**Proceedings of the ASME 2014 International Design Engineering Technical Conferences &
Computers and Information in Engineering Conference
IDETC/CIE 2014
August 17-20, 2014, Buffalo, New York, USA**

DETC2014-34322

EVALUATION OF COMPLEMENTARITY TECHNIQUES FOR MINIMAL COORDINATE CONTACT DYNAMICS

Harshavardhan Mylapilli
Department of Aerospace and Mechanical
Engineering, University of Southern California
Los Angeles, California 90089
Email: mylapill@usc.edu

Abhinandan Jain*
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California, 91109
Email: Abhinandan.Jain@jpl.nasa.gov

ABSTRACT

In this article, the non-smooth contact dynamics of multibody systems is formulated as a complementarity problem. Minimal coordinates operational space formulation is used to derive the dynamics equations of motion. Depending on the approach used for modeling Coulomb's friction, the complementarity problem can be posed either as a linear or a nonlinear problem. Both formulations are studied in this paper. An exact modeling of the friction cone leads to a nonlinear complementarity problem (NCP) formulation whereas a polyhedral approximation of the friction cone results in a linear complementarity problem (LCP) formulation. These complementarity problems are further recast as non-smooth unconstrained optimization problems, which are solved by employing a class of Levenberg-Marquardt algorithms. The necessary theory detailing these techniques is discussed and five schemes are implemented to solve contact dynamics problems. A simple test case of a sphere moving on a plane surface is used to validate these schemes, while a twelve-link pendulum example is chosen to compare the speed and accuracy of the schemes presented in this paper.

LIST OF ABBREVIATIONS

LCP	Linear Complementarity Problem
MLCP	Mixed Linear Complementarity Problem
NCP	Nonlinear Complementarity Problem
MNCP	Mixed Nonlinear Complementarity Problem
MC	Minimal Coordinates
RC	Redundant Coordinates
FB	Fisher-Burmeister Function
LMA	Levenberg-Marquardt Algorithm
RLM	Regular Levenberg-Marquardt Algorithm
PLM	Projected Levenberg-Marquardt Algorithm
MLCP-FB-RLM	Contact dynamics solver, where the dynamics is formulated as an MLCP, which is further recast as an optimization problem using the FB function, and solved with the help of the RLM algorithm
MLCP-FB-PLM	Similar to the MLCP-FB-RLM solver except that the PLM algorithm is used to tackle the optimization problem
MLCP-PATH	Contact dynamics solver, where the dynamics is formulated as an MLCP, and solved using the PATH [1] algorithm
MNCP-RLM	Contact dynamics solver, where the dynamics is formulated as an optimization problem using the implicit MNCP approach, and solved using the RLM algorithm
MNCP-PLM	Similar to the MNCP-RLM solver except that the PLM algorithm is used to tackle the optimization problem

*Address all correspondence to this author.

INTRODUCTION

In the past two decades, researchers have been developing complementarity based formulations to solve contact and collision dynamics problems. Complementarity based methods are an alternative to classical penalty based methods, that rely on a virtual spring-damper model to apply restoring forces at the point of deepest penetration between two bodies in contact [2]. Penalty methods notoriously suffer from oscillatory effects and become numerically unstable when bodies collide with a high velocity. Small time steps and excessively damped implicit integrators used to counter this make the method slow and computationally expensive [3].

Complementarity based methods, on the other hand, assume that the bodies are truly rigid and compute contact forces at each time step to prevent inter-penetration. Complementarity methods use impulsive dynamics to handle collision and contact interactions. They avoid the small time step and stiffening issues encountered in penalty methods by impulsively "stepping" over non-smooth events [4]. There are two variants of the complementarity formulation - the linear complementarity problem (LCP) formulation and the nonlinear complementarity problem (NCP) formulation. In the LCP formulation, the dynamics is cast as a linear complementarity problem by discretizing the friction cone using a polyhedral approximation. On the other hand, no such approximations are made in the NCP case leading to an exact modeling of the friction cone.

Considerable research effort [2,4–6] has been devoted to efficiently posing contact dynamics problems as solvable LCPs. Nevertheless, the LCP method can lead to inaccuracies because it relies on a discretized approximation of the friction cone. Increasing the accuracy of the LCP solution requires increasing the number of sides of the polygon used to approximate the friction cone, leading to an increase in the number of ancillary variables in the problem. An increased number of ancillary variables leads to a larger-sized LCP problem and a slowing down of the approach. Moreover, the degree of misalignment of friction cone direction vectors in the contact tangent space with the tangential friction impulse has a significant effect on the accuracy of the solution [2]. In contrast, the NCP method does not require the use of direction vectors and has only three unknown variables per contact leading to a more compact formulation compared to the LCP approach [3].

Finding the solution to these complementarity problems is in general a non-trivial problem. Classical approaches for solving linear complementarity problems include pivoting methods such as Lemke's or Dantzig's algorithm, whereas iterative methods such as projected-SOR or projected Gauss-Siedel methods are used for solving nonlinear complementarity problems. More recent approaches (including the approach in this paper) recast these complementarity problems as unconstrained (non-smooth) optimization problems, which can be solved using Levenberg-Marquardt type of algorithms. This approach of reformulating the complementarity problem as an unconstrained optimization problem has been shown to perform exceedingly well [1,7–10].

The aim of the present paper is to compare and contrast the linear and nonlinear complementarity approaches to solving contact dynamics problems. We use a minimal coordinates approach [6,11] to set up the contact dynamics problem of the multibody system. Additionally, the operational space formulation [12] is employed to take advantage of low order structure based recursive algorithms that significantly reduce computational costs [4]. The linear and nonlinear complementarity problems are reformulated as unconstrained optimization problems which are handled using Levenberg-Marquardt type of optimization algorithms. We develop four contact dynamics solvers, namely the MLCP–FB–RLM, MLCP–FB–PLM, MNCP–RLM, MNCP–PLM solvers based on this solution approach. We validate these solvers using the example of a sphere moving on a fixed horizontal plane, for which closed form analytical solutions are available [2]. Subsequently, we use the example of a twelve-link pendulum falling under gravity and colliding with its surrounding environment to compare the speed and accuracy of these four solvers with the MLCP-PATH method (which makes use of the PATH [1] solver).

Similar work comparing the performance of different complementarity solvers has been done by Lacoursiere et al. [13] for a different type of NCP solver. Their NCP solver uses a proximal function based approach for modeling the friction conditions while we employ Todorov's approach [7]. In the rest of this introduction, we review the important ideas pertaining to complementarity problems, constraints, contact dynamics formulations and Coulomb friction modeling.

Linear and Nonlinear Complementarity Problems

The *nonlinear complementarity problem* (NCP) seeks a vector $\mathrm{z} \in \Re^n$ satisfying the following system of equations and inequalities

$$\mathrm{z_i} \geq 0, \;\; \mathrm{f_i(z)} \geq 0, \;\; \mathrm{z_i\, f_i(z)} = 0 \;\; \text{for} \;\; i = 1, 2, \dots, n. \tag{1}$$

where $\mathrm{f} : \Re^n \to \Re^n$ is any smooth nonlinear function. The *mixed nonlinear complementarity problem* (MNCP) is defined by the mapping $\mathrm{f} : \Re^n \to \Re^n$, lower bounds $\mathfrak{l}_\mathrm{i} \in \Re \cup \{-\infty\}$ and upper bounds $\mathfrak{u}_\mathrm{i} \in \Re \cup \{+\infty\}$, where the solution of the MNCP is a vector $\mathrm{z} \in \Re^n$ such that for each $\mathrm{i} \in \{1,\ 2,\ \dots,\ n\}$, one of the following alternative holds

$$\begin{aligned} \mathrm{f_i(z)} &\geq 0 \quad \text{for } \mathrm{z_i} = \mathfrak{l}_\mathrm{i} \\ \mathrm{f_i(z)} &\leq 0 \quad \text{for } \mathrm{z_i} = \mathfrak{u}_\mathrm{i} \\ \mathrm{f_i(z)} &= 0 \quad \text{for } \mathfrak{l}_\mathrm{i} < \mathrm{z_i} < \mathfrak{u}_\mathrm{i} \end{aligned} \tag{2}$$

When f is an affine function of z i.e. $\mathrm{f(z)} = \mathfrak{M}\mathrm{z} + \mathrm{q}$, where $\mathfrak{M}$ is an $n \times n$ matrix and q is an n-vector, the NCP and the MNCP reduce to the LCP and the mixed LCP (MLCP) problems respectively.

Constraints

In contact dynamics, the constraints between rigid links can be either unilateral contact constraints that are defined by inequality relationships of the form $\mathfrak{d}(\mathrm{x},\mathrm{t}) \geq 0$, or inter-link bilateral constraints defined by equality relationships of the form $\mathfrak{b}(\mathrm{x},\mathrm{t}) = 0$, where x denotes the vector of generalized coordinates of the system and t denotes time [12].

In unilateral constraints, $\mathfrak{d}(\mathrm{x},\mathrm{t}) \geq 0$ represents the non-penetration condition between the surfaces of rigid bodies. Contact occurs at the constraint boundary when $\mathfrak{d}(\mathrm{x},\mathrm{t}) = 0$ and the surface normals at the contact point are parallel for the bodies in contact. For a pair of bodies A and B in contact, we use the convention where the i^{th} contact normal $\hat{\eta}(\mathrm{i})$ is defined as pointing from body B towards body A, such that the motion of A in the direction of the normal leads to a separation of the bodies [12]. A unilateral constraint is said to be active when there is contact, and the contact persists, i.e.,

$$\mathfrak{d}(\mathrm{x},\mathrm{t}) = \dot{\mathfrak{d}}(\mathrm{x},\mathrm{t}) = \ddot{\mathfrak{d}}(\mathrm{x},\mathrm{t}) = 0. \tag{3}$$

The contact is said to be inactive when Eqn.(3) is violated. Contact separation occurs when the relative linear velocity of the contact points along the normals becomes positive and the contact points drift apart. A separating constraint is in the process of losing contact and transitioning to an inactive state. At the start of a separation event, we have

$$\mathfrak{d}(\mathrm{x},\mathrm{t}) = \dot{\mathfrak{d}}(\mathrm{x},\mathrm{t}) = 0 \quad \text{and} \quad \ddot{\mathfrak{d}}(\mathrm{x},\mathrm{t}) \geq 0. \tag{4}$$

Only active unilateral constraints generate constraint forces on the system.

Redundant Coordinates vs Minimal Coordinates

The dynamics of a multibody system is often modeled using redundant coordinates (RC) [2, 12] The RC approach treats all bodies in the multi-link system as independent and uses $6n$ absolute coordinates for an n-link system. This system is further subject to unilateral contact constraints and explicit bilateral constraints associated with the inter-link hinges that restrict the relative motion of the of bodies. The advantages of this method include the relative ease with which the equations of motion can be set up, and the fact that the mass matrix of the system is block diagonal and constant, facilitating the use of sparse matrix solution techniques. However, due to the large number of redundant coordinates, the differential-algebraic nature of the equations of motion and the need for constraint error management at each integration time step, the RC approach is considerably slower [11].

An alternative to the RC approach is the minimal coordinates (MC) approach, where the inter-link bilateral constraints are eliminated from the equations of motion by using a minimal set of coordinates that parameterize the permissible motion for the hinges between pairs of bodies. In this approach, the system is regarded as being composed of a tree-topology sub-system together with a minimal set of bilateral constraints arising from any loop closure constraints. The advantage of this approach is that the number of generalized coordinates and number of closure constraints is much smaller compared to the RC approach. The underlying formulation is still of a differential-algebraic nature and constraint error management is still required, albeit for the smaller set of closure constraints. The mass matrix of the MC approach is however dense and configuration dependent. Using structure based recursive algorithms that do not require mass matrix inversion to solve for the system dynamics, references [11, 12] have shown that the MC approach is considerably faster than the RC approach for smooth dynamics. References [4, 6] further evaluated the RC and MC MLCP formulations for contact and collision non-smooth dynamics and found that the MC approach leads to much smaller-sized complementarity problems and is once again considerably faster compared to the RC approach. In this paper, we build upon the minimal coordinates approach to develop and compare the different complementarity formulations and their solution techniques for contact and collision dynamics modeling.

Minimal Coordinates Formulation

Let $\mathcal{N}$ denote the number of degrees of freedom for the tree sub-system. The minimal coordinates equations of motion for the tree-topology sub-system can be expressed as

$$\mathcal{M}(\boldsymbol{\theta})\,\ddot{\boldsymbol{\theta}} + \mathcal{C}(\boldsymbol{\theta},\dot{\boldsymbol{\theta}}) = \boldsymbol{\tau} \tag{5}$$

where $\boldsymbol{\theta} \in \mathfrak{R}^{\mathcal{N}}$ is the vector of hinge coordinates, $\mathcal{M}(\boldsymbol{\theta}) \in \mathfrak{R}^{\mathcal{N}\times\mathcal{N}}$ is the configuration dependent, symmetric and positive definite inertia matrix, $\mathcal{C}(\boldsymbol{\theta},\dot{\boldsymbol{\theta}}) \in \mathfrak{R}^{\mathcal{N}}$ is the vector of Coriolis, gyroscopic and gravitational forces acting on the system, and $\boldsymbol{\tau} \in \mathfrak{R}^{\mathcal{N}}$ denotes the applied generalized forces.

Let n_{b} denote the dimension of bilateral constraints arising from loop closures in the system. Since n_{b} in the MC approach corresponds only to the loop bilateral constraints, it is much smaller than the n_{b} in the RC approach. There exists a full-rank matrix $\mathrm{G_b}(\boldsymbol{\theta},\mathrm{t}) \in \mathfrak{R}^{n_{\mathrm{b}}\times\mathcal{N}}$ and a vector $\mathfrak{U}(\mathrm{t}) \in \mathfrak{R}^{n_{\mathrm{b}}}$ that defines the velocity domain constraint equation which can be expressed as

$$\mathrm{G_b}(\boldsymbol{\theta},\mathrm{t})\dot{\boldsymbol{\theta}} = \mathfrak{U}(\mathrm{t}) \tag{6}$$

The bilateral constraints effectively reduce the independent degrees of freedom of the system from $\mathcal{N}$ to $\mathcal{N} - n_{\mathrm{b}}$. The smooth dynamics of closed-chain systems can be obtained by modifying the tree system dynamics in Eqn.(5) to include the effect of the bilateral constraints via Lagrange multipliers, $\lambda \in \Re^{n_{\mathrm{b}}}$ as follows

$$\begin{aligned} \mathcal{M}(\theta)\,\ddot{\theta} \;+\; \mathcal{C}(\theta,\dot{\theta}) - \mathrm{G}_{\mathrm{b}}^{\mathrm{T}}(\theta,\mathrm{t})\lambda \;&=\; \tau \\ \mathrm{G}_{\mathrm{b}}(\theta,\mathrm{t})\dot{\theta} \;&=\; \mathfrak{U}(\mathrm{t}) \end{aligned} \tag{7}$$

where $-\mathrm{G}_{\mathrm{b}}^{\mathrm{T}}(\theta,\mathrm{t})\lambda$ term in the first equation of Eqn.(7) represents the internal generalized constraint forces from the closure constraints.

We now introduce unilateral contact constraints to this formulation. Let n_{u} denote the number of unilateral contact nodes and $\nu_{\mathrm{u}} \in \Re^{3n_{\mathrm{u}}}$ denote the vector of relative linear velocities across the contact nodes. The mapping between the contact velocities ν_{u} to the body spatial velocities $\dot{\theta}$ is defined by a matrix $\mathrm{G}_{\mathrm{u}} \in \Re^{3n_{\mathrm{u}} \times \mathcal{N}}$ such that

$$\nu_{\mathrm{u}} \;=\; \mathrm{G}_{\mathrm{u}}\,\dot{\theta} \tag{8}$$

The matrix G_{u} also maps the impulses at the contact node pairs, $\mathrm{F}_{\mathrm{u}} \in \Re^{3n_{\mathrm{u}}}$, to the corresponding generalized impulses, $p_u \in \Re^{\mathcal{N}}$, by means of the following dual mapping

$$p_u \;=\; \mathrm{G}_{\mathrm{u}}^{\mathrm{T}}\,\mathrm{F}_{\mathrm{u}} \tag{9}$$

The smooth dynamics equations of motion in Eqn. 7 can be extended to include the effect of these contact impulses p_u as follows

$$\begin{bmatrix} \mathcal{M} & -\mathrm{G}_{\mathrm{b}}^{\mathrm{T}} \\ \mathrm{G}_{\mathrm{b}} & 0 \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \lambda \end{bmatrix} = \begin{bmatrix} (\tau - \mathcal{C}) + p_u/\triangle_{\mathrm{t}} \\ \overline{\mathfrak{U}} \end{bmatrix} \tag{10}$$

where $\overline{\mathfrak{U}} = \dot{\mathfrak{U}}(\mathrm{t}) - \dot{\mathrm{G}}_{\mathrm{b}}\dot{\theta} \in \Re^{\mathrm{n_b}}$ and $\triangle_{\mathrm{t}}$ is the time step. A key requirement for solving the equations of motion is to solve for the unknown impulses, F_{u} at the contact node pairs.

Contact Impulses and Coulomb Friction Modeling

To describe the rolling and sliding phenomena at the i^{th} active contact constraint node, the 3-dimensional contact impulse $\mathrm{F}_{\mathrm{u}}(\mathrm{i}) \in \Re^3$ and contact velocity $\nu_{\mathrm{u}}(\mathrm{i}) \in \Re^3$ vectors can be decomposed into normal and tangential components as

$$\mathrm{F}_{\mathrm{u}}(\mathrm{i}) = \mathrm{F}_{\mathrm{n}}(\mathrm{i})\;\hat{\eta}(\mathrm{i}) + \mathrm{F}_{\mathrm{t}}(\mathrm{i})\;\hat{t}(\mathrm{i}) \tag{11}$$

$$\nu_{\mathrm{u}}(\mathrm{i}) = \nu_{\mathrm{n}}(\mathrm{i})\;\hat{\eta}(\mathrm{i}) + \nu_{\mathrm{t}}(\mathrm{i})\;\hat{t}(\mathrm{i}) \tag{12}$$

where $\mathrm{F}_{\mathrm{n}}(\mathrm{i}) \in \Re$ is the normal component of the contact impulse, $\hat{\eta}(\mathrm{i})$ is the contact normal, $\mathrm{F}_{\mathrm{t}}(\mathrm{i}) \in \Re^2$ is the tangential component of the friction impulse and $\hat{t}(\mathrm{i})$ is the tangent plane vector in the contact tangent plane, which is further spanned by two orthogonal vectors $\hat{f}(\mathrm{i})$ and $\hat{o}(\mathrm{i})$. Furthermore, $\nu_{\mathrm{n}}(\mathrm{i})$ and $\nu_{\mathrm{t}}(\mathrm{i})$ represent the normal and tangential components of the linear relative velocity of the body at the i^{th} contact pair. Specifically, $\nu_{\mathrm{n}}(\mathrm{i})$ denotes the relative velocity which is normal to the constraint and $\nu_{\mathrm{t}}(\mathrm{i})$ represents the relative velocities that are unconstrained but are resisted by friction.

As discussed earlier, an active i^{th} contact is defined by $\mathfrak{d}(\mathrm{i}) = 0$ (the bodies are touching) and $\nu_{\mathrm{n}}(\mathrm{i}) = 0$. Moreover, when $\nu_{\mathrm{n}}(\mathrm{i}) = 0$ and $\nu_{\mathrm{t}}(\mathrm{i}) \neq 0$, the contact is said to be *sliding*. On the other hand, when $\nu_{\mathrm{n}}(\mathrm{i}) = \nu_{\mathrm{t}}(\mathrm{i}) = 0$, the contact is said to be *rolling* [14]. Having defined the concepts of rolling and sliding, we can now state Coulomb's law of friction as

$$\begin{gathered} \mathrm{F}_{\mathrm{n}}(\mathrm{i}) \geq 0, \quad \nu_{\mathrm{n}}(\mathrm{i}) \geq 0, \quad \mathrm{F}_{\mathrm{n}}(\mathrm{i})\;\nu_{\mathrm{n}}(\mathrm{i}) = 0 \\ \nu_{\mathrm{t}}(\mathrm{i}) \;\text{ parallel to }\; \mathrm{F}_{\mathrm{t}}(\mathrm{i}), \;\; \langle \nu_{\mathrm{t}}(\mathrm{i}), \mathrm{F}_{\mathrm{t}}(\mathrm{i}) \rangle \leq 0 \\ ||\mathrm{F}_{\mathrm{t}}(\mathrm{i})|| \leq \mu(\mathrm{i})\mathrm{F}_{\mathrm{n}}(\mathrm{i}) \end{gathered} \tag{13}$$

The first line of Eqn.(13) states that the normal force and the normal contact velocity cannot both be simultaneously positive. The normal force is zero when the bodies are separating, and positive when there is sustained contact. The second line of (13) implies that if there is sliding between the bodies in contact, then the tangential friction impulse is in the direction opposite to that of the tangential relative linear velocity. The last line states that the tangential friction impulse must lie inside the friction cone. The tangential friction impulse is on the boundary of the cone when the bodies are sliding and in the interior of the cone when the bodies are rolling. The coefficient of friction is denoted by μ. Notice that only the first line of Eqn.(13) is a strict complementarity condition whereas additional work needs to be done to bring the other two conditions into the complementarity framework.

MLCP FORMULATION

In this section, we summarize the minimal coordinate, MLCP formulation for contact and collision dynamics [4, 6].

Friction Cone Discretization

The friction cone at the i^{th} contact is approximated by a friction polyhedron consisting of a finite number, n_{f}, of unit direction vectors $\hat{d_{\mathrm{j}}}(\mathrm{i})$ in the tangent plane. For notational simplicity, we assume that n_{f} is the same across all the contact points. The tangential friction impulse for the i^{th} contact is expressed as the linear combination of these direction vectors as

$$F_t(i)\hat{t}(i) = \sum_{j=1}^{n_f} \beta_j(i)\hat{d}_j(i) = D(i)\beta(i) \tag{14}$$

where $D(i) = \left[\, \hat{d}_1(i),\, \hat{d}_2(i),\, \dots\, ,\, \hat{d}_{n_f}(i) \,\right] \in \Re^{3\times n_f}$ and $\beta(i) = \mathrm{col}\{\beta_j(i)\}_{j=1}^{n_f} \in \Re^{n_f}$. Combining Eqn.(11) and Eqn.(14), we have

$$F_u(i) = \underline{D}(i)\underline{\beta}(i) \tag{15}$$

where $\underline{D}(i) = [\, \hat{\eta}(i),\, D(i) \,] \in \Re^{3\times(n_f+1)}$ and $\underline{\beta}(i) = \left[F_n(i),\, \beta^T(i)\right]^T \in \Re^{n_f+1}$. During sliding, the $\beta_j(i)$ component is non-zero and is equal to $\mu(i)F_n(i)$ for just the single direction j that corresponds to the closest direction opposing the tangential relative linear velocity. Denoting $\sigma(i) = \|v_t(i)\|$,

$$\beta_k(i) = \begin{cases} \mu(i)F_n(i), & \text{if } \sigma(i) > 0 \text{ and } k = j \\ 0, & \text{if } \sigma(i) > 0 \text{ and } k \neq j \end{cases}$$

The sliding and rolling contact relationships of Eqn.(13) can now be rephrased as the following complementarity conditions

$$\begin{aligned} \hat{\eta}^T(i)v_u^+(i) &\perp F_n(i) \\ \sigma(i)E(i) + D^T(i)v_u^+(i) &\perp \beta(i) \\ \mu(i)F_n(i) - E^T(i)\beta(i) &\perp \sigma(i) \end{aligned} \tag{16}$$

where $E(i) = \mathrm{col}\{1\}_{j=1}^{n_f} \in \Re^{n_f}$ and the component of the relative linear velocity along the contact normal is $v_n^+(i) = \hat{\eta}^T(i)v_u^+(i)$, where the superscript + (-) denotes the value of the quantity just after (before) the application of an impulse. Using Eqn.(15), the above complementarity conditions can be more compactly expressed as

$$\begin{aligned} \hat{E}(i)\sigma(i) + \underline{D}^T(i)v_u^+(i) &\perp \underline{\beta}(i) \\ \overline{E}(i)\underline{\beta}(i) &\perp \sigma(i) \end{aligned} \tag{17}$$

where $\hat{E}(i) = \left[0,\, E^T(i)\right]^T \in \Re^{n_f+1}$ and $\overline{E}(i) = \left[\mu(i),\, -E^T(i)\right] \in \Re^{1\times(n_f+1)}$. At the system level, these conditions across all the contacts can be expressed as

$$\begin{aligned} \hat{E}\sigma + \underline{D}^T v_u^+ &\perp \underline{\beta} \\ \overline{E}\underline{\beta} &\perp \sigma \end{aligned} \tag{18}$$

where $\underline{\beta} = \mathrm{col}\{\underline{\beta}(i)\}_{i=1}^{n_u} \in \Re^{n_u(n_f+1)}$, $\sigma = \mathrm{col}\{\sigma(i)\}_{i=1}^{n_u} \in \Re^{n_u}$, $\underline{D} = \mathrm{diag}\{\underline{D}(i)\}_{i=1}^{n_u} \in \Re^{3n_u\times n_u(n_f+1)}$, $\hat{E} = \mathrm{diag}\{\hat{E}(i)\}_{i=1}^{n_u} \in \Re^{n_u(n_f+1)\times n_u}$, $\overline{E} = \mathrm{diag}\{\overline{E}(i)\}_{i=1}^{n_u} \in \Re^{n_u\times n_u(n_f+1)}$ and $v_u^+ = \mathrm{col}\{v_u^+(i)\}_{i=1}^{n_u} \in \Re^{3n_u}$. Furthermore, the contact impulses at the system level can be written as

$$F_u = \underline{D}\underline{\beta} \quad \text{where} \quad F_u = \mathrm{col}\{F_u(i)\}_{i=1}^{n_u} \in \Re^{3n_u} \tag{19}$$

In principle, there are no restrictions on the number of direction vectors that can be chosen or their orientation in the contact tangent plane. Reducing the number of direction vectors reduces the size of the LCP problem but increases the approximation error. However, when choosing direction vectors, one can help reduce the approximation errors for the MLCP case by aligning *one* of the direction vectors in our set at each time step to coincide with the opposite of the tangential relative linear velocity vector (i.e. $-v_t^-$) at that time step. This ensures that at least one of the direction vectors lines up closely with tangential friction impulse (F_t^+) thereby reducing approximation errors. It should be noted however that we can only mitigate but not entirely eliminate the approximation errors arising from the misalignment of direction vectors and the tangential friction impulse. This is one of the main drawbacks of the linear complementarity approach.

Setting up the MLCP

We now set up the operational space (OS) MLCP formulation [12]. Eqn.(10) can be rearranged as

$$\begin{aligned} G_b\, \mathcal{M}^{-1} \left[\tau - \mathcal{C} + G_b^T\lambda + G_u^T\underline{D}\underline{\beta}/\triangle_t \right] - \overline{\mathfrak{U}} &= 0 \\ G_b\, \mathcal{M}^{-1}G_b^T\, p_b + G_b\, \mathcal{M}^{-1}G_u^T\, \underline{D}\underline{\beta} + \alpha_b^f \triangle_t &= 0 \end{aligned} \tag{20}$$

where $p_b = \lambda\triangle_t$ and $\alpha_b^f = G_b\mathcal{M}^{-1}(\tau - \mathcal{C}) - \overline{\mathfrak{U}}$. The relative linear acceleration of the contact nodes, $\dot{v}_u$, is obtained by differentiating Eqn.(8) with respect to time, which can be used to compute v_u^+ as follows

$$\begin{aligned} v_u^+ &= v_u^- + \dot{v}_u\triangle_t = v_u^- + (G_u\ddot{\theta} + \dot{G}_u\dot{\theta})\triangle_t \\ &= v_u^- + G_u\mathcal{M}^{-1}G_b^T\, p_b + G_u\mathcal{M}^{-1}G_u^T\, \underline{D}\underline{\beta} + \alpha_u^f\triangle_t \end{aligned} \tag{21}$$

where $\alpha_u^f = G_u\mathcal{M}^{-1}(\tau - \mathcal{C}) + \dot{G}_u\dot{\theta}$. Denoting n_c as the number of the combined set of nodes associated with the unilateral and bilateral constraints of the system, the spatial velocities of these nodes is given by the stacked vector $\mathcal{V}_c \in \Re^{6n_c}$, which is related to $\dot{\theta}$ by $\mathcal{V}_c = \mathcal{J}\dot{\theta}$, where $\mathcal{J} \in \Re^{6n_c\times\mathcal{N}}$ is the Jacobian of the constraint nodes. Now, there exist matrices $\mathcal{Q}_u \in \Re^{3n_u\times 6n_c}$ and $\mathcal{Q}_b \in \Re^{3n_b\times 6n_c}$ such that $v_u = \mathcal{Q}_u\mathcal{V}_c = \mathcal{Q}_u\mathcal{J}\dot{\theta}$ and $\mathfrak{U} = \mathcal{Q}_b\mathcal{V}_c = \mathcal{Q}_b\mathcal{J}\dot{\theta}$. Comparing these expressions with Eqn.(6) and Eqn.(8), we obtain $G_u = \mathcal{Q}_u\mathcal{J}$ and $G_b = \mathcal{Q}_b\mathcal{J}$. Denoting

$\Lambda = \mathcal{J}\mathcal{M}^{-1}\mathcal{J}^T \in \Re^{6n_\mathrm{c}\times 6n_\mathrm{c}}$, Eqns.(18 – 21) can be expressed as

$$\begin{aligned} \mathcal{Q}_\mathrm{b}\Lambda\mathcal{Q}_\mathrm{b}^\mathrm{T}\, p_b \;+\; \mathcal{Q}_\mathrm{b}\Lambda\mathcal{Q}_\mathrm{u}^\mathrm{T}\, \underline{\mathrm{D}}\underline{\beta} \;+\; \alpha_\mathrm{b}^\mathrm{f}\triangle_\mathrm{t} = 0 \\ \underline{\mathrm{D}}^\mathrm{T}\mathcal{Q}_\mathrm{u}\Lambda\mathcal{Q}_\mathrm{b}^\mathrm{T}\, p_b + \underline{\mathrm{D}}^\mathrm{T}\mathcal{Q}_\mathrm{u}\Lambda\mathcal{Q}_\mathrm{u}^\mathrm{T}\, \underline{\mathrm{D}}\underline{\beta} + \hat{\mathrm{E}}\sigma + \underline{\mathrm{D}}^\mathrm{T}(\alpha_\mathrm{u}^\mathrm{f}\triangle_\mathrm{t} + \nu_\mathrm{u}^-) \perp \underline{\beta} \\ \overline{\mathrm{E}}\underline{\beta} \perp \sigma \end{aligned} \tag{22}$$

or more compactly, we have $\mathrm{f(z)} = \mathfrak{M}\mathrm{z} + \mathrm{q}$, where

$$\mathfrak{M} = \begin{bmatrix} \mathrm{X}\Lambda\mathrm{X}^\mathrm{T} & \mathrm{E}_1 \\ \mathrm{E}_2 & \mathbf{0} \end{bmatrix}, \quad \mathrm{z} = \begin{bmatrix} p_b \\ \underline{\beta} \\ \sigma \end{bmatrix} \text{ and } \mathrm{q} = \begin{bmatrix} \alpha_\mathrm{b}^\mathrm{f}\triangle_\mathrm{t} \\ \underline{\mathrm{D}}^\mathrm{T}(\alpha_\mathrm{u}^\mathrm{f}\triangle_\mathrm{t} + \nu_\mathrm{u}^-) \\ \mathbf{0} \end{bmatrix}$$

where $\mathrm{X} = \begin{bmatrix} \mathcal{Q}_\mathrm{b} \\ \underline{\mathrm{D}}^\mathrm{T}\mathcal{Q}_\mathrm{u} \end{bmatrix}$, $\mathrm{E}_1 = \begin{bmatrix} \mathbf{0} \\ \hat{\mathrm{E}} \end{bmatrix}$ and $\mathrm{E}_2 = \begin{bmatrix} \mathbf{0} & \overline{\mathrm{E}} \end{bmatrix}$.

Equation (22) is a $(n_\mathrm{b} + n_\mathrm{u}(n_\mathrm{f} + 2))$ sized mixed LCP where the first equation is an equality condition while the bottom two equations are complementarity conditions. Structure based recursive algorithms of order $\mathrm{O}(\mathcal{N}) + \mathrm{O}(n_\mathrm{c}^2)$ can be used to compute the configuration dependent matrix Λ as shown in [6, 11].

Casting the MLCP as an Optimization Problem

The mixed complementarity conditions can be reformulated as a nonlinear non-smooth system of equations which can then be solved using unconstrained minimization algorithms [15]. The advantage of this conversion is the multitude of optimization solvers available for solving such unconstrained minimization problems. To cast the linear complementarity conditions as a nonlinear equation, we make use of *NCP functions* [16] which have the property that

$$\phi(a,b) = 0 \iff a \geq 0,\; b \geq 0,\; ab = 0 \tag{23}$$

An example of an NCP function is given by

$$\phi_{\mathrm{FB}} := \phi_\mathrm{i}(\mathrm{z_i}, \mathrm{f_i(z)}) = \sqrt{\mathrm{z_i^2} + \mathrm{f_i^2(z)}} - \mathrm{z_i} - \mathrm{f_i(z)} \tag{24}$$

which is called as the Fisher-Burmeister (FB) function [17]. The FB function has the property that the square of Eqn.(24) is continuously differentiable [15]. This fact can be used to build a cost function $\psi : \Re^n \to \Re^+$ as follows

$$\psi(\mathrm{z,f}) = \frac{1}{2}\sum_{\mathrm{i=1}}^{\mathrm{n}} \phi_\mathrm{i}^2(\mathrm{z_i}, \mathrm{f_i}) \tag{25}$$

Minimizing the cost function ψ gives us the solution to the system of nonlinear equations $\phi_{FB} = 0$, which owing to Eqn.(23) gives us the solution to the complementarity problem.

On the other hand, to solve a mixed linear complementarity problem, the NCP function ϕ is modified as [18]

$$\phi_\mathrm{i} = \begin{cases} \sqrt{\mathrm{f_i^2(z)} + \mathrm{z_i^2}} - \mathrm{z_i} - \mathrm{f_i(z)} & \text{if } \mathfrak{l}_\mathrm{i} = 0 \text{ and } \mathfrak{u}_\mathrm{i} = \infty \\ -\mathrm{f_i(z)} & \text{if } \mathfrak{l}_\mathrm{i} = -\infty \text{ and } \mathfrak{u}_\mathrm{i} = +\infty \end{cases}$$

where $\mathfrak{l}_\mathrm{i} = 0$ and $\mathfrak{u}_\mathrm{i} = \infty$ correspond to complementarity conditions and $\mathfrak{l}_\mathrm{i} = -\infty$ and $\mathfrak{u}_\mathrm{i} = \infty$ correspond to equality conditions. The cost function and its gradient for this MLCP are given by ψ (see Eqn.(25)) and $\triangledown\psi = \phi^\mathrm{T} J$ respectively, where J is the Jacobian matrix whose entries $J_\mathrm{ij} = \left[\frac{\partial\phi_\mathrm{i}}{\partial z_\mathrm{j}}\right]$ are given as follows:

a. If $\mathfrak{l}_\mathrm{i} = 0$ and $\mathfrak{u}_\mathrm{i} = \infty$, then

$$\frac{\partial\phi_\mathrm{i}}{\partial z_\mathrm{i}} = \begin{cases} \frac{\mathrm{z_i + f_i}\ \mathfrak{M}(\mathrm{i,i})}{\sqrt{\mathrm{z_i^2 + f_i^2}}} - 1 - \mathfrak{M}(\mathrm{i,i}), & \text{if } \sqrt{\mathrm{z_i^2} + \mathrm{f_i^2(z)}} \neq 0 \\ \sqrt{1 + (\mathfrak{M}(\mathrm{i,i}))^2} - 1 - \mathfrak{M}(\mathrm{i,i}), & \text{if } \sqrt{\mathrm{z_i^2} + \mathrm{f_i^2(z)}} = 0 \end{cases}$$

and

$$\frac{\partial\phi_\mathrm{i}}{\partial z_\mathrm{j}} = \begin{cases} \frac{\mathrm{f_i}\ \mathfrak{M}(\mathrm{i,j})}{\sqrt{\mathrm{z_i^2 + f_i^2}}} - \mathfrak{M}(\mathrm{i,j}) \quad , & \text{if } \sqrt{\mathrm{z_i^2} + \mathrm{f_i^2(z)}} \neq 0 \\ 0 \quad , & \text{if } \sqrt{\mathrm{z_i^2} + \mathrm{f_i^2(z)}} = 0 \end{cases}$$

b. If $\mathfrak{l}_\mathrm{i} = -\infty$ and $\mathfrak{u}_\mathrm{i} = \infty$, then $\frac{\partial\phi_\mathrm{i}}{\partial z_\mathrm{j}} = -\mathfrak{M}(\mathrm{i,j})$.

With this optimization reformulation, we can now employ unconstrained minimization algorithms to find the minimum point, and hence the solution to the complementarity problem.

MNCP FORMULATION

We now turn our attention to the nonlinear complementarity formulations of contact dynamics that avoid the friction cone approximations required for the LCP formulation.

Exact Modeling of Friction Cone.

Until now, we have only dealt with an approximate model of the friction cone. However, to model the friction cone exactly, an alternative approach is required that captures Coulomb's friction conditions (see Eqn.(13)) exactly. In the present work, we use Todorov's approach [7] to model the friction conditions, which we summarize here for the convenience of the reader. A similar approach has also been described by Drumwright and Shell [19].

Realizing that the contact relative linear velocity $\nu_\mathrm{u}(\mathrm{i})$ and the contact impulse $\mathrm{F_u(i)}$ are not independent but are instead coupled through laws of contact and friction, Todorov [7] parameterizes both $\mathrm{F_u(i)}$ and $\nu_\mathrm{u}(\mathrm{i})$ using *unconstrained* non-physical variables $\mathfrak{z}(\mathrm{i}) = [\mathfrak{z}_{\mathrm{f}}(\mathrm{i})\ \mathfrak{z}_{\mathrm{o}}(\mathrm{i})\ \mathfrak{z}_{\mathrm{n}}(\mathrm{i})]^{\mathrm{T}} \in \Re^3$ as follows:

$$\begin{aligned} \mathrm{F_n}(\mathrm{i}) &= \max(0, -\mathfrak{z}_{\mathrm{n}}(\mathrm{i})) \quad \text{and} \quad \nu_{\mathrm{n}}(\mathrm{i}) = \max(0, \mathfrak{z}_{\mathrm{n}}(\mathrm{i})) \\ \mathrm{F_t}(\mathrm{i}) &= -s(\mathrm{i})\ \mathfrak{z}_{\mathrm{t}}(\mathrm{i}) \qquad \text{and} \quad \nu_{\mathrm{t}}(\mathrm{i}) = [1 - s(\mathrm{i})]\mathfrak{z}_{\mathrm{t}}(\mathrm{i}) \end{aligned} \tag{26}$$

where $s(\mathrm{i}) = \min\left(1, \frac{\mu(\mathrm{i})\mathrm{F_n}(\mathrm{i})}{\|\mathfrak{z}_{\mathrm{t}}(\mathrm{i})\|}\right)$. The normal force $\mathrm{F_n}(\mathrm{i})$ and normal velocity $\nu_{\mathrm{n}}(\mathrm{i})$ are encoded by $\mathfrak{z}_{\mathrm{n}}(\mathrm{i})$ whereas the tangential force $\mathrm{F_t}(\mathrm{i}) = [\mathrm{F_f}(\mathrm{i}),\ \mathrm{F_o}(\mathrm{i})]^{\mathrm{T}} \in \Re^2$ and tangential velocity $\nu_{\mathrm{t}}(\mathrm{i}) = [\nu_{\mathrm{f}}(\mathrm{i}), \nu_{\mathrm{o}}(\mathrm{i})]^{\mathrm{T}} \in \Re^2$ are encoded by the tangential components of $\mathfrak{z}$ i.e. $\mathfrak{z}_{\mathrm{t}}(\mathrm{i}) = [\mathfrak{z}_{\mathrm{f}}(\mathrm{i}),\ \mathfrak{z}_{\mathrm{o}}(\mathrm{i})]^{\mathrm{T}} \in \Re^2$. The role of these parameters can be succinctly understood as follows:

1. **Non-penetration condition**: When $\mathfrak{z}_{\mathrm{n}}(\mathrm{i}) > 0$, the i^{th} contact constraint is inactive and $\mathfrak{z}_{\mathrm{n}}(\mathrm{i})$ represents the non-zero relative normal linear velocity $\nu_{\mathrm{n}}(\mathrm{i}) = \mathfrak{z}_{\mathrm{n}}(\mathrm{i})$, with normal force $\mathrm{F_n}(\mathrm{i}) = 0$. Consequently, $\mathrm{F_t}(\mathrm{i}) = [0,\ 0]^{\mathrm{T}}$. On the other hand, when $\mathfrak{z}_{\mathrm{n}}(\mathrm{i}) < 0$, the contact constraint is active, and $\mathfrak{z}_{\mathrm{n}}(\mathrm{i})$ represents the non-zero normal force $\mathrm{F_n}(\mathrm{i}) = -\mathfrak{z}_{\mathrm{n}}(\mathrm{i})$ with the normal velocity $\nu_{\mathrm{n}}(\mathrm{i}) = 0$. Thus, $\mathfrak{z}_{\mathrm{n}}(\mathrm{i})$ characterizes the non-penetration condition and the complementarity relation between the $\mathrm{F_n}(\mathrm{i})$ and $\nu_{\mathrm{n}}(\mathrm{i})$ variables.

2. **Rolling contact**: When the i^{th} contact is active and $s(\mathrm{i}) = 1$ (i.e. when $\|\mathfrak{z}_{\mathrm{t}}(\mathrm{i})\| \leq \mu(\mathrm{i})\mathrm{F_n}(\mathrm{i})$), we have a rolling contact. In this case, $\nu_{\mathrm{t}}(\mathrm{i}) = [0,\ 0]^{\mathrm{T}}$. Furthermore, $\mathrm{F_t}(\mathrm{i}) = -\mathfrak{z}_{\mathrm{t}}(\mathrm{i})$. Since $\|\mathfrak{z}_{\mathrm{t}}(\mathrm{i})\| = \| - \mathrm{F_t}(\mathrm{i}))\| \leq \mu(\mathrm{i})\mathrm{F_n}(\mathrm{i})$, we are inside the friction cone, which corresponds to a rolling contact.

3. **Sliding contact**: When the i^{th} contact is active and $s(\mathrm{i}) < 1$ (i.e when $\|\mathfrak{z}_{\mathrm{t}}(\mathrm{i})\| > \mu(\mathrm{i})\mathrm{F_n}(\mathrm{i})$), we have a sliding contact. The tangential velocity $\nu_{\mathrm{t}}(\mathrm{i}) = (1 - s(\mathrm{i}))\ \mathfrak{z}_{\mathrm{t}}(\mathrm{i})$ is in a direction opposite to the tangential friction $\mathrm{F_t}(\mathrm{i}) = -s(\mathrm{i})\ \mathfrak{z}_{\mathrm{t}}(\mathrm{i})$ and $\|\mathrm{F_t}(\mathrm{i})\| = \mu(\mathrm{i})\mathrm{F_n}(\mathrm{i})$. Thus, we are on the boundary of the friction cone which corresponds to a sliding contact.

For an in depth discussion and a schematic depiction of these functions, the interested reader is referred to [7]. The set of equations in Eqn.(26), which model Coulomb's friction exactly, can be more compactly represented in terms of $\mathfrak{z}(\mathrm{i})$ as

$$\nu_{\mathrm{u}}(\mathrm{i}) = \mathrm{F_u}(\mathrm{i}) + \mathfrak{z}(\mathrm{i}) \tag{27}$$

Setting up the implicit MNCP

We now develop the operational space implicit MNCP formulation. Using the system level contact impulse, Eqn. (10) can be rearranged as

$$\mathrm{G_b}\ \mathcal{M}^{-1}\mathrm{G_b^T}\ p_b\ +\ \mathrm{G_b}\ \mathcal{M}^{-1}\mathrm{G_u^T}\ \mathrm{F_u}\ +\ \alpha_{\mathrm{b}}^{\mathrm{f}} \triangle_{\mathrm{t}}\ =\ 0 \tag{28}$$

which can be further simplified using $\mathrm{G_u} = \mathcal{Q}_{\mathrm{u}}J$ and $\mathrm{G_b} = \mathcal{Q}_{\mathrm{b}}J$ as

$$\mathcal{Q}_{\mathrm{b}}\Lambda\mathcal{Q}_{\mathrm{b}}^{\mathrm{T}}\ p_b\ +\ \mathcal{Q}_{\mathrm{b}}\Lambda\mathcal{Q}_{\mathrm{u}}^{\mathrm{T}}\ \mathrm{F_u}\ +\ \alpha_{\mathrm{b}}^{\mathrm{f}}\triangle_{\mathrm{t}} = 0 \tag{29}$$

The relative linear velocity of the contact nodes after the application of the contact impulse, ν_{u}^{+}, is given by

$$\begin{aligned} \nu_{\mathrm{u}}^{+}\ &=\ \nu_{\mathrm{u}}^{-}\ +\ \dot{\nu}_{\mathrm{u}}\triangle_{\mathrm{t}}\ =\ \nu_{\mathrm{u}}^{-}\ +\ (\mathrm{G_u}\ddot{\theta}\ +\ \dot{\mathrm{G}}_{\mathrm{u}}\dot{\theta})\triangle_{\mathrm{t}} \\ &=\ \nu_{\mathrm{u}}^{-}\ +\ \mathrm{G_u}\mathcal{M}^{-1}\mathrm{G_b^T}\ p_b\ +\ \mathrm{G_u}\mathcal{M}^{-1}\mathrm{G_u^T}\ \mathrm{F_u}\ +\ \alpha_{\mathrm{u}}^{\mathrm{f}}\triangle_{\mathrm{t}} \\ &=\ \nu_{\mathrm{u}}^{-}\ +\ \mathcal{Q}_{\mathrm{u}}\Lambda\mathcal{Q}_{\mathrm{b}}^{\mathrm{T}}\ p_b\ +\ \mathcal{Q}_{\mathrm{u}}\Lambda\mathcal{Q}_{\mathrm{u}}^{\mathrm{T}}\ \mathrm{F_u}\ +\ \alpha_{\mathrm{u}}^{\mathrm{f}}\triangle_{\mathrm{t}} \end{aligned} \tag{30}$$

Using Eqn.(27), Eqns.(29 - 30) can be rewritten as

$$\begin{aligned} \Phi_{\mathrm{b}} &= \mathcal{Q}_{\mathrm{b}}\Lambda\mathcal{Q}_{\mathrm{b}}^{\mathrm{T}}\ \ p_b + \mathcal{Q}_{\mathrm{b}}\Lambda\mathcal{Q}_{\mathrm{u}}^{\mathrm{T}}\ \ \mathrm{F_u}(\mathfrak{z}) + \alpha_{\mathrm{b}}^{\mathrm{f}}\triangle_{\mathrm{t}} = 0 \\ \Phi_{\mathrm{u}} &= \mathcal{Q}_{\mathrm{u}}\Lambda\mathcal{Q}_{\mathrm{b}}^{\mathrm{T}}\ \ p_b + (\mathcal{Q}_{\mathrm{u}}\Lambda\mathcal{Q}_{\mathrm{u}}^{\mathrm{T}} - I)\ \mathrm{F_u}(\mathfrak{z}) - \mathfrak{z} + \alpha_{\mathrm{u}}^{\mathrm{f}}\triangle_{\mathrm{t}} + \nu_{\mathrm{u}}^{-} = 0 \end{aligned} \tag{31}$$

which can be expressed in matrix form as

$$\Phi(p_b, \mathfrak{z}) = \begin{bmatrix} \mathcal{Q}_{\mathrm{b}}\Lambda\mathcal{Q}_{\mathrm{b}}^{\mathrm{T}} & \mathcal{Q}_{\mathrm{b}}\Lambda\mathcal{Q}_{\mathrm{u}}^{\mathrm{T}} \\ \mathcal{Q}_{\mathrm{u}}\Lambda\mathcal{Q}_{\mathrm{b}}^{\mathrm{T}} & (\mathcal{Q}_{\mathrm{u}}\Lambda\mathcal{Q}_{\mathrm{u}}^{\mathrm{T}} - I) \end{bmatrix} \begin{bmatrix} p_b \\ \mathrm{F_u}(\mathfrak{z}) \end{bmatrix} + \begin{bmatrix} \alpha_{\mathrm{b}}^{\mathrm{f}}\triangle_{\mathrm{t}} \\ \alpha_{\mathrm{u}}^{\mathrm{f}}\triangle_{\mathrm{t}} + \nu_{\mathrm{u}}^{-} - \mathfrak{z} \end{bmatrix} \tag{32}$$

where Φ is called the residual and I is the identity matrix.

Optimization Reformulation of the implicit MNCP

The problem now reduces to solving a set of nonlinear nonsmooth equations $\Phi = 0$, which can be rephrased as an unconstrained minimization problem with the cost given by

$$\psi = \frac{1}{2}\Phi^T\Phi \tag{33}$$

The gradient of the cost is given by $\bigtriangledown\psi = \Phi^{\mathrm{T}}J$ where the Jacobian J can be computed as

$$J = \begin{bmatrix} \frac{\partial\Phi_{\mathrm{b}}}{\partial p_b} & \frac{\partial\Phi_{\mathrm{b}}}{\partial\mathfrak{z}} \\ \frac{\partial\Phi_{\mathrm{u}}}{\partial p_b} & \frac{\partial\Phi_{\mathrm{u}}}{\partial\mathfrak{z}} \end{bmatrix} = \begin{bmatrix} \mathcal{Q}_{\mathrm{b}}\Lambda\mathcal{Q}_{\mathrm{b}}^{\mathrm{T}} & \mathcal{Q}_{\mathrm{b}}\Lambda\mathcal{Q}_{\mathrm{u}}^{\mathrm{T}} \\ \mathcal{Q}_{\mathrm{u}}\Lambda\mathcal{Q}_{\mathrm{b}}^{\mathrm{T}} & (\mathcal{Q}_{\mathrm{u}}\Lambda\mathcal{Q}_{\mathrm{u}}^{\mathrm{T}} - I) \end{bmatrix} \begin{bmatrix} I & \mathbf{0} \\ \mathbf{0} & \frac{\partial\mathrm{F_u}}{\partial\mathfrak{z}} \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -I \end{bmatrix} \tag{34}$$

where $\frac{\partial\mathrm{F_u}}{\partial\mathfrak{z}} = \mathrm{diag}\left[\frac{\partial\mathrm{F_u}(1)}{\partial\mathfrak{z}(1)}, \frac{\partial\mathrm{F_u}(2)}{\partial\mathfrak{z}(2)}, \dots \frac{\partial\mathrm{F_u}(\mathrm{n_u})}{\partial\mathfrak{z}(\mathrm{n_u})}\right]$, and n_{u} is the number of unilateral contacts at the current integration time step. Focusing on a single typical contact for convenience, the expressions for its individual 3×3 block matrix is given by

$$\frac{\partial\mathrm{F_u}}{\partial\mathfrak{z}} = \frac{\partial}{\partial\mathfrak{z}}\begin{bmatrix} \mathrm{F_f} \\ \mathrm{F_o} \\ \mathrm{F_n} \end{bmatrix} = \begin{bmatrix} \frac{\partial\mathrm{F_f}}{\partial\mathfrak{z}_{\mathrm{f}}} & \frac{\partial\mathrm{F_f}}{\partial\mathfrak{z}_{\mathrm{o}}} & \frac{\partial\mathrm{F_f}}{\partial\mathfrak{z}_{\mathrm{n}}} \\ \frac{\partial\mathrm{F_o}}{\partial\mathfrak{z}_{\mathrm{f}}} & \frac{\partial\mathrm{F_o}}{\partial\mathfrak{z}_{\mathrm{o}}} & \frac{\partial\mathrm{F_o}}{\partial\mathfrak{z}_{\mathrm{n}}} \\ \frac{\partial\mathrm{F_n}}{\partial\mathfrak{z}_{\mathrm{f}}} & \frac{\partial\mathrm{F_n}}{\partial\mathfrak{z}_{\mathrm{o}}} & \frac{\partial\mathrm{F_n}}{\partial\mathfrak{z}_{\mathrm{n}}} \end{bmatrix} \tag{35}$$

where the entries of $\frac{\partial \mathrm{F_u}}{\partial \mathfrak{z}}$ are as follows

$$\frac{\partial \mathrm{F_f}}{\partial \mathfrak{z}_{\mathrm{f}}} = \begin{cases} -s & \text{if} \quad m_f \geq 1 \\ \frac{-s\mathfrak{z}_{\mathrm{o}}^2}{\|\mathfrak{z}_{\mathrm{f}},\mathfrak{z}_{\mathrm{o}}\|^2} & \text{if} \quad m_f < 1 \end{cases}, \quad \frac{\partial \mathrm{F_f}}{\partial \mathfrak{z}_{\mathrm{o}}} = \begin{cases} 0 & \text{if} \quad m_f \geq 1 \\ \frac{s\mathfrak{z}_{\mathrm{o}}\mathfrak{z}_{\mathrm{f}}}{\|\mathfrak{z}_{\mathrm{f}},\mathfrak{z}_{\mathrm{o}}\|^2} & \text{if} \quad m_f < 1 \end{cases}$$

$$\frac{\partial \mathrm{F_f}}{\partial \mathfrak{z}_{\mathrm{n}}} = \begin{cases} 0 & \text{if} \quad m_f \geq 1 \\ \frac{-\mu \frac{\partial \mathrm{F_n}}{\partial \mathfrak{z}_{\mathrm{n}}} \mathfrak{z}_{\mathrm{f}}}{\|\mathfrak{z}_{\mathrm{f}},\mathfrak{z}_{\mathrm{o}}\|} & \text{if} \quad m_f < 1 \end{cases}, \quad \frac{\partial \mathrm{F_o}}{\partial \mathfrak{z}_{\mathrm{f}}} = \begin{cases} 0 & \text{if} \quad m_f \geq 1 \\ \frac{s\mathfrak{z}_{\mathrm{f}}\mathfrak{z}_{\mathrm{o}}}{\|\mathfrak{z}_{\mathrm{f}},\mathfrak{z}_{\mathrm{o}}\|^2} & \text{if} \quad m_f < 1 \end{cases}$$

$$\frac{\partial \mathrm{F_o}}{\partial \mathfrak{z}_{\mathrm{o}}} = \begin{cases} -s & \text{if} \quad m_f \geq 1 \\ \frac{-s\mathfrak{z}_{\mathrm{f}}^2}{\|\mathfrak{z}_{\mathrm{f}},\mathfrak{z}_{\mathrm{o}}\|^2} & \text{if} \quad m_f < 1 \end{cases}, \quad \frac{\partial \mathrm{F_o}}{\partial \mathfrak{z}_{\mathrm{n}}} = \begin{cases} 0 & \text{if} \quad m_f \geq 1 \\ \frac{-\mu \frac{\partial \mathrm{F_n}}{\partial \mathfrak{z}_{\mathrm{n}}} \mathfrak{z}_{\mathrm{o}}}{\|\mathfrak{z}_{\mathrm{f}},\mathfrak{z}_{\mathrm{o}}\|} & \text{if} \quad m_f < 1 \end{cases}$$

$$\frac{\partial \mathrm{F_n}}{\partial \mathfrak{z}_{\mathrm{f}}} = 0, \quad \frac{\partial \mathrm{F_n}}{\partial \mathfrak{z}_{\mathrm{o}}} = 0, \quad \frac{\partial \mathrm{F_n}}{\partial \mathfrak{z}_{\mathrm{n}}} = \begin{cases} -1 & \text{if} \quad \mathfrak{z}_{\mathrm{n}} < 0 \\ 0 & \text{if} \quad \mathfrak{z}_{\mathrm{n}} \geq 0 \end{cases}, \quad m_f = \frac{\mu \mathrm{F_n}}{\|\mathfrak{z}_{\mathrm{f}},\mathfrak{z}_{\mathrm{o}}\|}$$

Such optimization problems are usually handled using some form of Gauss-Newton methods e.g. the Levenberg-Marquardt algorithms which we consider in more detail in the next section.

UNCONSTRAINED OPTIMIZATION ALGORITHMS

In this section, we discuss some methods for solving unconstrained optimization problems of the form

minimize $\psi(x)$ *where* $\psi : \Re^n \rightarrow \Re^+$
is convex and twice continuously differentiable.

We assume that the optimization problem above is solvable, in other words, there exists an optimal value at x^* denoted by p^* such that $p^* = \psi(x^*)$ is the minimum. If ψ is differentiable and convex, a necessary and sufficient condition for x^* to be optimal is that the gradient of ψ at x^* is zero i.e. $\bigtriangledown\psi(x^*) = 0$ which gives us n equations in n variables, the solution to which gives us the optimal value of $\psi(x)$. The most practical methods to solve convex minimization problems involve iterative algorithms where a sequence of points $x_1, x_2, ..., x_k \in$ dom ψ are produced with $\psi(x_k) \rightarrow p^*$ as $k \rightarrow \infty$. The algorithm terminates when $\psi(x_k) - p^* \leq \varepsilon$ where ε is some specified tolerance very close to zero. Extensions for non-smooth problems, where the function ψ is continuous but may be non-differentiable at finitely many points are discussed in references [7, 8].

Levenberg-Marquardt Algorithm

Newton-like methods often fail to work properly as the Hessian matrix computed at each iteration step can become singular during optimization and stall the algorithm. Moreover, in some cases, the Newton step being computed may not be a step in the descent direction leading to erroneous results. The Levenberg-Marquardt algorithm has been suggested in the literature to safeguard against these issues and to also converges irrespective of the starting position [8, 20].

The Levenberg-Marquardt algorithm (LMA), also known as the damped least squares algorithm, interpolates between Newton's method and the method of gradient descent. The LMA method combines the advantages of both worlds. Gradient methods are guaranteed to reach the minimum point (although they can take a long time) whereas Newton-like methods are known for their speed of convergence to the minimum. By combining both methods, LMA reaches the minimum point and has similar performance to other Newton-like methods. The LMA method is more robust compared to Newton's method because in many cases we do end up finding the minimum point even when we start off far away from it.

One disadvantage of the LMA method is that the Hessian needs to be computed, stored and inverted. If the size of the Hessian is large (more than 100), the LMA method can become slow. Regardless, the LMA method is widely used by researchers attempting to solve complementarity problems reposed as optimization problems [7–9]. Below we discuss two such LMA algorithms - the regular LMA and the projected LMA.

Regular Levenberg-Marquardt Algorithm: The following implementation of the regular LMA algorithm has been adopted from Bazaraa et al. [20].

```
ALGORITHM : Regular LMA (RLM)
given $\psi(x) = \frac{1}{2}\phi(x)^T\phi(x)$, initial $x \in dom\ \psi$, tol $\varepsilon = 10^{-30} > 0$,
$\lambda_{\mathrm{LM}} = 10^{-3}, \lambda_{step} = 20, \lambda_{min} = 10^{-25}$

repeat
  1. Compute the Cost $\psi$, Jacobian $J$, approximate Hessian $H$
     and the step $\triangle x$ as $\psi(x_k)$, $J(x_k) = \frac{\partial\phi}{\partial x}$, $H(x_k) = J^TJ + \lambda_{\mathrm{LM}}I$, $\triangle x_k = H^{-1}(J^T\phi)$ respectively.
  2. Stopping Criteria: If $\psi(x_k) < \varepsilon$, then STOP
  3. Update $x_{k+1} := x_k + \triangle x_k$
  4. Compute the Reduction Factor $r = \frac{\psi(x_{k+1})-\psi(x_k)}{q(x_{k+1})-q(x_k)}$
  5. Adaptively change $\lambda$ based on the reduction $r$:
     if $r < 0.3$, then $\lambda_{\mathrm{LM}} := \lambda_{\mathrm{LM}} * \lambda_{step}$
     if $r > 0$, then accept $\triangle x$ and if $r > 0.8$, then $\lambda_{\mathrm{LM}} := \frac{\lambda_{\mathrm{LM}}}{\lambda_{step}}$
```

The regular LMA adaptively keeps switching between a gradient descent method and a Newton's method based on the reduction factor r. The reduction factor r is defined as the ratio of the actual decrease of the function ψ and its quadratic approximation q at $x = x_k$. The closer the reduction factor r is to unity, the more reliable is the quadratic approximation, and the smaller we can allow λ_{LM} to be. Thus, if r is small, we increase λ_{LM} and if r is large, λ_{LM} is reduced. If λ_{LM} is close to zero, then the LMA becomes a Newton's method and if λ_{LM} is large, then this makes the Hessian diagonally dominant and we end up with a gradient descent method. Typical parameters that work well empirically with this method are $r_{min} = 0.3, r_{max} = 0.8, \lambda_{\mathrm{LM}} = 10^{-3}$ and $\lambda_{step} = 20$ [20].

Projected Levenberg-Marquardt Algorithm: The following basic implementation of the projected LMA has been adopted from Kanzow and Petra [10].

```
ALGORITHM: Projected LMA (PLM)
given $\psi(x) = \frac{1}{2}\phi(x)^T\phi(x)$, initial $x \in dom\ \psi$, tol $\varepsilon = 10^{-30} > 0$, $\lambda_{\text{LM}} = 10^{-16}$

repeat
  1. Compute the Cost $\psi$, Jacobian $J$, appended-Jacobian $H$, appended-$\phi$ $B$ and the step $\triangle x$ as $\psi(x_k)$, $J(x_k) = \frac{\partial \phi}{\partial x}$, $H(x_k) = \begin{bmatrix} J \\ \sqrt{\lambda_{\text{LM}}}\, I \end{bmatrix}$, $B(x_k) = \begin{bmatrix} -\phi(x_k) \\ 0 \end{bmatrix}$, $\triangle x_k = H^{-1}B$ respectively.
  2. Stopping Criteria: If $\psi(x_k) < \varepsilon$, then STOP
  3. Update $x_{k+1} := x_k + t \triangle x_k$
```

The main difference between the projected LMA and the regular LMA is in the computation of the search direction. In the projected LMA, the search direction is computed as $\triangle x_k = H^{-1}B$, where $H_{2n\times n}$ is generated by stacking the $n \times n$ Jacobian matrix J with the $n \times n$ diagonal matrix whose (diagonal) entries are given by $\sqrt{\lambda_{\text{LM}}}$ (where λ_{LM} is the Levenberg-Marquardt damping parameter). Similarly, the $2n \times 1$ vector B is generated by stacking the vector $(-\phi)$ with a zero vector of size n. To this basic algorithm, additional features have been added such as the non-monotone line search technique [21] together with watchdog stabilization [22] to make the algorithm more robust. For example, if the best function value found so far has not been sufficiently reduced within a fixed number of iterations, we restart from that point using a monotone line search.

We make use of these RLM and PLM optimization routines and develop four schemes to solve contact dynamics problems. The first two schemes i.e. MLCP-FB-RLM and the MLCP-FB-PLM solvers make use of the MLCP FB optimization reformulation, where the optimization problem is solved through the regular LMA and projected LMA algorithms, respectively. The last two schemes i.e. MNCP-RLM and the MNCP-PLM solvers use the implicit MNCP formulation, where the optimization problem is once again solved using the regular LMA and projected LMA algorithms, respectively. We validate these routines and test the speed and accuracy of these schemes in the next section.

SIMULATION RESULTS

Sphere on a fixed horizontal plane

Consider a uniform sphere placed on a fixed horizontal plane in a uniform gravitational field as shown in Figure 1. The mass and radius of the sphere is unity whereas the initial configuration and initial velocity of the sphere is specified by $\theta_o = [\,0\;0\;1 \mid 1\;0\;0\;0\,]$ and $\dot{\theta}_o = [\,\mathrm{v}_{x_o}\;\mathrm{v}_{y_o}\;\mathrm{v}_{z_o} \mid \omega_{x_o}\;\omega_{y_o}\;\omega_{z_o}\,] = [\,2\;0\;0 \mid 0\;0\;0\,]$, respectively. The coefficient of friction is assumed to have a constant value of $\mu = 0.2$. The numerical solution is computed for $\mathrm{t} = 0.6s$ with a time step $\triangle_{\mathrm{t}} = 1$ ms. Furthermore, unlike [2], the frictional $\hat{f}$, orthogonal $\hat{o}$ and normal $\hat{\eta}$ directions of the contact frame have been chosen to lie exactly along the x, y and z directions of the inertial frame, respectively. For the MLCP set of solvers, we choose four direction vectors at right angles to each other in the contact tangent plane to discretize the friction cone.

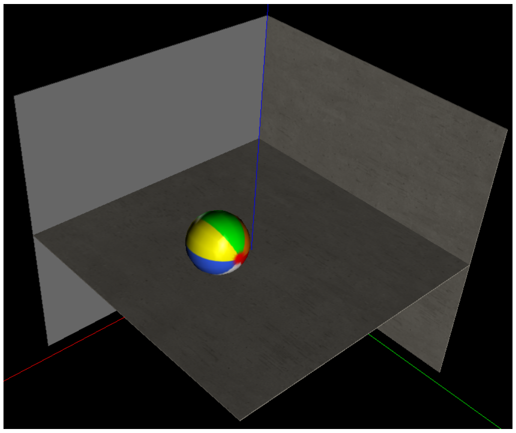
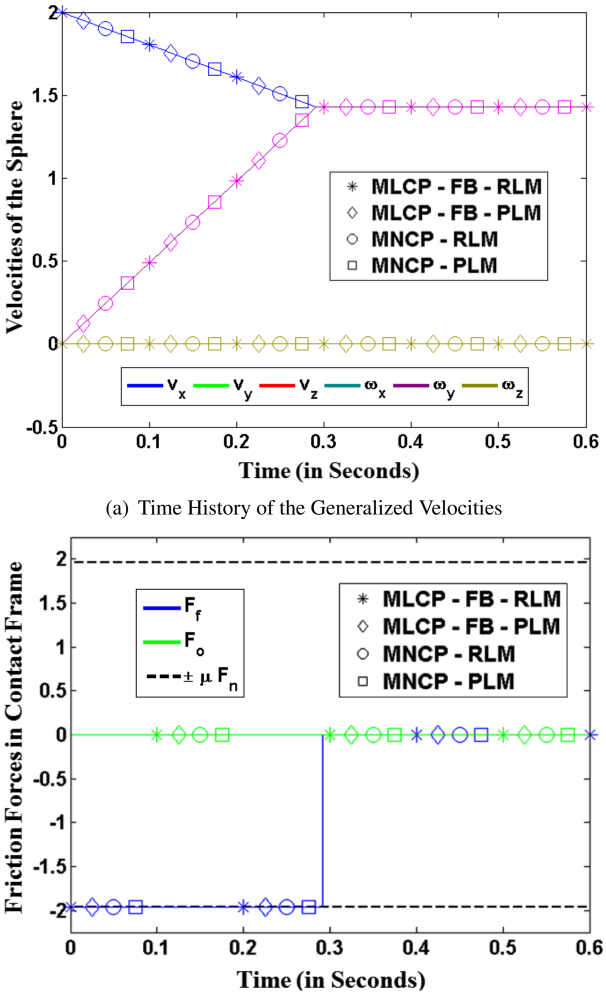


FIGURE 1. Sphere moving on a plane surface (RGB colors correspond to XYZ axes)

The sphere example for the initial conditions described above has been analytically solved by Trinkle [2]. For this initial condition set, the motion of the sphere is limited to sliding and rolling along the x-axis. The existence of the analytical solution allows us to use the sphere test case to validate the schemes presented in this paper. We do this validation in Example 1. To further study the effect of the alignment of the direction vectors with the motion of the sphere in the MLCP solvers case, we consider two examples. In Example 1, one of the direction vector is aligned perfectly with the motion of the sphere whereas in Example 2, the direction vectors are oriented at 45 degrees to the motion of the sphere.

Example 1: For the MLCP solvers, if one of the direction vectors in the contact tangent space is chosen to lie exactly along the friction force direction (-x direction in this example as the sphere moves only along the +x direction), then the effect of the friction cone discretization disappears and we expect to see our simulations match the analytically predicted solution. Figures 2(a) and 2(b) show time history plots of the velocities of the sphere and the friction forces in contact coordinates, respectively, computed using the MLCP–FB–RLM, MLCP–FB–PLM, MNCP–RLM and the MNCP–PLM solvers. From the figures, we observe that the sphere initially slides in the x-direction until $\mathrm{t}_{sr} = \frac{2\mathrm{v}_{x_o}}{7\mu g} \approx 0.291$s while gradually losing translational velocity in the x-direction and gaining angular velocity in the



(a) Time History of the Generalized Velocities

(b) Time History of Friction Forces in Contact Coordinates

FIGURE 2. Motion data for the sphere with friction cone direction vectors aligned with the tangential contact velocity.

y-direction during this time (see Fig.2(a)). After $t_{sr} = 0.291$s, the sphere starts rolling with a constant linear velocity in the x-direction (v_x) and constant angular velocity in the y-direction of $\omega_y = \frac{5v_{x_o}}{7} \approx 1.429$ units. The remaining velocity components are all zero (see Fig.2(a)). Sliding and rolling phases can be confirmed by the fact that in Fig.2(b), we observe that $\mu F_n = \|F_f,\ F_o\|$ until $t_{sr} = 0.291$s, after which $\mu F_n > \|F_f,\ F_o\|$ when the sphere starts rolling. Further, from Fig.2(b), we observe that the friction force acts in the $-\hat{f}$ (or $-x$) direction alone with $F_o \equiv 0$ throughout the simulation. Thus, as can be inferred from the figures, the solutions of both the MLCP as well as the MNCP set of solvers closely match the predicted analytical solution described in the reference [2], thus validating their solutions.

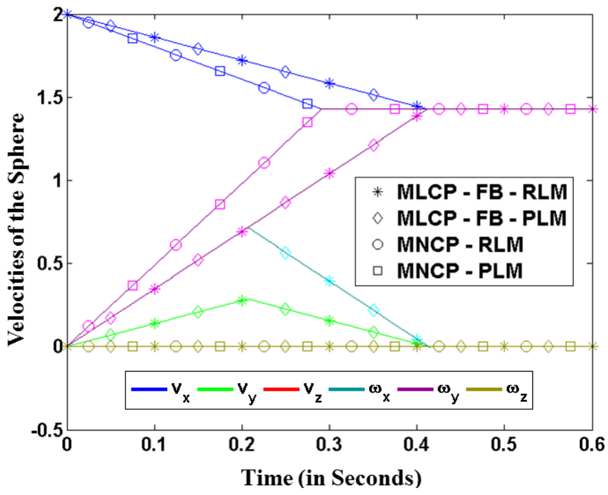


FIGURE 3. Velocities of the sphere when the friction cone direction vectors are not aligned with the the tangential contact velocity.

Example 2: Ideally, the choice of the friction direction vectors in the contact tangent plane should have no bearing on our final solution. However, they do so in the MLCP case. The degree to which the friction direction vectors in the contact tangent plane are aligned with the tangential frictional impulse affects the accuracy of the MLCP solution. This can be seen in the present example where the direction vectors are oriented at an angle of 45 degrees to the motion of the sphere (and hence the tangential friction impulse vector). The rest of the set up of the problem is chosen to be exactly the same as the previous example. From Fig. 3, the MLCP solvers predict that in addition to v_x and ω_y as seen in Example 1, the sphere also possesses non-zero translational velocity v_y and angular velocity ω_x, when in reality both of these quantities should be zero for all time. Furthermore, the sphere continues to slide until $t \approx 0.4$s (much longer than the previous case), after which it starts to roll. These results depart considerably from the analytically predicted motion of the sphere seen in Example 1. Thus, any misalignment between the friction direction vectors and the tangential friction impulse vector produces erroneous results in the MLCP case, which is one of the major drawbacks of the approach. To avoid these errors in the MLCP case, either the number of direction vectors has to be increased such that the polygon better approximates the circle (not desired as it increases the number of ancillary variables) or the direction vectors have to be chosen such that they align exactly with the tangential frictional impulse at each integration time step of the simulation (which lies beyond our control). In practice, the best one can do to minimize these errors is to align one of the the direction vectors at the current integration time step to coincide with the opposite of the tangential contact velocity vector at that time step. The MNCP solvers, on the other hand, have no such concept of direction vectors and they continue to give us the

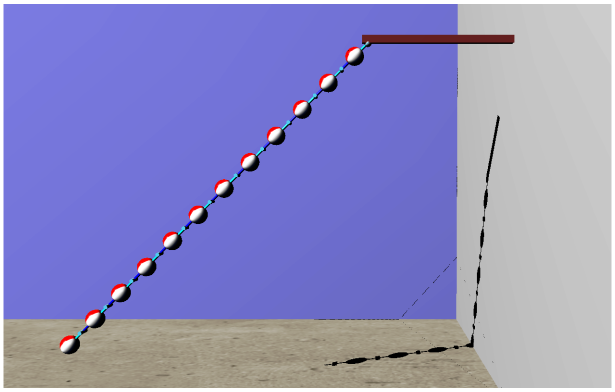


FIGURE 4. Initial set up of the twelve-link pendulum

correct solutions as can be seen in Fig. 3.

Twelve-link Pendulum

To evaluate the speed and accuracy of the complementarity schemes presented in this paper, we consider the more complex example of a twelve-link swinging pendulum colliding with itself and the surrounding environment as shown in Figure 4. The pendulum consists of twelve identical 1kg spherical masses connected together with pin hinges. The environment consists of a floor of thickness 0.2m and a colliding wall on the right side located 4m away. The overall length of the pendulum is 12m with each of the spheres having a diameter of 0.5m. The pendulum is located at a height of 10m above the ground. The coefficient of friction is assumed to be 0.5 for all surfaces and to simulate inelastic collisions the coefficient of restitution is assumed to be 0.7. The open source software Bullet [23] is used for collision detection. The pendulum makes an initial angle of $\pi/4$ radians with the vertical and has initial angular velocity of $\omega_x = 1$ radians/s. Uniform gravitational acceleration of 9.81m/s^2 is assumed. For the MLCP solvers, the number of direction vectors in the contact tangent space is chosen to be four.At each time step, one of the direction vectors at the current time step is aligned with the opposite of the tangential contact velocity vector at that time step. The simulations are run for a time span of 20s with a time step of 1 ms. As the pendulum swings from left to right, it collides with the ground, bounces off of the ground, collides with the wall on the right, swings back and collides with the ground once again. Over the course of the simulation, multiple links are at times in collision with ground, the wall and with each other.

Minimal coordinate operational space formulation is used to model the dynamics of this system. Since minimal coordinates are used, the inter-link constraints are automatically eliminated. No loop-closure bilateral constraints exist for the multilink pendulum. Hence, the only constraints acting on the pendulum system are unilateral contact constraints. The size of the complementarity problem is defined by the number of contacts, with the size of the MNCP problem being smaller compared to the MLCP problem. Using the methodologies presented in this paper, the ensuing complementarity problems are recast as optimization problems, which are further solved using the RLM and PLM optimization routines. Figs. 5 and 7 show time history plots of the last link's height and the linear z-velocity (v_z), respectively, computed using the MLCP-PATH, MLCP-FB-RLM,

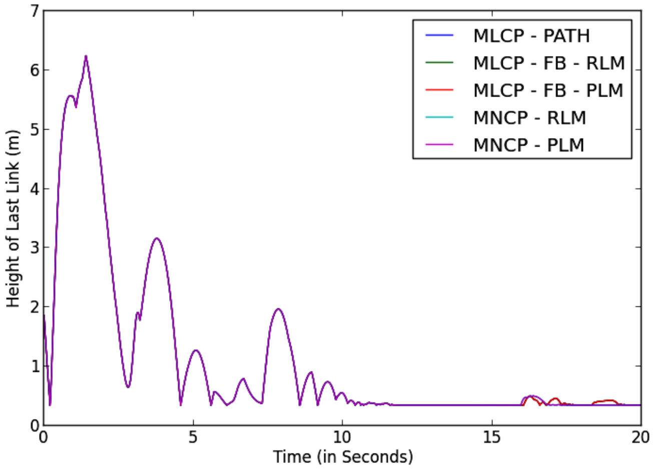


FIGURE 5. A time history plot of the height of the last link of the pendulum computed using the five different solvers. The plots of the five solvers are superposed on top of one another with minor differences showing up at the end of the 20s simulation. The height of the last link is measured from the origin, which is located at the mid-section of the floor of thickness 0.2m.

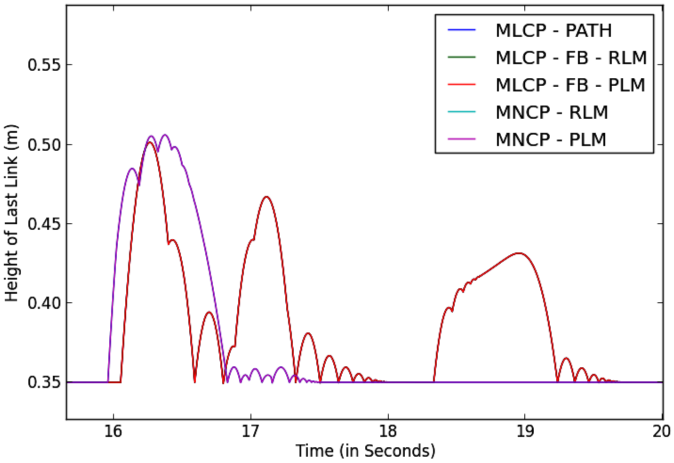


FIGURE 6. A zoomed-in plot depicting the height of the last link of the pendulum at the end of the 20s simulation. The solvers show a grouping behavior, with the MLCP set of solvers showing a similar solution and MNCP set of solvers showing a similar solution. The height of the last link is measured from the origin, which is located at the mid-section of the floor of thickness 0.2m.

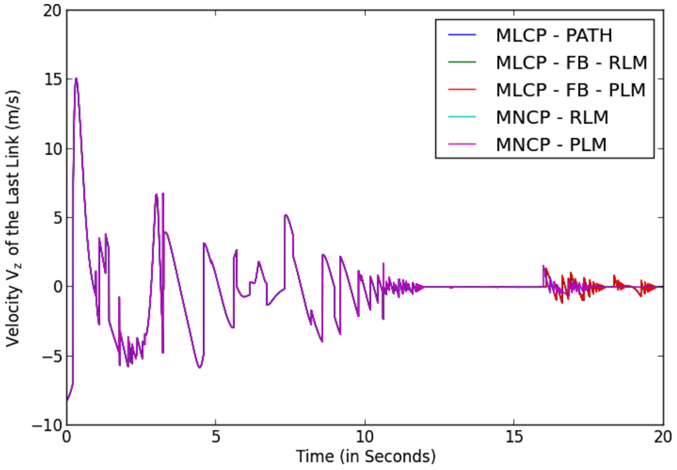


FIGURE 7. A time history plot of the z-component of the linear velocity (v_z) of the last link of the pendulum computed using the five different solvers. The plots of the five solvers are superposed on top of one another with minor differences showing up at the end of the 20s simulation.

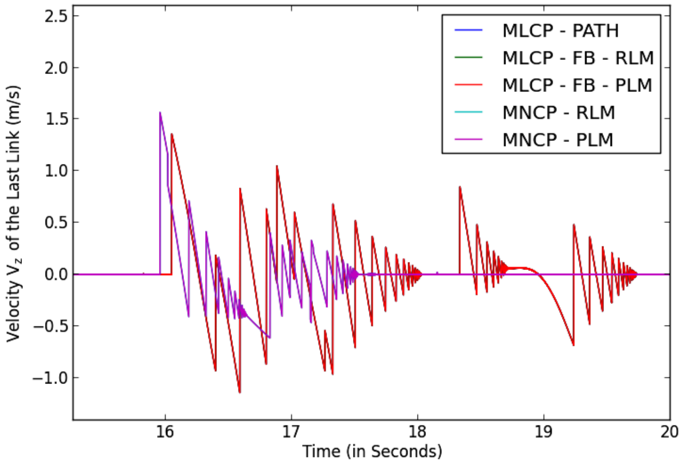


FIGURE 8. A zoomed-in plot depicting v_z of the last link of the pendulum at the end of the 20s simulation. The solvers show a grouping behavior, with the MLCP set of solvers showing a similar solution and MNCP set of solvers showing a similar solution.

MLCP-FB-PLM, MNCP-RLM and MNCP-PLM solvers. Figs. 6 and 8 depict zoomed-in plots of Figs. 5 and 7, respectively, at the end of the 20s simulation. As can be seen from figures, the plots show a close match between the solutions of all the five solvers with minor differences appearing at the end of the 20s simulation. The computation times of the solvers appear to be evenly matched with the MLCP-PATH, MLCP-FB-RLM, MLCP-FB-PLM, MNCP-RLM, MNCP-PLM solvers taking 23.92s, 22.78s, 24.24s, 22.85s, and 22.82s, respectively, to complete the simulation.

Conclusions

In this paper, the contact dynamics of multibody systems is approached using the minimal coordinate operational space formulation. The dynamics is cast as a complementarity problem, which is further reformulated as an unconstrained optimization problem. Levenberg-Marquardt type algorithms are employed to solve these optimization problems. Employing these techniques, three linear (MLCP–FB–RLM, MLCP–FB–PLM, MLCP–PATH) and two nonlinear (MNCP–RLM, MNCP–PLM) complementarity schemes have been developed for solving general contact dynamics problems. These solvers have been validated using the example of a sphere moving on a horizontal fixed plane, for which analytical solutions are available in [2].

Furthermore, an example of a twelve-link pendulum interacting with its surrounding environment is used to evaluate the performance of the different solvers. We found that all of the schemes had similar speed and accuracy performance and closely matched that of the MLCP–PATH algorithm. Elaborating on this observation, recall that the size of the complementarity problem itself is small for minimal coordinate approaches, and the overall solution cost is dominated by the cost of setting up the complementarity problem, with the complementarity solver itself contributing little to the cost. This is in contrast with our experience with redundant coordinate approaches where the complementarity problem size is large and the cost of solving the complementarity problem dominates the overall cost [4]. Our overall observation is that the cost savings from reducing the size of the complementarity problem using the minimal coordinate approach far outweigh the additional costs of setting up the complementarity problem, and have the additional benefit that the speed and accuracy of the solution is relatively insensitive to the specific choice of the complementarity solver.

We have also observed that accurate solutions can be obtained with the linear complementarity solvers by aligning one of the friction cone direction vectors in the contact tangent plane (at the current time integration step) to coincide with the opposite of the tangential contact velocity vector (at that time step). Using such an alignment procedure it is possible to use linear, rather than nonlinear complementarity solvers, with just a small number of friction cone direction vectors without incurring large accuracy or cost penalties.

Future work will be focused on testing these complementarity schemes on wider set of problems while simultaneously improving the speed, accuracy and robustness of these complementarity schemes.

Acknowledgments

The research described in this paper was performed at the Jet Propulsion Laboratory (JPL), California Institute of Technology, under a contract with the National Aeronautics and Space Administration.[1]

REFERENCES

[1] The PATH Solver, 2012. URL http://pages.cs.wisc.edu/ ferris/path.html.

[2] Trinkle, J. C., 2003. "Formulation of Multibody Dynamics as Complementarity Problems". In ASME International Design Engineering Technical Conference.

[3] Todorov, E., Erez, T., and Tassa, Y., 2012. "MuJoCo: A physics engine for model-based control". In 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 5026–5033.

[4] Jain, A., Crean, C., Kuo, C., von Bremen, H., and Myint, S., 2012. "Minimal Coordinate Formulation of Contact Dynamics in Operational Space". In Robotics Science and Systems.

[5] Anitescu, M., and Potra, F. A., 1997. "Formulating dynamic multi-rigid-body contact problems with friction as solvable linear complementarity problems". *Nonlinear Dynamics,* **14**(3), pp. 231–247.

[6] Jain, A., 2013. "Minimal Coordinates Formulation of Contact Dynamics". In Multibody Dynamics 2013, ECCOMAS Thematic Conference.

[7] Todorov, E., 2010. "Implicit nonlinear complementarity: A new approach to contact dynamics". In 2010 IEEE International Conference on Robotics and Automation, no. 5, Ieee, pp. 2322–2329.

[8] Kanzow, C., and Petra, S., 2004. "On a semismooth least squares formulation of complementarity problems with gap reduction". *Optimization Methods and Software,* **19**(5), pp. 507–525.

[9] Kanzow, C., and Petra, S., 2005. "LMMCP: A Levenberg-Maquardt Type MATLAB Solver for Mixed Complementarity Problems".

[10] Kanzow, C., and Petra, S., 2005. *Projected Filter Trust Region Methods for a Semismooth Least Squares Formulation of Mixed Complementarity Problems*. Inst. of Applied Math. and Statistics.

[11] Jain, A., Crean, C., Kuo, C., and Quadrelli, M. B., 2012. "Efficient Constraint Modeling for Closed-Chain Dynamics". In The 2nd Joint International Conference on Multibody System Dynamics.

[12] Jain, A., 2011. *Robot and Multibody Dynamics: Analysis and Algorithms*. Springer.

[13] Lacoursiere, C., Lu, Y., Williams, J., and Trinkle, J.C., 2013. "Standard Interface for data analysis of solvers in multibody dynamics". In The 4th Canadian Conference on Nonlinear Solid Mechanics (CanCNSM 2013), McGill University, July 23-26, 2013, Montreal, Canada.

[14] Trinkle, J. C., and Pang, J., 1997. "Dynamic multi-rigid-body systems with concurrent distributed contacts". In IEEE International Conference on Robotics and Automation, pp. 2276–2281.

[15] Geiger, C., and Kanzow, C., 1996. "On the resolution of monotone complementarity problems". *Computational Optimization and Applications,* **5**(2), pp. 155–173.

[16] Sun, D., and Qi, L., 1999. "On NCP-functions". *Computational Optimization and Applications,* **13**(1-3), pp. 201–220.

[17] Fischer, A., 1995. "A Newton-type method for positive-semidefinite linear complementarity problems". *Journal of Optimization Theory and Applications,* **86**(3), pp. 585–608.

[18] Ferris, M. C., Kanzow, C., and Munson, T. S., 1999. "Feasible descent algorithms for mixed complementarity problems". *Mathematical Programming,* **86**(3), pp. 475–497.

[19] Drumwright, E., and Shell, D. A., 2009. "A Robust and Tractable Contact Model for Dynamic Robotic Simulation". In Proceedings of the 2009 ACM Symposium on Applied Computing (SAC '09), ACM, New York, NY, USA, pp. 1176–1180.

[20] Bazaraa, M. S., Sherali, H. D., and Shetty, C. M., 1993. *Nonlinear programming: theory and algorithms*. John Wiley & Sons.

[21] Grippo, L., Lampariello, F., and Lucidi, S., 1986. "A nonmonotone line search technique for Newton's method". *SIAM Journal on Numerical Analysis,* **23**(4), pp. 707–716.

[22] Grippo, L., Lampariello, F., and Lucidi, S., 1991. "A class of nonmonotone stabilization methods in unconstrained optimization". *Numerische Mathematik,* **59**(1), pp. 779–805.

[23] Bullet Physics Library, 2013. URL http://bulletphysics.org.

[1] ©2014 California Institute of Technology. Government sponsorship acknowledged.