mathbf{d} = \mathbf{p}_{ij} - \mathbf{p}_{\alpha\beta}$ and differencing respect to time twice the equation 3.17 transforms to

$${}^{\alpha\beta}_{ij}\mathbf{d}^T\,{}^{\alpha\beta}_{ij}\mathbf{d} - l = 0 \Rightarrow {}^{\alpha\beta}_{ij}\mathbf{d}^T\,{}^{\alpha\beta}_{ij}\dot{\mathbf{d}} = 0 \Rightarrow {}^{\alpha\beta}_{ij}\mathbf{d}^T\,{}^{\alpha\beta}_{ij}\ddot{\mathbf{d}} + {}^{\alpha\beta}_{ij}\dot{\mathbf{d}}^T\,{}^{\alpha\beta}_{ij}\dot{\mathbf{d}} = 0 \tag{3.18}$$

The equation 3.18 can be expresed via generalized coordinates

$${}^{\alpha\beta}_{ij}\mathbf{d}^T\,{}^{\alpha\beta}_{ij}\ddot{\mathbf{d}} + {}^{\alpha\beta}_{ij}\dot{\mathbf{d}}^T\,{}^{\alpha\beta}_{ij}\dot{\mathbf{d}} = 0 \Rightarrow {}^{\alpha\beta}_{ij}\mathbf{d}^T({}^{v}_{ij}J\ddot{\mathbf{q}}_s + {}^{v}_{ij}\dot{J}\dot{\mathbf{q}}_s) + \dot{\mathbf{q}}_s^T\,{}^{v}_{ij}J^T\,{}^{v}_{ij}J\dot{\mathbf{q}}_s = 0 \tag{3.19}$$

where ${}^{v}_{ij}J \equiv {}^{v}_{ij}J(\mathbf{q}_s)$ is the j-th frame velocity Jacobian of the i-th system. Now, it is possible to convert 3.17 constraint to **??** form

$$A_p(\mathbf{q}_s, \dot{\mathbf{q}}_s) = {}^{\alpha\beta}_{ij}\mathbf{d}^T\,{}^{v}_{ij}J \tag{3.20}$$

$$b_p(\mathbf{q}_s, \dot{\mathbf{q}}_s) = -\,{}^{\alpha\beta}_{ij}\mathbf{d}^T\,{}^{v}_{ij}\dot{J}\dot{\mathbf{q}}_s - \dot{\mathbf{q}}_s^T\,{}^{v}_{ij}J^T\,{}^{v}_{ij}J\dot{\mathbf{q}}_s \tag{3.21}$$

Now, let suppose that $\gamma_1, \gamma_2, \ldots, \gamma_w$ system connected by rigid body. Here γ_i is an index of the system, and $w \leq p$.

Chapter 4

Implementation

Chapter 5

Evaluation and Discussion

Chapter 6

Conclusion

Appendix A

Brief Lie theory

The Lie group, a mathematical concept dating back to the 19th century, was first proposed by Sophus Lie, laying the foundation for continuous transformation groups. Although it was initially abstract, over time its influence has spread to various scientific and technological fields. Before proceeding further, it is necessary to consider the basics of Lie theory. The mathematical tools discussed in this section are crucial for the research discussed here.

Let $\mathcal{G}$ is smooth manifold that satisfies the group axioms. For any $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{Z}$ from $\mathcal{G}$ the following statements are true:

$$
\begin{aligned}
&\mathcal{X} \circ \mathcal{Y} \in \mathcal{G} && \text{(I)} \\
&\exists\, \mathcal{E} :\; \mathcal{E} \circ \mathcal{X} = \mathcal{X} \circ \mathcal{E} = \mathcal{X} && \text{(II)} \\
&\exists\, \mathcal{X}^{-1} :\; \mathcal{X}^{-1} \circ \mathcal{X} = \mathcal{X} \circ \mathcal{X}^{-1} = \mathcal{E} && \text{(III)} \\
&(\mathcal{X} \circ \mathcal{Y}) \circ \mathcal{Z} = \mathcal{X} \circ (\mathcal{Y} \circ \mathcal{Z}) && \text{(IV)}
\end{aligned}
\tag{A.1}
$$

For such group $T_{\mathcal{E}}\mathcal{G}$ is the Lia Algebra defined at $\mathcal{E}$ element. The geometric interpretation of algebra is tangent plane that touchs the smooth manifold (group). $\mathfrak{m} \equiv T_{\mathcal{E}}\mathcal{G}$ is always a vector space.

The next crucial defenition is a group action

$$f : \mathcal{G} \times \mathcal{V} \to \mathcal{V} \tag{A.2}$$

where $\mathcal{V}$ is some set. The operation defined above should satisfy the axioms ($v \in \mathcal{V}$; $\mathcal{X}, \mathcal{Y} \in \mathcal{G}$),

$$\begin{aligned} &\mathcal{E} \cdot v = v && \text{(I)} \\ &(\mathcal{X} \circ \mathcal{Y}) \cdot v = \mathcal{X} \cdot (\mathcal{Y} \cdot v) && \text{(II)} \end{aligned} \tag{A.3}$$

For instance, if the SO(n) rotations is considered as Lie group, than the transformation $R \cdot x \equiv Rx$ ($x \in \mathbb{R}^n$) is a group action.

The exponential map $\exp : \mathfrak{m} \to \mathcal{G}$ converts elements of Lie algebra to corresponding group. The log map do inverse operation. However, it is neccesary to remember that this map applicable only for "identity" algebra. However, there is a linear transformation between $T_{\mathcal{X}}\mathcal{G}$ and $T_{\mathcal{E}}\mathcal{G}$. It is called adjoin.

It is known that $\mathfrak{m}$ is isomorphic to the vector space $\mathbb{R}^m$. One can write it as $\mathfrak{m} \cong \mathbb{R}^m$. Due to convinience this isomorphism would be highly utilized. Moreover, in the bellow sections only algebras constructed on $\mathbb{R}^m$ are considered. Thus, a maping between sets can be defined in the following manner (hat-vee notation)

$$\begin{aligned} &[*]^{\wedge} : \ \mathbb{R}^m \to \mathfrak{m} && \boldsymbol{\tau}^{\wedge} = \sum_{i=1}^{m} \tau_i E_i \\ &[*]^{\vee} : \mathfrak{m} \to \mathbb{R}^m && \boldsymbol{\tau} = \sum_{i=1}^{m} \tau_i \mathbf{e}_i \end{aligned} \tag{A.4}$$

where $\mathbf{e}_i$ are a base of $\mathbb{R}^m$ and E_i are base vectors of $\mathfrak{m}$. Obviously, $\mathbf{e}_i^{\wedge} = E_i$. Using this mapping, the exponential / log map can be modified

$$\begin{aligned} \exp: \quad & \mathcal{X} = \exp(\boldsymbol{\tau}^{\wedge}) \\ \log: \quad & \boldsymbol{\tau} = \log(\mathcal{X})^{\vee} \end{aligned} \tag{A.5}$$

Or, in the most convinient form

$$\begin{aligned} \mathrm{Exp}: \quad & \mathcal{X} = \exp(\boldsymbol{\tau}^{\wedge}) = \mathrm{Exp}(\boldsymbol{\tau}) \\ \mathrm{Log}: \quad & \boldsymbol{\tau} = \log(\mathcal{X})^{\vee} = \mathrm{Log}(\mathcal{X}) \end{aligned} \tag{A.6}$$

Through this definitions it easy to introduce plus and minus operations

$$\begin{aligned} \text{right-}\oplus: \mathcal{X} \oplus {}^{\mathcal{X}}\boldsymbol{\tau} \equiv \mathcal{X} \circ \mathrm{Exp}({}^{\mathcal{X}}\boldsymbol{\tau}) \in \mathcal{G} \\ \text{right-}\ominus: \mathcal{Y} \ominus \mathcal{X} \equiv \mathrm{Log}(\mathcal{X}^{-1} \circ \mathcal{Y}) \in T_{\mathcal{X}}\mathcal{G} \end{aligned} \tag{A.7}$$

$\mathbf{TODO}$: continue section

Bibliography cited

[1] F. Udwadia and R. Kalaba, "A new perspective on constrained motion," *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, vol. 439, no. 1906, pp. 407–410, Nov. 1992. DOI: 10.1098/rspa.1992.0158.

[2] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033, Oct. 2012. DOI: 10.1109/IROS.2012.6386109.

[3] J. Carpentier, G. Saurel, G. Buondonno, *et al.*, "The Pinocchio C++ library – A fast and flexible implementation of rigid body dynamics algorithms and their analytical derivatives," *International Symposium on System Integration SII*, Jan. 2019. DOI: 10.1109/SII.2019.8700380.

[4] S. Diamond and S. Boyd, "CVXPY: A Python-embedded modeling language for convex optimization," *Journal of Machine Learning Research*, vol. 17, no. 83, pp. 1–5, 2016.

[5] H. S. et al., "Application of the udwadia-kalaba approach to tracking control of mobile robots," *Nonlinear Dynamics*, vol. 83, no. 1–2, pp. 389–400, Aug. 2015. DOI: 10.1007/s11071-015-2335-3.

[6] J. Peters, M. Mistry, F. Udwadia, J. Nakanishi, and S. Schaal, "A unifying framework for robot control with redundant dofs," *Autonomous Robots*, vol. 24, no. 1, pp. 1–12, Oct. 2007. DOI: 10.1007/s10514-007-9051-x.

[7] F. E. Udwadia, "Constrained motion of hamiltonian systems," *The Computer Journal*, vol. 84, no. 3, pp. 1135–1145, Dec. 2015. DOI: 10.1007/s11071-015-2558-3.

[8] M. Chignoli, N. Adrian, and P. M. Wensing, "Recursive rigid-body dynamics algorithms for systems with kinematic loops," *arXiv.org*, Nov. 2023. DOI: 10.48550/arXiv.2311.13732.

[9] S. Kuindersma, R. Deits, M. Fallon, and et al., "Optimization-based locomotion planning, estimation, and control design for the atlas humanoid robot," *Autonomous Robots*, vol. 40, pp. 429–455, Jul. 2015. DOI: 10.1007/s10514-015-9479-3.

[10] S. Sovukluk, J. Englsberger, and C. Ott, "Whole body control formulation for humanoid robots with closed/parallel kinematic chains: Kangaroo case study," *International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2023. DOI: 10.1109/IROS55552.2023.10341391.

[11] J. Baumgarte, "Stabilization of constraints and integrals of motion in dynamical systems," *Computer Methods in Applied Mechanics and Engineering*, vol. 1, no. 1, pp. 1–16, Jun. 1972. DOI: 10.1016/0045-7825(72)90018-7.

[12] B. T. Lopez and J.-J. E. Slotine, "Sliding on Manifolds: Geometric Attitude Control with Quaternions," *IEEE International Conference on Robotics and Automation (ICRA)*, May 2021. DOI: 10.1109/ICRA48506.2021.9561867.

[13] T. Lee, M. Leok, and N. H. McClamroch, "Geometric tracking control of a quadrotor UAV on SE(3)," *IEEE Conference on Decision and Control (CDC)*, Dec. 2010. DOI: 10.1109/CDC.2010.5717652.

[14] N. A. Chaturvedi, A. K. Sanyal, and N. H. McClamroch, "Rigid-Body Attitude Control," *IEEE Control Systems Magazine*, vol. 31, pp. 30–51, May 2011. DOI: 10.1109/MCS.2011.940459.

[15] H. Mishra, M. D. Stefano, A. M. Giordano, and C. Ott, "Output feedback stabilization of an orbital robot," *IEEE Conference on Decision and Control (CDC)*, Dec. 2020. DOI: 10.1109/CDC42340.2020.9304044.

[16] H. Mishra, M. D. Stefano, A. M. Giordano, and C. Ott, "A Nonlinear Observer for Free-Floating Target Motion using only Pose Measurements," *American Control Conference (ACC)*, Jul. 2019. DOI: 10.23919/ACC.2019.8814815.

[17] F. Udwadia and A. Schutte, "Equations of motion for general constrained systems in lagrangian mechanics," *Acta Mechanica*, vol. 213, pp. 111–129, Feb. 2010. DOI: 10.1007/s00707-009-0272-2.