A Generalization of Bellman’s Equation for Path Planning, Obstacle Avoidance and Invariant Set Estimation

Morgan Jones a, Matthew M. Peet a,

aSEMTE at Arizona State University, Tempe, AZ, 85287-6106 USA

Abstract

The standard Dynamic Programming (DP) formulation can be used to solve Multi-Stage Optimization Problems (MSOP's) with additively separable objective functions. In this paper we consider a larger class of MSOP's with monotonically backward separable objective functions; additively separable functions being a special case of monotonically backward separable functions. We propose a necessary and sufficient condition, utilizing a generalization of Bellman’s equation, for a solution of a MSOP, with a monotonically backward separable cost function, to be optimal. Moreover, we show that this proposed condition can be used to efficiently compute optimal solutions for two important MSOP's; the optimal path for Dubin’s car with obstacle avoidance, and the maximal invariant set for discrete time systems.

Key words: Dynamic Programming, Path Planning, Maximal Invariant Sets, GPU-accelerated computing.

1 Introduction

Throughout Engineering, Economics, and Mathematics many problems can be formulated as Multi-Stage Optimization Problems (MSOP’s):

\[
\min \left\{ J(u(0),...,u(T-1),x(0),...,x(T)) \right\}
\]

\[
x(0) = x_0, \ x(t+1) = f(x(t),u(t),t) \text{ for } t = 0,...,T-1
\]

\[
x(t) \in X_t \subset \mathbb{R}^n, \ u(t) \in U \subset \mathbb{R}^m \text{ for } t = 0,...,T.
\]

Such problems consist of 1) a cost function \( J : \mathbb{R}^m \times T \times \mathbb{R}^{n \times (T+1)} \to \mathbb{R} \), 2) an underlying discrete-time dynamical system governed by the plant equation \( f : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{N} \to \mathbb{R}^n \), 3) a state space \( X_t \subset \mathbb{R}^n \), 4) an admissible input space \( U \subset \mathbb{R}^m \), and 5) a terminal time \( T > 0 \). Examples of such optimization problems include: optimal battery scheduling to minimize consumer electricity bills [10]; energy-optimal speed planning for road vehicles [29]; optimal maintenance of manufacturing systems [19]; etc.

MSOP’s are members of the class of constrained nonlinear optimization problems. Such optimization problems can be solved using nonlinear solvers such as SNOPT [7] over small time horizons. However, the most commonly used class of methods for solving MSOP’s is Dynamic Programming (DP) [2]. DP methods exploit the structure of MSOP’s to decompose the optimization problem into lower dimensional sub-problems that can be solved recursively to give the solution to the original higher dimensional MSOP. Typically, DP is used to solve problems with cost functions of the form \( J(u,x) = \sum_{t=0}^{T-1} c_t(x(t),u(t)) + c_T(x(T)) \). These functions, defined in Definition 2, are called additively separable functions, as they can be additively separated into sub-functions, each of which only depend on a single time-stage, \( t \in \{0,...,T\} \). In the additively separable case it was shown in [1] that if we can find a function \( F(x,t) \) that satisfies Bellman’s Equation,

\[
F(x,t) = c_T(x) \quad \forall x \in X_T
\]

\[
F(x,t) = \inf_{u \in \Gamma_{x,t}} \left\{ c_t(x,u) + F(f(x,u,t),t+1) \right\}
\]

\[\forall x \in X_t, t \in \{0,...,T-1\},\]

where \( \Gamma_{x,t} := \{ u \in U : f(x,u,t) \in X_t \} \), then a necessary and sufficient condition for a feasible input and state sequence, \( u = (u(0),...,u(T-1)) \) and \( x = (x(0),...,x(T)) \), to be optimal is

\[
u(t) \in \arg \inf_{u \in \Gamma_{x(t),t}} \left\{ c_t(x(t),u) + F(f(x(t),u,t),t+1) \right\} \quad \forall t \in \{0,...,T-1\}.
\]

We consider MSOP’s with cost functions of the more general form \( J(u,x) = \phi_0(x(0),u(0),\phi_1(x(1),u(1),...),\phi_T(x(T))) \), where maps \( \phi_t : X \times U \times \mathbb{R} \to \mathbb{R} \) are monotonic in their...
third argument for $t = 0, \ldots, T - 1$. Such functions are called monotonically backward separable, defined in Definition 3, and shown to contain the class of additively separable functions in Lemma 4. For MSOP’s with monotonically backward separable cost functions we show in Theorem 7 that if we can find a function $V(x, t)$ that satisfies

$$V(x, t) = \phi_t(x) \quad \forall x \in X_T$$

(1)

$$V(x, t) = \inf_{u \in \mathcal{U}_{x,t}} \left\{ \phi_t(x, u, V(f(x, u, t + 1)|t + 1)) \right\}$$

$$\forall x \in X_t, t \in \{0, \ldots, T - 1\},$$

where \( \mathcal{U}_{x,t} := \{u \in U : f(x, u, t) \in X_t\} \), then a necessary and sufficient for a feasible input and state sequence, \( u = (u(0), \ldots, u(T - 1)) \) and \( x = (x(0), \ldots, x(T)) \), to be optimal is

$$u(t) \in \arg \inf_{u \in \mathcal{U}_{x(t),t}} \left\{ \phi_t(x(t), u, V(f(x(t), u, t + 1)|t + 1)) \right\}$$

$$\forall t \in \{0, \ldots, T - 1\}.$$  

Equation (1) can be thought of as a generalization of Bellman’s Equation: as it is shown in Corollary 8 that in the special case when the cost function is additively separable Equation (1) reduces to Bellman’s Equation. We therefore refer to Equation (1) as the Generalized Bellman’s Equation (GBE). Through several examples we show a solution, \( V \), to the GBE can be obtained numerically by recursively solving the GBE backwards in time for each element of \( X_t \), the same way Bellman’s Equation is solved, thereby extending traditional DP methods to solve a larger class of MSOP’s with non-additively separable cost functions. Moreover, in Section 3 it is shown how Approximate Dynamic Programming (ADP) methods can be modified to solve the GBE.

By recursively solving the GBE it is possible to synthesize optimal input sequences for many important practical problems. In this paper we consider two such problems; path planning with obstacle avoidance and maximal invariant sets. First, we define the path planning problem as the search for a sequence of inputs that drives a dynamical system to a target set in minimum time while avoiding obstacles defined by subsets of the state-space. In Section 4 we show that such problems can be formulated as an MSOP with monotonically backward separable objective, of form \( J(u, x) = \min \{\inf \{t \in [0, T] : x(t) \in S\}, T\} \), implying that the solution to the path planning problem can be found using the solution to the GBE. Similarly, in Section 5 we show that computation of maximal invariant sets can be formulated as an MSOP with monotonically backward separable objective of form \( J(u, x) = \max \{\max_{0 \leq k \leq T - 1} \{c_{u}(x(k), x(k))\}, c_{f}(x(T))\} \). Path planning with obstacle avoidance has been extensively studied (see surveys [4] [6]) and has many applications; including UAV surveillance [27]. In [22] the path planning problem is separated into two separate problems: the “geometric problem”, in which the shortest curve, \( \tilde{x}(t) \), between the initial set and target set is calculated, and the “tracking problem”, in which a controller, \( u(t) \), is synthesized so that \( \sum_{t=0}^{T}||x(t) - \tilde{x}(t)||^2 \) is minimized, where \( x(t + 1) = f(x(t), u(t), t) \) and \( ||\cdot||^2 \) is the Euclidean norm. Separating the path planning problem allows for the use of efficient algorithms such as \( A^* \)-search to solve the “geometric problem” and LQR control to solve the “tracking problem”, however, there is no guaranteed that this method will produce the true solution to the original path planning problem. The same approach is used in [3], where it is shown through numerical examples that a controller closer to optimality can be derived when the state space is augmented with historic trajectory information. Our approach of using the GBE to solve the path planning does separate the problem into the “geometric” or “tracking” problem and thus does not require any state augmentation.

The GBE can also be used in the application of computing the Finite Time Horizon Maximal Invariant Set (FTHMIS), defined as the largest set of initial conditions for a discrete time process such that there exists a feasible input sequence for which the state of the system never violates a time-varying constraint. Knowledge of this set can be used to design controllers that ensure the system never violates given safety constraints. We show that FTHMIS’s are equivalent to the sublevel set of solutions to the GBE. To the best of the authors knowledge the problem of computing FTHMIS’s has not previously been addressed in the literature. However, a proposed methodology for computing maximal invariant sets over infinite time horizons can be found in [28,25,26]. Similar continuous-time formulations of this problem can be found in [14,13].

Other examples in the literature of MSOP’s with non-additively separable cost functions can be found in the pioneering work of Li [18,17,16,15]. Li considered MSOP’s with \( k \)-separable cost functions; functions of the form \( J(u, x) = H(J_1(u, x), \ldots, J_k(u, x)) \), where \( H : \mathbb{R}^k \rightarrow \mathbb{R} \) is strictly increasing and differentiable, and each of the functions, \( J_i \), are differentiable monotonically backward separable functions. Li showed that for problems in this class of MSOP, an equivalent multi-objective optimization problem with \( k \)-separable cost functions can be constructed. The multi-objective optimization problem can then be analytically solved, using methods relying of the differentiability of the cost function, to find the optimal input sequence for the MSOP. We do not assume, as in Li, that the cost function is differentiable or \( k \)-separable and our solution does not require the solution of a multi-objective optimization problem.

In related work, coherent risk measures, from [25,24,23], result in MSOP’s with non-additively separable cost functions of the form \( J(u, x) = c_0(x(0), u(0)) + \rho_1(c_1(x(1), u(1)) + \rho_2(c_2(x(2), u(2)) + \ldots + \rho_T(c_T(x(T)), u(T)))) \). Such MSOP’s are solved recursively using a modified Bellman’s Equation. Coherent risk measure functions are a special case of monotonically backward separable functions; in this case
our GBE reduces to the previously proposed modified Bellman’s equation.

2 Multi-Stage Optimization Problems With Backward Separable Cost Functions

In this section we will introduce a class of general Multi-Stage Optimization Problems (MSOP’s). We show this class contains problems that classical DP theory is able to solve; MSOP’s with additively separable cost functions Eqn. (3). We then propose a more general class of cost functions called monotonically backward separable functions, Eqn. (4), that contain the class of additively separable functions. Using this framework we are then able to derive necessary and sufficient conditions for an input sequence to solve an MSOP with monotonically backward separable cost function. Such conditions are shown to reduce to the classical conditions proposed by Bellman [1] in the special case when the cost function is additively separable.

Definition 1 For a given initial condition \( x_0 \in \mathbb{R}^n \), for every tuple of the form \( \{J, f, \{X_t\}_{0 \leq t \leq T}, U, T \} \), where \( J: \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R} \), \( f: \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{N} \to \mathbb{R}^n \), \( X_t \subset \mathbb{R}^n \), \( U \subset \mathbb{R}^m \), and \( T \in \mathbb{N} \), we associate a MSOP of the following form

\[
\begin{align*}
(u^*, x^*) & \in \text{arg min}_{u, x} J(u, x) \text{ subject to:} \\
x(t+1) & = f(x(t), u(t), t) \text{ for } t = 0, \ldots, T-1 \\
x(0) & = x_0, \quad x(t) \in X_t \subset \mathbb{R}^n \text{ for } t = 0, \ldots, T \\
u(t) & \in U \subset \mathbb{R}^m \text{ for } t = 0, \ldots, T-1 \\
u & = (u(0), \ldots, u(T-1)) \text{ and } x = (x(0), \ldots, x(T))
\end{align*}
\]

For a given tuple \( \{J, f, \{X_t\}_{0 \leq t \leq T}, U, T \} \), the function \( J \) represents the cost function, \( f \) represents the plant dynamics, \( X_t \) represents the set of admissible states at time step \( t \in \{0, \ldots, T\} \), and \( U \) represents the set of admissible inputs.

Classical DP theory is concerned with the special case when the cost function, \( J: \mathbb{R}^m \times \mathbb{T} \times \mathbb{R}^n \times (T+1) \to \mathbb{R} \), has an additively separable structure defined as follows.

Definition 2 The function \( J: \mathbb{R}^m \times \mathbb{T} \times \mathbb{R}^n \times (T+1) \to \mathbb{R} \) is said to be additively separable if there exists functions, \( c_T(x): \mathbb{R}^n \to \mathbb{R} \), and \( g_t(x, u): \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R} \) for \( t = 0, \ldots, T-1 \) such that,

\[
J(u, x) = \sum_{t=0}^{T-1} g_t(x(t), u(t)) + c_T(x(T)),
\]

where \( u = (u(0), \ldots, u(T-1)) \) and \( x = (x(0), \ldots, x(T)) \).

We consider the class of “monotonic backward separable” cost functions defined next. The definition of this class of functions uses the image set of a function. Specifically, for a function \( f: X \to Y \) we denote the image set of the function as \( \text{Image}\{f\} := \{y \in Y : \exists x \in X \text{ such that } f(x) = y\} \).

Definition 3 The function \( J: U^T \times \prod_{t=0}^T X_t \to \mathbb{R} \), where \( U \subset \mathbb{R}^m \) and \( X_t \subset \mathbb{R}^n \) is said to be monotonically backward separable if there exists representation maps, \( \phi_t: X_t \to \mathbb{R} \), and \( \phi: X_0 \times U \times \text{Image}\{\phi_{t+1}\} \to \mathbb{R} \) for \( t = 0, \ldots, T-1 \) such that the following holds:

1. \( J(u, x) = \phi_0(x(0), u(0), \phi_1(x(1), u(1), \ldots, \phi_T(x(T)))) \),

where \( u = (u(0), \ldots, u(T-1)) \) and \( x = (x(0), \ldots, x(T)) \).

2. Each representation map, \( \phi_t \), satisfies the following upper semi-continuous and monotonic property. For any \( t \in \{0, \ldots, T-1\} \), \( x \in X_t \), \( u \in U \) and any monotonically decreasing sequence \( \{z_n\}_{n \in \mathbb{N}} \subset \mathbb{R} \), such that \( z_{n+1} \leq z_n \) for all \( n \in \mathbb{N} \), then

\[
\phi_t(x, u, z_{n+1}) \leq \phi_t(x, u, z_n) \quad \forall n \in \mathbb{N}
\]

3. For all \( t \in \{0, \ldots, T-1\} \) and \( (x, u, z) \in X_t \times U \times \text{Image}\{\phi_{t+1}\} \) we have \( |\phi_t(x, u, z)| < \infty \) and for all \( x \in X_T \) we have \( |\phi_T(x)| < \infty \); that is each set \( \text{Image}\{\phi_t\} \subset \mathbb{R} \) for \( \forall t \in \{0, \ldots, T\} \) is bounded.

Monotonically backward separable functions have the special property that the order of an infimum and composition of representation maps can be interchanged. To see this note, for \( t \in \{0, \ldots, T-1\} \) if \( \{z_n\}_{n \in \mathbb{N}} \subset \text{Image}\{\phi_{t+1}\} \) is a bounded monotonically decreasing sequence then by the monotone convergence theorem \( \inf_{n \to \infty} z_n \).

If (5) is satisfied then \( \phi_t(x, u, z_{n+1}) \leq \phi_t(x, u, z_n) \) implying, by the monotone convergence theorem, since \( \phi_t(x, u, z_n) \) is bounded, that \( \inf_{n \to \infty} \phi_t(x, u, z_n) = \lim_{n \to \infty} \phi_t(x, u, z_n) \). Thus by the upper semi-continuity property of the representation maps, \( \inf_{n \to \infty} \phi_t(x, u, z_n) = \phi_t(x, u, \inf_{n \to \infty} z_n) \) for all \( t \in \{0, \ldots, T-1\} \), \( x \in X_t \), \( u \in U \).

We next show the class of DP problems with monotonically backward separable objective functions includes the class of DP problems with additively separable objective functions as a special case.

Lemma 4 Every additively separable function is a monotonically backward separable function.

PROOF. Given an additively separable function, \( J: \mathbb{R}^m \times (T+1) \to \mathbb{R} \), we know there exists functions \( \{g_t\}_{0 \leq t \leq T} \) such that (3) holds. To prove J is monotonically backward separable we construct representation maps \( \{\phi_t\}_{0 \leq t \leq T} \) such that (4) and (5) holds.

\[
\phi_t(x, u, z) = c_t(x, u) + z \quad \text{for } i = 1, \ldots, T-1
\]

\[
\phi_T(x, w) = c_T(x).
\]

Now, \( \frac{\partial \phi_t(x, u, z)}{\partial z} = 1 > 0 \) for all \( t \in \{0, \ldots, T-1\} \), \( x \in X_t \) and \( u \in U \), implying (5).
Further examples of monotonically backward separable functions, including instances where the representation maps are non-differentiable, are given in Section 2.2.

2.1 Main Result: A Generalization Of Bellman’s Equation

When J is additively separable, the MSOP, given in (2), associated with the tuple \(\{J, f, X_t\}_{t \leq T} U, T\), can be solved recursively using Bellman’s Equation [1]. In this section we show that a similar approach can be used to solve MSOP’s with monotonically backward separable cost functions. First, however, we introduce notation for the set of feasible controls. Given a tuple \(\{J, f, X_t\}_{t \leq T} U, T\) for \(x \in X_t\) and \(t \in \{0, T - 1\}\) we denote

\[
\Gamma_{t,x} := \{u \in U : f(x, u, t) \in X_{t+1}\}.
\]

Moreover we say

\[
u = (u(0), \ldots, u(T - 1)) \in \Gamma_{x,0}[0, T - 1]
\]

if \(u(t) \in \Gamma_{x,t} \) for all \(t \in \{0, T - 1\}\), where \(x(0) = x_0\) and \(x(k + 1) = f(x(k), u(k), k)\) for \(k \in \{t, \ldots, T - 1\}\).

We next define conditions under which a function, \(V\), is said to be a value function for an associated MSOP.

**Definition 5** Consider a monotonically backward separable function \(J : \mathbb{R}^n \times X \times [0, T] \rightarrow \mathbb{R}\) with representation functions \(\{\phi_t\}_{t \leq T}\), \(f : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{N} \rightarrow \mathbb{R}^n, X \subset \mathbb{R}^n, U \subset \mathbb{R}^m, T \in \mathbb{N}\). We say the function \(V : \mathbb{R}^n \times [0, T] \rightarrow \mathbb{R}\) is a value function of the MSOP associated with the tuple \(\{J, f, X_t\}_{0 \leq t \leq T} U, T\) if for all \(x \in X_T\)

\[
V(x, T) = \phi_T(x),
\]

and for all \(x \in X_t\) and \(t \in \{0, T - 1\}\)

\[
V(x, t) = \inf_{u(t) \in \Gamma_{t,x}} \{\phi(t, u, f(x, u, t), k + 1)\},
\]

where \(x(t) = x\) and \(x(k + 1) = f(x(k), u(k), k)\) for \(k \in \{t, \ldots, T - 1\}\).

We note that the value function has the special property that \(V(x_0, 0) = J^*\), where \(J^*\) is the minimum value of the cost function of the MSOP (2). In the special case when \(J\) is an additively separable function the value function defined in this way reduces to the optimal cost-to-go function.

**Proposition 6 (Generalized Bellman’s Equation (GBE))**

Suppose \(\phi_t : X_T \rightarrow \mathbb{R}, \phi_t : X_T \times U \times \mathbb{R} \rightarrow \mathbb{R}\) for \(t = 0, \ldots, T - 1\), \(f : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{N} \rightarrow \mathbb{R}^n, X_t \subset \mathbb{R}^n, U \subset \mathbb{R}^m, T \in \mathbb{N}\), and \(\Gamma_{t,x} \neq \emptyset\) for all \(t \in \{0, \ldots, T - 1\}\) \(x \in X_t\), If \(F : \mathbb{R}^n \times [0, T] \rightarrow \mathbb{R}\) satisfies

\[
F(x, T) = \phi_T(x) \quad \forall x \in X_T \quad \text{and} \quad F(x, t) = \inf_{u \in \Gamma_{t,x}} \{\phi(t, u, f(x, u, t), t + 1)\}
\]

\[
\forall x \in X_t, t \in \{0, \ldots, T - 1\},
\]

then \(F\) is a value function of the MSOP associated with \(\{J, f, X_t\}_{0 \leq t \leq T} U, T\), where \(J\) is the monotonically backward separable objective function with representation maps \(\{\phi_t\}_{0 \leq t \leq T}\), as in Definition 3.

**PROOF.** Suppose \(F\) satisfies (10). To show \(F\) is a value function of the MSOP associated with the tuple \(\{J, f, X_t\}_{0 \leq t \leq T} U, T\) we must show it satisfies Equations (8) and (9). We prove this using backward induction in the time variable of \(F\). Clearly \(F(x, k)\) satisfies (8) for \(k = T\). Now, for our induction hypothesis, let us assume for some \(k \in \{0, \ldots, T - 1\}\) that \(F\) satisfies (9) at time-stage \(k + 1\) for all \(x \in X_{k+1}\). We will now show that the induction hypothesis implies \(F\) must also satisfy (9) at time-stage \(k\) for all \(x \in X_k\). Letting \(x \in X_k\) we have

\[
F(x, k) = \inf_{u \in \Gamma_{k,x}} \{\phi_k(x, u, F(f(x, u, k), k + 1))\}
\]

\[
= \inf_{u \in \Gamma_{k,x}} \left\{\phi_k(x, u, u(k + 1)) \inf_{u(k + 2) \in \Gamma_{k+1,x}} \{\phi_{k+1}(x(k + 1), u(k + 2) \ldots \phi_T(x(T))\})\right\}
\]

\[
= \inf_{u \in \Gamma_{k,x}} \{\phi_k(x, u, k + 1) \ldots \phi_T(x(T))\}
\]

where \(x(k) = x\) and \(x(t + 1) = f(x(t), u(t), t)\) for \(t \in \{k, T - 1\}\). The first equality follows as \(F\) satisfies (10); the second equality follows from the induction hypothesis; the third equality follows since the representation maps satisfy the monotonic property in (5).

Therefore, by backward induction, we conclude \(F(x, t)\) satisfies (8) and (9) and hence is a value function for the MSOP associated with the tuple \(\{J, f, X_t\}_{0 \leq t \leq T} U, T\). ■

We next propose necessary and sufficient conditions showing an input sequence is optimal iff it recursively minimizes the right hand side of the GBE (10).

**Theorem 7** Suppose \(J\) is a monotonically backward separable function with representation maps \(\{\phi_t\}_{0 \leq t \leq T}\), as in (4), \(f : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{N} \rightarrow \mathbb{R}^n, X \subset \mathbb{R}^n, U \subset \mathbb{R}^m, T \in \mathbb{N}\), \(\Gamma_{t,x} \neq \emptyset\) for all \(t \in \{0, \ldots, T - 1\}\) \(x \in X_t\), and \(V : \mathbb{R}^n \times [0, T] \rightarrow \mathbb{R}\) satisfies the GBE (10). The state sequence \(x^* = (x^*(0), \ldots, x^*(T - 1))\) solve the MSOP
Therefore, we will now show the above equation implies (12) and (11). It follows the pair \((u^*, x^*)\) is satisfied let us suppose for contradiction the negation and thus Equation (12) is satisfied. To prove Equation (11) then \((u^*, x^*)\) solve the MSOP given in (2). Where the first equality follows as it was shown in Proposition 6 that \(V(x, t)\) is a value function of the MSOP, the second equality follows using the GBE given in Equation 10 and using \(x^0(0) = x_0\), the third equality follows by (13), the fourth inequality follows again using the GBE, and the fifth inequality follows by recursively using the GBE together with (13). Thus if \((u^*, x^*)\) satisfy Equations (12) and (11) then \((u^*, x^*)\) solve the MSOP given in (2).

Let us now assume \(u^*\) and \(x^*\) solve the MSOP (2) associated with the tuple \(\{J, f, \{X_t\}_{0 \leq t \leq T}, U, T\}\). As we have assumed \(u^*\) and \(x^*\) is a solution then it follows \(u^*\) and \(x^*\) is feasible and thus Equation (12) is satisfied. To prove Equation (11) is also satisfied let us suppose for contradiction the negation of Equation (11), that there exists \(k \in \{0, \ldots, T - 1\}\) such that

\[
u^*(k) \notin \arg \inf_{u \in \Gamma_x(k), k} \left\{ \phi_k(x^*(k), u, V(f(x^*(k), u, k), k + 1)) \right\}
\]

and hence it follows

\[
\begin{align*}
&\inf_{u \in \Gamma_x(k), k} \left\{ \phi_k(x^*(k), u, V(f(x^*(k), u, k), k + 1)) \right\} < \phi_k(x^*(k), u^*(k), V(f(x^*(k), u^*(k), k), k + 1)) \\quad (14)
\end{align*}
\]

Using (14) it follows,

\[
\begin{align*}
J(u^*, x^*) &= \inf_{u \in \Gamma_x(k), k} J([u(0), \ldots, u(T - 1)], [x(0), \ldots, x(T)]) \\
&\leq \inf_{w \in \Gamma_x(k), k} J([u^*(0), \ldots, u^*(k - 1), w(k), \ldots, w(T - 1)], [x^*(0), \ldots, x^*(k), z(k + 1), \ldots, z(T)]) \\
&= \phi_0(x^*(0), u^*(0), \ldots, \inf_{w \in \Gamma_x(k), k} \phi_k(x^*(k), w(k), V(f(x^*(k), w(k), k, k + 1))) \ldots) \\
&< \phi_0(x^*(0), u^*(0), \ldots, \phi_k(x^*(k), u^*(k), V(f(x^*(k), u^*(k), k, k + 1))) \ldots) \\
&= \phi_0(x^*(0), u^*(0), \ldots, \inf_{w \in \Gamma_x(k), k} \phi_k(x^*(k), u^*(k), w(k + 1), \ldots, \phi_T(z(T))) \ldots) \\
&\leq \phi_0(x^*(0), u^*(0), \ldots, \phi_k(x^*(k), u^*(k), \phi_{k + 1}(x^*(k + 1), u^*(k + 1), \ldots, \phi_T(x^*(T))) \ldots) \\
&= J(u^*, x^*)
\end{align*}
\]

where the first inequality follows as the pair \((u^*, x^*)\) is assumed to solve the MSOP. The first inequality follows by taking the infimum only over the input and state sequences from time stage \(k + 1\) onwards and fixing the first \(k\) input and state sequences as \((u^*(0), \ldots, u^*(k - 1))\) and \((x^*(0), \ldots, x^*(k))\) (which are known to be feasible as the pair \((u^*, x^*)\) is assumed to solve the MSOP). The second equality follows by the monotonicity property of the representation maps given in (5). The third equality follows by Proposition 6 that shows \(V(x, t)\) is the value function. The second inequality follows from (14). The fourth equality follows using Proposition 6, that shows \(V(x, t)\) is the value function. The third inequality follows by fixing the decision variables of the infimum to \((u^*(k), \ldots, u^*(T + 1))\) and \((x^*(k + 1), \ldots, x^*(T))\) (which are known to be feasible as the pair \((u^*, x^*)\) is assumed to solve the MSOP).

We therefore get a contradiction; showing if \((u^*, x^*)\) solve the MSOP then Equations (12) and (11) must hold.

In the next corollary we show that when the cost function, \(J(u, x)\), is additively separable, the GBE (10) reduces to Bellman’s Equation (15); thus showing Bellman’s Equation is an implication of the GBE. Therefore we have generalized...
the necessary and sufficient conditions for optimality, encapsulated in Bellman’s Equation, to the GBE that provides such optimality conditions for a larger class of MSOP’s with monotonically backward separable cost functions; that no longer need be additively separable.

### Corollary 8

Suppose \( f: \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{N} \rightarrow \mathbb{R}^n, c_i: \mathbb{R}^n \times U \rightarrow \mathbb{R}, X \subset \mathbb{R}^n, U \subset \mathbb{R}^m, \) and \( T \in \mathbb{N} \). If \( F: \mathbb{R}^n \times [0,T] \rightarrow \mathbb{R} \) satisfies

\[
F(x,T) = c_T(x),
\]

\[
F(x,t) = \inf_{u \in \mathcal{X}_t} \left\{ c_i(x,u) + F(f(x,u,t),t+1) \right\}
\]

\[\forall x \in \mathcal{X}_t, t \in \{0,..,T-1\},\]

then \( F \) is a value function for the MSOP, associated with the tuple \( \{J,f,\{x_t\}_{0 \leq t \leq T},U,\} \), where \( J \) is additively separable as defined in (3).

#### PROOF.

As \( J \) is additively separable by Lemma 4 it is also monotonically backward separable and can be written in the form (4) using the representation maps given in (6). Substituting the representation maps in (6) into Equation (10), we obtain (15). Proposition 6 then guarantees optimality. ■

### 2.2 Examples: Backward Separable Functions

In Subsection 2.1, we have shown that MSOP’s with cost functions that are monotonically backward separable, Definition 3, can be solved efficiently using the GBE in Eqn. (10). We now give examples of non-additively separable, yet monotonically backward separable cost functions which may be of significant interest.

The first function we consider is the point-wise maximum function. This function occurs in MSOP’s when demand charges are present [11] and in maximal invariant set estimation [28].

### Example 1 (Point wise maximum function)

Suppose \( J: U^T \times \Pi_t=0^T \rightarrow \mathbb{R} \) is of the form

\[
J(u,x) = \max \left\{ \max_{0 \leq k \leq T-1} \{c_k(u(k),x(k))\}, c_T(x(T)) \right\},
\]

where \( u = (u(0),...,u(T-1)), x = (x(0),...,x(T)) \), \( U \subset \mathbb{R}^m \) and \( X_t \subset \mathbb{R}^n \) are compact sets, \( c_i: \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R} \) for \( 0 \leq k \leq T-1 \) and \( c_T: \mathbb{R}^n \rightarrow \mathbb{R} \). Then \( J \) is a monotonically backward separable function.

#### PROOF.

We can write \( J(u,x) \) in Form (4) using the representation functions

\[
\phi_T(x) = \int_{x(T)} c_T(x) p_T(x,w) dw,
\]

\[
\phi_i(x,u,z) = \int_{x(u)} z p_i(x,u,w) c_i(x,u) dw \forall i \in \{0,..,T-1\}.
\]

The monotonicity property (5) follows since if \( y \geq z \) then for all \( i \in \{0,..,T-1\} \)

\[
\phi_i(x,u,y) = \max\{c_i(x,u),y\} \geq \max\{c_i(x,u),z\} = \phi_i(x,u,z),
\]

where the above inequality follows by considering separately the cases \( c_i(x,u) \geq y \) and \( c_i(x,u) < y \). Boundedness and semi-continuity properties follow as the point-wise max function is Lipschitz continuous and the sets \( U \subset \mathbb{R}^m \) and \( X_t \subset \mathbb{R}^n \) are compact. ■

In the next example we consider multiplicative costs. A special case of this cost function, of the form \( J(u,x) = \mathbb{E}_w[\exp(\sum_{t=0}^{T-1} c_t(x(t),u(t),w(t)) + c_T(x(T),w(t)))] = \int_{x(0),...,x(T)} \exp(\sum_{t=0}^{T-1} c_t(x(t),u(t),w(t)) + c_T(x(T),w(t))) p(w) dw, \)

where \( p(w) \) is the probability density function of \( w = (w(0),...,w(T)) \), has previously appeared [9][8].

### Example 2 (Multiplicative Cost Functions)

Suppose \( J: U^T \times \Pi_t=0^T \rightarrow \mathbb{R} \) is of the form

\[
J(u,x) = \mathbb{E}_w[c_T(x(T),w(T))\prod_{t=0}^{T-1} c_t(x(t),u(t),w(t))]
\]

\[
= \int_{x(0),...,x(T)} c_T(x(T),w(T))\prod_{t=0}^{T-1} c_t(x(t),u(t),w(t))
\]

\[
p_T(x(T),w(T))\prod_{t=0}^{T-1} p_t(x(t),u(t),w(t)) dw(0)dw(T),
\]

where \( u = (u(0),...,u(T-1)), x = (x(0),...,x(T)), w = (w(0),...,w(T)) \), \( U \subset \mathbb{R}^m \) and \( X_t \subset \mathbb{R}^n \) are compact sets, \( l_t \subset \mathbb{R}^k, c_t: \mathbb{R}^m \times \mathbb{R}^n \rightarrow \mathbb{R} \) for \( 0 \leq t \leq T-1 \), \( c_T: \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R} \), and \( p_t: \mathbb{R}^m \times \mathbb{R}^n \rightarrow \mathbb{R} \), \( p_T: \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R} \).

\[
\int_{x(t)} p_t(x,u,w) dw = 1 \text{ and } \int_{x(t)} p_T(x,w) dw = 1 \text{ for } 0 \leq t \leq T-1.
\]

Then \( J \) is a monotonically backward separable function.

#### PROOF.

We can write \( J(u,x) \) in Form (4) using the representation functions

\[
\phi_T(x) = \int_{x(T)} c_T(x) p_T(x,w) dw,
\]

\[
\phi_i(x,u,z) = \int_{x(u)} z p_i(x,u,w) c_i(x,u) dw \forall i \in \{0,..,T-1\}.
\]

The monotonicity property (5) follows as \( c_i(x,u,w) \geq 0 \) and \( p_i(x,u,w) \geq 0 \) for all \( (x,u,w) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^k \) and \( i \in \{0,..,T-1\} \). Moreover, for fixed \( i \in \{0,..,T-1\} \) and \( (x,u) \in X_t \times U \) it follows \( \phi_i(x,u,z) = c_z \), where \( c \in \mathbb{R}^+ \) is some constant that depends on \( (x,u,i) \), is clearly continuous and bounded over compact sets. ■

In the next example we consider a function that can be interpreted as the expectation of cumulative additive costs, where at each time stage, \( t \in \{0,..,T-1\} \), a cost \( c_i(x(t),u(t)) \) is added and there is an independent
probability, \( p_t(x(t),u(t)) \in [0,1] \), of stopping, incurring no further future costs. For a state and input trajectory, \((u,x) \in \mathbb{R}^{m \times T} \times \mathbb{R}^{n \times (T+1)}\), let us denote the stopping time by \( T(u,x) \); it then follows the distribution of this random variable is given as

\[
\mathbb{P}(T(u,x) = T) = p_T(x(T))\Pi_{i=1}^{T-1}(1 - p_i(x(i),u(i))),
\]

and \( \forall t \in \mathbb{N} \),

\[
\mathbb{P}(T(u,x) = t) = p_t(x(t),u(t))\Pi_{i=1}^{T-1}(1 - p_i(x(i),u(i))),
\]

where we slightly abuse notation to write \( \Pi_{i=1}^{T-1}(1 - p_i(x(i),u(i))) = 1 \) so \( \mathbb{P}(T(u,x) = 0) = p_0(x(0),u(0)). \)

The stopped additive function is then given as

\[
J(u,x) = \mathbb{E}_{T(u,x)} \left[ \min\{T(u,x),T-1\} \sum_{t=0}^{T} c_t(x(t),u(t)) \right] + \mathbb{1}_{\{(u,x) \in \mathbb{R}^{m \times T} \times \mathbb{R}^{n \times (T+1)} : T(u,x) = T\}}(u,x)c_T(x(T)).
\]

(17)

To show (17) is monotonically backward separable we will assume the probability of the stopping time occurring inside the finite time horizon \( \{0, \ldots, T\} \) is one; this gives us the following “law of total probability” equation \( \sum_{t=0}^{T} \mathbb{P}(T(u,x) = t) = 1 \) \( \forall (u,x) \in \mathbb{R}^{m \times T} \times \mathbb{R}^{n \times (T+1)} \), which can be rewritten in terms of its probability density functions as,

\[
\sum_{t=0}^{T-1} p_t(x(t),u(t))\Pi_{i=1}^{T-1}(1 - p_i(x(i),u(i)))
\]

\[+ p_T(x(T))\Pi_{i=1}^{T-1}(1 - p_i(x(i),u(i))) \equiv 1. \]

(18)

Note, if \( p_T(x(T)) \equiv 1 \) then trivially (18) holds for any functions \( p_i : \mathbb{R}_+ \times \mathbb{R}_+ \to [0,1] \).

Assuming (18) holds and using the law of total expectation, conditioning on the probability of each stopping time, it follows

\[
J(u,x) = \mathbb{E}_{T(u,x)} \left[ \min\{T(u,x),T-1\} \sum_{t=0}^{T} c_t(x(t),u(t)) \right]
\]

\[+ \mathbb{1}_{\{(u,x) \in \mathbb{R}^{m \times T} \times \mathbb{R}^{n \times (T+1)} : T(u,x) = T\}}(u,x)c_T(x(T)).
\]

(19)

\[
\sum_{t=0}^{T-1} \left( \sum_{s=0}^{T} c_s(x,u(s)) \right) \mathbb{P}(T(u,x) = t)
\]

\[+ \left( \sum_{s=0}^{T} c_T(x,u(s)) + c_T(x(T)) \right) \mathbb{P}(T(u,x) = T)
\]

\[= \sum_{i=0}^{T-1} \left( \sum_{s=0}^{T} c_s(x,u(s)) \right) p_i(x(t),u(t))\Pi_{i=1}^{T-1}(1 - p_i(x(i),u(i)))
\]

\[+ \left( \sum_{s=0}^{T-1} c_T(x,u(s)) + c_T(x(T)) \right) \mathbb{P}(T(u,x) = T)
\]

\[= \sum_{i=0}^{T-1} \left( \sum_{s=0}^{T} c_s(x,u(s)) \right) p_i(x(t),u(t))\Pi_{i=1}^{T-1}(1 - p_i(x(i),u(i)))
\]

\[+ \left( \sum_{s=0}^{T-1} c_T(x,u(s)) + c_T(x(T)) \right) \mathbb{P}(T(u,x) = T)
\]

\[+ c_T(x(T))\Pi_{i=0}^{T-1}\mathbb{P}(T(u,x) \neq i)
\]

\[= \sum_{i=0}^{T-1} \left( \sum_{s=0}^{T} c_s(x,u(s)) \right) p_i(x(t),u(t))\Pi_{i=1}^{T-1}(1 - p_i(x(i),u(i)))
\]

\[+ c_T(x(T))\Pi_{i=0}^{T-1}\mathbb{P}(T(u,x) \neq i)
\]

\[+ c_T(x(T))\mathbb{P}(T(u,x) = T)\]

\[+ c_T(x(T))\mathbb{P}(T(u,x) \neq T).
\]

It then follows \( J(u,x) \) satisfies (4) using the representation maps

\[
\phi_t(x,u,z) = c_t(x,u) + z(1 - p_t(x,u)) \quad \forall t \in \{0, \ldots, T-1\},
\]

\[
\phi_T(x) = c_T(x)p_T(x).
\]

(20)

The monotonicity property (5) follows as \( (1 - p_t(x,u)) \geq 0 \) for all \( (x,u) \in X_t \times U \) and \( t \in \{0, \ldots, T-1\} \). Moreover, for fixed \( i \in \{0, \ldots, T-1\} \) and \( (x,u) \in X_t \times U \) it follows \( \phi_t(x,u,z) = c_0 + c_1z \), where \( c_0, c_1 \in \mathbb{R} \) are constants that depend on \((x,u,i)\), is clearly continuous and bounded over compact sets. ■
In the next example we introduce a function representing the number of time-steps a trajectory spends outside some target set. Later, in Section 4, we will use this function as the cost function for path planning problems.

**Example 4 (Minimum time set entry function)** Suppose \( J : \mathbb{R}^m \times T \times \mathbb{R}^{(T+1)} \to \mathbb{R} \) is of the form

\[
J(u, x) = \min \left\{ \inf \left\{ t \in [0, T] : x(t) \in S \right\} , T \right\},
\]

where \( u = (u(0), \ldots, u(T - 1)) \), \( u(t) \in \mathbb{R}^m \), \( x = (x(0), \ldots, x(T)) \), \( x(t) \in \mathbb{R}^n \), \( S \subseteq \mathbb{R}^n \), and if the set \( \{ t \in [0, T] : x(t) \in S \} \) is empty, we define the infimum to be infinity. Then \( J \) is a monotonically backward separable function.

**PROOF.** The function given in (21) is actually a special case of the function given in (19) with

\[
p_T(x) = 1, \quad p_t(x, u) = \mathbb{I}_S(x),
\]

\[
c_T(x) = T, \quad c_t(x, u) = t.
\]

Note, the functions \( \{ p_k \}_{0 \leq k \leq T} \) trivially satisfy (18) as \( p_T(x) = 1 \). \( \blacksquare \)

### 3 Comparison With State Augmentation Methods

We proposed an alternative method for solving MSOP’s with non-additively separable costs in [11]; where cost functions are forward separable:

\[
J(u, x) = \psi_T(x(T)), \psi_{T-1}(x(T-1), u(T-1)), \ldots, \psi_1(x(1), u(1)), \psi_0(x(0), u(0))), \ldots,)
\]

where \( \psi_0 : X_0 \times U \to \mathbb{R}^k, \psi_t : X_{t-1} \times U \times \text{Image} \{ \psi_{t-1} \} \to \mathbb{R}^k \) \( \forall t \in \{1, \ldots, T-1\} \), and \( \psi_T : X_{T-1} \times \text{Image} \{ \psi_{T-1} \} \to \mathbb{R}^k \).

It was shown that for \( \{ J, f, \{ X_t \}_{0 \leq t \leq T}, U, T \} \), where \( J \) is of the form (22), an equivalent MSOP with additively separable cost function, \( \{ J, f, \{ X_t \}_{0 \leq t \leq T}, U, T \} \), can be constructed, where \( J(u, x) = \psi_T(x(T)), f(x(1), x_2) \psi_t(x_1, u, x_2) \psi_{t+1}(x_2, u) \psi_{t+2}(x_2, u) \psi_{t+3}(x_2, u) \).

Consider the MSOP

\[
\min_{u, x} \sqrt{x(0) + u(0) + \ldots + \sqrt{x(T-1) + u(T-1) + \sqrt{x(T)}}}
\]

subject to:

\[
x(t + 1) = \begin{cases} 2 & \text{if } u = 0.5 \\ 1 & \text{if } u = 1 \end{cases} \quad \text{for } t = 0, \ldots, T,
\]

\[
x(0) = 0, \quad x(t) \in \{1, 2\} \quad \text{for } t = 0, \ldots, T,
\]

\[
u(t) \in \{0.5, 1\} \quad \text{for } t = 0, \ldots, T - 1.
\]

The cost function in the above MSOP is monotonically backward separable and can be written in the form (4) with representation maps

\[
\phi_T(x) = \sqrt{x}, \quad \psi(x, u, z) = \sqrt{x + u + z}, \quad \forall t \in \{0, \ldots, T - 1\}.
\]

Moreover the cost function is also forward separable and can be written in the form (22) with representation maps

\[
\psi_0(x, u) = [x, u]^T, \quad \psi_t(x, u, z) = [z, x, u]^T, \quad \psi_T(x, z) = \sqrt{z_1 + z_2 + \ldots + z_{2T-1} + z_{2T} + \sqrt{x}}.
\]

We solved (23) using both the GBE and the state augmentation method, plotting the computation time results in Figure 1. The green points represent the computation time required to construct the value function by solving the GBE, given in Equation (10) with representation maps given in (24), and then to synthesize the optimal input sequence using (11). The red points represent the computation time required to construct the value function by solving Bellman’s Equation (15) for the state augmented MSOP and then to construct the optimal input sequence. The green points increases linearly as a function of the terminal time, \( T \in \mathbb{N} \), of order \( O(T) \), whereas the red points increases exponentially with respect to \( T \), of order \( O(2^T) \) (due to the fact that using representation maps, given in (25), results in an augmented state space of size \( 2^T \)). Moreover, Figure 1 also includes blue dots representing computation times required to solve the GBE approximately, as discussed in the next section.

#### 3.1 Approximate Dynamic Programming Using The GBE

Rather than solving the MSOP (23) exactly using the GBE, as we did in the previous section, we now use an Approximate Dynamic Programming (ADP)/Reinforcement Learning (RL) algorithm to heuristically solve the MSOP and numerically show these algorithms can result in lower computational times when compared to methods that solve the GBE exactly. This demonstrates that MSOP’s with monotonically backward separable cost functions can be heuristically solved using the same methods developed in the ADP literature with the aid of the methodology developed in this paper.
Log scale of computation seconds
10
10

10
and approximates the value function as follows
\[
\phi \left( \tilde{x}, u, V \left( \tilde{\phi} \left( f \left( \tilde{x} \left( k \right), u, k \right), k+1 \right) \right) \right)
\]
for \( k \in \{0, \ldots, T-1\} \).
\[
\tilde{x} \left( 0 \right) = x_0, \quad \tilde{x} \left( k+1 \right) = f \left( \tilde{x} \left( k \right), \tilde{u} \left( k \right), k \right)
\]
for \( k \in \{0, \ldots, T-1\} \). \hspace{1cm} (26)

One way to obtain an approximate value function, \( V \left( x, t \right) \), is to use the rollout algorithm found in the textbook [2]. This algorithm supposes a base policy is known \( \mu_{base} : \mathbb{R}^n \times \mathbb{N} \rightarrow U \) and approximates the value function as follows
\[
\tilde{V} \left( x, t \right) = \phi \left( x(t), u(t), \phi_{t+1} \left( x(t+1), u(t+1), \ldots, x(t) \right) \right),
\]
where \( x(t) = x \) and \( \forall s \in \{t, \ldots, T-1\} \),
\[
x \left( s+1 \right) = f \left( x(s), u(s), t \right), \quad u(s) = \mu_{base} \left( x(s), s \right).
\]

Using the base policy \( \mu_{base} \left( x, t \right) = \begin{cases} 1 & \text{if } t/4 \in \mathbb{N} \\ 0.5 & \text{otherwise} \end{cases} \) we used the rollout algorithm to solve (23) for terminal times \( T = 8 \) to \( 10^6 \). Computation times are plotted as the blue points in Figure 1 showing better performance than solving the GBE exactly or using state augmentation.

4 Application: Path Planning And Obstacle Avoidance

In this section we design a full state feedback controller (Markov Policy) for a discrete time dynamical system with the objective of reaching a target set in minimum time while avoiding moving obstacles.

4.1 MSOP’s For Path Planning

We say the MSOP, associated with tuple \( \left( J, f, \{X_1 \}_{0 \leq t \leq T}, U, T \right) \), defines a Path Planning DP problem if
\[
\cdot \quad J(u, x) = \inf \{ t \in [0, T]: x(t) \in S \},
\]
\[
\cdot \quad S = \{ x \in \mathbb{R}^n : g(x) < 0 \}, \text{ where } g : \mathbb{R}^n \rightarrow \mathbb{R}.
\]
\[
\cdot \quad X_r = \mathbb{R}^n / (\cup_{i=1}^{N} O_i), \text{ where } O_i = \{ x \in \mathbb{R}^n : h_{i,t}(x) < 0 \}
\]
and \( h_{i,t} : \mathbb{R}^n \rightarrow \mathbb{R} \).
\[
\cdot \quad \text{There exits a feasible solution, } (u, x), \text{ to the MSOP (2) associated with the tuple } \{ J, f, \{X_1 \}_{0 \leq t \leq T}, U, T \} \text{ such that }
\]
\[
x(k) \in S \text{ for some } k \in \{0, \ldots, T\}.
\]

Clearly, solving an the MSOP (2) associated with a path planning problem tuple, \( \{ J, f, \{X_1 \}_{0 \leq t \leq T}, U, T \} \), is equivalent to finding the input sequence that drives a discrete time system, governed by the vector field \( f \), to a target \( S \) in minimum time while avoiding the moving obstacles, represented as sets \( O_{i,t} \subseteq \mathbb{R}^n \).

4.2 Path Planning for Dubin’s Car

We now solve the path planning problem with dynamics as defined in [20]; also known as the Dubin’s car dynamics.
\[
f \left( x, u, t \right) = \left[ x_1 + v \cos(x_3), x_2 + v \sin(x_3), x_3 + \frac{v}{L} \tan(u) \right]^T,
\]
where \( (x_1, x_2) \in \mathbb{R}^2 \) is the position of the car, \( x_3 \in \mathbb{R} \) denotes the angle the car is pointing, \( u \in \mathbb{R} \) is the steering angle input, \( v \in \mathbb{R} \) is the fixed speed of the car, and \( L \) is a parameter that determines the turning radius of the car.

We solve the path planning problem using a discretization scheme, similar to [11]; such discretization schemes are known to be parallelizable [21]. The target set, obstacles, state space, and input constraint sets are given by
\[
S = \{ (x_1, x_2) \in \mathbb{R}^2 : -0.25 < x_1 - 0.75 < 0.25, -0.25 < x_2 + 0.75 < 0.25 \}
\]
\[
O_{i,t} = \{ (x_1, x_2) \in \mathbb{R}^2 : (x_1 - X_1)^2 + (x_2 - Y_1)^2 - R_1^2 < 0 \}
\]
for \( i \in \{1, \ldots, 15\} \) and \( t \in \{0, \ldots, T\} \)
\[
X_i = [-1, 1] \times \mathbb{R} \quad \forall i \in \{0, \ldots, T\}, \quad U = [-1, 1],
\]
where \( X, Y, R \in \mathbb{R}^{15} \) are randomly generated vectors. The parameters of the system are set to \( v = 0.1 \) and \( L = 1/6 \).
The Finite Time Horizon Maximal Invariant Set (FTHMIS) is the largest set of initial conditions such that there exists

wards between two obstacles and into the target set, taking 18 steps. The third trajectory was chosen to have initial condition $[-0.2, 0.95, 0.5\pi]^T \in \mathbb{R}^3$-starting very closely to an obstacle facing upwards. This trajectory had to use the full turning radius of the car to navigate around the obstacle towards the target set and took 10 steps.

4.3 Path Planning in 3D

We now solve a three dimensional path planning problem with dynamics given by

$$f(x,u,t) = [x_1 + u_1, x_2 + u_2, x_3 + u_3]^T.$$  \hfill (28)

The target set, obstacles, state space and input constraint set were respectively are given by

$$S = \{(x_1,x_2,x_3) \in \mathbb{R}^2 : -0.25 < x_1 - 0.75 < 0.25, -0.25 < x_2 + 0.75 < 0.25, -0.25 < x_3 + 0.75 < 0.25\}$$

$$O_{i,t} = \{(x_1,x_2,x_3) \in \mathbb{R}^3 : (x_1 - A_i - \alpha_i t)^2 + (x_2 - B_i - \beta_i t)^2 + (x_3 - C_i - \gamma_i t)^2 - R_i^2 < 0 \} \quad \forall i \in \{1,\ldots,35\}, t \in \{0,\ldots,T\}$$

$$X_t = [-1,1]^3 \quad \forall t \in \{0,\ldots,T\}, \quad U = [-0.05,0.05]^3,$$

where $A,B,C,\alpha,\beta,\gamma,R \in \mathbb{R}^{35}$ are randomly generated vectors. Note, when $\alpha,\beta,\gamma$ are non-zero the center of the spherical obstacles moves with time. For presentation purposes we select $\alpha = \beta = \gamma = 0$.

This path planning problem can be numerically solved by computing the solution to the GBE, given in Equation (10), using $\phi_1$ as given in (20). To numerically solve the GBE we discretized the state and input space, $X_t \subset \mathbb{R}$ and $U \subset \mathbb{R}^3$, as a $40 \times 40 \times 40$ uniform grid on $[-1,1]^3$ and a $5 \times 5 \times 5$ uniform grid on $[-0.05,0.05]^3$ respectively. Figure 3 shows four optimal state sequences, shown as green lines, starting from various initial conditions. All trajectories successfully avoid the obstacles, represented as red spheres, and reach the target set, shown as a blue cube.

GPU Implementation All DP methods involving discretization fall prey to the curse of dimensionality, where the number of points required to sample a space increases exponentially with respect to the dimension of the space. For this reason solving MSOP’s in dimensions greater than three can be computationally challenging. Fortunately, our discretization approach to solving the GBE (Equation (10)), can be parallelized at each time-step. To improve the scalability of the proposed approach, we have therefore constructed in Matlab a GPU accelerated DP algorithm for solving the 3D path planning problem. This code is available for download at Code Ocean [12].

5 Application: Maximal Invariant Sets

The Finite Time Horizon Maximal Invariant Set (FTHMIS) is the largest set of initial conditions such that there exists
an input sequence that produces a feasible state sequence over a finite time period. Computation of the maximal robust invariant sets over infinite time horizons was considered in [28]. Before we define the FTHMIS we introduce some notation.

For \( f : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{N} \rightarrow \mathbb{R}^n \) we say the map \( \psi_f : \mathbb{R}^n \times \mathbb{R} \times \mathbb{R}^m \times \{T-1\} \rightarrow \mathbb{R}^n \) is the solution map associated with \( f \) if for any \( T > 0 \) the following holds for all \( t \in \{0, ..., T\} \)

\[
\psi_f(x(t), u) = x(t),
\]

where \( u = (u(0), ..., u(T-1)) \), \( x(k+1) = f(x(k), u(k)) \) for all \( k \in \{0, ..., k-1\} \), and \( x(0) = x_0 \).

**Definition 9** For \( f : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{N} \rightarrow \mathbb{R}^n \), \( X_T \subseteq \mathbb{R}^n \), \( U \subseteq \mathbb{R}^m \), \( T \in \mathbb{N} \), and \( \mathcal{A}_T \subseteq \mathbb{R}^n \) we define the Finite Time Horizon Maximal Invariant Set (FTHMIS), denoted by \( \mathcal{R} \), by

\[
\mathcal{R} := \{ x_0 \in \mathbb{R}^n : \exists u \in \Gamma_{x_0,0,T-1} \text{ such that } \psi_f(x_0, t, u) \in \mathcal{A}_T \quad \forall t \in \{0, ..., T\} \},
\]

where the notation \( \Gamma_{x_0,0,T-1} \) is as in (7).

We next show that the sublevel set of the value function associated with a certain DP problem can completely characterize the FTHMIS.

**Theorem 10** Consider the sets \( \mathcal{A}_T = \{ x \in \mathbb{R}^n : g_r(x) < 0 \} \), where \( g_r : \mathbb{R}^n \rightarrow \mathbb{R} \). Suppose \( V(x, t) \) is a value function associated with the MSOP, defined by the tuple \( \{ J, f, \{ X_t \}_{0 \leq t \leq T}, U, T \} \), where \( J(\mathbf{u}, \mathbf{x}) = \max_{0 \leq k \leq T} g_k(x(k)) \). Then

\[
\mathcal{R} = \{ x \in \mathbb{R}^n : V(x, 0) < 0 \},
\]

where the set \( \mathcal{R} \subseteq \mathbb{R}^n \) is the FTHMIS as in Definition 9.

**Proof.** The function \( J(\mathbf{u}, \mathbf{x}) = \max_{0 \leq k \leq T} g_k(x(k)) \) is monotonically backward separable as shown in Example 1 using representation maps given by

\[
\phi_r(x, u, z) = \max \{ g_r(x), z \} \quad \text{for all } i \in \{0, ..., T-1\}
\]

Therefore by Definition 5 any value function, \( V : \mathbb{R}^n \rightarrow \mathbb{R} \), associated with \( \{ J, f, \{ X_t \}_{0 \leq t \leq T}, U, T \} \) satisfies for all \( x \in X_T \)

\[
V(x, T) = g_T(x),
\]

and for all \( t \in \{0, 1, ..., T-1\} \) and \( x \in X_t \)

\[
V(x, t) = \inf_{u \in \Gamma_{x,0,T-1; \leq k \leq T}} \max_{0 \leq k \leq T} g_k(\psi_f(x, k, u)).
\]

We will first show that \( \mathcal{R} \subseteq \{ x \in \mathbb{R}^n : V(x, 0) < 0 \} \). Let \( x_0 \in \mathcal{R} \) then by Definition 9 there exists \( w_0 \in \Gamma_{x_0,0,T-1} \) such that

\[
\psi_f(x_0, t, u_0) \in \mathcal{A}_T \quad \forall t \in \{0, ..., T\}.
\]

As \( \mathcal{A}_T = \{ x \in \mathbb{R}^n : g_r(x) < 0 \} \) we deduce from the above equation that

\[
g_r(\psi_f(x_0, t, u_0)) < 0 \quad \forall t \in \{0, ..., T\}.
\]

Therefore,

\[
V(x_0, 0) = \inf_{u \in \Gamma_{x_0,0,T-1; 0 \leq k \leq T}} \max_{0 \leq k \leq T} g_k(\psi_f(x_0, k, u)) \leq \max_{0 \leq k \leq T} g_k(\psi_f(x_0, k, u_0)) < 0,
\]

where the second inequality follows by (32). We therefore deduce \( x_0 \in \{ x \in \mathbb{R}^n : V(x, 0) < 0 \} \) and hence \( \mathcal{R} \subseteq \{ x \in \mathbb{R}^n : V(x, 0) < 0 \} \).

We next show \( \{ x \in \mathbb{R}^n : V(x, 0) < 0 \} \subseteq \mathcal{R} \). Let \( x_0 \in \{ x \in \mathbb{R}^n : V(x, 0) < 0 \} \) then,

\[
\inf_{u \in \Gamma_{x_0,0,T-1; 0 \leq k \leq T}} \max_{0 \leq k \leq T} g_k(\psi_f(x_0, k, u)) = V(x_0, 0) < 0.
\]

Therefore as the above inequality is strict, there exists some \( \varepsilon > 0 \) such that

\[
\inf_{u \in \Gamma_{x_0,0,T-1; 0 \leq k \leq T}} \max_{0 \leq k \leq T} g_k(\psi_f(x_0, k, u)) = V(x_0, 0) < -\varepsilon.
\]

By the definition of the infimum for any \( \delta > 0 \) there exists \( w \in \Gamma_{x_0,0,T-1} \) such that

\[
\max_{0 \leq k \leq T} g_k(\psi_f(x_0, k, w)) < \inf_{u \in \Gamma_{x_0,0,T-1; 0 \leq k \leq T}} \max_{0 \leq k \leq T} g_k(\psi_f(x_0, k, u)) + \delta.
\]

Hence by letting \( 0 < \delta < \varepsilon \) we get

\[
\max_{0 \leq k \leq T} g_k(\psi_f(x_0, k, w)) < \inf_{u \in \Gamma_{x_0,0,T-1; 0 \leq k \leq T}} \max_{0 \leq k \leq T} g_k(\psi_f(x_0, k, u)) + \delta < -\varepsilon + \delta < 0,
\]

where the first inequality follows by (34), the second inequality follows from (33), and the third inequality follows from selecting \( \delta < \varepsilon \).

Therefore by (35) there exists \( w \in \Gamma_{x_0,0,T-1} \) such that \( \max_{0 \leq k \leq T} g_k(\psi_f(x_0, k, w)) < 0 \). We now deduce that for any \( t \in \{0, ..., T\} \)

\[
g_r(\psi_f(x_0, t, w)) < \max_{0 \leq k \leq T} g_k(\psi_f(x_0, k, w)) < 0.
\]

Thus \( \psi_f(x_0, t, w) \in \mathcal{A}_T \), implying \( x_0 \in \mathcal{R} \). Therefore \( \{ x \in \mathbb{R}^n : V(x, 0) < 0 \} \subseteq \mathcal{R} \).
5.1 Numerical Example: Maximal Invariant Sets

Value functions can characterize FTHMIS’s, as shown by Theorem 10. We now approximate a FTHMIS by computing a value function using a discretization scheme for solving the GBE, given in Equation (10). Let us consider a discrete time switching system, whose Robust Maximal Invariant Set (RMIS) was previously computed in [28].

Let us consider

\[ f(x, u, t) = \begin{cases} 
\frac{x_1}{(0.5 + u)x_1 - 0.1x_2} & \text{if } 1 - (x_1 - 1)^2 - x_2^2 \leq 0 \\
\frac{x_2}{0.2x_1 - (0.1 + u)x_2 + x_2^2} & \text{otherwise.} 
\end{cases} \]

We now compute the FTHMIS, denoted by \( \mathcal{R} \), associated with

\[ \mathcal{R}_t = \{ x \in \mathbb{R}^2 : g_t(x) \leq 0 \} \text{ for all } t \in \{0, \ldots, T \}, \]
\[ g_t(x) = \left( x_1 - \frac{(t-1)}{4} \right)^2 + \left( x_2 - \frac{(t+1)}{4} \right)^2 - 1.5, \]
\[ X_t = [-1, 1]^2 \text{ for all } t \in \{0, \ldots, T \}, \]
\[ U = \{ u \in \mathbb{R} : u^2 - 0.01 \leq 0 \}, \quad T = 4. \]

Figure 4 shows the FTHMIS, \( \mathcal{R} \), found by using a discretization scheme to solve the GBE (10) for 5 x 5 state grid points in \([-1, 1]^2\). To represent \( \mathcal{R} \) in \( \mathbb{R}^2 \), once the value function, \( V(x,t) \), is found at each grid point a polynomial function is fitted and its zero-sublevel set, shown as the orange shaded region, approximately gives \( \mathcal{R} \).

6 Conclusion

For MSOP’s with monotonically backward separable cost functions we have derived necessary and sufficient conditions for solutions to be optimal. We have shown that by solving the Generalized Bellman’s Equation (GBE) one can derive an optimal input sequence. Furthermore, we have demonstrated the GBE can be numerically solved using a discretization scheme and Approximate Dynamic Programming (ADP) techniques such as Rollout. We have shown our numerical methods can solve current practical problems of interest; such as path planning and the computation of maximal invariant sets.

References

[1] Richard Bellman. Dynamic programming. Science, 153(3731):34–37, 1966.
[2] Dimitri P Bertsekas. Dynamic programming and optimal control, volume 1. Athena scientific Belmont, MA, 1995.
[3] Raghvendra V Cowlagi and Panagiotis Tsiotras. Hierarchical motion planning with dynamical feasibility guarantees for mobile robotic vehicles. IEEE Transactions on Robotics, 28(2):379–395, 2011.
[4] Stuart E Dreyfus. An appraisal of some shortest-path algorithms. Operations research, 17(3):395–412, 1969.
[5] Willem Esterhuizen, Tim Aschenbruck, and Stefan Streif. On maximal robust positively invariant sets in constrained nonlinear systems. arXiv preprint arXiv:1904.01985, 2019.
[6] Giorgio Gallo and Stefano Pallottino. Shortest path algorithms. Annals of operations research, 13(1):1–79, 1988.
[7] Philip E Gill, Walter Murray, and Michael A Saunders. Snopt: An sqp algorithm for large-scale constrained optimization. SIAM review, 47(1):99–131, 2005.
[8] Keith Glover and John C Doyle. State-space formulae for all stabilizing controllers that satisfy an \( h_\infty \)-norm bound and relations to relations to risk sensitivity. Systems & control letters, 11(3):167–172, 1988.
[9] David Jacobson. Optimal stochastic linear systems with exponential performance criteria and their relation to deterministic differential games. IEEE Transactions on Automatic control, 18(2):124–131, 1973.
[10] Morgan Jones and Matthew M Peet. Solving dynamic programming with supremum terms in the objective and application to optimal battery scheduling for electricity consumers subject to demand charges. In Conference on Decision and Control (CDC), pages 1323–1329. IEEE, 2017.
[11] Morgan Jones and Matthew M Peet. Extensions of the dynamic programming framework: Battery scheduling, demand charges, and renewable integration. arXiv preprint arXiv:1812.00792, 2018.
[12] Morgan Jones and Matthew M Peet. GPU accelerated 3D dynamic programming path planning and obstacle avoidance. https://codeocean.com/capsule/3639299/, 2019.
[13] Morgan Jones and Matthew M Peet. Relaxing the Hamilton Jacobi Bellman equation to construct inner and outer bounds on reachable sets. arXiv preprint arXiv:1903.07274, 2019.
[14] Morgan Jones and Matthew M Peet. Using SOS and sublevel set volume minimization for estimation of forward reachable sets. arXiv preprint arXiv:1901.11174, 2019.
[15] Duan Li. Multiple objectives and non-separability in stochastic dynamic programming. *International Journal of Systems Science*, 21(5):933–950, 1990.

[16] Duan Li and Yacov Y Haimes. Multilevel methodology for a class of non-separable optimization problems. *International Journal of Systems Science*, 21(11):2351–2360, 1990.

[17] Duan Li and Yacov Y Haimes. New approach for nonseparable dynamic programming problems. *Journal of Optimization Theory and Applications*, 64(2):311–330, 1990.

[18] Duan Li and Yacov Y Haimes. Extension of dynamic programming to nonseparable dynamic optimization problems. *Computers & Mathematics with Applications*, 21(11-12):51–56, 1991.

[19] Qiming Liu, Ming Dong, Wenyuan Lv, and Chunming Ye. Manufacturing system maintenance based on dynamic programming model with prognostics information. *Journal of Intelligent Manufacturing*, 30(3):1155–1173, 2019.

[20] John Maidens, Axel Barrau, Silvère Bonnabel, and Murat Arcak. Symmetry reduction for dynamic programming. *Automatica*, 97:367–375, 2018.

[21] John Maidens, Andrew Packard, and Murat Arcak. Parallel dynamic programming for optimal experiment design in nonlinear systems. In *Conference on Decision and Control (CDC)*, pages 2894–2899. IEEE, 2016.

[22] Eran Rippel, Aharon Bar-Gill, and Nahum Shimkin. Fast graph-search algorithms for general-aviation flight trajectory generation. *Journal of Guidance, Control, and Dynamics*, 28(4):801–811, 2005.

[23] Andrzej Ruszczyński. Risk-averse dynamic programming for markov decision processes. *Mathematical programming*, 125(2):235–261, 2010.

[24] Alexander Shapiro. On a time consistency concept in risk averse multistage stochastic programming. *Operations Research Letters*, 37(3):143–147, 2009.

[25] Alexander Shapiro and Kerem Ugurlu. Decomposability and time consistency of risk averse multistage programs. *Operations Research Letters*, 44(5):663–665, 2016.

[26] Zheming Wang, Raphaël M Jungers, and Chong-Jin Ong. Computation of the maximal invariant set of discrete-time systems subject to quasi-smooth non-convex constraints. *arXiv preprint arXiv:1912.09727*, 2019.

[27] Junfei Xie, Lei Jin, and Luis Rodolfo Garcia Carrillo. Optimal path planning for unmanned aerial systems to cover multiple regions. In *AIAA Scitech 2019 Forum*, page 1794, 2019.

[28] Bai Xue and Naijun Zhan. Robust invariant sets computation for switched discrete-time polynomial systems. *arXiv preprint arXiv:1811.11454*, 2018.

[29] Xiangrui Zeng and Junmin Wang. Globally energy-optimal speed planning for road vehicles on a given route. *Transportation Research Part C: Emerging Technologies*, 93:148–160, 2018.

Morgan Jones received the B.S. and MMath in mathematics from The University of Oxford, England in 2016. He is a research associate with Cybernetic Systems and Controls Laboratory (CSCL) in the School for Engineering of Matter, Transport and Energy (SEMTE) at Arizona State University. His research primarily focuses on the estimation of attractors, reachable sets and regions of attraction for nonlinear ODE’s. Furthermore, He has been studying the effects of optimal energy storage in smart grid environments.

Matthew M. Peet received the B.S. degree in physics and in aerospace engineering from the University of Texas, Austin, TX, USA, in 1999 and the M.S. and Ph.D. degrees in aeronautics and astronautics from Stanford University, Stanford, CA, in 2001 and 2006, respectively. He was a Postdoctoral Fellow at INRIA, Paris, France from 2006 to 2008. He was an Assistant Professor of Aerospace Engineering at the Illinois Institute of Technology, Chicago, IL, USA, from 2008 to 2012. Currently, he is an Associate Professor of Aerospace Engineering at Arizona State University, Tempe, AZ, USA. Dr. Peet received a National Science Foundation CAREER award in 2011.