

Hamilton–Jacobi–Bellman Equations and Approximate Dynamic Programming on Time Scales

John Seiffertt, *Student Member, IEEE*, Suman Sanyal, and Donald C. Wunsch, II, *Fellow, IEEE*

Abstract—The time scales calculus is a key emerging area of mathematics due to its potential use in a wide variety of multi-disciplinary applications. We extend this calculus to approximate dynamic programming (ADP). The core backward induction algorithm of dynamic programming is extended from its traditional discrete case to all isolated time scales. Hamilton–Jacobi–Bellman equations, the solution of which is the fundamental problem in the field of dynamic programming, are motivated and proven on time scales. By drawing together the calculus of time scales and the applied area of stochastic control via ADP, we have connected two major fields of research.

Index Terms—Approximate dynamic programming (ADP), dynamic equations, Hamilton–Jacobi–Bellman (HJB) equation, reinforcement learning, time scales.

I. INTRODUCTION

THE MATHEMATICS of time scales bridges the divide between the discrete and the continuous [31]. This calculus provides a unified framework for the analysis of difference and differential equations. Such dynamic equations on time scales [17], [18] have been applied in population biology [12], quantum calculus [13], geometric analysis [30], real-time communication networks [26], intelligent robotic control [27], adaptive sampling [28], approximation theory [45], financial engineering [44], and switched linear circuits [37] among others. These fields are ideal for approximate dynamic programming (ADP) [40]. ADP seeks the solutions of the Hamilton–Jacobi–Bellman (HJB) equation [7]. In discrete time, backward induction is often used. We extend this method to all isolated time scales and then prove versions of the HJB equation on general time scales.

The organization of this paper is as follows. Section II presents the core definitions and concepts of the time-scale calculus. Section III focuses on the Bellman’s optimality principle and the dynamic programming backward induction algorithm. The HJB equations on time scales are proven in Section IV, and Section V concludes this paper with perspectives on future directions.

Manuscript received September 10, 2007; revised February 23, 2008. This work was supported in part by the Mary K. Finley Missouri Endowment, the Brian Blaha Memorial Scholarship, the Intelligent Systems Center, and the National Science Foundation. This paper was recommended by Guest Editor D. Liu.

J. Seiffertt and D. C. Wunsch, II are with the Applied Computational Intelligence Laboratory, Department of Electrical and Computer Engineering, Missouri University of Science and Technology, Rolla, MO 65409 USA (e-mail: jes0b4@mst.edu; Wunsch@ieee.org).

S. Sanyal is with the Department of Mathematics and Statistics, Missouri University of Science and Technology, Rolla, MO 65409 USA (e-mail: ssrgd@mst.edu).

Digital Object Identifier 10.1109/TSMCB.2008.923532

II. TIME SCALES PRELIMINARIES

A. Fundamental Definitions

A time scale \mathbb{T} is any nonempty closed subset of the real line \mathbb{R} . Examples include the integers \mathbb{Z} , the scaled integers $h\mathbb{Z}$ (where $h > 0$ is a scaling factor), the quantum calculus time scale $\mathbb{T} = q^{\mathbb{N}_0}$ (with $q > 1$) [41], as well as more exotic constructs such as the Cantor set. The quantum time scale is of particular importance as it yields the continuous case in the limit as $q \rightarrow 1^+$. It is often used as an initial time scale in the analysis of a particular equation or modeling framework.

Two characteristic functions are defined on a time scale \mathbb{T} . The forward jump operator σ is defined as $\sigma(t) = \inf\{x \in \mathbb{T} : x > t\}$. This function returns the “next” element of the time scale in the sense given by its definition. When $\mathbb{T} = \mathbb{R}$, $\sigma(t) = t$; if $\mathbb{T} = \mathbb{Z}$, $\sigma(t) = t + 1$; and on the q -time scale $\mathbb{T} = q^{\mathbb{N}_0}$, $\sigma(t) = qt$. The backward jump operator ρ is defined as $\rho(t) = \sup\{x \in \mathbb{T} : x < t\}$. A point $t \in \mathbb{T}$ is said to be right scattered if $\sigma(t) > t$ and said to be left scattered if $\rho(t) < t$. If a point is right and left scattered, then it is said to be isolated. Discrete points are isolated. If $t < \sup \mathbb{T}$ and $\sigma(t) = t$, then t is right dense. If $t > \inf \mathbb{T}$ and $\rho(t) = t$, then t is left dense. If t is both right and left dense, then t is said to be dense. Finally, it is necessary to define \mathbb{T}^κ as follows: If \mathbb{T} has a left-scattered maximum point m , then $\mathbb{T}^\kappa = \mathbb{T} - \{m\}$; otherwise, $\mathbb{T}^\kappa = \mathbb{T}$. This set is used in the definition of the delta derivative on time scales.

A time scale \mathbb{T} has an associated graininess function μ given by $\mu(t) = \sigma(t) - t$. Note that when $\mathbb{T} = \mathbb{Z}$, $\mu(t) \equiv 1$; if $\mathbb{T} = \mathbb{R}$, $\mu(t) \equiv 0$; for $\mathbb{T} = q^{\mathbb{N}_0}$, $\mu(t) = (q - 1)t$; and when $\mathbb{T} = h\mathbb{Z}$, $\mu(t) = h$. Many formulas of the time-scale calculus will involve the graininess function μ .

B. Time Scales Calculus of a Single Variable

Let $f : \mathbb{T} \rightarrow \mathbb{R}$ be a function. Then, the delta derivative $f^\Delta(t)$ of f at a point $t \in \mathbb{T}^\kappa$ is defined to be the number such that given $\epsilon > 0$, there is a neighborhood U of t such that

$$|[f(\sigma(t)) - f(s)] - f^\Delta(t)[\sigma(t) - s]| \leq \epsilon|\sigma(t) - s|$$

for all $s \in U$, where neighborhood is defined such that $U = (t - \delta, t + \delta)$ for some $\delta > 0$. Note that this follows the classical definition of the derivative, with the traditional $x + h$ increment replaced by the forward jump operator $\sigma(t)$. This sort of translation is a common theme in the calculus of time scales. The delta derivative $f^\Delta(t)$ becomes $f'(t)$ when $\mathbb{T} = \mathbb{R}$ and becomes the standard difference operator on $\mathbb{T} = \mathbb{Z}$. Other derivatives, such as the alpha, nabla, and diamond- α , can also be defined on time scales [2], [17], [42]. Whereas the role of these other derivatives is still emerging in computational optimization theory, our results will focus on the delta derivative.

Many of the classic rules for differentiation obtain for the delta derivative [17]. However, the traditional chain rule fails to extend to general time scales. Instead, this calculus admits a suite of chain rules. The most general and applicable chain rule for time scales is due to Potzsche [18], [39]. Let $f: \mathbb{R} \rightarrow \mathbb{R}$ be continuously differentiable and let $g: \mathbb{T} \rightarrow \mathbb{R}$ be delta differentiable. Then, $f(g(t))$ is delta differentiable, with $(f \circ g)^\Delta(t)$ given by the following:

$$g^\Delta(t) \int_0^1 f'(g(t) + h\mu(t)g^\Delta(t)) dh. \quad (1)$$

When $\mathbb{T} = \mathbb{R}$, then $\mu(t) \equiv 0$ and the equation reduces to the chain rule of traditional calculus. The fundamental theorem of calculus is valid on time scales and takes a familiar form. Provided a function f satisfies certain conditions (regulated and rd continuous), then

$$\int_a^b f^\Delta(\tau) \Delta\tau = f(b) - f(a). \quad (2)$$

For a thorough overview of integration theory on time scales, including definitions of regulated and rd continuity, the interested reader is directed to [15], [17], and [29].

C. Time-Scale Calculus of Multiple Variables

We use a definition of partial derivatives on time scales given by Jackson [34]. Let $\mathbb{T}_1, \mathbb{T}_2, \dots, \mathbb{T}_n$ be time scales, set $\mathbb{T} = \mathbb{T}_1 \times \mathbb{T}_2 \times \dots \times \mathbb{T}_n$, and let $f: \mathbb{T} \rightarrow \mathbb{R}$ be a function. Define the operators on \mathbb{T} as $\sigma(t) = (\sigma(t_1), \sigma(t_2), \dots, \sigma(t_n))$ and $\rho(t) = (\rho(t_1), \rho(t_2), \dots, \rho(t_n))$. Also, define $\mathbb{T}^\kappa = \mathbb{T}_1^\kappa \times \mathbb{T}_2^\kappa \times \dots \times \mathbb{T}_n^\kappa$, $f^{\sigma_i(t)} = f(t_1, \dots, t_{i-1}, \sigma_i(t_i), t_{i+1}, \dots, t_n)$, and $f_i^s = f(t_1, \dots, t_{i-1}, s, t_{i+1}, \dots, t_n)$.

The partial delta derivative of f at t with respect to t_i is the number f^{Δ_i} , provided that it exists, such that given any $\epsilon > 0$, there exists a neighborhood U of t_i for $\delta > 0$ such that

$$|[f^{\sigma_i(t)} - f_i^s] - f^{\Delta_i}(t) [\sigma_i(t) - s]| \leq \epsilon |\sigma_i(t) - s|$$

for all $s \in U$, where neighborhood is defined such that $U = (t_i - \delta, t_i + \delta) \cap \mathbb{T}_i$.

Necessary for our proof of the HJB equation on time scales will be the chain rule for partial derivatives given by Bohner and Guseinov [16]. Let $t \in \mathbb{T}$, $x: \mathbb{T} \rightarrow \mathbb{R}$, $y: \mathbb{T} \rightarrow \mathbb{R}$, $x(\mathbb{T}) = \mathbb{T}_x$, $y(\mathbb{T}) = \mathbb{T}_y$, and $F(x(t), y(t))$. Assume that $x(\sigma(t)) = \sigma_x(x(t))$ and $y(\sigma(t)) = \sigma_y(y(t))$. If $F = f(x(t), y(t))$ is σ -completely differentiable and x and y are differentiable, then

$$F^\Delta(t) = f^{\Delta_x}(x(t), y(t)) x^\Delta(t) + f^{\Delta_y}(\sigma_x(x(t)), y(t)) y^\Delta(t). \quad (3)$$

For more details on partial derivatives on time scales, including a definition of σ_x -complete differentiability, the reader is directed to [3] and [16].

D. Principle of Induction on Time Scales

A form of backward induction exists on time scales [18]. Let $t_0 \in \mathbb{T}$ and $S(t)$ be a statement for each $t \in [-\infty, t_0)$ such that the following four conditions hold.

- 1) $S(t_0)$ is true.

- 2) $S(t)$, being true at a left-scattered t , forces $S(\rho(t))$ to be true.
- 3) $S(t)$, being true at a left-dense t , forces $S(t')$ to be true for all t' in a left neighborhood of t .
- 4) $S(t')$, being true for all $t' \in (t_0, t]$ when t is right-dense, forces $S(t)$ to be true.

Then, it can be concluded that $S(t)$ is true for all $t \in [t_0, \infty)$. There is also a forward version involving right-scattered and left-dense intervals and the forward jump operator $\sigma(t)$, but it is this backward form which we use in the next section.

III. DYNAMIC PROGRAMMING ALGORITHM

We extend the backward induction algorithm of dynamic programming to time scales. We define the problem, discuss the principle of optimality, and then prove our result.

A. System Definitions

The elements of our dynamic programming problem are as follows: a time scale \mathbb{T} housing our decision points, controls $c(x(t), t)$, a stochastic disturbance $w(t)$, states $x(t)$ which evolve according to a rule $f(x(t), c(x(t), t), w(t), t)$, and a cost/reward $r(x(t), c(x(t), t), w(t), t)$ where the cost at a terminal decision point T is piecewise defined as $r_T(x(T))$. A policy π is a set of state-control pairs for each point in \mathbb{T} such that each control is valid for both the state and time. We denote by π^t the tail of the policy π beginning with time step t . We also require a cost-to-go function given by the following:

$$J_\pi(x(t_0)) = \mathbb{E} \left\{ \int_{t_0}^T r(x(\tau), c(x(\tau), \tau), w(\tau), \tau) \Delta\tau \right\} \quad (4)$$

which measures the expected cost of a policy π . We assume that the expected values are finite and well defined.

We limit $w(t)$ to take on values in a countable set. Whereas this constraint prohibits disturbances such as Gaussian noise or Brownian motion, it does permit useful application of this model. For example, the representation of state-space systems in Markov decision processes [41] gives the $w(t)$ the form of transition probabilities $P(i, j, c)$, which indicate the probability that the system evolves from state x_i to state x_j in response to control signal c . Such $w(t)$'s are countable.

We consider the following dynamical system defined on a time scale \mathbb{T} :

$$x^\Delta(t) = f(x(t), c(x(t), t), w(t), t) \quad (5)$$

where $t \in \mathbb{T}$. Our task is to calculate a policy π^* that minimizes the cost-to-go function J_π . We call such a π^* an optimal policy and denote the optimal cost to go as $J^*(x(t), t_0) = \min_\pi J_\pi(x(t_0), t_0)$, where the minimum is considered over all policies.

B. Principle of Optimality

Bellman's principle of optimality [7], [8] aids in the solution to the aforementioned optimization problem. This principle can be stated in the following way. Let π be an optimal policy. Then,

the optimal policy for the tail problem starting at time t_n , which is to minimize

$$\mathbb{E} \left\{ \int_{t_n}^T r(x(\tau), c(x(\tau), \tau), w(\tau), \tau), \Delta\tau) \right\} \quad (6)$$

is equal to the portion of π^* which overlaps π^{t_n} . To justify this principle, note that if it were not true, then the tail of π^* could be replaced by a more optimal π^{t_n} , thus contradicting the claim of optimality of π^* .

C. Backward Induction

The dynamic programming algorithm, which is a form of backward induction, involves the stochastic optimization of control selection starting from the terminal time point T . Beginning with setting $J(x(T), T)$, the algorithm proceeds via the following update rule:

$$J(x(t), t) = \min_c \mathbb{E} \left\{ r(x(t), c(x(t), t), t, w(t), t) + J(f(x(t), c(x(t), t), w(t), t), \sigma(t)) \right\}$$

for $t \in \mathbb{T}$. This rule says that the cost to go of the current state $x(t)$ under a control $c(x(t), t)$ equals the expected value of the immediate cost $r(x(t), c(x(t), t), t, w(t), t)$ plus the future costs $J(f(x(t), c(x(t), t), w(t), t), \sigma(t))$. Recall that $\sigma(t)$ is the “next” point in our time scale \mathbb{T} .

It is standard to discuss the optimality of this algorithm in terms which assume convergence [9], [10]. Our proof, following [9], declares controls that are optimal if they minimize the update rule. Convergence issues occupy a vast literature and are thoroughly explored in the classic text [41].

The classical version of this update rule is true for the discrete time scale $\mathbb{T} = \{1, 2, \dots, T\}$. We extend this result to any isolated time scale \mathbb{T} in advance of our derivation of the HJB equation on more general time scales.

Theorem 1 (Dynamic Programming Algorithm): If $c^*(x(t), t)$ minimizes the update expression given previously for each state and for all $t \in \mathbb{T}$, then the policy $c^*(x(t), t)$ is optimal.

Proof: We set $J^*(x(T), T) = r_T(x(T))$ and proceed via time-scale induction to show that the application of the dynamic programming algorithm’s recursive update equations yields the optimal policy at each stage, i.e., that $J^*(x(t), t) = J(x(t), t)$ for all $t \in \mathbb{T}$. Letting $t = T$ yields, by definition

$$J^*(x(T), T) = r_T(x(T)) = J(x(T), T). \quad (7)$$

Now, assume $J^*(x(t), t) = J(x(t), t)$ for some time point $t \in \mathbb{T}$ and all states $x(t)$. To apply the backward induction algorithm, recall that in a time scale, $\rho(t)$ is the point that comes just “before” point t . Therefore, the quantity $\rho(t)$ plays a central role in our discussion. To wit, (8) gives the immediate one-step application of our update rule

$$J^*(x(\rho(t)), \rho(t)) = \min_{c, \pi} \mathbb{E} \left\{ r(x(t), c(x(t), t), w(t), \rho(t)) + \int_t^T r(x(\tau), c(x(\tau), \tau), w(\tau), \tau), \Delta\tau) \right\}. \quad (8)$$

The integral represents the value of the cost-to-go function J at the “next” time step after $\rho(t)$, which is t . Also, note that the minimization is taken term by term over all controls and policies. We now use the principle of optimality to distribute the min through the expectation, as the tail problem is indeed an optimal policy for the problem contained within the tail. This yields the second equation

$$J^*(x(\rho(t)), \rho(t)) = \min_c \mathbb{E} \left\{ r(x(t), c(x(t), t), w(t), \rho(t)) + \min_{\pi} \mathbb{E} \left\{ \int_t^T r(x(\tau), c(x(\tau), \tau), w(\tau), \tau), \Delta\tau) \right\} \right\}.$$

Using the definition of $J^*(x(t), t)$, which subsumes the term minimized over the policy, we can reduce this expression to the following:

$$J^*(x(\rho(t)), \rho(t)) = \min_c \mathbb{E} \left\{ r(x(t), c(x(t), t), w(t), \rho(t)) + J^*(x(t), t) \right\}. \quad (9)$$

By the induction hypothesis, we know that the optimal cost to go $J^*(x(t), t)$ is equivalent to the approximation $J(x(t), t)$ due to the dynamic programming algorithm. Thus, we write $J^*(x(\rho(t)), \rho(t))$ as follows:

$$\min_c \mathbb{E} \{ r(x(t), c(x(t), t), w(t), \rho(t)) + J(x(t), t) \} \quad (10)$$

which, by definition, is simply

$$J^*(x(\rho(t)), \rho(t)) = J(x(\rho(t)), \rho(t)). \quad (11)$$

We have now satisfied conditions 1 and 2 of the principle of backward induction on time scales given in Section II. Because we assume \mathbb{T} to be isolated, conditions 3 and 4 do not apply, and we conclude that, by backward induction on time scales, we have proven our claim. ■

Thus, the dynamic programming algorithm is extended to time scales. The computational requirements for implementing this algorithm, particularly for industrial-scale optimization problems common in operations research, are great [40]. It is the task of ADP to calculate suboptimal policies in an efficient manner while simultaneously satisfying the needs of a given application. Within a time-scale framework, this approach is also valid as the optimal update rule underlying the approximations holds.

IV. HJB EQUATIONS

We prove the HJB equation on time scales and discuss other forms that this equation may take. Note that whereas our proof of the dynamic programming algorithm holds on isolated time scales, no such restriction is required for the HJB equation proper. Instead, we may consider decision problems on arbitrary time scales using the results of this section.

A. HJB Equation on Time Scales

Consider the dynamical system given by the following:

$$x^\Delta(t) = f(x(t), c(t)) \quad (12)$$

where x represents states and c is the control. Let $t \in \mathbb{T}$, $x: \mathbb{T} \rightarrow \mathbb{R}$, and $x(\mathbb{T}) = \mathbb{T}_x$. The cost-to-go function $J: \mathbb{T}_x \times \mathbb{T} \rightarrow \mathbb{R}$ is given by the following:

$$J(x(t_0), t_0) = \int_{t_0}^T r(x(\tau), c(\tau)) \Delta\tau \quad (13)$$

where t_0 is the initial decision point and $r(x(t), c(t))$ is the cost.

Assume that J is delta differentiable and x is σ_x -completely delta differentiable. Furthermore, require x to satisfy the following condition of forward compositional commutativity:

$$x(\sigma(t)) = \sigma_x(x(t)). \quad (14)$$

Then, the HJB equation on time scales is given by the following:

$$0 = \min_c \left\{ r(x(t), c(x(t), t)) + J^{\Delta t}(x(t), t) + J^{\Delta x}(x(t), \sigma(t)) f(x(t), t) \right\}. \quad (15)$$

This is an equation that any optimal policy of our minimization problem must satisfy. Because precious few industrial-scale applications admit an analytic solution of this equation, ADP is employed to develop approximation techniques for this purpose. The proof of this equation is our next theorem. For the classical proofs, see a reference such as [5], [9], and [24].

The Hamilton–Jacobi equation is a result of the calculus of variations [25], [43], and work extending this calculus to time scales is only just beginning [4], [11], [14], [23], [32]. These problems typically take the general form of minimizing the cost functional given by the following integral [11]:

$$J(y) = \int_a^b L(\tau, y^\sigma(\tau), y^\Delta(\tau)) \Delta\tau. \quad (16)$$

From this, the usual Euler and Legendre conditions can be derived on time scales. Our next result takes this a step further and proves the Hamilton–Jacobi equation for an alternate version, which is given by (13), of the aforementioned integral (16). Because (13) is the common cost functional of dynamic programming, the resulting equation is given the name Hamilton–Jacobi–Bellman. In this way, the following theorem is a contribution to the development of the calculus of variations on time scales as well as to ADP. However, as we prove the HJB equation for a form other than that given by (16), there is still work to be done on Hamilton–Jacobi equations for more generalized cost functionals.

Such variational problems are central to control theory, as are the issues of controllability and observability. The interested reader is directed to [6], [19]–[22], [33], [36], and [38] for more

details on these important topics both in the real case and for a general time scale \mathbb{T} .

Theorem 2 (HJB Equation): Let $V(x(t), t)$ be a solution to (15) such that

$$0 = \min_c \left\{ r(x(t), c(x(t), t)) + V^{\Delta t}(x(t), t) + V^{\Delta x}(x(t), \sigma(t)) f(x(t), t) \right\}. \quad (17)$$

Assume that the boundary condition $V(x(T), T) = r_T(x(T))$ and $\hat{x}(t_0) = x(t_0)$, and suppose $c^*(x(t), t)$ attains the minimum called for in (15) for all states and time. Let $x^*(t)$ be the state trajectory, subject to the condition $x^*(t_0) = x(t_0)$, which corresponds to applying the controls $c^*(x(t), t)$ at each decision point t . Then, the function $V(x(t), t)$ is the optimal cost-to-go function $J^*(x(t), t)$, and the control $c^*(x(t), t)$ is optimal.

Proof: Let $\hat{c}(x(t), t)$ be a control policy with corresponding state trajectory $\hat{x}(t)$. We will show that the policy $c^*(x(t), t)$ achieves a cost that is no greater than this arbitrary $\hat{c}(x(t), t)$, thus forcing $c^*(x(t), t)$ to be our optimal control. We begin by invoking (17) to give us

$$0 \leq r(\hat{x}(t), \hat{c}(\hat{x}(t), t)) + V^{\Delta t}(\hat{x}(t), t) + V^{\Delta x}(\hat{x}(t), \sigma(t)) f(\hat{x}(t), t). \quad (18)$$

Noting that, via (12), we have $x^\Delta(t) = f(x(t), c(t))$, we can rewrite (18) as follows:

$$0 \leq r(\hat{x}(t), \hat{c}(\hat{x}(t), t)) + V^{\Delta t}(\hat{x}(t), t) + V^{\Delta x}(\hat{x}(t), \sigma(t)) x^\Delta(t) \quad (19)$$

and by reversing the chain rule implicit in this formulation, we arrive at the following:

$$0 \leq r(\hat{x}(t), \hat{c}(\hat{x}(t), t)) + V^\Delta(t). \quad (20)$$

Integrating over our time horizon yields

$$0 \leq \int_{t_0}^T r(\hat{x}(\tau), \hat{c}(\hat{x}(\tau), \tau)) \Delta\tau + \int_{t_0}^T V^\Delta(\tau) \Delta\tau. \quad (21)$$

Using the fundamental theorem (2), we arrive at the following:

$$0 \leq \int_{t_0}^T r(\hat{x}(\tau), \hat{c}(\hat{x}(\tau), \tau)) \Delta\tau + V(\hat{x}(T), T) - V(\hat{x}(t_0), t_0).$$

Substituting in our boundary conditions $V(x(T), T) = r_T(x(T))$ and $\hat{x}(t_0) = x(t_0)$ gives us

$$0 \leq \int_{t_0}^T r(\hat{x}(\tau), \hat{c}(\hat{x}(\tau), \tau)) \Delta\tau + r_T(\hat{x}(T)) - V(x(t_0), t_0)$$

which is equal to the following:

$$V(x(t_0), t_0) \leq \int_{t_0}^T r(\hat{x}(\tau), \hat{c}(\hat{x}(\tau), \tau)) \Delta\tau + r_T(\hat{x}(T)).$$

From our hypothesis, we assume that the controls $c^*(x(t), t)$ and their corresponding state trajectory $x^*(t)$ minimize the value function $V(x(t), t)$. Using this information and the initial condition $x^*(t_0) = x(t_0)$, we can replace the inequality with equality in the case of these quantities

$$V(x(t_0), t_0) = \int_{t_0}^T r(x^*(\tau), c^*(x^*(\tau), \tau)) \Delta\tau + r_T(x^*(T)).$$

Combining with the previous equation, we have

$$\begin{aligned} \int_{t_0}^T r(x^*(\tau), c^*(x^*(\tau), \tau)) \Delta\tau + r_T(x^*(T)) \\ \leq \int_{t_0}^T r(\hat{x}(\tau), \hat{c}(\hat{x}(\tau), \tau)) \Delta\tau + r_T(\hat{x}(T)). \end{aligned}$$

This equation tells us that the cost of the policy $c^*(x(t), t)$ is less than or equal to the cost of any admissible policy $\hat{c}(x(t), t)$. We conclude that the policy $c^*(x(t), t)$ is optimal and that, because $\hat{c}(x(t), t)$ is arbitrary, we have $V(x(t), t) = J^*(x(t), t)$. Therefore, any optimal policy must satisfy the HJB equation given by (15). ■

B. Other Forms of the HJB Equation on Time Scales

The calculus of time scales admits many different chain rules depending on various conditions on the functions of interest. The key step in our proof of the HJB equation was in the reversal of the chain rule. In principle, given any chain rule we can derive a different form of the HJB equation and the proof from (20) onward will remain unchanged. For example, we assume the σ -complete differentiability of $x(t)$. If we instead assume that $x(t)$ is σ_x -completely differentiable, we obtain, by a different chain rule of Bohner and Guseinov [16], the following form of the HJB equation:

$$0 = \min_c [r(x(t), c(x(t), t)) + J^{\Delta t}(\sigma_x(x(t), t)) + J^{\Delta x}(x(t), t) f(x(t), t)]. \quad (22)$$

V. CONCLUSION AND FUTURE DIRECTIONS

The time scales calculus is an increasingly relevant and developed area of mathematics with wide-ranging opportunities for application. We have established that the dynamic programming algorithm, which is derived from Bellman's principle of optimality, obtains on time scales. We have also derived the HJB equation on time scales and demonstrated that a family of such equations exists. The solution of such an equation is the fundamental goal in ADP. We identify three significant directions that the investigation of ADP on time scales can take. First, as the derivation of the HJB equation was dependent on the mathematics of the time-scale calculus of multiple variables in general, and the chain rule in particular, further variations and extensions in this area will prove critical. The generalized Stokes theorem, principles of the variational calculus, and more complete chain rules are three areas where new contributions are of exceptional need.

Second, numerical approximation work in time scales remains a promising endeavor. With the availability of computational resources such as the Time Scales MatLab Toolbox from the Baylor University Time Scales Group, both applied and theoretical investigations into the numerics of time-scale calculus can be pursued. Numerical differentiation and integration techniques on time scales would provide significant value, as would time-scale extensions of optimization algorithms, be they population-based models from the computational intelligence literature or provably convergent methods from applied mathematics [1]. Also, of need are demonstrations of ADP-based controllers operating in a time-scale framework. This brings us to our third direction for growth: applications.

In addition to the electric circuit and population biology models to which time scales have been applied [18], the field of time-scale control needs to show significant capability to upgrade to larger scale problems. Analysis of technical trading rules, macroeconomic dynamical learning models, and monetary policy are areas in economics and finance where controllers operating on time scales would be of great interest. Financial portfolio management may be a particularly natural fit, as control is applied during intervals broken up by overnight periods during which the system evolves but the controller has no direct influence.

It is our position that while the study of time scales can provide a concise theoretical unification of control theory in the discrete and continuous cases, it can also provide utility gains in certain problem domains. We believe that there are important application areas where dealing simultaneously with discrete and continuous variables is critical [46], [47] and that the time-scale calculus provides a natural and powerful framework for such exploration.

ACKNOWLEDGMENT

The authors would like to thank Martin Bohner, Department of Mathematics and Statistics, Missouri University of Science and Technology for the helpful discussions.

REFERENCES

- [1] M. A. Abramson and C. Audet, "Convergence of mesh adaptive direct search to second-order stationary points," *SIAM J. Optim.*, vol. 17, no. 2, pp. 606–619, Aug. 2006.
- [2] C. D. Ahlbrandt, M. Bohner, and J. Ridenhour, "Hamiltonian systems on time scales," *J. Math. Anal. Appl.*, vol. 250, no. 2, pp. 561–578, Oct. 2000.
- [3] C. D. Ahlbrandt and C. Morian, "Partial differential equations on time scales," *J. Comput. Appl. Math.*, vol. 141, no. 1, pp. 35–55, Apr. 2002.
- [4] F. M. Atici, D. C. Biles, and A. Lebedinsky, "An application of time scales to economics," *Math. Comput. Model.*, vol. 43, no. 7/8, pp. 718–726, Apr. 2006.
- [5] M. Bardi and I. Capuzzo-Dolcetta, *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*, ser. Systems & Control: Foundations & Applications. Boston, MA: Birkhäuser, 1997. With Appendices by Maurizio Falcone and Pierpaolo Soravia.
- [6] T. Başar and G. J. Olsder, *Dynamic Noncooperative Game Theory*, ser. Classics in Applied Mathematics, vol. 23. Philadelphia, PA: SIAM, 1999. reprint of the second (1995) edition.
- [7] R. Bellman, *Dynamic Programming*. Princeton, NJ: Princeton Univ. Press, 1957.
- [8] R. E. Bellman and S. E. Dreyfus, *Programarea Dinamică Aplicată*, ser. Translated from the English. Bucharest, Romania: Editura Tehnică, 1967.
- [9] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 2nd ed, vol. I. Belmont, MA: Athena Scientific, 2001.
- [10] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*, ser. Optimization and Neural Computation Series, 3. Belmont, MA: Athena Scientific, 1996.

- [11] M. Bohner, "Calculus of variations on time scales," *Dyn. Syst. Appl.*, vol. 13, no. 3/4, pp. 339–349, 2004.
- [12] M. Bohner, M. Fan, and J. Zhang, "Periodicity of scalar dynamic equations and applications to population models," *J. Math. Anal. Appl.*, vol. 330, no. 1, pp. 1–9, Jun. 2007.
- [13] M. Bohner and T. Hudson, "Euler-type boundary value problems in quantum calculus," *Int. J. Appl. Math. Stat.*, vol. 9, no. J07, pp. 19–23, Jun. 2007.
- [14] M. Bohner and G. S. Guseinov, "Double integral calculus of variations on time scales," *Comput. Math. Appl.*, vol. 54, no. 1, pp. 45–57, Jul. 2007.
- [15] M. Bohner and G. S. Guseinov, "Multiple integration on time scales," *Dyn. Syst. Appl.*, vol. 14, no. 3/4, pp. 579–606, 2005.
- [16] M. Bohner and G. S. Guseinov, "Partial differentiation on time scales," *Dyn. Syst. Appl.*, vol. 13, no. 3/4, pp. 351–379, 2004.
- [17] M. Bohner and A. Peterson, Eds., *Advances in Dynamic Equations on Time Scales*, Boston MA: Birkhäuser 2003.
- [18] M. Bohner and A. Peterson, *Dynamic Equations on Time Scales: An Introduction With Applications*. Boston, MA: Birkhäuser, 2001.
- [19] J. J. DaCunha, "Instability results for slowly time varying linear dynamic systems on time scales," *J. Math. Anal. Appl.*, vol. 328, no. 2, pp. 1278–1289, Apr. 2007.
- [20] J. J. DaCunha, "Stability for time varying linear dynamic systems on time scales," *J. Comput. Appl. Math.*, vol. 176, no. 2, pp. 381–410, Apr. 2005.
- [21] J. J. DaCunha, J. M. Davis, and P. K. Singh, "Existence results for singular three point boundary value problems on time scales," *J. Math. Anal. Appl.*, vol. 295, no. 2, pp. 378–391, Jul. 2004.
- [22] P. W. Eloe, S. Hilger, and Q. Sheng, "A qualitative analysis on nonconstant graininess of the adaptive grids via time scales," *Rocky Mt. J. Math.*, vol. 36, no. 1, pp. 115–133, 2006.
- [23] R. Ferreira and D. Torres, "Remarks on the calculus of variations on time scales," *Int. J. Econ. Ecol. Stat.*, vol. 9, no. F07, pp. 65–73, Sep. 2007.
- [24] W. H. Fleming and R. W. Rishel, "Deterministic and stochastic optimal control," in *Applications of Mathematics*, vol. 1. Berlin, Germany: Springer-Verlag, 1975.
- [25] I. M. Gelfand and S. V. Fomin, *Calculus of Variations*, ser. Revised English Edition Translated, R. A. Silverman, Ed. Englewood Cliffs, NJ: Prentice-Hall, 1963.
- [26] I. Gravagne, R. J. Marks, II, J. Davis, and J. DaCunha, "Application of time scales to real world time communications networks," in *Proc. Amer. Math. Soc. West. Sect. Meeting*, 2004.
- [27] I. Gravagne, J. Davis, R. J. Marks, II, "How deterministic must a real-time controller be?" in *Proc. IEEE/RSJ Int. Conf.*, Edmonton, AB, Canada, Aug. 2005, pp. 3856–3861.
- [28] I. Gravagne, J. Davis, J. DaCunha, and R. J. Marks, II, "Bandwidth reduction for controller area networks using adaptive sampling," in *Proc. Int. Conf. Robot. Autom.*, New Orleans, LA, Apr. 2004, pp. 2–6.
- [29] G. S. Guseinov, "Integration on time scales," *J. Math. Anal. Appl.*, vol. 285, no. 1, pp. 107–127, Sep. 2003.
- [30] G. S. Guseinov and E. Özyý İmaz, "Tangent lines of generalized regular curves parametrized by time scales," *Turkish J. Math.*, vol. 25, no. 4, pp. 553–562, 2001.
- [31] S. Hilger, "Analysis on measure chains—A unified approach to continuous and discrete calculus," *Results Math.*, vol. 18, no. 1/2, pp. 18–56, 1990.
- [32] R. Hilscher and V. Zeidan, "Calculus of variations on time scales: Weak local piecewise C_{rd}^1 solutions with variable endpoints," *J. Math. Anal. Appl.*, vol. 289, no. 1, pp. 143–166, Jan. 2004.
- [33] D. G. Hull, *Optimal Control Theory for Applications*, ser. Mechanical Engineering Series. New York: Springer-Verlag, 2003.
- [34] B. Jackson, "Partial dynamic equations on time scales," *J. Comput. Appl. Math.*, vol. 186, no. 2, pp. 391–415, Feb. 2006.
- [35] V. Kac and P. Cheung, *Quantum Calculus*, ser. Universitext. New York: Springer-Verlag, 2002.
- [36] F. L. Lewis, *Optimal Control*, ser. A Wiley-Interscience Publication. New York: Wiley, 1986.
- [37] R. J. Marks, II, I. A. Gravagne, J. M. Davis, and J. J. DaCunha, "Non-regressivity in switched linear circuits and mechanical systems," *Math. Comput. Model.*, vol. 43, no. 11/12, pp. 1383–1392, Jun. 2006.
- [38] K. R. Messer, "Riccati techniques on a time scale," *Panam. Math. J.*, vol. 13, no. 2, pp. 1–17, 2003.
- [39] C. Pötzsche, "Chain rule and invariance principle on measure chains," *J. Comput. Appl. Math.*, vol. 141, no. 1/2, pp. 249–254, Apr. 2002.
- [40] W. B. Powell, *Approximate Dynamic Programming: Solving the Curses of Dimensionality*, ser. Wiley Series in Probability and Statistics. Hoboken, NJ: Wiley, 2007.
- [41] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, ser. Wiley Series in Probability and Mathematical Statistics: Applied Probability and Statistics. New York: Wiley, 1994.
- [42] J. Rogers and Q. Sheng, "Notes on the diamond- α dynamic derivative on time scales," *J. Math. Anal. Appl.*, vol. 326, no. 1, pp. 228–241, 2007.
- [43] H. Rund, *The Hamilton-Jacobi Theory in the Calculus of Variations: Its Role in Mathematics and Physics*. New York: Van Nostrand, 1966.
- [44] S. Sanyal, "Stochastic dynamic equations," Ph.D. dissertation, Missouri Univ. Sci. Technol., Rolla, MO, 2008.
- [45] Q. Sheng, M. Fadag, J. Henderson, and J. M. Davis, "An exploration of combined dynamic derivatives on time scales and their applications," *Nonlinear Anal.: Real World Appl.*, vol. 7, no. 3, pp. 395–413, Jul. 2006.
- [46] P. J. Werbos, "Backwards differentiation in AD and neural nets: Past links and new opportunities," in *Automatic Differentiation: Applications, Theory, and Implementations*, vol. 50. Berlin, Germany: Springer-Verlag, 2006, pp. 15–34.
- [47] D. A. White and D. A. Sofage, Eds., *Handbook of Intelligent Control*, New York: Van Nostrand, 1992.



John Seiffertt (S'04) received the B.S. and M.S. degrees in applied mathematics from the University of Missouri, Rolla, and the M.A. degree in economics from the University of Missouri, St. Louis. He is currently working toward the Ph.D. degree in computer engineering at the Applied Computational Intelligence Laboratory, Department of Electrical and Computer Engineering, Missouri University of Science and Technology.

He has been a Pension Actuary, Economics Research Analyst, and Institutional Asset Allocation Analyst and was with the faculty of the Department of Mathematics and Computer Science, University of Missouri, St. Louis. His research focuses on the multiagent modeling of biosocioeconomic systems, including dynamic programming, computational intelligence, control theory, adaptive resonance theory, monetary policy, asset pricing, and game theory.



Suman Sanyal received the B.Sc. degree with major in mathematics and minor in physics and chemistry from Presidency College, Calcutta, India, in 2000, the M.Sc. degree in applied mathematics from the Indian Institute of Technology, Kharagpur, in 2002, and the Ph.D. degree in mathematics from the Missouri University of Science and Technology, Rolla, in 2008. His dissertation is titled "Stochastic Dynamic Equations."

He is currently with the Department of Mathematics and Statistics, Missouri University of Science and Technology. His research interests include time-scale calculus, time series forecasting, neural networks, control theory, and stochastic analysis with application to mathematical finance.



Donald C. Wunsch, II (S'86–M'91–SM'94–F'05) received the B.S. degree in applied mathematics from the University of New Mexico, Albuquerque, the M.S. and Ph.D. degrees in applied mathematics from the University of Washington, Seattle, and the EMBA degree from Washington University, St. Louis, MO.

He was with Texas Tech, Boeing, Rockwell International, and International Laser Systems. He is an M. K. Finley Missouri Distinguished Professor with the Applied Computational Intelligence Laboratory, Department of Electrical and Computer Engineering, Missouri University of Science and Technology, Rolla. He has produced 13 Ph.D.'s in three disciplines and over 275 publications. His contributions include adaptive resonance and reinforcement learning hardware and applications, neurofuzzy regression, traveling salesman problem heuristics, and bioinformatics.

Dr. Wunsch, II was an International Neural Networks Society (INNS) President, International Joint Conference on Neural Networks General Chair, and a Board Member of the IEEE Neural Network Council, INNS, and the University Missouri Bioinformatics Consortium. He is an INNS Fellow.