

Selection of Input Primitives for the Generalized Label Correcting Method

Brian Paden^a and Emilio Frazzoli.^a

Abstract—The generalized label correcting method is an efficient search-based approach to trajectory optimization. It relies on a finite set of control primitives that are concatenated into candidate control signals. This paper investigates the principled selection of this set of control primitives. Emphasis is placed on a particularly challenging input space geometry, the n -dimensional sphere. We propose using controls which minimize a generalized energy function and discuss the optimization technique used to obtain these control primitives. A numerical experiment is presented showing a factor of two improvement in running time when using the optimized control primitives over a random sampling strategy.

I. INTRODUCTION

Kinodynamic motion planning and trajectory optimization problems consist of finding an open loop control signal from an infinite dimensional signal space which minimizes a cost functional. This challenging problem is approached by borrowing techniques from numerical analysis to approximate the input signal space by a subset over which an optimal solution can be efficiently computed.

A traditional approach is to approximate the signal space by a finite dimensional vector space. This enables the use of well developed finite-dimensional nonlinear optimization methods to compute locally optimal solutions to the approximated problem [1]. This is not always an acceptable strategy since many kinodynamic motion planning problems have unsatisfactory local minima.

The alternative is search-based methods which approximate the input signal space by strings of control signal primitives (or dynamically feasible trajectories) which can be efficiently searched using graph search techniques. Some examples of search based methods include Lattice-planners [2], the kinodynamic variant of the Rapidly Exploring Random Tree (RRT*) [3], the Stable Sparse RRT (SST) method [4], and the generalized label correcting (GLC) method [5]. There are additionally some hybrid approaches utilizing ideas from both finite-dimensional optimization and search-based methods [6], [7], [8].

The GLC method, which is the focus of this paper, is a general search-based algorithm for efficiently generating feasible trajectories solving a trajectory optimization or kinodynamic motion planning problem. The advantage to this method is the ability to compute feasible trajectories and control signals whose cost approximates the globally optimal cost to the problem in finite time. In contrast to related algorithms the GLC method relies on weaker technical

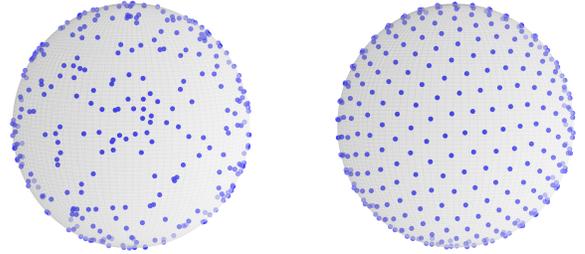


Fig. 1. Side-by-side comparison of the distribution of 500 points generated by randomly sampling from the uniform distribution (left) and by obtaining a local solution to the Thomson problem (right).

assumptions and fewer subroutines such as a local point-to-point planning solution.

In this paper we continue the development of the GLC method by investigating a selection strategy for the control primitives used by the algorithm. Without prior knowledge of the problem there is no reason to bias the selection of control primitives around any point in the input space. This suggests evenly dispersing the control primitives on the set of allowable control inputs. One technique for obtaining approximately evenly dispersed control primitives is random sampling (cf. Figure 1). This is essentially the approach taken with some randomized methods such as SST. In contrast, a more principled deterministic construction of control primitives can yield a more evenly distributed collection of points on the input space.

Sukharev grids [9] are an optimal (they minimize L_∞ dispersion) arrangement of points on hypercubes and are easily computed. In contrast, control input spaces described by the n -dimensional sphere are frequently encountered in control input constrained systems and are a challenging geometry for distributing points evenly. The principal contribution of this paper is a procedure for computing optimal configurations of control primitives on the n -sphere along with an open source implementation [10]. This is accomplished by defining an interaction potential between the points representing control primitives and computing a local minima of the total energy. This problem arises in many scientific fields and is classically known as Thomson’s problem [11].

Before describing the control primitive selection strategy in detail, a review of the kinodynamic motion planning problem is presented in Section II together with a high level overview of the GLC method. This is followed by an empirical comparison of the proposed selection strategy with a randomized selection strategy in Section III. The

^aThe authors are with the Laboratory for Information and Decision Systems at MIT. Email: bapaden@mit.edu, frazzoli@mit.edu

optimization, known as Thomson's problem, used to select control primitives is presented in Section IV. Lastly, Section V discusses the nonlinear programming technique used to find locally optimal solutions to Thomson's problem.

II. KINODYNAMIC MOTION PLANNING

Kinodynamic motion planning is a form of open loop trajectory optimization with differential and point-wise constraints. Differential constraints can be expressed by a classical nonlinear control system

$$\dot{x}(t) = f(x(t), u(t)), \quad (1)$$

with $x(t) \in \mathbb{R}^n$, $u(t) \in \Omega$, and $f : \mathbb{R}^n \times \Omega \rightarrow \mathbb{R}^n$. The control input space Ω is a subset of \mathbb{R}^m . A dynamically feasible trajectory over the time interval $[0, T]$ is a continuous time history of states $x : [0, T] \rightarrow \mathbb{R}^n$ for which there exists a measurable, essentially bounded control signal $u : [0, T] \rightarrow \Omega$ satisfying (1) almost everywhere.

The problem has three point-wise constraints. The first is an initial state constraint, $x(0) = x_0$. The second is a constraint enforced along the entire trajectory, $x(t) \in X_{free}$ for all $t \in [0, T]$. Lastly, there is a terminal constraint, $x(T) \in X_{goal}$. A trajectory that satisfies the differential and point-wise constraints is said to be a *feasible* trajectory.

Next, a cost functional J is used to measure the relative quality of trajectories,

$$J(u) = \int_{[0, T]} g(x(t), u(t)) d\mu(t). \quad (2)$$

In the above expression x is the unique solution to (1) with input u and initial condition x_0 . Since the minimum of this functional is not attained in general, the goal of computational methods for optimal kinodynamic motion planning is to return a sequence of feasible trajectories and control signals which converge to the optimal value of J subject to the feasibility constraints.

A. The Generalized Label Correcting Method

The GLC method uses a single resolution parameter R to balance an approximation of the control signal space and state space. An approximate shortest path algorithm is applied to the approximation so that the output of the algorithm converges, in cost, to the optimal cost of the problem with increasing resolution.

The control signal space is approximated by a tree of piecewise constant control signals taking values from a finite subset Ω_R of the input space Ω . The duration of constant control primitives is $1/R$ and Ω_R is assumed to converge to a dense subset of Ω as $R \rightarrow \infty$. The total duration of a control signal is limited to $h(R)/R$ where $h : \mathbb{N} \rightarrow \mathbb{R}$ defines a horizon limit that must grow unbounded as $R \rightarrow \infty$. This simple construction ensures an optimal control signal can be approximated arbitrarily well with sufficiently high resolution [5, Lemma 3].

The number of alternatives in the above construction grows exponentially with $h(R)$. Thus, the GLC method performs an approximate search of for the optimal control

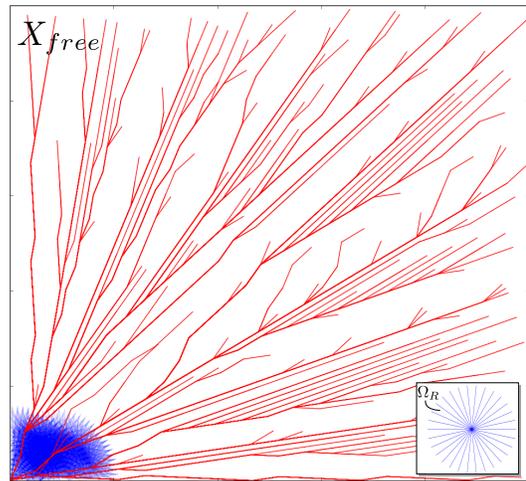


Fig. 2. 2D Single integrator example. The first 3000 iterations of a breadth first exhaustive search over all strings of control primitives is shown in blue. The subset of 416 strings satisfying the pruning condition and remaining in the illustrated rectangle are shown in red.

signal within this finite set. This is accomplished by defining a hyper-rectangular grid on the state space and considering controls producing trajectories terminating within the same hyper-rectangle as equivalent. This is denoted $u_1 \overset{R}{\sim} u_2$. The "label" for a hyper-rectangular region is the lowest cost signal producing a trajectory terminating in that region discovered by the search at any given iteration. Like the approximation of the signal space, the grid is controlled by the resolution. Each cell in the grid must be contained in a ball of radius $\eta(R)$ where $\eta(R)$ satisfies

$$\lim_{R \rightarrow \infty} \frac{R}{L_f \eta(R)} \left(e^{\frac{L_f h(R)}{R}} - 1 \right) = 0, \quad (3)$$

for a global Lipschitz constant L_f on the system dynamics with respect to x .

Then if $u_1 \overset{R}{\sim} u_2$ and

$$J(u_1) + \frac{\sqrt{n}}{\eta(R)} \frac{L_g}{L_f} \left(e^{\frac{L_f h(R)}{R}} - 1 \right) \leq J(u_2), \quad (4)$$

the signal u_2 and subsequent concatenations of signals beginning with u_2 can be omitted or pruned from the search. L_g denotes a global Lipschitz constant for the running cost g in (2). Among the remaining signals is a signal with approximately the optimal cost. With increasing resolution this signal converges to the optimal cost [5, Theorem 1].

As a simple illustration of the pruning operation consider a 2D single integrator with $\Omega = \{u \in \mathbb{R}^2 : \|u\|_2 = 1\}$ and a minimum time objective. Figure 2 shows how the pruned subset explores the space effectively from the initial condition in the lower left corner. In comparison, an exhaustive search over all strings of control primitives evaluates many paths that zig-zag around the initial condition.

Algorithm 1 describes the general search procedure which is a standard uniform cost search together with the pruning operation. A set \mathcal{U}_{feas} denotes signals producing trajectories remaining in X_{free} . Similarly, \mathcal{U}_{goal} denotes signals producing feasible trajectories terminating in the goal. The empty

string, denoted Id_U , has no cost and the NULL control has infinite cost.

The method $expand(u)$ returns the set of all signals consisting of u concatenated with one more input primitive from Ω_R . A queue Q contains candidate signals for future expansion. The method $pop(Q)$ returns a signal \hat{u} in Q satisfying

$$\hat{u} \in \underset{w \in Q}{\operatorname{argmin}} \{J(w)\}. \quad (5)$$

The method $find(w, \Sigma)$ returns a signal z belonging to the same hyper-rectangle as w from the set of labels Σ ; if no such signal is present in Σ the method $find(w, \Sigma)$ returns NULL. The method $depth(w)$ returns the number of piecewise constant segments making up w . If z prunes w in the sense of equation (4) we write $z \prec_R w$.

Algorithm 1 Generalized Label Correcting (GLC) Method

```

1:  $Q \leftarrow \{Id_U\}, \Sigma \leftarrow \emptyset, S \leftarrow \emptyset$ 
2: while  $Q \neq \emptyset$ 
3:    $u \leftarrow pop(Q)$ 
4:    $S \leftarrow expand(u)$ 
5:   for  $w \in S$ 
6:     if  $w \in \mathcal{U}_{goal}$ 
7:       return  $(J(w), w)$ 
8:      $z = find(w, \Sigma)$ 
9:     if  $(w \notin \mathcal{U}_{feas.} \vee (z \prec_R w) \vee depth(w) \geq h(R))$ 
10:       $S \leftarrow S \setminus \{w\}$ 
11:     else if  $J(w) < J(z)$ 
12:        $\Sigma \leftarrow (\Sigma \setminus \{z\}) \cup \{w\}$ 
13:    $Q \leftarrow Q \cup S$ 
14: return  $(\infty, \text{NULL})$ 

```

III. MOTIVATING EXAMPLE

Consider an agile aerial robot navigating an indoor environment. The environment, depicted in Figure 3, consists of two $5m \times 5m \times 10m$ rooms connected by a $1m \times 1m$ window in an upper corner of each room. The task is to plan a dynamically feasible and collision free trajectory between a starting state and a goal set in minimum time.

The robot is modeled with six states; three each for position and velocity. The mobility of the robot is described by the following equations

$$\begin{aligned} \dot{x} &= v, \\ \dot{v} &= -0.1v\|v\|_2 + 5u. \end{aligned} \quad (6)$$

The states are $x, v \in \mathbb{R}^3$ and the control is $u \in \mathbb{R}^3$. In this representation the zero control is defined about a hover state negating the effect of gravity. The term $-0.1v\|v\|_2$ reflects a quadratic aerodynamic drag, and the control u is a thrust vector which can be directed in any direction. The control is limited to a maximum thrust which is modeled by the constraint $\|u\|_2 \leq 1$ so that the robot's acceleration is limited to $5m/s^2$ and speed is limited to $\sqrt{50}m/s$. It follows from Pontryagin's minimum principle [12] that the minimum time objective will yield saturated control inputs

Resolution range (R)	$R = 8, 9, \dots, 13$
Horizon limit ($h(R)$)	$10R \log(R)$
State space partition scaling ($\eta(R)$)	$65R^{3/2}$
Control primitive duration (c/R)	$10/R$
Number of controls from n -sphere (Ω_R)	$3R^{3/2}$

TABLE I
TUNING PARAMETER SELECTION FOR THE GLC METHOD.

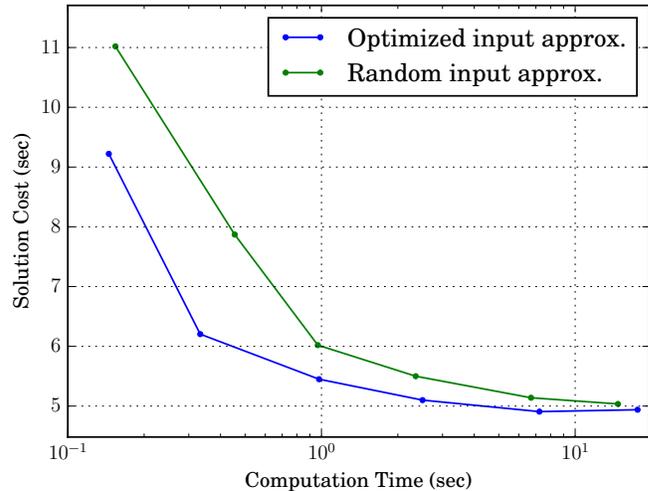


Fig. 4. Average cost and running times (10 trials each) of solutions returned by the GLC method for the two input approximation schemes. The optimized selection (blue) improves the running time required to obtain a solution of a given cost over the random selection (green) by roughly a factor of two.

at all times, $\|u(t)\|_2 = 1$ (i.e. the control is restricted to the sphere).

An implementation of the GLC method in C++ was run on an Intel i7 processor at 2.6GHz. Parameters of the algorithm are provided in Table I. In the first set of trials randomly generated control primitives on the 2-sphere are obtained by sampling from the uniform distribution. In the second set of trials minimum energy arrangements of points were used. Figure 1 illustrates configurations of 500 points generated by the two strategies.

The average running time and solution cost of trajectories returned by the GLC method are summarized in Figure 4. We observe that the optimized input approximation strategy improves the running time required to obtain a trajectory of a given cost by roughly a factor of two for this problem.

IV. THOMSON'S PROBLEM

We have proposed using minimum energy configurations of points on the n -sphere as the control primitives for the GLC method. This is classically referred to as Thomson's problem. This section provides a more detailed description of the problem and existence of optimal solutions.

Thomson's problem is an optimization problem seeking to find a minimum energy configuration of N points in \mathbb{R}^n whose Euclidean norm is 1. The generalized energy is given by a superposition of pair-wise interactions parameterized by scalar α . The energy between two points $p_1, p_2 \in \mathbb{R}^n$ is

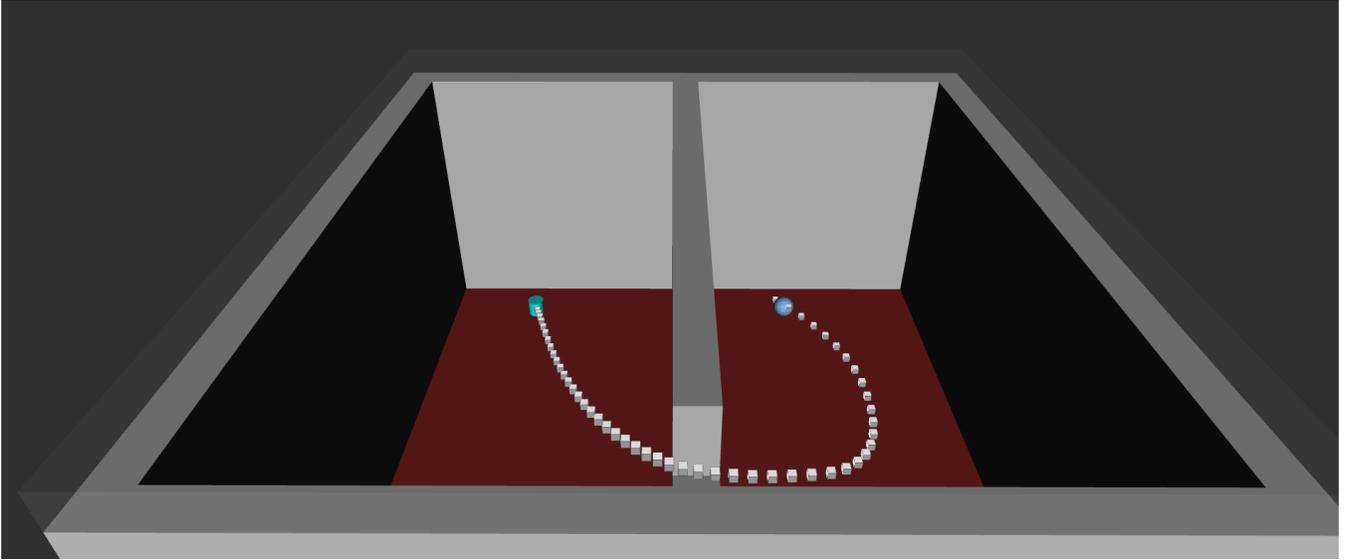


Fig. 3. Visualization of the trajectory generated by the GLC method using the optimized input space approximation. The point robot accelerates from the initial stationary position (left) directly to the window followed by a wide cornering maneuver and continued acceleration towards the goal (right). Note that the goal set leaves the terminal velocity unspecified. Thus, the optimal trajectory intercepts the goal at high speed which causes the asymmetry in the solution depicted.

given by $\|p_1 - p_2\|_2^\alpha$ if $\alpha \neq 0$, and $\log(\|p_1 - p_2\|_2^{-1})$ if $\alpha = 0$.

The configuration of the N points is represented by a matrix $X \in \mathbb{R}^{n \times N}$ with each column representing coordinates of a point in \mathbb{R}^n . The notation $X_{:,j}$ will be used to identify the coordinates of the j^{th} point in \mathbb{R}^n . In our analysis we need to distinguish between the Frobenius and Euclidean norms and inner products,

$$\begin{aligned} \langle X, Y \rangle_F &:= \text{Tr}(XY^T), \\ \|X\|_F &:= \sqrt{\text{Tr}(XX^T)}, \\ \langle x, y \rangle_2 &:= x^T y, \\ \|x\|_2 &:= \sqrt{x^T x}. \end{aligned} \quad (7)$$

The set of feasible configurations S is defined

$$S := \{X \in \mathbb{R}^{n \times N} : \|X_{:,j}\|_2 = 1, \quad j = 1, \dots, N\}. \quad (8)$$

It follows from this definition that

$$\|X\|_F = \sqrt{N} \quad \forall X \in S. \quad (9)$$

Using the above notation, the total generalized energy is given by

$$\begin{aligned} E_\alpha(X) &= \sum_{i=1}^N \sum_{j < i}^N (\|X_{:,i} - X_{:,j}\|_2^\alpha) \quad \text{if } \alpha \neq 0, \\ E_\alpha(X) &= \sum_{i=1}^N \sum_{j < i}^N \log(\|X_{:,i} - X_{:,j}\|_2^{-1}) \quad \text{if } \alpha = 0, \end{aligned} \quad (10)$$

and the optimization objective is

$$\begin{aligned} \min_{X \in S} E_\alpha(X) & \quad \text{if } \alpha \leq 0, \\ \max_{X \in S} E_\alpha(X) & \quad \text{if } \alpha > 0. \end{aligned} \quad (11)$$

A. Existence of Optimal Configurations

For $\alpha > 0$, the energy is a sum of differentiable functions. Thus, the energy is also differentiable and continuous. The set of configurations S is compact in $(\mathbb{R}^{n \times N}, \|\cdot\|_F)$. Then by Weierstrass' theorem the maximum value is attained on S .

For $\alpha \leq 0$ the continuity of E_α is broken as a result of the negative exponent and becomes unbounded from above; configurations with overlapping points have infinite energy. However, E_α remains continuous on any subset in which it is bounded.

Take $X_0 \in S$ such that $E_\alpha(X_0) < \infty$, the set $\{X \in \mathbb{R}^{n \times N} : E_\alpha(X) \leq E_\alpha(X_0)\}$ is closed and by construction contains a minimizer over S if one exists. Since S is compact, $S \cap \{X \in \mathbb{R}^{n \times N} : f(X) \leq f(X_0)\}$ is also a compact set on which E_α is continuous. Thus, the minimum over this subset is attained and is the minimum over S .

V. THE GRADIENT PROJECTION METHOD

This section reviews the gradient projection method with the adaptation of the Armijo rule proposed by Bertsekas in [13] for closed convex sets. We then address applying this method to the Thomson problem.

Consider a general minimization problem where the feasible set C is a nonempty *closed convex* subset of \mathbb{R}^m . The gradient of f at x is denoted $\nabla f(x)$, and the projection of $x \in \mathbb{R}^m$ into C is denoted $[x]^+$ and satisfies

$$\{[x]^+\} = \underset{y \in C}{\text{argmin}} \{\|y - x\|_2\}. \quad (12)$$

For a closed and convex C , there is a unique $y \in C$ minimizing $\|y - x\|_2$. The convergence of the method relies

on a non-expansiveness property,

$$\|[x]^+ - [y]^+\|_2 \leq \|x^+ - y^+\|_2, \quad (13)$$

which requires that C be closed and convex.

Using (12), the recursion of the gradient projection method is of the form

$$x_{k+1} = [x_k - \gamma_k \nabla f(x)]^+. \quad (14)$$

The subsequent point x_{k+1} is obtained by moving along the direction of steepest descent scaled by a step-size γ_k , and then projecting the result into C .

In contrast to the gradient descent step for the classical Armijo rule, which is taken along the ray through x_k in the direction of steepest descent, the gradient projection step is taken along the projection of that ray onto C . Three tuning parameters define the step size selection; $s > 0$, $\sigma > 0$, and $\beta \in (0, 1)$. The step size is given by $\alpha_k = s\beta^m$ where m is the smallest natural number such that

$$\begin{aligned} f(x_k) - f([x_k - s\beta^m \nabla f(x_k)]^+) &\geq \\ \langle \sigma \nabla f(x_k), (x_k - [x_k - s\beta^m \nabla f(x_k)]^+) \rangle. \end{aligned} \quad (15)$$

In contrast to carrying out an exact minimization along the steepest descent direction, the Armijo-rule has much less computational overhead and simply finds a step size with a sufficient decrease in f . Intuitively, s is the initial large step size which is rapidly reduced as m is increased from 0.

It was shown in [13] that limit points x^* of the sequence $\{x_k\}$ produced by (14) satisfy the necessary (but not sufficient) condition for local optimality:

$$\langle \nabla f(x^*), (x - x^*) \rangle_2 \geq 0, \quad \forall x \in C. \quad (16)$$

It follows from (15) that $f(x^{k+1}) < f(x^k)$ so these limit points are generally local minima.

A. Application to the Thomson Problem

To adapt the standard theory, we have to address the issue that the feasible set S for the Thomson problem is not convex. We replace feasible set S by its convex hull to ensure that the gradient projection iteration (20) converges to a local solution on the convex hull. We then prove that S is invariant under the gradient projection iteration so an initial configuration of points on S with bounded energy will converge to a local solution on S .

Let $\text{conv}(S)$ denote the convex hull of S . While $\|X\|_F = \sqrt{N}$ for every $X \in S$, we now have $\|X\|_F \leq \sqrt{N}$ for every $X \in \text{conv}(S)$.

The relaxed problem is

$$\begin{aligned} \min_{X \in \text{conv}(S)} E_\alpha(X) &\quad \text{if } \alpha \leq 0, \\ \max_{X \in \text{conv}(S)} E_\alpha(X) &\quad \text{if } \alpha > 0. \end{aligned} \quad (17)$$

This problem admits optimal values for X by the same argument as the original problem. The projection onto B is given by

$$([X]^+)_{:,j} := \begin{cases} \frac{X_{:,j}}{\|X_{:,j}\|_2}, & \text{if } \|X_{:,j}\|_2 > 1 \\ X_{:,j} & \text{otherwise} \end{cases}. \quad (18)$$

Note that if $X \in \text{conv}(S)$, the projection is the identity map. However, if $X \notin \text{conv}(S)$, the projection takes X into S . That is

$$\begin{aligned} [X]^+ &= X \quad \forall X \in \text{conv}(S), \\ [X]^+ &\in S \quad \forall X \notin \text{conv}(S). \end{aligned} \quad (19)$$

The gradient projection iteration is then

$$\begin{aligned} X^{k+1} &= [X^k - \gamma_k \nabla E(X^k)]^+ \quad \text{if } \alpha \leq 0, \\ X^{k+1} &= [X^k + \gamma_k \nabla E(X^k)]^+ \quad \text{if } \alpha > 0. \end{aligned} \quad (20)$$

The existence of optimal solutions to the objective over $\text{conv}(S)$ together with the standard theory ensures that limit points of the iteration will satisfy the optimality condition (16). However, we are not interested in solutions on $\text{conv}(S) \setminus S$. To ensure that the recursion converges to a stationary point on S , we make sure the initial configuration is in S . The justification for this is provided below.

Proposition 1. S is an invariant set under the gradient projection iteration.

Proof. Suppose $\alpha < 0$ and $X \in S$ (The essentially identical derivations for $\alpha > 0$ and $\alpha = 0$ are omitted for brevity). We will first show that that $X - \gamma \nabla E_\alpha(X) \notin \text{conv}(S)$ for any $\gamma > 0$. Then by (19) we will have $[X - \gamma \nabla E_\alpha(X)]^+ \in S$.

Select an index k and consider the motion of the coordinates $X_{:,k}$ in a step of the gradient projection iteration. The partial derivative with respect to $X_{:,k}$ is

$$\frac{\partial E(X)}{\partial X_{:,k}} = \sum_{j \neq k} \alpha (X_{:,k} - X_{:,j}) \|X_{:,k} - X_{:,j}\|_2^{\alpha-1}. \quad (21)$$

Since $X_{:,j} \neq X_{:,k}$ and $\|X_{:,k}\|_2 = \|X_{:,k}\|_2$, we have the inequality

$$\langle X_{:,k}, (X_{:,k} - X_{:,j}) \rangle_2 > 0, \quad (22)$$

which is derived in the appendix. Since $\alpha < 0$ and equation (22) is true for all $j \neq k$ we obtain

$$\left\langle X, -\frac{\partial E(X)}{\partial X} \right\rangle_F > 0. \quad (23)$$

The interpretation (23) is that the steepest descent direction is directed out of $\text{conv}(S)$. For any step size $\gamma > 0$ we have

$$\begin{aligned} &\|X - \gamma \nabla E_\alpha(X)\|_F \\ &= \sqrt{\|X\|_F^2 + 2 \langle X, -\gamma \nabla E_\alpha(X) \rangle_F + \|\gamma \nabla E_\alpha(X)\|_F^2} \\ &> \sqrt{\|X\|_F^2 + 2 \langle X, -\gamma \nabla E_\alpha(X) \rangle_F} \\ &> \|X\|_F \end{aligned} \quad (24)$$

By assumption $X \in S$ so $\|X\|_F = \sqrt{N}$ and $\|X - \gamma \nabla E_\alpha(X)\|_F > \sqrt{N}$ by equation (24). Thus, $\|X - \gamma \nabla E_\alpha(X)\|_F \notin \text{conv}(S)$ so in reference to (19) the projection will take $X - \gamma \nabla E_\alpha(X)$ into S which is the stated result. \square

In contrast, if $X \notin S$ it is not necessarily true that successive iterations of the gradient projection map will

converge to S . It is not difficult to construct fixed points of the map on $\text{conv}(S) \setminus S$.

B. Open Source Implementation

A lightweight open source implementation in C++ has been made available [10]. The code has no external dependencies so that it can be put into use quickly and is easily integrated into larger projects.

The initial configuration is sampled randomly from the uniform distribution on S . The method terminates at iteration k if $|E(X^{k+1}) - E(X^k)| < \varepsilon_{\text{tol}}$ or $k = k_{\text{max}}$. A configuration file allows the user to specify ε_k and k_{max} as well as the number of points N , the dimension of the space n and the power law in the generalized energy α . Additionally, the user can specify the Armijo step parameters σ , β , and s .

Figure 5 illustrates several point configurations generated by the released code in \mathbb{R}^3 for $\alpha = -1$ and various N .

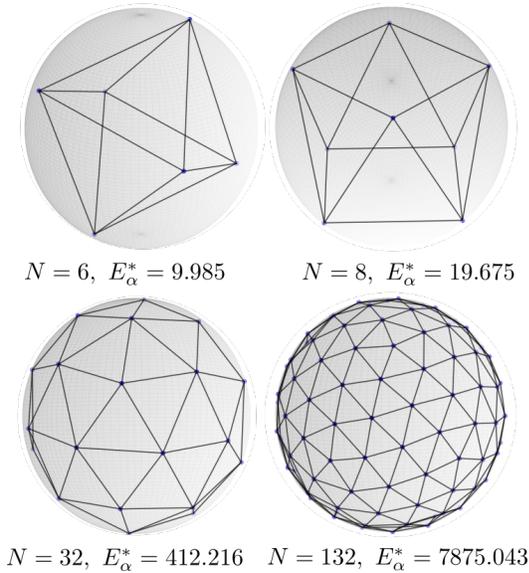


Fig. 5. Visualizations of various locally optimal configurations for the 2-sphere with $\alpha = -1$. The objective values E_α^* above coincide with the best known values found in [14] and [15].

VI. CONCLUSION

This paper demonstrated that an optimized selection of control primitives with respect to a general energy function improved performance of the GLC method by a factor of two in comparison to randomly sampled controls. Optimization of the control primitives was addressed with the gradient projection method. While the resulting optimization does not meet the standard assumptions of the gradient projection method, a rigorous analysis showed that it remains applicable to this problem with an appropriately selected initial configuration of control inputs. An open source implementation of the gradient projection method applied to Thomson’s problem has been made available to generate control primitives on the n -sphere.

REFERENCES

[1] J. T. Betts, “Survey of numerical methods for trajectory optimization,” *Journal of guidance, control, and dynamics*, vol. 21, no. 2, pp. 193–207, 1998.

[2] S. R. Lindemann and S. M. LaValle, “Incremental low-discrepancy lattice methods for motion planning,” in *International Conference on Robotics and Automation*, vol. 3, pp. 2920–2927, IEEE, 2003.

[3] S. Karaman and E. Frazzoli, “Optimal kinodynamic motion planning using incremental sampling-based methods,” in *49th Conference on Decision and Control*, pp. 7681–7687, IEEE, 2010.

[4] Y. Li, Z. Littlefield, and K. E. Bekris, “Asymptotically optimal sampling-based kinodynamic planning,” *The International Journal of Robotics Research*, vol. 35, no. 5, pp. 528–564, 2016.

[5] B. Paden and E. Frazzoli, “A generalized label correcting method for optimal kinodynamic motion planning,” *arXiv preprint arXiv:1607.06966*, 2016.

[6] S. Stoneman and R. Lampariello, “Embedding nonlinear optimization in rrt* for optimal kinodynamic planning,” in *53rd Conference on Decision and Control*, pp. 3737–3744, IEEE, 2014.

[7] C. Xie, J. van den Berg, S. Patil, and P. Abbeel, “Toward asymptotically optimal motion planning for kinodynamic systems using a two-point boundary value problem solver,” in *International Conference on Robotics and Automation*, pp. 4187–4194, IEEE, 2015.

[8] S. Choudhury, J. D. Gammell, T. D. Barfoot, S. S. Srinivasa, and S. Scherer, “Regionally accelerated batch informed trees (RABIT*): A framework to integrate local information into optimal path planning,” in *International Conference on Robotics and Automation*, pp. 4207–4214, IEEE, 2016.

[9] A. G. Sukharev, “Optimal strategies of the search for an extremum,” *USSR Computational Mathematics and Mathematical Physics*, vol. 11, no. 4, pp. 119–137, 1971.

[10] B. Paden and E. Frazzoli, “Gradient projection based solver for the generalized thomson problem,” Available at: https://github.com/bapaden/thomson_solver/releases/tag/v1.0.

[11] E. B. Saff and A. B. Kuijlaars, “Distributing many points on a sphere,” *The mathematical intelligencer*, vol. 19, no. 1, pp. 5–11, 1997.

[12] M. Athans and P. L. Falb, *Optimal control: an introduction to the theory and its applications*. Courier Corporation, 2013.

[13] D. Bertsekas, “On the goldstein-levitin-polyak gradient projection method,” *Transactions on Automatic Control*, vol. 21, no. 2, pp. 174–184, 1976.

[14] T. Erber and G. Hockney, “Equilibrium configurations of n equal charges on a sphere,” *Journal of Physics A: Mathematical and General*, vol. 24, no. 23, p. L1369, 1991.

[15] E. L. Altschuler, T. J. Williams, E. R. Ratner, R. Tipton, R. Stong, F. Dowla, and F. Wooten, “Possible global minimum lattice configurations for thomson’s problem of charges on a sphere,” *Physical Review Letters*, vol. 78, no. 14, p. 2681, 1997.

APPENDIX

The strict positivity of $\langle X_{:,j^*}, (X_{:,j^*} - X_{:,k}) \rangle_2$ in (22) is a consequence of the following Lemma which is true for any inner product space.

Recall that $X_{:,j^*} \neq X_{:,k}$ and $\|X_{:,k}\|_2 = \|X_{:,j^*}\|_2$.

Lemma. *If $y \neq x$ and $\|y\| \leq \|x\|$, then $\langle x, x - y \rangle > 0$.*

Proof. We have the strict inequality $0 < \|x - y\|^2$ since $y \neq x$. Then

$$\begin{aligned} 0 &< \|x - y\|^2 \\ &= \langle x, x \rangle - 2\langle x, y \rangle + \langle y, y \rangle \\ &\leq 2\langle x, x \rangle - 2\langle x, y \rangle, \end{aligned}$$

where we used $\|y\| \leq \|x\|$ in the last step. Rearranging the expression yields

$$\begin{aligned} 0 &< \langle x, x \rangle - \langle x, y \rangle \\ &= \langle x, x - y \rangle, \end{aligned}$$

which is the desired inequality. \square