CMC-Opt: Constraint Manifold with Corners for Inequality-Constrained Optimization

1st Yetong Zhang
College of Computing
Georgia Institute of Technology
Atlanta, USA
yetongz.gatech@gmail.com

2nd Frank Dellaert
College of Computing
Georgia Institute of Technology
Atlanta, USA
frank.dellaert@cc.gatech.edu

***Abstract*—We introduce a manifold-based framework for addressing optimization problems with equality and inequality constraints found in robotics. Our approach transforms the original problem into an unconstrained optimization problem directly on the constrained state space. To achieve this, we introduce "constraint manifolds with corners" to represent the state space satisfying mixed nonlinear equality and inequality constraints. We further extend manifold optimization algorithms to operate on this new topological structure. We demonstrate the power and robustness of our framework in the context of a large-scale kinodynamic planning problem, successfully generating dynamically feasible trajectories where standard methods fail.**

***Index Terms*—constrained optimization, manifold optimization**

I. Introduction

Modern robotic systems are inherently governed by complex hard constraints. They arise from diverse sources: rigid-body kinodynamic constraints [14], contact constraints [16] (e.g., Coulomb friction, complementarity constraints[20]), hardware limitations (e.g., joint angle, speed, and torque limits), and task-specific requirements (e.g., collision avoidance). While discrete-valued constraints also exist in scenarios like footstep planning [12], they fall outside the scope of this paper, which focuses exclusively on continuous state spaces.

Standard constrained optimization methods struggle to balance cost reduction with strict constraint satisfaction in highly nonlinear robotic state spaces. Methods that incorporate constraints into the cost function via penalties or Augmented Lagrangians [10, 27, 21, 17, 2, 18, 24] must scale weights on penalty terms toward infinity to enforce exact feasibility, which can lead to numerical conditioning issues or slow convergence. For algorithms that attempt to satisfy optimality conditions, like sequential quadratic programming (SQP) and primal-dual interior point methods [6, 23, 5, 3], devising a merit function for global convergence is a nontrivial task.

A compelling alternative to constrained optimization methods is *manifold optimization*, offering lower complexity and better numerical properties [1, 4, 15]. Manifolds are devised to represent feasible state spaces defined by equality constraints. Optimization is then directly conducted on these manifolds, which constitute a lower-dimensional state space. [26] demonstrated the applicability of manifold optimization in solving equality-constrained nonlinear problems.

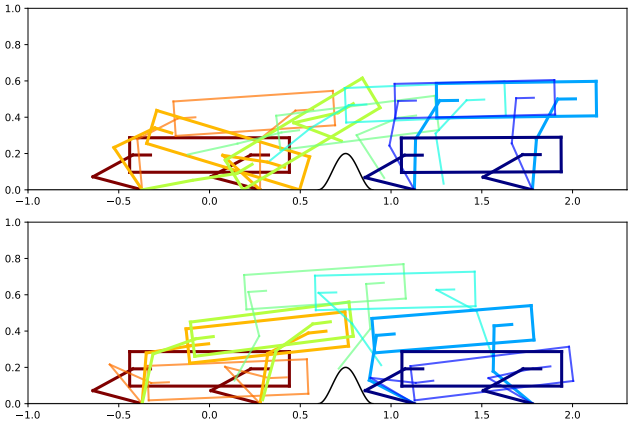

Fig. 1. The optimized quadruped jumping trajectory with the penalty method (top) and the CMC-Opt method (bottom). The trajectory produced by the CMC-Opt method exhibits a smooth and natural jumping motion. In contrast, the trajectory generated by the penalty method renders dynamically infeasible due to significant "collocation leakage" where physics are slightly violated to minimize torque costs.

However, the standard manifold theory breaks down when introduced with inequality constraints. The feasible state spaces contain sharp "boundaries" and "corners" that are not locally homeomorphic to the Euclidean space, which breaks the smoothness assumptions used in manifold optimization.

In this paper, we extend the constraint manifold optimization framework [26] to operate on state spaces defined by both nonlinear equality and inequality constraints. We make the following core contributions:

- We formulate *constraint manifolds with corners* (CMC) to represent state spaces governed by mixed nonlinear equality and inequality constraints, and provide their local geometric parameterization.
- We present a general framework that transforms the standard constrained optimization problem into an unconstrained problem on the constrained state space. This is achieved by leveraging factor graphs to isolate constraint-connected components and turn them into CMCs.
- We extend manifold optimization techniques to operate on CMCs, adapting Riemannian gradient descent to handle sharp manifold boundaries and corners.

II. Problem Formulation

We consider a general nonlinear constrained optimization problem (1), and model it using a factor graph framework [11, 8]. The continuous state space is defined by variable nodes $X = \{x_1, \cdots, x_{n_\mathrm{v}}\} \in \mathbb{R}^N$. The three types of factor nodes encode smooth objective functions $f_j(X_{I_j^\mathrm{f}})$, nonlinear equality constraints $h_k(X_{I_k^\mathrm{h}}) = 0$, and nonlinear inequality constraints $g_\ell(X_{I_\ell^\mathrm{g}}) \geq 0$, where I_j^f, I_k^h, I_ℓ^g represent the multi-index sets indicating the variables involved in the cost function or constraint expression. Traditional solvers attempt to solve (1) directly in $\mathbb{R}^N$.

$$X^\star = \arg\min_{X \in \mathbb{R}^N} \sum_{j=1}^{n_\mathrm{f}} f_j(X_{I_j^\mathrm{f}}) \tag{1a}$$

$$\text{s.t.} \quad h_k(X_{I_k^\mathrm{h}}) = 0 \quad \text{for } k = 1, \ldots, n_\mathrm{h} \tag{1b}$$

$$g_\ell(X_{I_\ell^\mathrm{g}}) \geq 0 \quad \text{for } \ell = 1, \ldots, n_\mathrm{g} \tag{1c}$$

Our approach aims to fold the constraints (1b) and (1c) into the geometry of the search space. With the factor graph representation, we can easily identify sets of variables connected by local constraints that involve only variables within the same set, as demonstrated in Figure 2. Our objective is to formulate the feasible state-space of such *constraint-connected components* and formulate optimization algorithms on them.

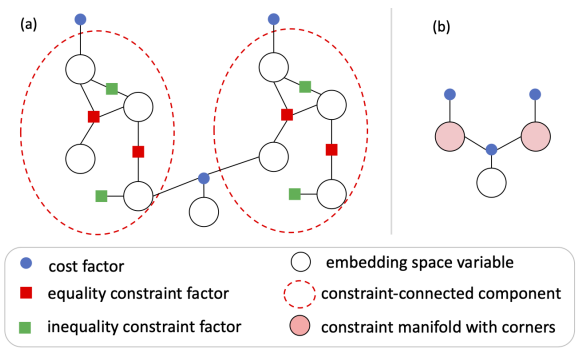

Fig. 2. (a) Factor graph representation of a constrained optimization problem. (b) Factor graph representation of the transformed unconstrained optimization problem on constraint manifolds with corners.

III. Constraint Manifold with Corners

We consider the state space constrained by both equality and inequality constraints, embedded in $\mathbb{R}^N$. Let $h : \mathbb{R}^N \to \mathbb{R}^{n_\mathrm{h}}$ and $g : \mathbb{R}^N \to \mathbb{R}^{n_\mathrm{g}}$ be functions representing the complete set of equality and inequality constraints, respectively, the constrained state space $\mathcal{M}$ is then expressed as (2).

$$\mathcal{M} = \left\{ X \in \mathbb{R}^N : h(X) = 0,\ g(X) \geq 0 \right\} \tag{2}$$

The constrained state space $\mathcal{M}$ exhibits "boundaries" and "corners" [1] where one or more inequality constraints are active, thus termed a *constraint manifold with corners* (CMC).

[1]In differential geometry, boundaries typically refer to the space where only one inequality constraint is active; corners refer to the state space where one or more inequality constraints are active.

At a point $X \in \mathcal{M}$, we classify the inequality constraints into active constraints $g^\mathcal{A} = g_{I_X^\mathcal{A}}$ (3) and inactive constraints $g^\mathcal{I} = g_{I_X^\mathcal{I}}$ (4) based on the evaluation of the constraint function $g(X)$. X is termed an *interior point* X^Int if no inequality constraint is active; otherwise, it is termed a *corner point* X^∂.

$$I_X^\mathcal{A} = \left\{ \ell \in \{1, \ldots, n_\mathrm{g}\} : g_\ell(X_{I_\ell^\mathrm{g}}) = 0 \right\} \tag{3}$$

$$I_X^\mathcal{I} = \left\{ \ell \in \{1, \ldots, n_\mathrm{g}\} : g_\ell(X_{I_\ell^\mathrm{g}}) > 0 \right\} \tag{4}$$

We assume that at any point $X \in \mathcal{M}$, all the active constraint functions $h(X)$, $g^\mathcal{A}(X)$ and the cost functions $f(X)$ are continuously differentiable. In addition, the Jacobian matrix of the equality and the active inequality constraint functions are always full rank as in (5).

$$\forall X \in \mathcal{M} : \quad \mathrm{rank}\left(\begin{bmatrix} \nabla h(X)^\mathsf{T} \\ \nabla g^\mathcal{A}(X)^\mathsf{T} \end{bmatrix} \right) = n_\mathrm{h} + \left| I_X^\mathcal{A} \right| \tag{5}$$

At an interior point X^Int, the constrained state space $\mathcal{M}$ is locally identical to a constraint manifold $\overline{\mathcal{M}}$ solely defined by equality constraints (6) [26]. Hence, $\mathcal{M}$ is locally homeomorphic to $\mathbb{R}^n$ at X^Int, where $n = N - n_\mathrm{h}$. Thus, around X^Int, $\mathcal{M}$ can be locally parameterized with $\theta \in \mathbb{R}^n$.

$$\overline{\mathcal{M}} = \left\{ X \in \mathbb{R}^N : h(X) = 0 \right\} \tag{6}$$

At a corner point X^∂, moving in any direction v such that $\nabla g_\ell(X^\partial)^\mathsf{T} v < 0$ will violate the inequality constraint, therefore going outside $\mathcal{M}$. The constrained state space $\mathcal{M}$ is homeomorphic to the model space H_m^n (7), resembling a corner [13, 9], where $m = n - \dim I_{X^\partial}^\mathcal{A}$. Therefore, around X^∂, $\mathcal{M}$ can be locally parameterized with $\theta \in H_m^n$.

$$H_m^n = \{\theta \in \mathbb{R}^n : \theta_1, \ldots, \theta_{n-m} \geq 0\} \tag{7}$$

We further define the connected subset of $\mathcal{M}$ where a subset I of inequality constraints remain active as a "corner", which is locally identical to the constraint manifold $\overline{\mathcal{M}}^I$ in (8).

$$\overline{\mathcal{M}}^I = \left\{ X \in \mathbb{R}^N : h(X) = 0, g_I(X) = 0 \right\} \tag{8}$$

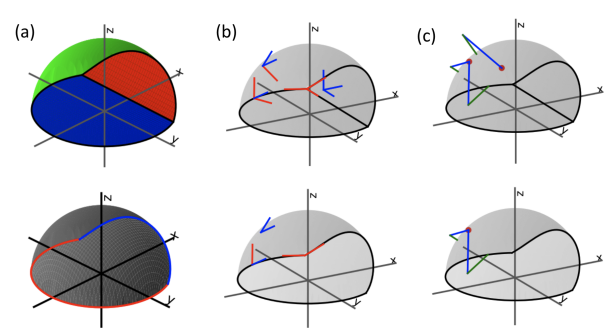

Fig. 3. (a) Example constraint manifolds with corners. Top is defined by $1 - x^2 - y^2 - z^2 \geq 0$, $z - x^3 + 0.1 \geq 0$, $z \geq 0$; bottom is defined by $1 - x^2 - y^2 - z^2 = 0$, $z - x^3 + 0.1 \geq 0$, $z \geq 0$. The "corners" are marked with colors. (b) The tangent spaces at various points on the CMCs, represent as a set of basis vectors. Blue basis vectors indicate the freedom to move in both directions, while the red ones are limited to only move in one direction. (c) Examples of retraction operation (green) given a tangent vector (blue).

A. Tangent Space

In order to run gradient-based optimization on CMCs, we need to compute the differential of functions on CMCs, which requires the definition of a "tangent space" that captures the local smoothness structure at any point on the CMC.

We define the tangent space of $\mathcal{M}$ following the tangent space definition of a regular manifold [4]. The tangent space of $\mathcal{M}$ at a point $X \in \mathcal{M}$ is the set of velocities of all smooth curves C_X on $\mathcal{M}$ starting at X (9), which can be derived as the set of vectors satisfying the linearized equality and active inequality constraints (10). Notably, the tangent space is a linear space at interior points, and a convex cone for corner points, as depicted in Figure 3b.

$$C_X = \{c : [0,1) \to \mathcal{M} \text{ is smooth}, c(0) = X\} \tag{9}$$

$$\begin{aligned} \mathrm{T}_X\mathcal{M} &= \{c'(0) : c \in C_X\} \\ &= \left\{v \in \mathbb{R}^N : \nabla h(X)^\mathsf{T} v = 0, \nabla g^{\mathcal{A}}(X)^\mathsf{T} v \geq 0\right\} \end{aligned} \tag{10}$$

The tangent space $\mathrm{T}_X\mathcal{M}$ is a subset of the tangent space of the constraint manifold $\overline{\mathcal{M}}$ defined by only the equality constraints [26] as (11). Therefore, it can be parameterized with a set of basis vectors written in matrix form as B. Notice that for a specially devised basis, we can have $\Omega = H^n_m$.

$$\begin{aligned} \mathrm{T}_X\mathcal{M} &= \left\{v \in \mathrm{T}_X\overline{\mathcal{M}} : \nabla g^{\mathcal{A}}(X)^\mathsf{T} v \geq 0\right\} \\ &= \{B \cdot \theta : \theta \in \Omega\} \end{aligned} \tag{11}$$

$$\Omega = \left\{\theta \in \mathbb{R}^n : \nabla g^{\mathcal{A}}(X)^\mathsf{T} B\theta \geq 0\right\} \tag{12}$$

B. Differential of Function on CMCs

We now derive the differential of functions on CMCs. Let's consider a real-valued function defined on a CMC $\mathcal{F} : \mathcal{M} \to \mathbb{R}$; the differential of the function $\mathcal{F}$ at $X \in \mathcal{M}$ is represented as a map $\mathrm{d}\mathcal{F} : \mathrm{T}_X\mathcal{M} \to \mathbb{R}$, indicating the rate of change of the function value when moving from X in the direction of a tangent vector $v \in \mathrm{T}_X\mathcal{M}$, as defined in Equation (13).

$$\mathrm{D}\mathcal{F}(X)[v] = \frac{\mathrm{d}}{\mathrm{d}t}\mathcal{F}\left(\mathrm{R}_X\left(t \cdot v\right)\right)\Big|_{t=0} \tag{13}$$

Using basis B of the tangent space, the differential can be elegantly expressed in matrix form (14). Here, $f : \mathbb{R}^N \to \mathbb{R}$ denotes a smooth extension of $\mathcal{F}$ to the ambient space, which corresponds to the cost function in the original constrained optimization problem. The differential of $\mathcal{F}$ in the direction of any tangent vector $v = B\theta \in \mathrm{T}_X\mathcal{M}$ adheres strictly to the property (15).

$$\nabla\mathcal{F}(X) = B^\mathsf{T}\nabla f(X) \tag{14}$$

$$\mathrm{D}\mathcal{F}(X)[B\theta] = \nabla\mathcal{F}(X)^\mathsf{T} \cdot \theta \tag{15}$$

The gradient of function $\mathcal{F}$ seeks a feasible direction in the tangent space that can most rapidly decrease the function value. It can be computed as (16).

$$\begin{aligned} \theta^\star &= \arg\min_{\theta \in \Omega} \|\nabla\mathcal{F}(X) - \theta\|^2 \\ \mathrm{grad}\mathcal{F}(X) &= B \cdot \theta^\star \end{aligned} \tag{16}$$

C. Retraction

We formulate the retraction operation to ensure staying on the CMC after each update step in optimization. Similar to the definition of retraction operation of a standard manifold [4], we define the retraction on a manifold with corners at $X \in \mathcal{M} \hookrightarrow \mathbb{R}^N$ as a smooth map $R_X : \mathrm{T}_X\mathcal{M} \to \mathcal{M}$ such that the zero tangent vector maps to X, and its differential at the zero tangent vector is the identity map (17).

$$\begin{aligned} \mathrm{R}_X(0) &= X \\ \mathrm{DR}_X(0)[v] &= v \end{aligned} \tag{17}$$

We define the Euclidean metric projection (Fig. 3c) as a retraction operation, which solves an optimization subproblem (18).

$$\begin{aligned} \mathrm{R}_X(v) &= \arg\min_{Y \in \mathbb{R}^N} \|Y - (X + v)\|^2 \\ \text{s.t.} \quad & h(Y) = 0,\ g(Y) \geq 0 \end{aligned} \tag{18}$$

D. Optimization on CMCs

Finally, we extend Riemannian gradient descent [4] to work on CMCs with the induced metric from the embedding space. The algorithm (Alg. 1) minimizes a real-valued function on CMCs $\mathcal{F} : \mathcal{M}_1 \times \ldots \times \mathcal{M}_n \to \mathbb{R}$, with initial values $\{X_1^{(0)} \in \mathcal{M}_1, \ldots, X_n^{(0)} \in \mathcal{M}_n\}$. In the k-th iteration of the algorithm, we first compute the differential of the cost function $\nabla\mathcal{F}(X^{(k)})$, then compute an approximation to the overall gradient by solving the QP subproblem (16) for each manifold. Finally, with a chosen step size α, we perform the retraction with (18) to perform the update and ensures staying on the CMCs. Fig. 4 provides an illustrate toy example.

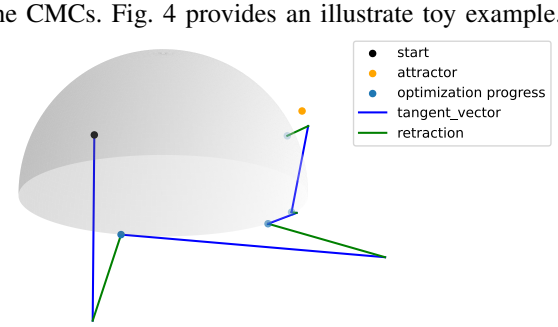

Fig. 4. Example of Riemannian gradient descent applied on a "half sphere" manifold defined by $x^2 + y^2 + z^2 = 1$ and $z \geq 0$. We consider a point constrained on the "half sphere" manifold subject to gravity force pointing downwards and attraction force from a remote point. The optimization minimizes the potential energy of the point.

Algorithm 1: Riemannian gradient descent on CMC

Given $X^{(0)}$, initialize $k = 0$.
while *not converged* **do**
  a). Compute the differential $\nabla\mathcal{F}(X^{(k)})$ with (14).
  b). **for** *each constraint manifold* $\mathcal{M}_i$ **do**
    Compute feasible descent direction $\theta_i^\star$ with (16).
  c). Choose step size α.
  d). **for** *each constraint manifold* $\mathcal{M}_i$ **do**
    Compute tangent vector $v_i^{(k)} = \alpha B_i \theta_i^*$.
    Apply retraction $X_i^{(k+1)} = \mathrm{R}_{X_i^{(k)}}(v_i^{(k)})$ with (18).
  e). $k = k + 1$

IV. Experiments and Results

A. Scenario Setup

We evaluate the robustness of CMC-Opt on a complex kinodynamic planning problem: a 13-link quadruped (a torso and four 3-link legs) leaping over an obstacle. The trajectory is discretized into 70 time steps and divided into four distinct phases using hybrid collocation [19]: (1) Stance (20 steps) with all four feet grounded, (2) Propulsion (10 steps) where the front legs lift and back legs drive, (3) Flight (20 steps) where all feet are airborne to clear the obstacle, and (4) Landing (20 steps) where all feet land simultaneously and the quadruped returns to its nominal stance pose.

To rigorously test our framework, we adopt the Newton-Euler formulation [14, 25] for kinodynamics. While highly expressive and general, this approach introduces a massive number of variables and constraints. For every link, the state variables include spatial poses, twists, and accelerations. For every joint, we explicitly model angles, angular velocities, angular accelerations, actuation torques, and transmitted wrenches. Furthermore, we introduce additional variables to explicitly represent contact forces at the feet.

Equality constraints include kinematic constraints (involving link poses, twists, and accelerations) at all joints, alongside Newton-Euler rigid-body dynamics for links. During ground contact phases, we also include equality constraints to ensure the contact points remain stationary. Inequality constraints include collision avoidance, Coulomb friction at the contact points, joint angle limits, and actuator torques limits.

Finally, the objective is to minimize a combined weighted sum of squared penalties: control effort (actuator torques), trajectory jerk, state targeting errors (which penalize deviations from desired goal state), and a collocation cost. This final collocation term ensures that the discrete integration of positions and velocities strictly adheres to the trapezoidal rule across all time steps. To initialize the optimization solvers, we provide a naive, dynamically infeasible initial guess where all twists and accelerations are uniformly set to zero, resulting in a massive collocation violation for the algorithms to overcome.

B. Results

We benchmark CMC-Opt against four baselines: the penalty method and augmented Lagrangian method are implemented following [24]; the trust-region SQP is implemented following [18], and CM-Opt [26] (where inequality constraints are formulated as penalty terms). All methods are implemented using the GTSAM library [7], employing the Levenberg-Marquardt (LM) algorithm [18] to solve their respective optimization subproblems. For CMC-Opt, although Riemannian gradient descent serves as an illustrative framework in earlier sections, the practical implementation deploys a custom LM-like manifold optimization algorithm. This advanced solver leverages Gauss-Newton Hessian approximations and trust-region techniques to ensure robust, rapid convergence. The performance comparisons are summarized in Table I with four key metrics: state space dimensionality, computational time (in seconds), the constraint violation (in SI units) and cost evaluated at final optimized values.

TABLE I
Comparison of methods in quadruped planning problem.

Method	Dimension	Time(s)	Violation	Cost
Penalty	32,194	1.2e+02	1.3e+01	2.7e+05
AugL	32,194	5.9e+01	7.7e+00	3.1e+04
SQP	32,194	8.4e+01	1.0e-01	1.3e+06
CM-Opt	2,260	8.1e+01	4.2e-01	7.0e+05
CMC-Opt	**2,260**	6.3e+02	**0**	**709.78**

As Table I shows, CMC-Opt is the only method that drives constraint violations exactly to zero while achieving a significantly lower final objective cost. By projecting the problem onto the feasible state space, CMC-Opt drastically reduces the search space dimensionality from 32,194 to 2,260. As illustrated in Fig. 1, CMC-Opt generates a dynamically feasible and natural jumping trajectory. In contrast, the baseline methods fail to yield viable solutions, instead converging to results with extremely high collocation costs.

V. Discussion

A noticeable advantage of CMC-Opt over classical constrained optimization methods lies in the significantly reduced problem scale. This is accomplished through the computation of updates as vectors within the lower-dimensional tangent space of the CMCs. Furthermore, because the framework internally enforces constraints through CMC retractions, the CMC-Opt approach simplifies the optimization task to a single objective of minimizing the cost function.

Conceptually, the CMC-Opt approach functions as a bi-level optimization framework [22]. The upper-level addresses the core manifold optimization problem, while the lower-level manages manifold-based mathematical operations, including the construction of the tangent space basis and the execution of the retraction operation. In the trajectory planning example, the variables of the same time step are formulated as a CMC, since the kinodyanmic constraints only connect variables of the same time step. With this formulation, the higher-level optimization can focus on solving the trajectory optimization problem directly on the lower-dimensional manifolds that reflect the robot's inherent topological structure, while the lower-level optimization is in charge of kinodynamic feasibility.

CMC-Opt shows significant potential for application to robots with highly complex structures. For systems with contacts or closed-loop kinematics, where defining generalized coordinates is nontrivial, CMC-Opt permits use of the general Euler-Newton formulation, delegating kinodynamic constraints entirely to the lower-level optimization.

CMC-Opt is yet unsuitable for managing cross-time-step constraints, because they cause the constrained-connected component to grow prohibitively large, rendering retraction subproblems intractable to solve. Such constraints are therefore relaxed as penalty terms in the cost function.

References

[1] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, Princeton, NJ, 2008. ISBN 978-0-691-13298-3.

[2] Dimitri P Bertsekas. Nonlinear programming. *Journal of the Operational Research Society*, 48(3):334–334, 1997.

[3] John T Betts. *Practical methods for optimal control and estimation using nonlinear programming*. SIAM, 2010.

[4] Nicolas Boumal. An introduction to optimization on smooth manifolds. Available online, Aug 2020. URL http://www.nicolasboumal.net/book.

[5] Richard H Byrd, Frank E Curtis, and Jorge Nocedal. An inexact sqp method for equality constrained optimization. *SIAM Journal on Optimization*, 19(1):351–369, 2008.

[6] Alexander Cunningham, Manohar Paluri, and Frank Dellaert. Ddf-sam: Fully distributed slam using constrained factor graphs. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3025–3030. IEEE, 2010.

[7] F. Dellaert. Factor graphs and GTSAM: A hands-on introduction. Technical report, Georgia Institute of Technology, 2012.

[8] F. Dellaert and M. Kaess. *Factor Graphs for Robot Perception*. Now Publishers Inc., August 2017.

[9] Dominic Joyce. On manifolds with corners. *arXiv preprint arXiv:0910.3518*, 2009.

[10] Mrinal Kalakrishnan, Sachin Chitta, Evangelos Theodorou, Peter Pastor, and Stefan Schaal. Stomp: Stochastic trajectory optimization for motion planning. In *2011 IEEE international conference on robotics and automation*, pages 4569–4574. IEEE, 2011.

[11] F. R. Kschischang, B. J. Frey, and H. . Loeliger. Factor graphs and the sum-product algorithm. *IEEE Transactions on Information Theory*, 47(2):498–519, 2001. doi: 10.1109/18.910572.

[12] Scott Kuindersma, Robin Deits, Maurice Fallon, Andrés Valenzuela, Hongkai Dai, Frank Permenter, Twan Koolen, Pat Marion, and Russ Tedrake. Optimization-based locomotion planning, estimation, and control design for the atlas humanoid robot. *Autonomous robots*, 40(3):429–455, 2016.

[13] John M Lee and John M Lee. *Smooth manifolds*. Springer, 2012.

[14] Kevin M. Lynch and Frank C. Park. *Modern Robotics: Mechanics, Planning, and Control*. Cambridge University Press, USA, 1st edition, 2017. ISBN 1107156300.

[15] Aaron Mahler, Tyrus Berry, Tom Stephens, Harbir Antil, Michael Merritt, Jeanie Schreiber, and Ioannis Kevrekidis. On-manifold projected gradient descent. *Frontiers in Computer Science*, 6:1274181, 2024.

[16] Carlos Mastalli, Rohan Budhiraja, Wolfgang Merkt, Guilhem Saurel, Bilal Hammoud, Maximilien Naveau, Justin Carpentier, Ludovic Righetti, Sethu Vijayakumar, and Nicolas Mansard. Crocoddyl: An efficient and versatile framework for multi-contact optimal control. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2536–2542. IEEE, 2020.

[17] Mustafa Mukadam, Jing Dong, Xinyan Yan, Frank Dellaert, and Byron Boots. Continuous-time gaussian process motion planning via probabilistic inference. *The International Journal of Robotics Research*, 37(11):1319–1340, 2018.

[18] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, New York, NY, USA, second edition, 2006.

[19] Diego Pardo, Michael Neunert, Alexander Winkler, Ruben Grandia, and Jonas Buchli. Hybrid direct collocation and control in the constraint-consistent subspace for dynamic legged robot locomotion. In *Robotics: Science and Systems*, 07 2017. doi: 10.15607/RSS.2017.XIII.042.

[20] Michael Posa, Cecilia Cantu, and Russ Tedrake. A direct method for trajectory optimization of rigid bodies through contact. *The International Journal of Robotics Research*, 33(1):69–81, 2014.

[21] John Schulman, Yan Duan, Jonathan Ho, Alex Lee, Ibrahim Awwal, Henry Bradlow, Jia Pan, Sachin Patil, Ken Goldberg, and Pieter Abbeel. Motion planning with sequential convex optimization and convex collision checking. *The International Journal of Robotics Research*, 33(9):1251–1270, 2014.

[22] Ankur Sinha, Pekka Malo, and Kalyanmoy Deb. A review on bilevel optimization: From classical to evolutionary approaches and applications. *IEEE transactions on evolutionary computation*, 22(2):276–295, 2017.

[23] Duy-Nguyen Ta, Marin Kobilarov, and Frank Dellaert. A factor graph approach to estimation and model predictive control on unmanned aerial vehicles. In *2014 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 181–188. IEEE, 2014.

[24] Lieven Vandenberghe. Constrained nonlinear least squares. http://www.seas.ucla.edu/~vandenbe/133B/lectures/nllseq.pdf, 2020.

[25] Mandy Xie, Alejandro Escontrela, and Frank Dellaert. A factor-graph approach for optimization problems with dynamics constraints. *arXiv preprint arXiv:2011.06194*, 2020.

[26] Yetong Zhang, Fan Jiang, Gerry Chen, Varun Agrawal, Adam Rutkowski, and Frank Dellaert. Constraint manifolds for robotic inference and planning. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3175–3181, 2023. doi: 10.1109/ICRA48891.2023.10161024.

[27] Matt Zucker, Nathan Ratliff, Anca D Dragan, Mihail Pivtoraiko, Matthew Klingensmith, Christopher M Dellin, J Andrew Bagnell, and Siddhartha S Srinivasa. Chomp: Covariant hamiltonian optimization for motion planning. *The International journal of robotics research*, 32(9-10): 1164–1193, 2013.