Implicit Trajectory Planning for Feedback Linearizable Systems: A Time-varying Optimization Approach

Tianqi Zheng, John Simpson-Porco, and Enrique Mallada

***Abstract*— We develop an optimization-based framework for joint real-time trajectory planning and feedback control of feedback-linearizable systems. To achieve this goal, we define a target trajectory as the optimal solution of a time-varying optimization problem. In general, however, such trajectory may not be feasible due to , e.g., nonholonomic constraints. To solve this problem, we design a control law that generates feasible trajectories that asymptotically converge to the target trajectory. More precisely, for systems that are (dynamic) full-state linearizable, the proposed control law implicitly transforms the nonlinear system into an optimization algorithm of sufficiently high order. We prove global exponential convergence to the target trajectory for both the optimization algorithm and the original system. We illustrate the effectiveness of our proposed method on multi-target or multi-agent tracking problems with constraints.**

***Index Terms*— Time-varying optimization, motion planning, feedback linearization**

I. Introduction

The ability to design and execute safe trajectories for nonlinear systems constitutes one of the major pillars towards the development of autonomous systems [1]–[9]. Thus, not surprisingly, motion planning and control has been an increasingly popular subject of research in both industry and academia [2]–[11]. In general, this problem is usually solved in a two stage-approach. The first stage, known as motion planning, designs trajectories — usually by solving an optimization problem — that are *feasible* in that they account for obstacles and system constraints [5]–[9]. In the second stage, feedback controllers are designed to track the designed trajectories and account for system uncertainties and disturbances [2], [3], [9], [10].

While in general this approach has been quite successful, it requires the planning problem to be solved quickly enough to account for time varying environments. Thus, it imposes limits on the complexity of the optimization problem that implements motion planning. In particular, when implemented in real-time, motion planning usually amounts to linear [6] or quadratic optimization problems [11], and rarely involves more than one agent at a time. In this work, we seek to alleviate these limitations by combining the planning and tracking stages.

T. Zheng and E. Mallada are with the Department of Electrical and Computer Engineering, Johns Hopkins University, Baltimore, MD 21218, USA. Email: {tzheng8, mallada}@jhu.edu. John Simpson-Porco is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo ON, N2L 3G1 Canada. Email:jwsimpson@uwaterloo.ca. This work was supported by ARO through contract W911NF-17-1-0092, NSF through grants CNS 1544771, EPCN 1711188, AMPS 1736448, and CAREER 1752362, and Johns Hopkins University Discovery Award.

More precisely, we seek to develop a time-varying optimization-based framework for joint real-time trajectory planning and feedback control of a nonlinear dynamical system. To achieve this goal, we first define a target trajectory as the optimal solution of a time-varying optimization problem. Although in principle the target trajectory may not be feasible, we overcome this problem by designing a control law that exponentially drives the system towards the target trajectory. For nonlinear systems that are dynamic full-state linearizable, we accomplish this by designing a control law that transforms the nonlinear system into an optimization algorithm.

Our work broadly aligns with the extensive research recently performed at the intersection of optimization and traditional control theory [12]–[14], and with recent works trying to eliminate the time-scale separation usually present between optimization and control [15]–[17]. In many practical settings of robot control, especially when designing control laws for multi-robot system tracking of moving targets [18], [19], the optimization problems are not stationary (i.e., time-invariant), as the objective function and/or the constraints depend explicitly on time. Such time-varying optimization problems with or without constraints have been studied in both continuous [20] and discrete time settings [21] using prediction-correction algorithms. Our work here can be understood as an extension of these ideas to accommodate non-trivial system dynamics.

The rest of the paper is organized as follows. Section II introduces some preliminary definitions, including feedback linearization, which means a system can be transformed into a linear system by a state diffeomorphism, its dynamic feedback extension, and elementary analysis of Hurwitz linear systems. Then, in Section III, we formally state the problem and present two motivating example with different system dynamics (integrator and wheeled mobile robot). The main contribution of this paper is contained in Section IV, where we use a prediction-correction algorithm for the time-varying optimization and feedback linearization to satisfy the design requirement. We design a control law which (i) implicitly defines a target trajectory as the optimal solution of a time-varying optimization problem, and (ii) asymptotically drives the system to the target trajectory. Finally, we illustrate the effectiveness of our approach in two examples, one where a wheeled mobile robot switches from tracking one moving target to another (Section V-A), and another where multiple agents must track multiple targets with internal distance constraints (Section V-B).

Notation: Given an n-tuple $(x_1, ..., x_n)$, $x \in \mathbb{R}^n$ is the associated column vector. The $n \times n$ identity matrix is denoted as I_n. For a square symmetric matrix A, is positive (semi-)definite, and write $A \succ 0$ ($A \succeq 0$), if and only if all the eigenvalues of A are positive (nonnegative). We further write $A \succ B$ ($A \succeq B$) whenever $A - B \succ 0$ ($A - B \succeq 0$). The Euclidean norm of a vector x is denoted by $\|x\|_2$, and the Euclidean norm of a matrix A by $\|A\|_2$.

Given a differentiable function $f(x,t)$ of state $x \in \mathbb{R}^n$ and time $t \in \mathbb{R}$, the gradient with respect to x (resp. t) is denoted by $\nabla_x f(x,t)$ (resp. $\nabla_t f(x,t)$). The total derivative of $\nabla_x f(x(t),t)$ with respect to t is denoted by $\dot{\nabla}_x f(x,t) := \frac{d}{dt}\nabla_x f(x,t)$, and the n-th total derivative with respect to t by $\nabla_x^{(n)} f(x,t)$. The partial derivatives of $\nabla_x f(x,t)$ with respect to x and t are denoted by $\nabla_{xx} f(x,t) := \frac{\partial}{\partial x}\nabla_x f(x,t) \in \mathbb{R}^{n\times n}$ and $\nabla_{xt} f(x,t) := \frac{\partial}{\partial t}\nabla_x f(x,t) \in \mathbb{R}^n$, respectively. The derivative $L_f h$ of a function $h : \mathbb{R}^n \to \mathbb{R}$ along the vector field $f : \mathbb{R}^n \to \mathbb{R}^n$ and is given by $(L_f h)(x) = \nabla h(x)^T f(x)$.

II. Preliminaries

A. Feedback Linearization

1) Static Feedback Linearization: We consider a square control-affine nonlinear system with m inputs $u \in \mathbb{R}^m$ and m outputs $y \in \mathbb{R}^m$, described in state-space form:

$$\dot{x} = f(x) + g(x)u \;, \tag{1a}$$
$$y = h(x) \;, \tag{1b}$$

where $x \in \mathbb{R}^n$ is the state and where $f : D \to \mathbb{R}^n$, $g : D \to \mathbb{R}^{n\times m}$, and $h : D \to \mathbb{R}^m$ are sufficiently smooth on a domain $D \subset \mathbb{R}^n$, with g and h expanded as

$$g(x) = \big[g_1(x), \dots, g_m(x)\big] \in \mathbb{R}^{n\times m} \;,$$
$$h(x) = (h_1(x), \dots, h_m(x)) \in \mathbb{R}^m.$$

Problem 1 (State-Space Exact Linearization)**.** *Given a point $x_0 \in \mathbb{R}^n$. For the control-affine nonlinear system* (1)*, find a feedback controller $u = \alpha(x) + \beta(x)v$ defined on a neighborhood U of x_0, a coordinate transformation $z = \Phi(x)$ also defined on U, and a controllable pair (A, B) $(A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times m})$ such that:*

$$\dot{z} = Az + Bv = \frac{\partial \Phi(x)}{\partial x}\Big(f(x) + g(x)(\alpha(x) + \beta(x)v)\Big). \tag{2}$$

The key condition on (1) for solvability of the State-Space Exact Linearization Problem is that the system possess vector relative degree.

Definition 1 (Vector Relative Degree)**.** *The control-affine system* (1) *is said to have* vector relative degree $\{r_1, r_2, \dots, r_m\}$ *at a point $x_0 \in \mathbb{R}^n$ if:*

(i) $L_{g_j} L_f^k h_i(x) = 0$ for all $1 \le i \le m$, for all $k < r_i - 1$, for all $1 \le j \le m$, and for all x in a neighborhood of x_0, and

(ii) the $m \times m$ matrix,

$$R(x) = \begin{bmatrix} L_{g_1} L_f^{r_1-1} h_1(x) & \dots & L_{g_m} L_f^{r_1-1} h_1(x) \\ L_{g_1} L_f^{r_2-1} h_2(x) & \dots & L_{g_m} L_f^{r_2-1} h_2(x) \\ \vdots & \dots & \vdots \\ L_{g_1} L_f^{r_m-1} h_m(x) & \dots & L_{g_m} L_f^{r_m-1} h_m(x) \end{bmatrix}, \tag{3}$$

is nonsingular at $x = x_0$.

Lemma 1 (Solution of Exact Linearization Problem [22, Lemma 5.2.1])**.** *Suppose the matrix $g(x_0)$ has rank m. Then the State-Space Exact Linearization Problem is solvable if and only if there exists a neighborhood of x_0 such that the system* (1) *has vector relative degree $\{r_1, r_2, \dots, r_m\}$ at x_0 and $r_1 + r_2 + \cdots + r_m = n$.*

In particular, one may choose

(i) the feedback as

$$u = -R(x)^{-1}P(x) + R(x)^{-1}v,$$

where $P(x) = \mathrm{col}(L_f^{r_1} h_1(x), \dots, L_f^{r_m} h_m(x)) \in \mathbb{R}^m$ and $R(x)$ is defined in (3)*,*

(ii) the coordinate transformation as

$$\Phi(x) = \mathrm{col}(h_1(x), \dots, L_f^{r_1-1}(x), \dots, L_f^{r_m-1}(x)),$$

(iii) (A, B) having the Brunovsky Canonical Form

$$A = \mathrm{diag}\,(A_1, \dots, A_m)\,,\; B = \mathrm{diag}\,(b_1, \dots, b_m)\,,$$

where $A_i \in \mathbb{R}^{r_i\times r_i}$ and $b_i \in \mathbb{R}^{r_i}$ are

$$A_i = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ . & . & . & \dots & . \\ 0 & 0 & 0 & \dots & 1 \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix}, \quad b_i = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}.$$

2) Dynamic Feedback Linearizaiton: For systems which do not have vector relative degree, one can sometimes achieve a vector relative degree by introducing auxiliary state variables ζ, e.g., for a system that is differentially flat [23], by using dynamic feedback of the form

$$u = \alpha(x,\zeta) + \beta(x,\zeta)w, \tag{4a}$$
$$\dot{\zeta} = \gamma(x,\zeta) + \delta(x,\zeta)w. \tag{4b}$$

Consider then the composite system formed by (1) and (4)

$$\begin{bmatrix}\dot{x} \\ \dot{\zeta}\end{bmatrix} = \tilde{f}(x,\zeta) + \tilde{G}(x,\zeta)w, \tag{5a}$$
$$y = h(x), \tag{5b}$$

where

$$\tilde{f}(x,\zeta) = \begin{bmatrix} f(x) + g(x)\alpha(x,\zeta) \\ \gamma(x,\zeta)\end{bmatrix}, \tilde{G}(x,\zeta) = \begin{bmatrix} g(x)\beta(x,\zeta) \\ \delta(x,\zeta)\end{bmatrix}.$$

If the nonlinear system (5) now has vector relative degree, then the results of Lemma 1 can be applied, and the dynamic feedback has the following form:

$$w = -R^{-1}(x,\zeta)P(x,\zeta) + R^{-1}(x,\zeta)v, \tag{6}$$

where $P(x,\zeta) = \text{col}(L_{\tilde{f}}^{r_1} h_1(x), \ldots, L_{\tilde{f}}^{r_m} h_m(x)) \in \mathbb{R}^m$ and $R(x,\zeta)$ is defined in (3). Further details on this approach, known as *dynamic extension*, can be found in [22] and [24].

B. Convergence Rate of Hurwitz Matrix

A square matrix H is called Hurwitz if

$$\mu(H) := \max_{\lambda \in \text{spec}(H)} \Re[\lambda] < 0 \ ,$$

where $\text{spec}(H) := \{\lambda_i\}$ denotes the set of eigenvalues of H. If H is Hurwitz, then $\lim_{t \to +\infty} e^{Ht} = 0$.

Theorem 2 (Exponential Convergence of Hurwitz Matrices [25, Theorem 8.1])**.** *If H is Hurwitz, then there exist constants $c, \lambda > 0$ such that*

$$\|e^{Ht}\|_2 \leq c e^{-\lambda t}, \quad \textit{for all } \ t \geq 0,$$

where $-\lambda := \max_{\lambda \in \text{spec}(H)} \Re[\lambda] + \epsilon$, for some $\epsilon > 0$ that are small enough.

When H is diagonalizable, i.e., when all Jordan blocks of H have size equal to 1, one can choose $-\lambda = \max_{\lambda \in \text{spec}(H)} \Re[\lambda]$.

III. Problem Statement

As mentioned before, our goal is to develop an optimization-based framework for joint real-time trajectory planning and feedback control of nonlinear systems. To achieve this goal we develop a two-stage design approach where we (i) implicitly define the desired trajectory as the optimal solution of a *time-varying* optimization problem, and (ii) design a control law that seeks to converge asymptotically to the optimal solution of the optimization problem.

Formally, we consider a nonlinear system with state $x \in \mathbb{R}^n$, input $u \in \mathbb{R}^m$ and output $y \in \mathbb{R}^m$ as described in (1). Let $t \geq 0$ be a continuous time index, and $f_0 : \mathbb{R}^m \times \mathbb{R}_+ \to \mathbb{R}$ be a time-varying function of the output y; i.e., $f_0(y,t)$. Using $f_0(y,t)$ we implicitly define our target trajectory:

$$y^*(t) = \arg \min_{y \in \mathbb{R}^m} \ f_0(y,t). \tag{7}$$

The goal is to generate a control input $u(t)$ such that $\|y(t) - y^*(t)\|_2 \to 0$ as $t \to \infty$ for all initial conditions; i.e., global asymptotic convergence. The following assumption will be used throughout this paper.

Assumption 1 (Objective Function)**.** *The objective function $f_0(y,t)$ is infinitely differentiable ($\mathbb{C}^\infty$) with respect to both y and t, and is uniformly strongly convex in y; i.e., $\nabla_{yy} f_0(y(t),t) \succeq m_f I_m$ for some $m_f > 0$.*

The remainder of this section provides two examples that help motivate both our goals and our solution approach.

A. Example #1: Integrator

We aim to design a control law for an integrator

$$\begin{aligned} \dot{x} &= u, \\ y &= x, \end{aligned} \tag{8}$$

such that y converges asymptotically to the optimal solution of time-varying optimization problem

$$y^*(t) = \arg \min_{y} f_0(y,t). \tag{9}$$

Since the initial condition $y(0)$ may be different from $y^*(0)$, $y^*(t)$ is not a feasible trajectory. Thus, we need to find a control law to converge to it asymptotically.

The main idea is to find a control law that transforms (8) into the following optimization dynamics

$$\dot{\nabla}_y f_0(y,t) = -P \nabla_y f_0(y,t), \quad P \succ 0, \tag{10}$$

where the gradient $\nabla_y f_0(y,t)$ is driven to zero exponentially fast. Thus, since by convexity (see Assumption 1), the optimal trajectory $y^*(t)$ is characterized by $\nabla_y f_0(y^*(t),t) = 0$, the controlled y asymptotically reaches $y^*(t)$.

To achieve this transformation, we first characterize the required evolution of y for (10) to hold, and then find the control properly controls y.

Using the chain rule to differentiate the gradient term with respect to time yields

$$\dot{\nabla}_y f_0(y,t) = \nabla_{yy} f_0(y,t) \dot{y} + \nabla_{yt} f_0(y,t). \tag{11}$$

Then, by combining (10) and (11), we find that $\dot{y}$ is implicitly defined by

$$\dot{y}_{\text{imp}} = -\nabla_{yy}^{-1} f_0(y,t)[P \nabla_y f_0(y,t) + \nabla_{yt} f_0(y,t)]. \tag{12}$$

Finally, since by (8), $u = \dot{y}$, equation (12) leads to the control:

$$u = -\nabla_{yy}^{-1} f_0(y,t)[P \nabla_y f_0(y,t) + \nabla_{yt} f_0(y,t)]. \tag{13}$$

The control law (13) implicitly transforms (8) into (10). Further, it has a nice optimization-based interpretation consisting of two terms [20], [21]:

1) a *prediction term* $-\nabla_{yy}^{-1} f_0(y,t) \nabla_{yt} f_0(y,t)$, which tracks the change of the optimal solution; i.e., target trajectory,
2) and a *correction term* $-\nabla_{yy}^{-1} f_0(y,t) P \nabla_y f_0(y,t)$, which acts as a proportional controller that cancels the optimality error and drives the system toward the optimum.

Unfortunately, the solution approach shown in this example critically relies on the integrator structure in (8) that allows to arbitrarily control $\dot{y}$ by choosing u. However, for a general nonlinear system, satisfying (10) may not be possible. This is shown in the next example.

B. Example #2: Wheeled Mobile Robot

We now show how to extend the approach described above for a more involved example where we aim to drive a nonholonomic wheeled mobile robot (WMR) [24], [26]:

$$\dot{x}_1 = \cos(x_3) u_1 \tag{14a}$$

$$\dot{x}_2 = \sin(x_3) u_1 \tag{14b}$$

$$\dot{x}_3 = u_2 \tag{14c}$$

$$y = (x_1, x_2), \tag{14d}$$

such that y converges asymptotically to the optimal solution of time-varying optimization problem

$$y^*(t) = \arg\min_y f_0(y,t). \tag{15}$$

If we once again want (14) to match the dynamics (10), we need (12) to hold. However, it follows from (14) that

$$\dot{y} = \begin{bmatrix} \cos(x_3)u_1 \\ \sin(x_3)u_1 \end{bmatrix}. \tag{16}$$

It is easy to see that one cannot control every direction of $\dot{y}$ and therefore we cannot derive a control law that ensures (12).

This motivates the search for an alternative to (10) that has the equivalent effect of driving y towards $y^*(t)$. Instead, we seek to transform (14) into

$$\begin{bmatrix} \dot{\nabla}_y f_0(y,t) \\ \ddot{\nabla}_y f_0(y,t) \end{bmatrix} = \begin{bmatrix} 0 & I_m \\ -k_\text{p} I_m & -k_\text{d} I_m \end{bmatrix} \begin{bmatrix} \nabla_y f_0(y,t) \\ \dot{\nabla}_y f_0(y,t) \end{bmatrix}, \tag{17}$$

where $k_\text{p}, k_\text{d} > 0$, and $\text{col}(\nabla_y f_0(y,t), \dot{\nabla}_y f_0(y,t))$ can be interpreted as the optimality error of y, and its time derivative. Since the matrix in (17) is Hurwitz, Theorem 2 guarantees its exponential convergence.

To find the control law that transforms (14) into (17), we can follow (11) and differentiate the gradient term with respect to time twice:

$$\begin{aligned} \ddot{\nabla}_y f_0(y,t) =& \nabla_{yy} f_0(y,t)\ddot{y} + \dot{\nabla}_{yy} f_0(y,t)\dot{y}(t) \\ &+ \dot{\nabla}_{yt} f_0(y,t). \end{aligned} \tag{18}$$

Now combining once again (17) and the second row of (18) leads to the following implicit condition for acceleration the $\ddot{y}$:

$$\begin{aligned} \ddot{y}_\text{imp} = -\nabla_{yy}^{-1} f_0(y,t) \big[& \dot{\nabla}_{yy} f_0(y,t)\dot{y} + \dot{\nabla}_{yt} f_0(y,t) \\ & + k_\text{p} \nabla_y f_0(y,t) + k_\text{d} \dot{\nabla}_y f_0(y,t) \big] \end{aligned} \tag{19}$$

Finally, by differentiating (16) with respect to time we notice that the matrix on the right-hand side of

$$\ddot{y} = \begin{bmatrix} \cos(x_3) & -\sin(x_3)u_1 \\ \sin(x_3) & \cos(x_3)u_1 \end{bmatrix} \begin{bmatrix} \dot{u}_1 \\ u_2 \end{bmatrix}. \tag{20}$$

is invertible for every nonzero u_1 and thus, can can use $(\dot{u}_1, u_2)$ to control $\ddot{y}$ to follow (19), leading to the control law:

$$\begin{bmatrix} \dot{u}_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} \cos(x_3) & -\sin(x_3)u_1 \\ \sin(x_3) & \cos(x_3)u_1 \end{bmatrix}^{-1} \ddot{y}_\text{imp}. \tag{21}$$

As long as $u_1 \neq 0$, the control law is well-defined by introducing u_1 as an auxiliary state.

We finalize this section showing a particular case of (19) that is familiar for most control audience. If the task is simply tracking a moving target, we can define the following time-varying problem:

$$y^*(t) = \arg\min_y \tfrac{1}{2}\|y - y_\text{d}(t)\|_2^2, \tag{22}$$

where $y_\text{d}(t)$ represents the target trajectory. And according to (19), the implicitly defined trajectory takes the form:

$$\ddot{y}_\text{imp} = \ddot{y}_\text{d}(t) - k_p(y - y_\text{d}(t)) - k_d(\dot{y} - \dot{y}_\text{d}(t)). \tag{23}$$

Thus, in this case equation (23) can be interpreted as a common Proportional-Derivative (PD) controller.

IV. Implicit Trajectory Planning for Feedback Linearizable Systems

The above motivating example shows how to extend the algorithm from a first-order system (an integrator) to a second-order system (a unicycle). We will require some technical assumptions to carry this procedure over to a more general setting.

A. Uniform Vector Relative Degree

We assume now that the system under consideration has a uniform vector relative degree, which will in general need to be achieved via dynamic extension.

Assumption 2 (Uniform Vector Relative Degree)**.** *The multivariable nonlinear system* (5) *has vector relative degree* $r_1 = \cdots = r_m = k$ *via dynamic extension* (4) *and* $m \times k = n$.

The following is immediate from Lemma 1.

Theorem 3 (Brunovsky Canonical Form For Uniform Relative Degree System)**.** *Suppose that both, assumptions 1 and 2 hold, and there exists a dynamic compensate state* ζ *satisfying* (4). *Then the feedback function* (6) *and a state diffeomorphism* $z = \Phi(x,\zeta)$ *will transform the composite system* (5) *into* $\dot{z} = Az + Bv$, *with* (A,B) *in Brunovsky Canonical Form.*

Based on Theorem 3, it is straightforward that for a multivariable nonlinear system that has vector relative degree $r_1 = \cdots = r_m = k$ (possibly via dynamic extension) and $m \times k = n$, we can implicitly design the trajectory for y by considering $\text{col}(\nabla_y f_0(y,t), \ldots, \nabla_y^{(k-1)} f_0(y,t))$ as the new optimality error state, where the goal is to construct the following dynamical system:

$$\begin{bmatrix} \dot{\nabla}_y f_0(y,t) \\ \vdots \\ \nabla_y^{(k)} f_0(y,t) \end{bmatrix} = H \begin{bmatrix} \nabla_y f_0(y,t) \\ \vdots \\ \nabla_y^{(k-1)} f_0(y,t) \end{bmatrix}, \tag{24}$$

where

$$H = \begin{bmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ a_0 & a_1 & a_2 & \ldots & a_{k-1} \end{bmatrix} \otimes I_m \tag{25}$$

is Hurwitz.

Lemma 4 (Gradient Time Differentiation)**.** *Differentiating the gradient* $\nabla_y f_0(y,t)$ *with respect to time* $k-$*times yields:*

$$\begin{aligned} \nabla_y^{(k)} f_0(y,t) = \sum_{m=0}^{k-1} \binom{k-1}{m} \nabla_{yy}^{(m)} f_0(y,t) y^{(k-m)} \\ + \nabla_{yt}^{(k-1)} f_0(y,t). \end{aligned} \tag{26}$$

Proof: See Appendix subsection VIII-A.

Combining (24) and (26), we can implicitly design the trajectory for y by:

$$y_{\text{imp}}^{(k)} = \nabla_{yy}^{-1} f_0(y,t)[\sum_{i=0}^{k-1} a_i \nabla_y^{(i)} f_0(y,t) - \sum_{m=1}^{k-1} \binom{k-1}{m} \nabla_{yy}^{(m)} f_0(y,t) y^{(k-m)} - \nabla_{yt}^{(k-1)} f_0(y,t)]. \tag{27}$$

Now, we formally provide our solution for systems with uniform relative degree.

Theorem 5 (Control Law for Uniform Vector Relative Degree Systems)**.** *Consider the multivariable system defined as* (1) *and the time-varying optimization problem defined as* (7)*. If both assumptions 1 and 2 are satisfied, then the system will globally exponentially converge to the optimal solution of* (7)*, by using the control law:*

$$u = \alpha(x,\zeta) + \beta(x,\zeta) R(x,\zeta)^{-1}[y_{\text{imp}}^{(k)} - P(x,\zeta)], \tag{28}$$

where $y_{\text{imp}}^{(k)}$ *is given in* (27) *and the dynamic feedback function defined in* (6)*. More specifically, the following inequalities hold:*

$$\|y(t) - y^*(t)\|_2 \le Ce^{-\alpha t}, \tag{29}$$

$$0 \le f_0(y(t),t) - f_0(y^*(t),t) \le m_f C^2 e^{-2\alpha t}, \tag{30}$$

$$0 \le C = \left(\frac{c^2}{m_f^2} \sum_{j=0}^{k-1} \|\nabla_y^{(j)} f_0(y(0),0)\|_2^2)\right)^{\frac{1}{2}} < \infty,$$

for some constant $c > 0$*,* $-\alpha = \max\{\Re(\lambda_i) + \epsilon, i \in [1...n]\}$*, for some* $\epsilon > 0$ *small enough.*

Proof: See Appendix subsection VIII-B.

Theorem 5 makes a strong assumption on the structure of the nonlinear system, which is that the system must have equal vector degree $\{r_1 = \cdots = r_m\}$. In the next section we relax this assumption.

B. Non-Uniform Vector Relative Degree

We now consider the less restrictive assumption.

Assumption 3 (Non-Uniform Vector Relative Degree)**.** *We assume the multivariable nonlinear system* (5) *has vector relative degree* $\{r_1, \ldots, r_m\}$ *via dynamic extension* (4) *and* $r_1 + r_2 + \cdots + r_m = n$*.*

As a result of Assumption 3, now the order with respect to differentiation of each channel of the output is different and we can not directly design the trajectory as what we did in (27). However, we can always introduce $k - r_i$ auxiliary states for each channel y_i (i.e., further construct the dynamic extension of (2) to a system with uniform vector relative degree) , where $k = \max\{r_1, r_2, \ldots, r_m\}$ and define the new input s_i accordingly. For example, for channel y_i, we introduce the following states $\xi_1^i = v_i, \xi_2^i = \dot{\xi}_1^i, \ldots, \dot{\xi}_{k-r_i}^i = s_i$.

Theorem 6 (Brunovsky Canonical Form For Unequal Relative Degree System)**.** *Suppose that both Assumption 1 and Assumption 3 hold, and the dynamic compensate state* ζ *satisfying* (4)*. We can introduce* $k - r_i$ *auxiliary states* $\xi_j^i, j \in \{1, \ldots, k - r_i\}$ *, where* $k = \max\{r_1, r_2, \ldots, r_m\}$*, for each output channel* y_i*. More specifically, the auxiliary states* ξ *should satisfy the following dynamic:*

$$v = \tilde{\alpha}(\xi) + \tilde{\beta}(\xi)s, \tag{31a}$$

$$\dot{\xi} = \tilde{\gamma}(\xi) + \tilde{\delta}(\xi)s. \tag{31b}$$

Then the feedback function (6)*, the auxiliary states dynamic of* ξ (31)*, and a state diffeomorphism* $z = \Phi(x,\zeta,\xi)$ *will transform the composite system* (5) *into*

$$\dot{z} = Az + Bs,$$

with A, B *in Brunovsky Canonical Form.*

Proof: It immediately follows from Theorem 3 by introducing auxiliary state ξ *and new input* $s \in \mathbb{R}^m$*.*

Theorem 7 (Control Law for General Vector Relative Degree System)**.** *Consider the multivariable system defined as* (1) *and the time-varying optimization problem defined as* (7)*. Suppose that both Assumption 1 and Assumption 3 are satisfied, then the system will globally exponentially converge to the optimal solution of* (7)*, by using the control law:*

$$u = \alpha(x,\zeta) + \beta(x,\zeta) R^{-1}(x,\zeta)[\tilde{\alpha}(\xi) + \tilde{\beta}(\xi) y_{\text{imp}}^{(k)} - P(x,\zeta)], \tag{32}$$

where $y_{\text{imp}}^{(k)}$ *be the solution of* (27)*, the dynamic feedback function defined in* (6) *and the auxiliary states* ξ *satisfy* (31)*. More specifically, the following inequalities hold:*

$$\|y(t) - y^*(t)\|_2 \le Ce^{-\alpha t}, \tag{33}$$

$$0 \le f_0(y(t),t) - f_0(y^*(t),t) \le m_f C^2 e^{-2\alpha t}, \tag{34}$$

$$0 \le C = \left(\frac{c^2}{m_f^2} \sum_{j=0}^{k-1} \|\nabla_y^{(j)} f_0(y(0),0)\|_2^2)\right)^{\frac{1}{2}} < \infty,$$

for some constant $c > 0$*,* $-\alpha = \max \Re(\lambda_i) + \epsilon, i \in [1...n]$*, for some* $\epsilon > 0$ *small enough.*

Proof: See Appendix subsection VIII-C.

V. Numerical Examples

In this section, we illustrate how to leverage the time-varying optimization algorithm to solve the following robot tracking problems.

A. Robot Switching Target

Consider a wheeled mobile robot (14) charged with the task of tracking two moving targets sequentially. In the first time interval $[t_0, t_s]$, the agent is required to track the first target and in the second time interval $[t_s, t_f]$ gradually switched to track the second target. The equivalent time-varying optimization problem takes the following form:

$$\min_y S(t)\|y - y_1^d(t)\|_2^2 + (1 - S(t))\|y - y_2^d(t)\|_2^2, \tag{35}$$

where $y(t)$ is the robot position satisfying (14), $y_1^d(t), y_2^d(t)$ represents the position of moving targets at time t respectively.

The smooth switch function $S(t)$ takes the form:

$$S(t) = \begin{cases} 1, & t \leq t_s, \\ \frac{e^{\frac{-1}{t_f - t}}}{e^{\frac{-1}{t_f - t}} + e^{\frac{-1}{t - t_s}}}, & t_s < t < t_f, \\ 0, & t \geq t_f. \end{cases} \tag{36}$$

The target trajectories are designed via time parametric representation, where we use differential flatness in this trajectory generation problem [27]. Specifically, we parametrize the components of the flat output $\phi_1 = y = [x_1, x_2], \phi_2 = \dot{y}$, by

$$\phi_i(t) = \sum_{j=0}^{n-1} A_{ij} \lambda_j(t), \tag{37}$$

where the $\lambda_j(t) = t^j$ are the standard polynomial basis functions and the degree of the polynomial is set to be $n = 4$. Thus, the trajectory generation problem reduces from finding a function to finding a set of parameters.

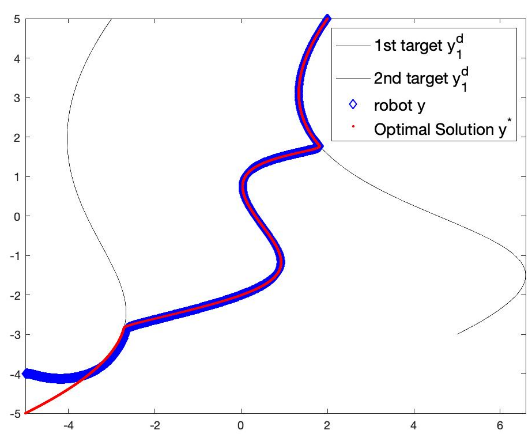

Fig. 1: Trajectory of the optimal solution $y^*(t)$ (in red), the robot (in blue) and the targets (in black). The robot converge to the optimal solution, which is to track the first target from $[0s, 5s]$ and gradually switch to track the second target in $[5s, 15s]$.

The resulting trajectories we proposed are illustrated in Figure 1, where the optimal solution $y^*(t)$ is in red, and the robot trajectory is in blue. It can be observed that the robot successfully tracks the first target from up to time $t_s = 5s$, gradually switching to the second target until $t_f = 15s$, and track the second target until simulation stops. Particularly, the random picked starting position and ending position for two targets are $[-5, -5]$ and $[5, -3]$ and the agent is positioned randomly near the starting position, which is $[-5, 4]$. We set $t_0 = 0s$ and the total simulation time is $20s$. For this implementation, the differential equation (14) is solved based on an explicit Runge-Kutta $(4, 5)$ formula, the Dormand-Prince pair.

B. Multi-robot Navigation

In this numerical example, two agents are required to track two moving targets respectively, but the maximum distance between two agents is limited (e.g., due to communication or formation constraints). We assume $y_1(t), y_2(t)$ representing the current position of the robot, whose dynamic are unicycles satisfying (14). We consider the following time-varying optimization problem for this task:

$$\min_{y_1, y_2} \|y_1 - y_1^d(t)\|_2^2 + \|y_2 - y_2^d(t)\|_2 + H(\|y_1 - y_2\|_2), \tag{38}$$

where $y_1^d(t), y_2^d(t)$ represents the current position of the moving target. $H(x) = \alpha \tan(\frac{x\pi}{2d})^2$ is a smooth penalty function, where the parameter d determines the maximum distance allowed for the two agents and α determines the flatness of penalty gain.

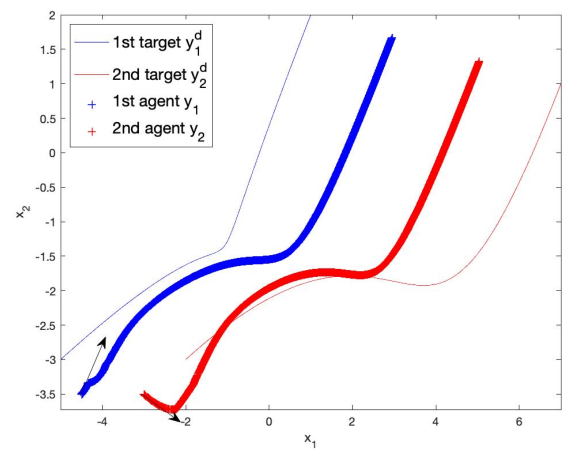

Fig. 2: Trajectories of two targets $y_1^d(t), y_2^d(t)$ (thin) and two agents y_1, y_2 (thick) with the arrows indicating their initial velocity. Agents succeed inracking targets while satisfying distance limitation between them.

The trajectories for the targets were also in time parametric representation, following the same computing procedure as in the previous section. Particularly, the random picked starting position and ending position for two targets are $[-5, -3]$ and $[-2, -3]$ respectively. The maximum allowed distance is set to be $d = 2$, and the gain is $\alpha = 1e - 8$. As to the agents, they are positioned randomly near the starting position while satisfying the distance limitation between them, which are $[-4.5, -3.5]$ and $[-3.5, -3.5]$. For this implementation, the differential equation (14) is solved using the same procedure as in Section V-A. The resulting trajectories are illustrated in Figure 2, where both robots start from arbitrary position succeed in tracking the moving target and keep the maximum distance within limits simultaneously.

VI. Conclusion

In this paper we develop an optimization-based framework for joint real-time trajectory planning and feedback control of feedback-linearizable systems. We implicitly define a target trajectory as the optimal solution of a time-varying optimization problem, which is strongly convex and smooth. For systems that are (dynamic) full-state linearizable, the proposed control law transforms the nonlinear system into an optimization algorithm of sufficiently high order. Under reasonable assumptions, our method globally asymptotically converges to the time-varying optimal solution of the original problem. Further work include: (i) adding equality and inequality time-varying constraints in the framework and (ii) considering more general nonlinear system that are not feedback linearizable.

VII. Acknowledgements

The authors would like thank Prof. George Pappas for motivating this work, and Prof. Marin Kobilarov and Mr. Rajni Bansal for their involvement in a class project that initiated this research.

VIII. Appendix

A. Proof of Lemma 4

We prove by mathematical induction. First we consider when $k = 1$ and 2.

$$\begin{aligned}\dot{\nabla}_y f_0(y,t) &= \frac{\partial \nabla_y f_0(y,t)}{\partial y}\dot{y} + \frac{\partial \nabla_y f_0(y,t)}{\partial t} \\ &= \nabla_{yy} f_0(y,t)\dot{y} + \nabla_{yt} f_0(y,t) \\ \ddot{\nabla}_y f_0(y,t) &= \frac{d}{dt}(\nabla_{yy} f_0(y,t)\dot{y} + \nabla_{yt} f_0(y,t)) \\ &= \nabla_{yy} f_0(y,t)\ddot{y} + \dot{\nabla}_{yy} f_0(y,t)\dot{y} + \dot{\nabla}_{yt} f_0(y,t)\end{aligned}$$

We want to show that for every $k \geq k_0$, $k_0 \geq 2$, if the statement holds for k, then it holds for $k+1$.

$$\begin{aligned}\nabla_y^{(k)} f_0(y,t) &= \sum_{m=0}^{k-1} \binom{k-1}{m} \nabla_{yy}^{(m)} f_0(y,t) y^{(k-m)} \\ &\quad + \nabla_{yt}^{(k-1)} f_0(y,t) \end{aligned} \tag{39}$$

Using the binomial theorem we obtain:

$$\begin{aligned}\nabla_y^{(k+1)} f_0(y,t) &= \frac{d}{dt}(\sum_{m=0}^{k-1} \binom{k-1}{m} \nabla_{yy}^{(m)} f_0(y,t) y^{(k-m)}) \\ &\quad + \frac{d}{dt}(\nabla_{yt}^{(k-1)} f_0(y,t)) \\ &= \sum_{m=0}^{k} \binom{k}{m} \nabla_{yy}^{(m)} f_0(y,t) y^{(k+1-m)} \\ &\quad + \nabla_{yt}^{(k)} f_0(y,t), \end{aligned} \tag{40}$$

which completes the proof.

B. Proof of Theorem 5

By uniformly strong convexity of $f_0(y,t)$ in y, the Hessian inverse $\nabla_{yy}^{-1} f_0(y,t)$ is defined for all $t \geq 0$. Because the vector relative degree of the nonlinear system is $r_1 = \cdots = r_m = k$, which means $y^{(k)}$ has a linear relationship with new input v. According to Lemma 4, we have (26). Furthermore, as a result of Theorem 3, feedback function of the form (28) results in $y^{(k)} = y_{\text{imp}}^{(k)}$, where $y_{\text{imp}}^{(k)}$ is the solution of (27).

Now, we are able to construct the desired dynamical system (24), where H is the designed Hurwitz matrix, and the solution of this ODE is:

$$\begin{bmatrix} \nabla_y f_0(y,t) \\ \vdots \\ \nabla_y^{(k-1)} f_0(y,t) \end{bmatrix} = e^{Ht} \begin{bmatrix} \nabla_y f_0(y(0),0) \\ \vdots \\ \nabla_y^{(k-1)} f_0(y(0),0) \end{bmatrix} \tag{41}$$

where $y(0) \in R^m$ is the initial point. By taking the Euclidean norms of both sides and applying Theorem 2 we obtain

$$\sum_{j=0}^{k-1} \|\nabla_y^{(j)} f_0(y,t)\|_2^2 \leq c^2 e^{-2\alpha t} (\sum_{j=0}^{k-1} \|\nabla_y^{(j)} f_0(y(0),0)\|_2^2) \tag{42}$$

for some constant $c > 0$, $-\alpha = \max \Re(\lambda_i) + \epsilon, i \in [1...n]$, for some $\epsilon > 0$ small enough.

Next, we use the mean-value theorem to expand $\nabla_y f_0(y,t)$ with respect to y as follows, where $\eta(t)$ is a convex combination of $y(t)$ and $y^*(t)$. Additionally using the fact that $\nabla_y f_0(y^*(t),t) = 0$ for all $t \geq 0$, we obtain:

$$y(t) - y^*(t) = \nabla_{yy}^{-1} f_0(\eta(t),t) \nabla_y f_0(y(t),t). \tag{43}$$

It follows from Assumption 1, that $\|\nabla_{yy}^{-1} f_0(y,t)\|_2 \leq m_f^{-1}$. Taking norm from both side together with equation (42) we have:

$$\begin{aligned} &\|y(t) - y^*(t)\|_2 \leq Ce^{-\alpha t}, \\ 0 \leq C = &\left(\tfrac{c^2}{m_f^2} \sum_{j=0}^{k-1} \|\nabla_y^{(j)} f_0(y(0),0)\|_2^2) \right)^{\frac{1}{2}} < \infty. \end{aligned} \tag{44}$$

On the other hand, convexity of $f_0(y(t),t)$ implies that for each $t \geq 0$

$$\begin{aligned} &0 \leq f_0(y(t),t) - f_0(y^*(t),t) \\ &\leq \nabla_y f_0(y(t),t)^T (y(t) - y^*(t)) \end{aligned} \tag{45}$$

By applying Cauchy-Swhartz inequality on the right hand side we obtain;

$$0 \leq f_0(y(t),t) - f_0(y^*(t),t) \leq m_f C^2 e^{-2\alpha t} \tag{46}$$

which completes the proof.

C. Proof of Theorem 7

According to Theorem 6, feedback function of the form (32) results in $\text{col}(y_1^{(k)}, \ldots, y_m^{(k)}) = y_{\text{imp}}^{(k)}$, where $y_{\text{imp}}^{(k)}$ is the solution of (27). Rest of the proof follows VIII-B.

References

[1] H.-D. Chiang, M. W. Hirsch, and F. F. Wu, "Stability regions of nonlinear autonomous dynamical systems," *IEEE Transactions on Automatic Control*, vol. 33, no. 1, pp. 16–27, 1988.

[2] P. Encarnacao and A. Pascoal, "3d path following for autonomous underwater vehicle," in *Proceedings of the 39th IEEE Conference on Decision and Control (Cat. No. 00CH37187)*, vol. 3. IEEE, 2000, pp. 2977–2982.

[3] M. Porfiri, D. G. Roberson, and D. J. Stilwell, "Tracking and formation control of multiple autonomous agents: A two-level consensus approach," *Automatica*, vol. 43, no. 8, pp. 1318–1328, 2007.

[4] R. M. Murray and S. S. Sastry, "Nonholonomic motion planning: Steering using sinusoids," *IEEE transactions on Automatic Control*, vol. 38, no. 5, pp. 700–716, 1993.
[5] M. Werling, J. Ziegler, S. Kammel, and S. Thrun, "Optimal trajectory generation for dynamic street scenarios in a frenet frame," in *2010 IEEE International Conference on Robotics and Automation*. IEEE, 2010, pp. 987–993.
[6] D. Mellinger and V. Kumar, "Minimum snap trajectory generation and control for quadrotors," in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 2520–2525.
[7] Y. Wang and G. S. Chirikjian, "A new potential field method for robot path planning," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, vol. 2. IEEE, 2000, pp. 977–982.
[8] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *The International Journal of Robotics Research*, vol. 17, no. 7, pp. 760–772, 1998.
[9] D. Mellinger, A. Kushleyev, and V. Kumar, "Mixed-integer quadratic program trajectory generation for heterogeneous quadrotor teams," in *2012 IEEE international conference on robotics and automation*. IEEE, 2012, pp. 477–483.
[10] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Transactions on intelligent vehicles*, vol. 1, no. 1, pp. 33–55, 2016.
[11] T. Kalmár-Nagy, R. DAndrea, and P. Ganguly, "Near-optimal dynamic trajectory generation and control of an omnidirectional vehicle," *Robotics and Autonomous Systems*, vol. 46, no. 1, pp. 47–64, 2004.
[12] U. Helmke and J. B. Moore, *Optimization and dynamical systems*. Springer Science & Business Media, 2012.
[13] A. Cherukuri, E. Mallada, and J. Cortés, "Asymptotic convergence of constrained primal–dual dynamics," *Systems & Control Letters*, vol. 87, pp. 10–15, 2016.
[14] M. Fazlyab, A. Koppel, V. M. Preciado, and A. Ribeiro, "A variational approach to dual methods for constrained convex optimization," in *2017 American Control Conference (ACC)*. IEEE, 2017, pp. 5269–5275.
[15] Z. E. Nelson and E. Mallada, "An integral quadratic constraint framework for real-time steady-state optimization of linear time-invariant systems," in *2018 Annual American Control Conference (ACC)*. IEEE, 2018, pp. 597–603.
[16] M. Colombino, E. DallÁnese, and A. Bernstein, "Online optimization as a feedback controller: Stability and tracking," *IEEE Transactions on Control of Network Systems*, 2019.
[17] L. S. P. Lawrence, J. W. Simpson-Porco, and E. Mallada, "The optimal steady-state control problem," 06 2019, revised, submitted Oct. 2018.
[18] K. Zhou and S. I. Roumeliotis, "Multirobot active target tracking with combinations of relative observations," *IEEE Transactions on Robotics*, vol. 27, no. 4, pp. 678–695, 2011.
[19] S. G. Lee, Y. Diaz-Mercado, and M. Egerstedt, "Multirobot control using time-varying density functions," *IEEE Transactions on Robotics*, vol. 31, no. 2, pp. 489–493, 2015.
[20] M. Fazlyab, S. Paternain, V. M. Preciado, and A. Ribeiro, "Prediction-correction interior-point method for time-varying convex optimization," *IEEE Transactions on Automatic Control*, vol. 63, no. 7, pp. 1973–1986, 2017.
[21] A. Simonetto and E. DallÁnese, "Prediction-correction algorithms for time-varying constrained optimization," *IEEE Transactions on Signal Processing*, vol. 65, no. 20, pp. 5481–5494, 2017.
[22] A. Isidori, *Nonlinear control systems*. Springer Science & Business Media, 2013.
[23] R. M. Murray, M. Rathinam, and W. Sluis, "Differential flatness of mechanical control systems: A catalog of prototype systems," in *ASME international mechanical engineering congress and exposition*. Citeseer, 1995.
[24] S. Sastry, *Nonlinear systems: analysis, stability, and control*. Springer Science & Business Media, 2013, vol. 10.
[25] J. P. Hespanha, *Linear systems theory*. Princeton university press, 2018.
[26] G. Oriolo, A. De Luca, and M. Vendittelli, "Wmr control via dynamic feedback linearization: design, implementation, and experimental validation," *IEEE Transactions on control systems technology*, vol. 10, no. 6, pp. 835–852, 2002.
[27] P. Martin, R. M. Murray, and P. Rouchon, "Flat systems, equivalence and trajectory generation," 2003.