# Policy Evaluation in Continuous MDPs with Efficient Kernelized Gradient Temporal Difference

Alec Koppel[1,2], Garrett Warnell[2], Ethan Stump[2], Peter Stone[3], Alejandro Ribeiro[1]
[1]University of Pennsylvania, [2]U.S. Army Research Laboratory, [3]The University of Texas at Austin
{akoppel,aribeiro}@seas.upenn.edu,
{garrett.a.warnell.civ,ethan.a.stump2.civ}@mail.mil, pstone@cs.utexas.edu

***Abstract*—We consider policy evaluation in infinite-horizon discounted Markov decision problems (MDPs) with continuous compact state and action spaces. We reformulate this task as a compositional stochastic program with a function-valued decision variable that belongs to a reproducing kernel Hilbert space (RKHS). We approach this problem via a new functional generalization of stochastic quasi-gradient methods operating in tandem with stochastic sparse subspace projections. The result is an extension of gradient temporal difference learning that yields nonlinearly parameterized value function estimates of the solution to the Bellman evaluation equation. We call this method Parsimonious Kernel Gradient Temporal Difference (PKGTD) Learning. Our main contribution is a memory-efficient non-parametric stochastic method guaranteed to converge exactly to the Bellman fixed point with probability $1$ with attenuating step-sizes under the hypothesis that it belongs to the RKHS. Further, with constant step-sizes and compression budget, we establish mean convergence to a neighborhood and that the value function estimates have finite complexity. In the Mountain Car domain, we observe faster convergence to lower Bellman error solutions than existing approaches with a fraction of the required memory.**

## I. Markov Decision Processes

We consider an autonomous agent acting in an environment defined by a Markov decision process (MDP) [1] with continuous spaces, which is increasingly relevant to emerging technologies such as robotics [2], power systems [3], and others. A MDP is a quintuple $(\mathcal{X}, \mathcal{A}, \mathbb{P}, r, \gamma)$, where $\mathbb{P}$ is the action-dependent transition probability of the process: when the agent starts in state $\mathbf{x}_t \in \mathcal{X} \subset \mathbb{R}^p$ at time $t$ and takes an action $\mathbf{a}_t \in \mathcal{A}$, a transition to next state $\mathbf{y}_t \in \mathcal{X}$ is distributed according to $\mathbf{y}_t \sim \mathbb{P}(\cdot \,|\, \mathbf{x}_t, \mathbf{a}_t)$. After the agent transitions to a particular $\mathbf{y}_t$, the MDP provides to it an instantaneous reward $r(\mathbf{x}_t, \mathbf{a}_t, \mathbf{y}_t)$, where the reward function is a map $r : \mathcal{X} \times \mathcal{A} \times \mathcal{X} \to \mathbb{R}$.

We focus on the problem of *policy evaluation*: control decisions $\mathbf{a}_t$ are chosen according to a fixed stationary stochastic policy $\pi : \mathcal{X} \to \rho(\mathcal{A})$, where $\rho(\mathcal{A})$ denotes the set of probability distributions over $\mathcal{A}$. Policy evaluation underlies methods that seek optimal policies through repeated evaluation and improvement [4]. In policy evaluation, we seek to compute the *value* of a policy when starting in state $\mathbf{x}$, quantified by the discounted expected sum of rewards, or value function $V^\pi(\mathbf{x})$:[1]

$$V^\pi(\mathbf{x}) = \mathbb{E}_{\mathbf{y}}\Big[\sum_{t=0}^{\infty}\gamma^t r(\mathbf{x}_t, \mathbf{a}_t, \mathbf{y}_t) \,\big|\, \mathbf{x}_0 = \mathbf{x}, \{\mathbf{a}_t = \pi(\mathbf{x}_t)\}_{t=0}^{\infty}\Big]. \quad (1)$$

For a single trajectory through the state space $\mathcal{X}$, $\mathbf{y}_t = \mathbf{x}_{t+1}$. The value function (1) is parameterized by a discount factor $\gamma \in (0, 1)$, which determines the agent's farsightedness. Decomposing the summand in (1) into its first and subsequent terms, and using both the stationarity of the transition probability and the Markov property yields the Bellman evaluation equation [5]:

$$V^\pi(\mathbf{x}) = \int_{\mathcal{X}} [r(\mathbf{x}, \pi(\mathbf{x}), \mathbf{y}) + \gamma V^\pi(\mathbf{y})] \mathbb{P}(d\mathbf{y} \,|\, \mathbf{x}, \pi(\mathbf{x})) \quad (2)$$

for all $\mathbf{x} \in \mathcal{X}$. The right-hand side of (2) defines a Bellman evaluation operator $\mathscr{B}^\pi : \mathcal{B}(\mathcal{X}) \to \mathcal{B}(\mathcal{X})$ over $\mathcal{B}(\mathcal{X})$, the space of bounded continuous value functions $V : \mathcal{X} \to \mathbb{R}$:

$$(\mathscr{B}^\pi V)(\mathbf{x}) = \int_{\mathcal{X}} [r(\mathbf{x}, \pi(\mathbf{x}), \mathbf{y}) + \gamma V(\mathbf{y})] \mathbb{P}(d\mathbf{y} \,|\, \mathbf{x}, \pi(\mathbf{x})) \quad (3)$$

for all $\mathbf{x} \in \mathcal{X}$. [6][Proposition 4.2(b)] establishes that the stationary point of (3) is $V^\pi$, i.e., $(\mathscr{B}^\pi V^\pi)(\mathbf{x}) = V^\pi(\mathbf{x})$. As a stepping stone to finding optimal policies in infinite MDPs, we seek here to find the fixed point of (3). Specifically, the goal of this work is stable value function estimation in infinite MDPs, with nonlinear parameterizations that are allowed to be infinite, but are nonetheless memory-efficient.

**Challenges** To solve (3), fixed point methods, i.e., value iteration ($V_{k+1} = \mathscr{B}^\pi V_k$), have been proposed [6], but only apply when the value function can be represented by a vector whose length is defined by the number of states and the state space is small enough that the expectation[2] in $\mathscr{B}$ can be computed. For large spaces, stochastic approximations of value iteration, i.e., temporal difference (TD) learning [7], have been utilized to circumvent this intractable expectation. Incremental methods (least-squares TD) provide an alternative when $V(\mathbf{x})$ has a finite linear parameterization [8], but their extensions to infinite representations require infinite memory [9] or elude stability [10].

Solving the fixed point problem defined by (3) requires surmounting the fact that this expression is defined for each $\mathbf{x} \in \mathcal{X}$, which for continuous $\mathcal{X} \subset \mathbb{R}^p$ has *infinitely many* unknowns. This phenomenon is one example of Bellman's curse of dimensionality [5], and it is frequently sidestepped

[1] In MDPs more generally, we choose actions $\{\mathbf{a}_t\}_{t=1}^{\infty}$ to maximize the reward accumulation starting from state $\mathbf{x}$, i.e., $V(\mathbf{x}, \{\mathbf{a}_t\}_{t=0}^{\infty}) = \mathbb{E}_{\mathbf{y}}\Big[\sum_{t=0}^{\infty}\gamma^t r(\mathbf{x}_t, \mathbf{a}_t, \mathbf{y}_t) \,\big|\, \mathbf{x}_0 = \mathbf{x}, \{\mathbf{a}_t\}_{t=0}^{\infty}\Big]$. For a fixed policy $\pi$, the setting of this work, this simplifies to (1).

[2] Observe that the integral in (2) defines a conditional expectation: $V^\pi(\mathbf{x}) = \mathbb{E}_{\mathbf{y}}[r(\mathbf{x}, \pi(\mathbf{x}), \mathbf{y}) + \gamma V^\pi(\mathbf{y})] \,|\, \mathbf{x}, \pi(\mathbf{x})]$.

by parameterizing the value function using a finite linear [11], [12] or nonlinear [13] basis expansion. Such methods have paved the way for the recent success of neural networks in value function-based approaches to MDPs, but combining TD learning with different parameterizations may cause divergence [11], [14]: in general, the representation must be tied to the stochastic update [15] to ensure both the parameterization and the stochastic process are stable.

**Contributions** Our main result is a memory-efficient, nonparametric, stochastic method that converges to the Bellman fixed point almost surely when it belongs to a reproducing kernel Hilbert space (RKHS). Our approach is to reformulate (2) as a compositional stochastic program (Section II), a topic studied in operations research [16] and probability [17], [18]. These problems motivate stochastic *quasi-gradient* (SQG) methods which use two time-scale stochastic approximation to mitigate the fact that the objective's stochastic gradient still requires evaluation of an intractable expectation [19]. Here, we use SQG for policy evaluation in infinite MDPs (finite MDPs addressed in [13], [20]).

In (2), the decision variable is a continuous function, which we address by hypothesizing the Bellman fixed point belongs to a RKHS [21], [22]. However, a function in a RKHS has comparable complexity to the number of training samples processed, which could be infinite (an issue ignored in many kernel methods for MDPs [9], [10], [23]–[27]). We will tackle this memory bottleneck by requiring memory efficiency in both the function sample path and in its limit.

To find a memory-efficient sample path in the function space, we generalize SQG to RKHSs (Section III), and combine this generalization with greedily-constructed sparse subspace projections (Section III-A). These subspaces are constructed via matching pursuit [28], [29], a procedure motivated by the facts that (a) kernel matrices induced by arbitrary data streams likely violate requirements for convex-relaxation-based sparsity [30], and (b) parsimony is more important than exact recovery since SQG iterates are not the target signal but rather a point along the convergence path to Bellman fixed point. Rather than unsupervised forgetting [31], we tie the projection-induced error to stochastic descent [32] which keeps only those dictionary points needed for convergence (Section IV).

As a result, we conduct functional SQG descent via sparse projections of the SQG. This maintains a moderate-complexity sample path exactly towards $V^*$, which may be made arbitrarily close to the Bellman fixed point by decreasing the regularizer. By generalizing the relationship between SQG and supermartingales in [33] to Hilbert spaces, we establish that the sparse projected SQG sequence converges almost surely to the Bellman fixed point with decreasing learning rates, and converges in mean while maintaining finite complexity when constant learning rates are used (Section IV). We then empirically evaluate the proposed value function approximation method on the discrete Mountain Car domain in Section V and summarize our findings in Section VI.

## II. Policy Evaluation as Compositional Stochastic Programming

We turn to reformulating the functional fixed point problem (3) defined by Bellman's equation so that it may be identified with a nested stochastic program. We note that the resulting domain of this problem is intractable, and address this by hypothesizing that the Bellman fixed point belongs to a RKHS, which, in turn, requires the introduction of regularization.

We proceed with reformulating (3): subtract the value function $V^\pi(\mathbf{x})$ that satisfies the fixed point relation from both sides, and then pull it inside the expectation:

$$0 = \mathbb{E}_{\mathbf{y}}[r(\mathbf{x},\pi(\mathbf{x}),\mathbf{y}) + \gamma V^\pi(\mathbf{y}) - V^\pi(\mathbf{x}) \,\big|\, \mathbf{x}, \pi(\mathbf{x})] \tag{4}$$

for all $\mathbf{x} \in \mathcal{X}$. Value functions satisfying (4) are equivalent to those which satisfy the quadratic expression $0 = \frac{1}{2}(\mathbb{E}_{\mathbf{y}}[r(\mathbf{x},\pi(\mathbf{x}),\mathbf{y}) + \gamma V^\pi(\mathbf{y}) - V^\pi(\mathbf{x}) \,\big|\, \mathbf{x}, \pi(\mathbf{x})])^2$ , which is null for all $\mathbf{x} \in \mathcal{X}$. Solving this expression for every $\mathbf{x}$ may be achieved by considering this expression in an initialization-independent manner. That is, integrating out $\mathbf{x}$, the starting point of the trajectory defining the value function (1), as well as policy $\pi(\mathbf{x})$, yields the compositional stochastic program:

$$\begin{aligned} V^\pi &= \operatorname*{argmin}_{V\in\mathcal{B}(\mathcal{X})} J(V) \qquad\qquad (5)\\ &:= \operatorname*{argmin}_{V\in\mathcal{B}(\mathcal{X})} \mathbb{E}_{\mathbf{x},\pi(\mathbf{x})}\Big\{\frac{1}{2}(\mathbb{E}_{\mathbf{y}}[r(\mathbf{x},\pi(\mathbf{x}),\mathbf{y})+\gamma V(\mathbf{y}) - V(\mathbf{x}) \big| \mathbf{x},\pi(\mathbf{x})])^2\Big\} \end{aligned}$$

whose solutions coincide exactly with the fixed points of (3).

(5) defines a functional optimization problem which is intractable when we search over all bounded continuous functions $\mathcal{B}(\mathcal{X})$. However, when we restrict $\mathcal{B}(\mathcal{X})$ to a Hilbert space $\mathcal{H}$ equipped with a unique *reproducing kernel*, i.e., an inner product-like map $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ such that

$$\begin{aligned} &(i)\ \langle f, \kappa(\mathbf{x}, \cdot)\rangle_{\mathcal{H}} = f(\mathbf{x}) \quad \text{for all } \mathbf{x} \in \mathcal{X},\\ &(ii)\ \mathcal{H} = \overline{\text{span}\{\kappa(\mathbf{x}, \cdot)\}} \quad \text{for all } \mathbf{x} \in \mathcal{X}\ , \end{aligned} \tag{6}$$

we may apply the Representer Theorem to transform the functional problem (5) into a parametric one [21], [34], [35] In a RKHS, the optimal function $f \in \mathcal{H}$ of (5) then takes the form

$$f(\mathbf{x}) = \sum_{n=1}^{N} w_n \kappa(\mathbf{x}_n, \mathbf{x})\ , \tag{7}$$

where $\mathbf{x}_n$ is a realization of the random variable $\mathbf{x}$. Thus, $f \in \mathcal{H}$ is an expansion of kernel evaluations *only* at training samples. We refer to the upper summand index $N$ in (7) in the kernel expansion of $f \in \mathcal{H}$ as the model order, which here coincides with the training sample size. Common kernel choices are polynomials and radial basis (Gaussian) functions, i.e., $\kappa(\mathbf{x}, \mathbf{x}') = \left(\mathbf{x}^T\mathbf{x}' + b\right)^c$ and $\kappa(\mathbf{x}, \mathbf{x}') = \exp\{-\|\mathbf{x} - \mathbf{x}'\|_2^2/2c^2\}$, respectively. In (6), property (i) is called the reproducing property, which follows from Riesz Representation Theorem [36]. Replacing $f$ by $\kappa(\mathbf{x}', \cdot)$ in (6) (i) yields the expression $\langle\kappa(\mathbf{x}', \cdot), \kappa(\mathbf{x}, \cdot)\rangle_{\mathcal{H}} = \kappa(\mathbf{x}, \mathbf{x}')$, the origin of the term "reproducing kernel." Moreover, property (6) (ii) states that functions $f \in \mathcal{H}$ admit a basis expansion in terms of kernel evaluations (7). Function spaces of this type are referred to as reproducing kernel Hilbert spaces (RKHSs). For universal kernels the kernel is universal [37], e.g., a Gaussian, a continuous function over a compact set may be approximated uniformly by one in a RKHS.

Subsequently, we seek to solve (5) with the restriction that $V \in \mathcal{H}$, and independent and identically distributed samples $(\mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)$ from the triple $(\mathbf{x}, \pi(\mathbf{x}), \mathbf{y})$ are sequentially available, yielding

$$V^* = \operatorname*{argmin}_{V \in \mathcal{H}} \mathbb{E}_{\mathbf{x},\pi(\mathbf{x})}\Big\{\frac{1}{2}(\mathbb{E}_{\mathbf{y}}[r(\mathbf{x},\pi(\mathbf{x}),\mathbf{y}) \\ + V(\mathbf{y}) - V(\mathbf{x}) \,\big|\, \mathbf{x}, \pi(\mathbf{x})])^2\Big\} + \frac{\lambda}{2}\|V\|^2_{\mathcal{H}} \tag{8}$$

Hereafter, define $L(V) := \mathbb{E}_{\mathbf{x},\pi(\mathbf{x})}\{\frac{1}{2}(\mathbb{E}_{\mathbf{y}}[r(\mathbf{x},\pi(\mathbf{x}),\mathbf{y}) + \gamma V(\mathbf{y}) - V(\mathbf{x}) \,\big|\, \mathbf{x}, \pi(\mathbf{x})])^2\}$ and $J(V) = L(V) + (\lambda/2)\|V\|^2_{\mathcal{H}}$. The regularization term $(\lambda/2)\|V\|^2_{\mathcal{H}}$ in (8) is needed to apply the Representer Theorem (7) [34]. Thus, policy evaluation in infinite MDPs (8) is both a specialization of compositional stochastic programming [33] to an objective defined by dynamic programming, and a generalization to the case where the decision variable is not vector-valued but is instead a function.

## III. Functional Stochastic Quasi-Gradient

To apply functional SQG to (8), we differentiate the compositional objective $L(V)$, which is of the form $L = g \circ h$, with $g(u) = \mathbb{E}_{\mathbf{x},\pi(\mathbf{x})}[(1/2)u^2]$ and $h(V) = \mathbb{E}_{\mathbf{y}}[r(\mathbf{x}, \pi(\mathbf{x}), \mathbf{y}) + \gamma V(\mathbf{y}) - V(\mathbf{x}) \,\big|\, \mathbf{x}, \pi(\mathbf{x})]$, and then consider its stochastic estimate. Consider the Frechét derivative of $L(V)$:

$$\begin{aligned} &\nabla_V L(V) \\ &=\nabla_V \mathbb{E}_{\mathbf{x},\pi(\mathbf{x})}\{\frac{1}{2}(\mathbb{E}_{\mathbf{y}}[r(\mathbf{x},\pi(\mathbf{x}),\mathbf{y})+\gamma V(\mathbf{y})-V(\mathbf{x})\big|\mathbf{x},\pi(\mathbf{x})])^2\} \\ &=\mathbb{E}_{\mathbf{x},\pi(\mathbf{x})}\{\nabla_V\frac{1}{2}(\mathbb{E}_{\mathbf{y}}[r(\mathbf{x},\pi(\mathbf{x}),\mathbf{y})+\gamma V(\mathbf{y})-V(\mathbf{x})\big|\mathbf{x},\pi(\mathbf{x})])^2\} \\ &= \mathbb{E}_{\mathbf{x},\pi(\mathbf{x})}\big\{\mathbb{E}_{\mathbf{y}}[\gamma\kappa(\mathbf{y},\cdot) - \kappa(\mathbf{x},\cdot) \,\big|\, \mathbf{x}, \pi(\mathbf{x})] \\ &\qquad \times \mathbb{E}_{\mathbf{y}}[r(\mathbf{x},\pi(\mathbf{x}),\mathbf{y})+\gamma V(\mathbf{y})-V(\mathbf{x})\big|\mathbf{x},\pi(\mathbf{x})]\big\} \end{aligned} \tag{9}$$

To get the second equality, we pull the differential operator inside the expectation, and, to get the third equality, we make use of the chain rule and reproducing property of the kernel (6)(i). For future reference, we define the expression $\mathbb{E}_{\mathbf{y}}[r(\mathbf{x}, \pi(\mathbf{x}), \mathbf{y}) + \gamma V(\mathbf{y}) - V(\mathbf{x}) \,\big|\, \mathbf{x}, \pi(\mathbf{x})] = \bar{\delta}$ as the average temporal difference [7]. To perform stochastic descent in function space $\mathcal{H}$, we need a stochastic approximate of (9) evaluated at a state-action-state triple $(\mathbf{x}, \pi(\mathbf{x}), \mathbf{y})$, which together with the regularizer yields

$$\begin{aligned} &\nabla_V J(V, \delta; \mathbf{x}, \pi(\mathbf{x}), \mathbf{y}) \\ &\quad =[\gamma\kappa(\mathbf{y},\cdot)-\kappa(\mathbf{x},\cdot)][r(\mathbf{x},\pi(\mathbf{x}),\mathbf{y})+\gamma V(\mathbf{y})-V(\mathbf{x})]+\lambda V \end{aligned} \tag{10}$$

where $\delta := r(\mathbf{x}, \pi(\mathbf{x}), \mathbf{y}) + \gamma V(\mathbf{y}) - V(\mathbf{x})$ is defined as the (instantaneous) temporal difference. Observe that we cannot obtain samples of $\nabla_V J(V, \delta; \mathbf{x}, \pi(\mathbf{x}), \mathbf{y})$ with a single query to a simulation oracle: stochastic gradient method would estimate one of the expected gradients by its instantaneous approximation, but would still leave a second expected value that depends on infinitely many realizations of either prior distribution and policy $(\mathbf{x}, \pi(\mathbf{x}))$ or MDP transition dynamics $\mathbf{y}$, a problem first identified in [20] for finite MDPs where it is called the *double sampling problem*. Therefore, we require a method that constructs a *coupled* stochastic descent procedure by considering noisy estimates of both terms in the product-of-expectations expression in (9).

Due to the fact that the first term $[\gamma\kappa(\mathbf{y}, \cdot) - \kappa(\mathbf{x}, \cdot)]$ in (10) is a difference of kernel maps, building up its total expectation will, in the limit, be of infinite complexity [38]. Thus, we propose instead to construct a sequence based on samples of the second term. That is, based on realizations of $\delta$, we consider a fixed point recursion that builds up an estimate of $\bar{\delta}$ by defining a scalar sequence $z_t$ as

$$\begin{aligned} \delta_t &= r(\mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t) + \gamma V_t(\mathbf{y}_t) - V_t(\mathbf{x}_t) \\ z_{t+1} &= (1 - \beta_t) z_t + \beta_t \delta_t \end{aligned} \tag{11}$$

where we define $\delta_t$ [7] as the temporal difference at time $t$ in (11) Thus, (11) approximately averages the temporal difference sequence $\delta_t$: $z_t$ estimates $\bar{\delta}_t$, and $\beta_t \in (0, 1)$ is a learning rate.

To define a stochastic descent step, we replace the first term inside the outer expectation in (9) with its instantaneous approximate, i.e., $[\gamma\kappa(\mathbf{y}_t, \cdot) - \kappa(\mathbf{x}_t, \cdot)]$, evaluated at a sample triple $(\mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)$, which yields the stochastic quasi-gradient step [19], [33]

$$\hat{V}_{t+1} = (1 - \alpha_t\lambda)\hat{V}_t - \alpha_t(\gamma\kappa(\mathbf{y}_t, \cdot) - \kappa(\mathbf{x}_t, \cdot)) z_{t+1} \ . \tag{12}$$

where the coefficient $(1 - \alpha_t\lambda)$ comes from the regularizer, and $\alpha_t$ is a positive scalar learning rate. This update is a stochastic quasi-gradient step because the true stochastic gradient of $J(V)$ is $(\gamma\kappa(\mathbf{y}_t, \cdot) - \kappa(\mathbf{x}_t, \cdot))\delta_t$, but this estimator is unavailable with a single trajectory of the MDP since the terms in this product are dependent. By replacing $\delta_t$ by auxiliary variable $z_{t+1}$ this issue may be circumvented in the construction of coupled supermartingales (Section IV).

**Kernel Parameterization** Suppose $V_0 = 0 \in \mathcal{H}$. Then the update in (12) at time $t$, making use of the Representer Theorem (7), implies the function $\hat{V}_t$ is a kernel expansion of past states $(\mathbf{x}_t, \mathbf{y}_t)$ as

$$\hat{V}_t(\mathbf{x}) = \sum_{n=1}^{2(t-1)} w_n \kappa(\mathbf{v}_n, \mathbf{x}) = \mathbf{w}_t^T \boldsymbol{\kappa}_{\mathbf{X}_t}(\mathbf{x}) \ . \tag{13}$$

On the right-hand side of (13) we introduce the notation $\mathbf{v}_n = \mathbf{x}_n$ for $n$ even and $\mathbf{v}_n = \mathbf{y}_n$ for $n$ odd, and: $\mathbf{w}_t = [w_1, \cdots, w_{2(t-1)}] \in \mathbb{R}^{2(t-1)}$ , $\mathbf{X}_t = [\mathbf{x}_1, \mathbf{y}_1, \ldots, \mathbf{x}_{t-1}, \mathbf{y}_{t-1}] \in \mathbb{R}^{p \times 2(t-1)}$ , and $\boldsymbol{\kappa}_{\mathbf{X}_t}(\cdot) = [\kappa(\mathbf{x}_1, \cdot),\ \kappa(\mathbf{y}_1, \cdot),\ \ldots\ , \kappa(\mathbf{x}_{t-1}, \cdot), \kappa(\mathbf{y}_{t-1}, \cdot)]^T$ . The kernel expansion in (13), together with the functional update (12), yields the fact that functional SQG in $\mathcal{H}$ amounts to the following updates on the kernel dictionary $\mathbf{X}$ and coefficient vector $\mathbf{w}$:

$$\begin{aligned} \mathbf{X}_{t+1} &= [\mathbf{X}_t\ ,\ \mathbf{x}_t\ ,\ \mathbf{y}_t], \\ \mathbf{w}_{t+1} &= [(1 - \alpha_t\lambda)\mathbf{w}_t\ ,\ \alpha_t z_{t+1}\ ,\ -\alpha_t\gamma z_{t+1}] \ , \end{aligned} \tag{14}$$

Observe that this update causes $\mathbf{X}_{t+1}$ to have two more columns than $\mathbf{X}_t$. We define the *model order* as number of data points $M_t$ in the dictionary at time $t$, which for functional stochastic quasi-gradient descent is $M_t = 2(t-1)$. Asymptotically, then, the complexity of storing $\hat{V}_t(\mathbf{x})$ is infinite.

### *A. Sparse Stochastic Subspace Projections*

Since the update (12) has complexity $\mathcal{O}(t)$ due to the parameterization induced by RKHS [32], [38], it is impractical in settings with streaming data or arbitrarily large training sets. We address this issue by replacing the stochastic descent step (12) with an orthogonally projected variant [32], where the projection is onto a low-dimensional functional subspace $\mathcal{H}_{\mathbf{D}_{t+1}}$ of $\mathcal{H}$, i.e.,

$$V_{t+1}=\mathcal{P}_{\mathcal{H}_{\mathbf{D}_{t+1}}}[(1-\alpha_t\lambda)V_t-\alpha_t(\gamma\kappa(\mathbf{y}_t,\cdot)-\kappa(\mathbf{x}_t,\cdot))z_{t+1}], \quad (15)$$

where $\alpha_t$ again is a scalar step-size, and $\mathcal{H}_{\mathbf{D}_{t+1}} = \text{span}\{\kappa(\mathbf{d}_n,\cdot)\}_{n=1}^{M_t}$ for some collection of sample instances $\{\mathbf{d}_n\} \subset \{\mathbf{x}_u\}_{u\leq t}$. Note that the un-projected function SQG method (12) may be interpreted as conducting a sequence of orthogonal projections, which motivates the design of (15). Specifically, rewrite (12) as the quadratic minimization

$$\begin{aligned}\hat{V}_{t+1}&=\underset{V\in\mathcal{H}}{\text{argmin}}\Big\|V-\Big((1-\alpha_t\lambda)\hat{V}_t-\alpha_t(\gamma\kappa(\mathbf{y}_t,\cdot)-\kappa(\mathbf{x}_t,\cdot))z_{t+1}\Big)\Big\|_{\mathcal{H}}^2\\&=\underset{V\in\mathcal{H}_{\mathbf{X}_{t+1}}}{\text{argmin}}\Big\|V-\Big((1-\alpha_t\lambda)\hat{V}_t-\alpha_t(\gamma\kappa(\mathbf{y}_t,\cdot)-\kappa(\mathbf{x}_t,\cdot))z_{t+1}\Big)\Big\|_{\mathcal{H}}^2,\end{aligned} \quad (16)$$

where the first equality in (16) comes from ignoring constant terms which vanish upon differentiation with respect to $V$, and the second comes from observing that $V_{t+1}$ can be represented using only the points $\mathbf{X}_{t+1}$, using (14). Notice now that (16) expresses $V_{t+1}$ as the orthogonal projection of the update $(1-\alpha_t\lambda)V_t-\alpha_t(\gamma\kappa(\mathbf{y}_t,\cdot)-\kappa(\mathbf{x}_t,\cdot))z_{t+1}$ onto the subspace defined by dictionary $\mathbf{X}_{t+1}$.

Rather than select dictionary $\mathbf{D} = \mathbf{X}_{t+1}$, we propose instead to select a different dictionary, $\mathbf{D} = \mathbf{D}_{t+1}$, which is extracted from the data points observed thus far, at each iteration. The process by which we select $\mathbf{D}_{t+1}$ is delayed for now, but is of dimension $p \times M_{t+1}$, with $M_{t+1} << \mathcal{O}(t)$. As a result, the sequence $V_t$ differs from the functional stochastic quasi-gradient method $\hat{V}_t$ presented at the outset of this section.

Specifically, suppose the function $V_{t+1}$ is parameterized dictionary $\mathbf{D}_{t+1}$ and weight vector $\mathbf{w}_{t+1}$. We denote columns of $\mathbf{D}_{t+1}$ as $\mathbf{d}_n$ for $n = 1, \ldots, M_{t+1}$, where the time index is dropped for notational clarity but may be inferred from the context. Setting aside how $\mathbf{D}_{t+1}$ is chosen for now, we replace the update (16) in which the dictionary grows at each iteration by the functional stochastic quasi-gradient sequence projected onto the subspace $\mathcal{H}_{\mathbf{D}_{t+1}} = \text{span}\{\kappa(\mathbf{d}_n,\cdot)\}_{n=1}^{M_{t+1}}$ as

$$\begin{aligned}V_{t+1}&=\underset{V\in\mathcal{H}_{\mathbf{D}_{t+1}}}{\text{argmin}}\Big\|V-\Big((1-\alpha_t\lambda)\hat{V}_t-\alpha_t(\gamma\kappa(\mathbf{y}_t,\cdot)-\kappa(\mathbf{x}_t,\cdot))z_{t+1}\Big)\Big\|_{\mathcal{H}}^2,\\&:=\mathcal{P}_{\mathcal{H}_{\mathbf{D}_{t+1}}}\Big[(1-\alpha_t\lambda)V_t-\alpha_t(\gamma\kappa(\mathbf{y}_t,\cdot)-\kappa(\mathbf{x}_t,\cdot))z_{t+1}\Big].\end{aligned} \quad (17)$$

where we define the projection operator $\mathcal{P}$ onto subspace $\mathcal{H}_{\mathbf{D}_{t+1}} \subset \mathcal{H}$ by the update (17). This orthogonal projection is the modification of the functional SQG iterate [cf. (12)] defined at the beginning of this subsection (15). Next we discuss how this update amounts to modifications of the parametric updates (14) defined by functional SQG.

**Coefficient update** The update (15), for a fixed dictionary $\mathbf{D}_{t+1} \in \mathbb{R}^{p\times M_{t+1}}$, may be expressed in terms of the parameter space of coefficients only. To do so, first define the stochastic quasi-gradient update *without projection*, given function $V_t$ parameterized by dictionary $\mathbf{D}_t$ and coefficients $\mathbf{w}_t$, as

$$\tilde{V}_{t+1} = (1-\alpha_t\lambda)V_t - \alpha_t(\gamma\kappa(\mathbf{y}_t,\cdot)-\kappa(\mathbf{x}_t,\cdot))z_{t+1}\ . \quad (18)$$

This update may be represented using dictionary and weight vector

$$\begin{aligned}\tilde{\mathbf{D}}_{t+1} &= [\mathbf{D}_t\ ,\ \mathbf{x}_t\ ,\ \mathbf{y}_t]\\ \tilde{\mathbf{w}}_{t+1} &= [(1-\alpha_t\lambda)\mathbf{w}_t\ ,\ \alpha_t z_{t+1}\ ,\ -\alpha_t\gamma z_{t+1}]\ ,\end{aligned} \quad (19)$$

We use the notation that $V_{t+1}$ is the sequence of projected quasi-FGSD iterates [cf. (15)] and $\tilde{V}_{t+1}$ is the update [cf. (18)] without projection in Section III-A. The later is parameterized by dictionary $\tilde{\mathbf{D}}_{t+1}$ and weights $\tilde{\mathbf{w}}_{t+1}$ (19).

When the dictionary defining $V_{t+1}$ is assumed fixed, we may use use of the Representer Theorem to rewrite (17) in terms of kernel expansions, and note that the coefficient vector is the only free parameter to write

$$\begin{aligned}&\underset{\mathbf{w}\in\mathbb{R}^{M_{t+1}}}{\text{argmin}}\ \frac{1}{2\eta_t}\Big\|\sum_{n=1}^{M_{t+1}} w_n\kappa(\mathbf{d}_n,\cdot) - \sum_{m=1}^{\tilde{M}} \tilde{w}_m\kappa(\tilde{\mathbf{d}}_m,\cdot)\Big\|_{\mathcal{H}}^2 \quad (20)\\&=\underset{\mathbf{w}\in\mathbb{R}^{M_{t+1}}}{\text{argmin}}\frac{1}{2\eta_t}\Big(\sum_{n,n'=1}^{M_{t+1}}w_n w_{n'}\kappa(\mathbf{d}_n,\mathbf{d}_{n'})-2\sum_{n,m=1}^{M_{t+1},\tilde{M}}w_n\tilde{w}_m\kappa(\mathbf{d}_n,\tilde{\mathbf{d}}_m)\\&\qquad+\sum_{m,m'=1}^{\tilde{M}}\tilde{w}_m\tilde{w}_{m'}\kappa(\tilde{\mathbf{d}}_m,\tilde{\mathbf{d}}_{m'})\Big)\\&=\underset{\mathbf{w}\in\mathbb{R}^{M_{t+1}}}{\text{argmin}}\frac{1}{2\eta_t}\Big(\mathbf{w}^T\mathbf{K}_{\mathbf{D}_{t+1},\mathbf{D}_{t+1}}\mathbf{w}-2\mathbf{w}^T\mathbf{K}_{\mathbf{D}_{t+1},\tilde{\mathbf{D}}_{t+1}}\tilde{\mathbf{w}}_{t+1}\\&\qquad+\tilde{\mathbf{w}}_{t+1}^T\mathbf{K}_{\tilde{\mathbf{D}}_{t+1},\tilde{\mathbf{D}}_{t+1}}\tilde{\mathbf{w}}_{t+1}\Big)\\&:=\mathbf{w}_{t+1}\ .\end{aligned}$$

In (20), the first equality comes from expanding the square, and the second comes from defining the cross-kernel matrix $\mathbf{K}_{\mathbf{D}_{t+1},\tilde{\mathbf{D}}_{t+1}}$ whose $(n,m)^{\text{th}}$ entry is $\kappa(\mathbf{d}_n,\tilde{\mathbf{d}}_m)$. Kernel matrices $\mathbf{K}_{\tilde{\mathbf{D}}_{t+1},\tilde{\mathbf{D}}_{t+1}}$ and $\mathbf{K}_{\mathbf{D}_{t+1},\mathbf{D}_{t+1}}$ are similarly defined. Here $M_{t+1}$ is the number of columns in $\mathbf{D}_{t+1}$, while $\tilde{M}_{t+1} = M_t + 2$ is that of in $\tilde{\mathbf{D}}_{t+1}$ [cf. (19)]. Observe that $\tilde{\mathbf{D}}_{t+1}$ has $\tilde{M}_{t+1} = M_t + 2$ columns, which is the length of $\tilde{\mathbf{w}}_{t+1}$. For a fixed dictionary $\mathbf{D}_{t+1}$, the stochastic projection in (17) is a least-squares problem on the coefficient vector, i.e.,

$$\mathbf{w}_{t+1} = \mathbf{K}_{\mathbf{D}_{t+1}\mathbf{D}_{t+1}}^{-1}\mathbf{K}_{\mathbf{D}_{t+1}\tilde{\mathbf{D}}_{t+1}}\tilde{\mathbf{w}}_{t+1}\ , \quad (21)$$

The explicit solution of (20) may be obtained by noting that the last term is a constant independent of $\mathbf{w}$, and thus by computing gradients and solving for $\mathbf{w}_{t+1}$ we obtain (21). We now turn to selecting the dictionary $\mathbf{D}_{t+1}$ from the MDP trajectory $\{\mathbf{x}_u, \pi(\mathbf{x}_u), \mathbf{y}_u\}_{u\leq t}$.

**Dictionary Update** We select kernel dictionary $\mathbf{D}_{t+1}$ via greedy compression, a topic studied in compressive sensing [39]. The function $\tilde{V}_{t+1} = (1-\alpha_t)V_t - \alpha_t(\gamma\kappa(\mathbf{y}_t,\cdot) - \kappa(\mathbf{x}_t,\cdot))z_{t+1}$ defined by SQG method without projection (18) is parameterized by dictionary $\tilde{\mathbf{D}}_{t+1}$ [cf. (19)]. We form

**Algorithm 1** PKGTD: Parsimonious Kernel Gradient Temporal Difference

```
Require: $\{\mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t, \alpha_t, \beta_t, \epsilon_t\}_{t=0,1,2,\dots}$
  initialize $V_0(\cdot) = 0, \mathbf{D}_0 = [], \mathbf{w}_0 = [], z_0 = 0$, i.e. initial dict., coeffs., and aux. variable null
  for $t = 0, 1, 2, \dots$ do
    Obtain trajectory realization $(\mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)$
    Compute the temporal difference and update the auxiliary sequence $z_{t+1}$ [cf. (11)]:
      $$\delta_t = r(\mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t) + \gamma V_t(\mathbf{y}_t) - V_t(\mathbf{x}_t),$$
      $$z_{t+1} = (1 - \beta_t) z_t + \beta_t \delta_t$$
    Compute unconstrained functional stochastic quasi-gradient step [cf. (12)]
      $\tilde{V}_{t+1}(\cdot) = (1 - \alpha_t\lambda)\tilde{V}_t(\cdot) - \alpha_t(\gamma\kappa(\mathbf{y}_t, \cdot) - \kappa(\mathbf{x}_t, \cdot))z_{t+1}$
    Revise dictionary $\tilde{\mathbf{D}}_{t+1} = [\mathbf{D}_t, \mathbf{x}_t\ , \mathbf{y}_t]$, weights $\tilde{\mathbf{w}}_{t+1} \leftarrow [(1 - \alpha_t\lambda)\mathbf{w}_t,\ \ \alpha_t z_{t+1}, -\alpha_t\gamma z_{t+1}]$
    Obtain greedy compression of function parameterization via Algorithm 2
      $$(V_{t+1}, \mathbf{D}_{t+1}, \mathbf{w}_{t+1}) = \textbf{KOMP}(\tilde{V}_{t+1}, \tilde{\mathbf{D}}_{t+1}, \tilde{\mathbf{w}}_{t+1}, \epsilon_t)$$
  end for
```

**Algorithm 2** Destructive Kernel Orthogonal Matching Pursuit (KOMP)

```
Require: function $\tilde{V}$ defined by dict. $\tilde{\mathbf{D}} \in \mathbb{R}^{p \times \tilde{M}}$, coeffs. $\tilde{\mathbf{w}} \in \mathbb{R}^{\tilde{M}}$, approx. budget $\epsilon_t > 0$
  initialize $V = \tilde{V}$, dictionary $\mathbf{D} = \tilde{\mathbf{D}}$ with indices $\mathcal{I}$, model order $M = \tilde{M}$, coeffs. $\mathbf{w} = \tilde{\mathbf{w}}$.
  while candidate dictionary is non-empty $\mathcal{I} \neq \emptyset$ do
    for $j = 1, \dots, \tilde{M}$ do
      Find minimal approximation error with dictionary element $\mathbf{d}_j$ removed
        $$\gamma_j = \min_{\mathbf{w}_{\mathcal{I}\setminus\{j\}} \in \mathbb{R}^{M-1}} \|\tilde{V}(\cdot) - \sum_{k \in \mathcal{I}\setminus\{j\}} w_k \kappa(\mathbf{d}_k, \cdot)\|_{\mathcal{H}}\ .$$
    end for
    Find dictionary index minimizing approximation error: $j^* = \operatorname{argmin}_{j \in \mathcal{I}} \gamma_j$
    if minimal approximation error exceeds threshold $\gamma_{j^*} > \epsilon_t$
      stop
    else
      Prune dictionary $\mathbf{D} \leftarrow \mathbf{D}_{\mathcal{I}\setminus\{j^*\}}$
      Revise set $\mathcal{I} \leftarrow \mathcal{I} \setminus \{j^*\}$ and model order $M \leftarrow M - 1$.
      Compute updated weights $\mathbf{w}$ defined by current dictionary $\mathbf{D}$
        $$\mathbf{w} = \underset{\mathbf{w} \in \mathbb{R}^M}{\operatorname{argmin}} \|\tilde{V}(\cdot) - \mathbf{w}^T \boldsymbol{\kappa}_{\mathbf{D}}(\cdot)\|_{\mathcal{H}}$$
    end
  end while
  return $V, \mathbf{D}, \mathbf{w}$ of model order $M \leq \tilde{M}$ such that $\|V - \tilde{V}\|_{\mathcal{H}} \leq \epsilon_t$
```

$\mathbf{D}_{t+1}$ by selecting a subset of $M_{t+1}$ columns from $\tilde{\mathbf{D}}_{t+1}$ that best approximate $\tilde{V}_{t+1}$ in terms of Hilbert norm error. To accomplish this, we use *kernel orthogonal matching pursuit* (KOMP) [40] with error tolerance $\epsilon_t$ to find a dictionary $\mathbf{D}_{t+1}$ based that which adds the latest samples $\tilde{\mathbf{D}}_{t+1}$. We tune $\epsilon_t$ to ensure both stochastic descent (Lemma 1(ii)) and finite model order (Corollary 1).

With respect to the KOMP procedure above, we specifically use a variant called destructive KOMP with pre-fitting (see [40], Section 2.3). This flavor of KOMP takes as an input a candidate function $\tilde{V}$ of model order $\tilde{M}$ parameterized by its dictionary $\tilde{\mathbf{D}} \in \mathbb{R}^{p \times \tilde{M}}$ and coefficients $\tilde{\mathbf{w}} \in \mathbb{R}^{\tilde{M}}$. The method then approximates $\tilde{V}$ by $V \in \mathcal{H}$ with a lower model order. Initially, the candidate is the original $V = \tilde{V}$ so that its dictionary is initialized with $\mathbf{D} = \tilde{\mathbf{D}}$, with coefficients $\mathbf{w} = \tilde{\mathbf{w}}$. Then, we sequentially and greedily remove model points from initial dictionary $\tilde{\mathbf{D}}$ until threshold $\|V - \tilde{V}\|_{\mathcal{H}} \leq \epsilon_t$ is violated. The result is a sparse approximation $V$ of $\tilde{V}$.

This process is executed via destructive KOMP. At each stage, a single dictionary element $j$ of $\mathbf{D}$ is selected to be removed which contributes the least to the Hilbert-norm approximation error $\min_{V \in \mathcal{H}_{\mathbf{D}\setminus\{j\}}} \|\tilde{V} - V\|_{\mathcal{H}}$ of the original function $\tilde{V}$, when dictionary $\mathbf{D}$ is used. Since at each stage the kernel dictionary is fixed, this amounts to a computation involving weights $\mathbf{w} \in \mathbb{R}^{M-1}$ only; that is, the error of removing dictionary point $\mathbf{d}_j$ is computed for each $j$ as $\gamma_j = \min_{\mathbf{w}_{\mathcal{I}\setminus\{j\}} \in \mathbb{R}^{M-1}} \|\tilde{V}(\cdot) - \sum_{k \in \mathcal{I}\setminus\{j\}} w_k \kappa(\mathbf{d}_k, \cdot)\|$. We use the notation $\mathbf{w}_{\mathcal{I}\setminus\{j\}}$ to denote the entries of $\mathbf{w} \in \mathbb{R}^M$ restricted to the sub-vector associated with indices $\mathcal{I} \setminus \{j\}$. Then, we define the dictionary element which contributes the least to the approximation error as $j^* = \operatorname{argmin}_j \gamma_j$. If the error associated with removing this kernel dictionary element exceeds the given approximation budget $\gamma_{j^*} > \epsilon_t$, the algorithm terminates. Otherwise, this dictionary element $\mathbf{d}_{j^*}$ is removed, the weights $\mathbf{w}$ are revised based on the pruned dictionary as $\mathbf{w} = \operatorname{argmin}_{\mathbf{w} \in \mathbb{R}^M} \|\tilde{f}(\cdot) - \mathbf{w}^T \boldsymbol{\kappa}_{\mathbf{D}}(\cdot)\|_{\mathcal{H}}$, and the process repeats as long as the current function approximation is defined by a nonempty dictionary. These steps are summarized in Algorithm 2

We summarize the proposed method, Parsimonious Kernel Gradient Temporal Difference (PKGTD) in Algorithm 1: we execute the stochastic projection of the functional SQG iterates onto sparse subspaces $\mathcal{H}_{\mathbf{D}_{t+1}}$ stated in (17). With initial function null $V_0 = 0$ (empty dictionary $\mathbf{D}_0 = []$ and coefficients $\mathbf{w}_0 = []$),at each step, given an i.i.d. sample $(\mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)$ and step-sizes $\alpha_t, \beta_t$, we compute the *unconstrained* functional SQG iterate $\tilde{V}_{t+1}(\cdot) = (1 - \alpha_t\lambda)\tilde{V}_t(\cdot) - \alpha_t(\gamma\kappa(\mathbf{y}_t, \cdot) - \kappa(\mathbf{x}_t, \cdot))z_{t+1}$ parameterized by $\tilde{\mathbf{D}}_{t+1}$ and $\tilde{\mathbf{w}}_{t+1}$ as stated in (19), which are fed into KOMP (Algorithm 2) with budget $\epsilon_t$, i.e., $(V_{t+1}, \mathbf{D}_{t+1}, \mathbf{w}_{t+1}) = \text{KOMP}(\tilde{V}_{t+1}, \tilde{\mathbf{D}}_{t+1}, \tilde{\mathbf{w}}_{t+1}, \epsilon_t)$.

## IV. Convergence Analysis

We now analyze the stability and memory requirements of Algorithm 1 developed in Section III. Our approach is fundamentally different from stochastic fixed point methods such as TD learning, which are not descent techniques, and thus exhibit delicate convergence. The interplay between the Bellman operator contraction [6] and expectations prevents the construction of supermartingales underlying stochastic descent stability [41]. Attempts to mitigate this issue, such as those based on stochastic backward-differences [42] ( [43], [44]) or Lyapunov approaches [45], e.g., [20], require the state space to be completely explored in the limit *per step* (intractable when $|\mathcal{X}| = \infty$), or stipulate that data dependent matrices be non-singular, respectively. Thus, there is a long-standing question of how to perform policy evaluation in MDPs under conditions applicable to practitioners while also guaranteeing stability. We provide an answer by connecting RKHS-valued stochastic quasi-gradient methods (Algorithm 1) with coupled supermartingale theory [46].

Before continuing, we introduce a few key assumptions and definitions which are required to establish convergence. In particular, for further reference, we define the functional stochastic quasi-gradient of the regularized objective as

$$\begin{aligned}\hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t) =& \\ (\gamma\kappa(\mathbf{y}_t, \cdot) - \kappa(\mathbf{x}_t, \cdot))z_{t+1} + \lambda V_t \ ,& \qquad (22)\end{aligned}$$

and its sparse-subspace projected variant as

$$\begin{aligned}&\tilde{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t) \qquad (23)\\ &= \frac{1}{\alpha_t}\Big(V_t - \mathcal{P}_{\mathcal{H}_{\mathbf{D}_{t+1}}}\Big[V_t - \alpha_t \hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)\Big]\Big) \ ,\end{aligned}$$

Note that the update (15), using (23), may be rewritten as a stochastic projected quasi-gradient step rather than a stochastic quasi-gradient step followed by set projection, i.e.,

$$V_{t+1} = V_t - \alpha_t \tilde{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t) \ , \qquad (24)$$

Now we are ready to state the technical conditions required for convergence.

**Assumption 1.** *The state space $\mathcal{X} \subset \mathbb{R}^p$ and action space $\mathcal{A} \subset \mathbb{R}^q$ are compact, and the reproducing kernel map may be bounded as*

$$\sup_{\mathbf{x}\in\mathcal{X}} \sqrt{\kappa(\mathbf{x}, \mathbf{x})} = X < \infty \qquad (25)$$

**Assumption 2.** *The temporal difference $\delta$ and auxiliary sequence $z$ [cf.* (11)*] satisfy the zero-mean, finite conditional variance, and Lipschitz continuity conditions, respectively,*

$$\begin{aligned}&\mathbb{E}\left[\delta \,\middle|\, \mathbf{x}, \pi(\mathbf{x})\right] = \bar{\delta} \ , \qquad \mathbb{E}\left[(\delta - \bar{\delta})^2\right] \leq \sigma_\delta^2 \ , \\ &\mathbb{E}\left[z^2 \,\middle|\, \mathbf{x}, \pi(\mathbf{x})\right] \leq G_\delta^2 \ . \qquad (26)\end{aligned}$$

*where $\sigma_\delta$ and $G_\delta$ are positive scalars.*

**Assumption 3.** *The functional gradient of the temporal difference is an unbiased estimate for $\nabla_V J(V)$ [cf.* (9)*], and the difference of reproducing kernels expression (the first term in the product expression* (10)*) has finite conditional variance:*

$$\begin{aligned}&\mathbb{E}\left[(\gamma\kappa(\mathbf{y}, \cdot) - \kappa(\mathbf{x}, \cdot))\delta\right] = \nabla_V J(V) \ , \\ &\mathbb{E}\left[\|\gamma\kappa(\mathbf{y}_t, \cdot) - \kappa(\mathbf{x}_t, \cdot)\|_{\mathcal{H}}^2 \,\middle|\, \mathcal{F}_t\right] \leq G_V^2 \ . \qquad (27)\end{aligned}$$

*Moreover, the projected stochastic quasi-gradient of the objective [cf.* (23)*] has finite second conditional moment as*

$$\mathbb{E}\Big[\|\tilde{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)\|_{\mathcal{H}}^2 \,\Big|\, \mathcal{F}_t\Big] \leq \sigma_V^2 \ , \qquad (28)$$

*and the conditional mean of the temporal difference $\bar{\delta}$ is Lipschitz continuous with respect to the value function $V$, i.e for any two distinct $\delta$ and $\tilde{\delta}$, we have*

$$|\bar{\delta} - \bar{\tilde{\bar{\delta}}}| \leq L_V \|V - \tilde{V}\|_{\mathcal{H}} \qquad (29)$$

*where $V, \tilde{V} \in \mathcal{H}$ are distinct value functions in the RKHS, $L_V > 0$ is a scalar, and $\bar{\delta} = \mathbb{E}_{\mathbf{y}}[r(\mathbf{x}, \pi(\mathbf{x}), \mathbf{y}) + \gamma V(\mathbf{y}) - V(\mathbf{x}) \,|\, \mathbf{x}, \pi(\mathbf{x})]$.*

Assumption 1 regarding the compactness of the state and action spaces of the Markov Decision Process intrinsically hold for most application settings and limit the radius of the set from which the MDP trajectory is sampled. Similar boundedness conditions on the reproducing kernel map have been considered in supervised learning applications [38]. The mean and variance properties of the temporal difference stated in Assumption 2 are necessary to bound the error in the descent direction associated with stochastic approximations, and are necessary to establish stability properties of stochastic methods. Assumption 3 is similar to Assumption 2 but instead of establishing bounds on the stochastic approximation error of the temporal difference, limits stochastic error variance in the reproducing kernel Hilbert space. These are natural extensions of the conditions needed for convergence of stochastic compositional gradient methods with vector-valued decision variables [33]. However, we note that (29) restricts the class of reward functions to be those which may be smoothly interpolated in a RKHS.

The stipulation that $V_t$ belongs to a RKHS means that it has finite Hilbert norm for all time [22], which allows us to write

$$\|V_t\|_{\mathcal{H}} \leq K \ , \qquad \|V^*\|_{\mathcal{H}} \leq K \ , \qquad \text{for all } t \qquad (30)$$

where $K > 0$ is some constant. The boundedness of $V^*$ follows from the fact that since $\mathcal{X}$ is compact and $J(V)$ is a continuous convex function over a compact set, its minimizer is achieved over this compact set [48][Corrolary 3.23].

**Iterate Convergence** Under the assumptions stated at the beginning of this section, it is possible to derive the fact that the auxiliary variable $z_t$ and value function estimate $V_t$ satisfy supermartingale-type relationships, but their behavior is intrinsically coupled to one another. We generalize recently developed coupled supermartingale tools in [46], i.e., Lemma 2 in Appendix A, to RKHSs in order to establish the following almost sure convergence result when the step-sizes and compression budget are diminishing.

**Theorem 1.** *Consider the sequence $z_t$ [cf.* (11)*] and $\{V_t\}$ [cf. 15] as stated in Algorithm 1. Assume the regularizer is positive $\lambda > 0$, Assumptions 1 - 3 hold, and the step-size conditions hold:*

$$\begin{aligned}&\sum_{t=1}^{\infty} \alpha_t = \infty \ , \quad \sum_{t=1}^{\infty} \beta_t = \infty, \\ &\sum_{t=1}^{\infty} \alpha_t^2 + \beta_t^2 + \frac{\alpha_t^2}{\beta_t} < \infty \ , \quad \epsilon_t = \alpha_t^2 \qquad (31)\end{aligned}$$

*Then $V_t \to V^*$ defined by* (8) *with probability* 1*, and thus achieves the regularized Bellman fixed point* (4) *restricted to the reproducing kernel Hilbert space.*

The proof is given in Appendix B. Theorem 1 states that the value functions generated by Algorithm 1 converge almost surely to the optimal $V^*$ defined by (8). With regularizer $\lambda$ made arbitrarily small but nonzero, using a universal kernel (e.g., a Gaussian), $V_t$ converges arbitrarily close to a function satisfying Bellman's equation in *infinite MDPs* (3). This is the first guarantee w.p.1 for a true stochastic descent method with an infinitely and nonlinearly parameterized value function. Theorem 1 requires attenuating step-sizes such that the stochastic approximation error approaches null. In contrast, constant learning rates allow for the perpetual revision of the value function estimates without diminishing algorithm adaptivity, motivating the following result.

One step-size sequence which satisfies the attenuation conditions (31) is $\alpha_t = \mathcal{O}(t^{-(3/4+\zeta/2)})$ ,$\beta_t = \mathcal{O}(t^{-(1+\zeta)/2})$ ,$\epsilon_t = \mathcal{O}(\alpha_t^2) = \mathcal{O}(t^{-(3/2+\zeta)})$, where $\zeta > 0$ is an arbitrarily small constant so that series $\sum_t \alpha_t$ and $\sum_t \beta_t$ diverge. Generally, satisfying (31), requires: $\alpha_t = \mathcal{O}(t^{-p_\alpha})$, $\beta_t = \mathcal{O}(t^{-p_\beta})$ with $p_\alpha \in (3/4, 1)$ and $p_\beta \in (1/2, 2p_\alpha - 1)$.

**Theorem 2.** *Suppose Algorithm 1 is run with constant positive learning rates $\alpha_t = \alpha$ and $\beta_t = \beta$ and constant compression budget $\epsilon_t = \epsilon$ with sufficiently large regularization, i.e.*

$$0 < \beta < 1 \,, \alpha = \beta, \epsilon = C\alpha^2, \lambda = G_V^2 \frac{\alpha}{\beta} + \lambda_0 \tag{32}$$

*where $C > 0$ is a scalar, and $0 < \lambda_0 < 1$. Then, under Assumptions 1 - 3, the sub-optimality sequence $\|V_t - V^*\|^2_{\mathcal{H}}$ converges in mean to a neighborhood:*

$$\limsup_{t\to\infty} \mathbb{E}\left[\|V_t - V^*\|^2_{\mathcal{H}}\right] = \mathcal{O}\left(\alpha + \alpha^2 + \alpha^3\right) \,. \tag{33}$$

Theorem 2 (proof in Appendix C) establishes that the value function estimates generated by Algorithm 1 converge in expectation to a neighborhood when constant step-sizes $\alpha$ and $\beta$ and sparsification budget $\epsilon$ in Algorithm 2 are small constants. In particular, the bias $\epsilon$ induced by sparsification does not cause instability even when it is *not going to null*. Moreover, this result only holds when the regularizer $\lambda$ is chosen large enough, which numerically induces a forgetting factor on past kernel dictionary weights (19). We may make the learning rates $\alpha$ and $\beta$ arbitrarily small, which yield a proportional decrease in the radius of convergence to a neighborhood of the Bellman fixed point (3).

**Remark 1.** (Aggressive Constant Learning Rates) In practice, one may obtain better performance by using larger constant step-sizes. To do so, the criterion (32) may be relaxed: we require $0 < \beta < 1$ but $\alpha > 0$ may be any positive scalar. Then, the radius of convergence is (see Appendix C)

$$\begin{aligned}\limsup_{t\to\infty} \mathbb{E}&\left[\|V_t - V^*\|^2_{\mathcal{H}}\right] \\ &= \mathcal{O}\Big(\alpha^2 + \beta^2 + \frac{\alpha^2}{\beta}\Big[1 + \alpha^2 + \frac{\alpha}{\beta} + \frac{\alpha^2}{\beta^2}\Big]\Big) \,.\end{aligned} \tag{34}$$

The ratios $\alpha^2/\beta$ and $\alpha^2/\beta^2$ dominate (34) and must be made small to obtain accurate solutions.

Theorem 2 is the first constant learning rate result for non-parametric compositional stochastic programming of which we are aware, and allows for repeatedly revising value function without the need for stochastic approximation error to approach null. Use of constant learning rates yields the fact that value function estimates have moderate complexity even in the worst case, as we detail next.

**Model Order Control** As noted in Section III, the complexity of functional stochastic quasi-gradient method in a RKHS is of order $\mathcal{O}(2(t-1))$ which grows without bound. To mitigate this issue, we develop the sparse subspace projection in Section III-A. We formalize here that this projection does indeed limit the complexity of the value function when constant learning rates and compression budget are used. This result is a corollary, since it is an extension of Theorem 3 in [32]. To obtain this result (proof in Appendix D), we require the reward function to be bounded, as we state next.

**Assumption 4.** *The reward function $r : \mathcal{X} \times \mathcal{A} \times \mathcal{X} \to \mathbb{R}$ is bounded for all $\mathbf{x}, \mathbf{a}, \mathbf{y}$, i.e.,*

$$r(\mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t) \leq R_{\max} \;\; \textit{for all } t \tag{35}$$

Assumption 4 holds whenever the reward function is continuous and the state and action spaces are compact, and thus is not restrictive as these conditions are met in most practical settings. In this setting, we have the following finite-memory property of Algorithm 1. Under this additional condition, we obtain that the complexity of Algorithm 1 is finite, as we state next.

**Corollary 1.** *Denote $V_t$ as the value function sequence defined by Algorithm 1 with constant step-sizes $\alpha_t = \alpha$ and $\beta_t = \beta \in (0, 1)$ with compression budget $\epsilon_t = \epsilon = C\alpha^2$ and regularization parameter $\lambda = (\alpha/\beta)G_V^2 + \lambda_0 = \mathcal{O}(\alpha\beta^{-1} + 1)$ as in Remark 1. Let $M_t$ be the model order of the value function $V_t$ i.e., the number of columns of the dictionary $\mathbf{D}_t$ which parameterizes $V_t$. Then there exists a finite upper bound $M^\infty$ such that, for all $t \geq 0$, the model order is always bounded as $M_t \leq M^\infty$. Consequently, the model order of the limiting function $V^\infty = \lim_t V_t$ is finite.*

The result above, whose proof is given in Appendix D, establishes that Algorithm 1 yields convergent behavior for the problem (8) in both diminishing and constant step-size regimes. With diminishing step-sizes [cf. (31)] and compression budget $\epsilon_t = \mathcal{O}(\alpha_t^2)$, we obtain exact convergence with probability 1 of the function sequence $\{V_t\}$ in the RKHS to that of the regularized Bellman fixed point of the evaluation equation $V^*$ (Theorem 1). This result holds for any positive regularizer $\lambda > 0$, and thus can be made arbitrarily close to the *true Bellman fixed point* $V^\pi$ [cf. (2)] by decreasing $\lambda$. However, an exact solution requires increasing the complexity of the function estimate such that its limiting memory becomes infinite. This drawback motivates us to consider the case where both the learning rates $\alpha_t = \alpha$, $\beta_t = \beta$ and the compression budget $\epsilon_t = \epsilon$ are constant. Under specific selections (32), the algorithm converges to a neighborhood of the optimal value function, whose radius depends on the step-sizes, and may be made small by decreasing $\alpha$ at the cost of a decreasing

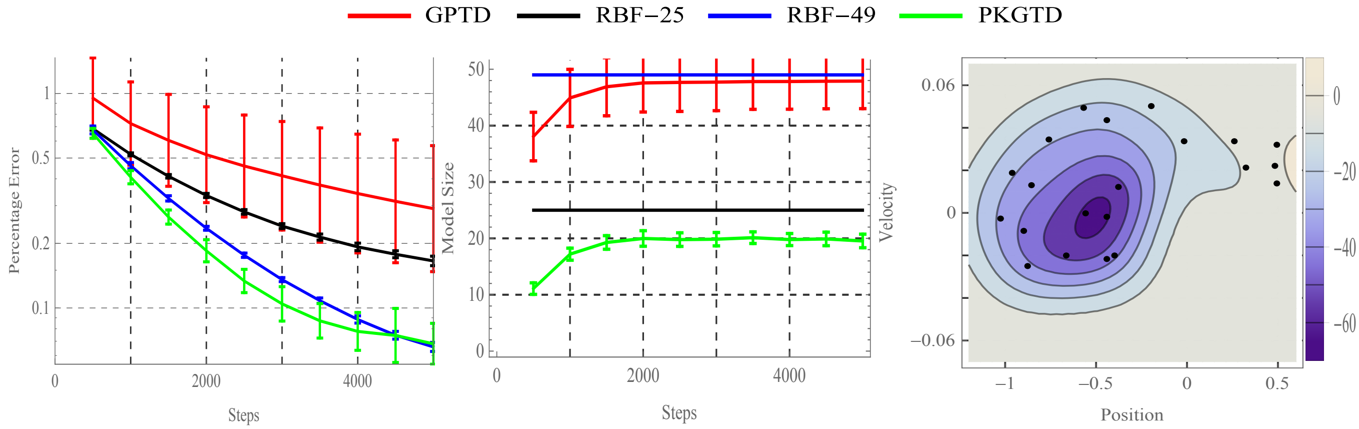

Fig. 1. Experimental comparison of PKGTD to existing kernel methods for policy evaluation on the Mountain Car task. Test set error (left), and the parameterization complexity (center) vs. iterations. PKGTD learns fastest and most stably with the least complexity (best viewed in color). We plot the contour of the learned value function (right): its minimal value is in the valley, and states near the goal are close to null. Bold black dots are kernel dictionary elements, or retained instances.

TABLE I
EXPERIMENT PARAMETERS

| | $\alpha$ | $\beta$ | $\lambda$ | $K$ | $\sigma_0$ | $\nu$ | $h_1$ | $h_2$ |
|---|---|---|---|---|---|---|---|---|
| PKGTD | 8.0 | 0.2 | 1e-6 | 0.02 | | | | |
| GPTD | | | 1e-6 | | 0.01 | 0.2 | | |
| RBF-25 | 10.0 | 0.25 | | | | | 0.44 | 0.0343 |
| RBF-49 | 1.5 | 0.35 | | | | | 0.26 | 0.0203 |

learning rate. Moreover, the use of constant step-sizes and compression budget with large enough regularization yields a value function parameterized by a dictionary whose model order is always bounded (Corollary 1).

## V. EXPERIMENTS

Our experiments aim to compare PKGTD to other policy evaluation techniques in this domain. Because it seeks memory-efficient solutions over an RKHS, we expect PKGTD to obtain accurate estimates of the value function using only a fraction of the memory required by the other methods. We perform experiments on the classical Mountain Car domain [1]: an agent applies discrete actions $\mathcal{A} = \{\texttt{reverse}, \texttt{coast}, \texttt{forward}\}$ to a car that starts at the bottom of a valley and attempts to climb up to a goal at the top of one of the mountain sides. The state space is continuous, consisting of the car's scalar position and velocity, i.e., $\mathcal{X} = \mathbb{R}^2$. The reward function $r(\mathbf{x}_t, \mathbf{a}_t, \mathbf{y}_t)$ is $-1$ unless $\mathbf{y}_t$ is the goal state at the mountain top, in which case it is 0 and the episode terminates.

Now we describe the configuration of the algorithms used for comparison. The Mountain Car environment has a two-dimensional state space (position and velocity) with bounds of $[-1.2, 0.6]$ in position, and $[-0.07, 0.07]$ in velocity. We chose not to normalize this state space to $[0, 1]$ intervals, choosing instead to handle the scale difference by using non-isotropic kernels. The ratio of the kernel variances is equal to the ratio of the lengths of their corresponding bounds, so they would be isotropic kernels if we normalized the state space.

We used a fixed non-isotropic kernel bandwidth of $\sigma_1 = 0.2, \sigma_2 = 0.0156$ in all cases. By fixing the kernel bandwidth across all algorithms, we are basically enforcing that the learned functions all belong to the same Kernel Hilbert Space.

For PKGTD, the relevant parameters are the step size, $\alpha$, the rate of expectation update, $\beta$, the regularizer, $\lambda$, and the approximation error, $K$. For GPTD, the relevant parameters are the gaussian process noise standard deviation, $\sigma_0$, the linear independence test bound, $\nu$, and the regularizer, $\lambda$. For the RBF grids fit using GTD, the relevant parameters are the grid spacing in the position and velocity directions, $h_1$ and $h_2$, respectively, the step size, $\alpha$, and the rate of expectation update, $\beta$. Our values are summarized in Table. I.

To obtain a benchmark policy for this task, we make use of trust region policy optimization [47]. To evaluate value function estimates, we form an offline training set of state transitions and associated rewards by running this policy through consecutive episodes until we had one training trajectory of 5000 steps and then repeat this for 100 training trajectories to generate sample statistics. For ground truth, we generate one long trajectory of 10000 steps and randomly sample 2000 states from it. From each of these 2000 states, we apply the policy until episode termination and use the observed discounted return as $\hat{V}_\pi(\mathbf{x})$. Since our policy was deterministic, we only performed this procedure once per sampled state. For value function $V$, we define the percentage error metric: Percentage Error$(V) = (1/2000)\sum_{i=1}^{2000} |(V(\mathbf{x}_i) - \hat{V}_\pi(\mathbf{x}_i))/\hat{V}_\pi(\mathbf{x}_i)|$ We compared PKGTD with a Gaussian kernel to two other techniques for policy evaluation that also use kernel-based value function representations: (1) Gaussian process temporal difference (GPTD) [31], and (2) gradient temporal difference (GTD) [20] using radial basis function (RBF) network features.

Figure 1 depicts the results of our experiment. We fix a kernel bandwidth across all techniques, and select parameter values that yield the best results for each method (see Table I). For RBF feature generation, we use two fixed grids with different spacing. The first was one for which GTD yielded a value function estimate with percentage error similar to that which we obtained using PKGTD (RBF-49), and the second was one which yielded a number of basis functions that was similar to what PKGTD selected (RBF-25). Observe that GTD with fixed RBF features requires a much denser grid in order to reach the same Percentage Error as Algorithm 1. Moreover, PKGTD's adaptive instance selection results in both faster initial learning and smaller error. Compared to GPTD, which chooses model points online according to a fixed linear-dependence criterion, PKGTD requires fewer model points and converges to a better estimate of the value function more quickly and stably.

## VI. Discussion

In this paper, we considered the problem of policy evaluation in infinite MDPs with value functions that belong to a RKHS. To solve this problem, we extended recent SQG methods for compositional stochastic programming to a RKHS, and used the result, combined with greedy sparse subspace projection, in a new policy-evaluation procedure called PKGTD (Algorithm 1). Under diminishing step sizes, PKGTD solves Bellman's evaluation equation exactly under the hypothesis that its fixed point belongs to a RKHS (Theorem 1). Under constant step sizes, we can further guarantee finite-memory approximations (Corollary 1) that still exhibit mean convergence to a neighborhood of the optimal value function (Theorem 2). In our Mountain Car experiments, PKGTD yields excellent sample efficiency and model complexity, and therefore holds promise for large state space problems common in robotics where fixed state-action space tiling may prove impractical.

## References

[1] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press Cambridge, 1998, vol. 1, no. 1.

[2] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *The International Journal of Robotics Research*, p. 0278364913495721, 2013.

[3] W. R. Scott, W. B. Powell, and S. Moazehi, "Least squares policy iteration with instrumental variables vs. direct policy search: Comparison against optimal benchmarks using energy storage," *arXiv preprint arXiv:1401.0843*, 2014.

[4] M. G. Lagoudakis and R. Parr, "Least-squares policy iteration," *Journal of Machine Learning Research*, vol. 4, no. Dec, pp. 1107–1149, 2003.

[5] R. Bellman, *Dynamic Programming*, 1st ed. Princeton, NJ, USA: Princeton University Press, 1957. [Online]. Available: http://books.google.com/books?id=fyVtp3EMxasC&pg=PR5&dq=dynamic+programming+richard+e+bellman&client=firefox-a#v=onepage&q=dynamic%20programming%20richard%20e%20bellman&f=false

[6] D. P. Bertsekas and S. E. Shreve, *Stochastic optimal control: The discrete time case*. Academic Press, 1978, vol. 23.

[7] R. S. Sutton, "Learning to predict by the methods of temporal differences," *Machine learning*, vol. 3, no. 1, pp. 9–44, 1988.

[8] S. J. Bradtke and A. G. Barto, "Linear least-squares algorithms for temporal difference learning," *Machine learning*, vol. 22, no. 1-3, pp. 33–57, 1996.

[9] W. B. Powell and J. Ma, "A review of stochastic algorithms with continuous value function approximation and some new approximate policy iteration algorithms for multidimensional continuous applications," *Journal of Control Theory and Applications*, vol. 9, no. 3, pp. 336–352, 2011.

[10] X. Xu, T. Xie, D. Hu, and X. Lu, "Kernel least-squares temporal difference learning," *International Journal of Information Technology*, vol. 11, no. 9, pp. 54–63, 2005.

[11] J. N. Tsitsiklis and B. Van Roy, "An analysis of temporal-difference learning with function approximation," *IEEE transactions on automatic control*, vol. 42, no. 5, pp. 674–690, 1997.

[12] F. S. Melo, S. P. Meyn, and M. I. Ribeiro, "An analysis of reinforcement learning with function approximation," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 664–671.

[13] S. Bhatnagar, D. Precup, D. Silver, R. S. Sutton, H. R. Maei, and C. Szepesvári, "Convergent temporal-difference learning with arbitrary smooth function approximation," in *Advances in Neural Information Processing Systems*, 2009, pp. 1204–1212.

[14] L. Baird, "Residual algorithms: Reinforcement learning with function approximation," in *In Proceedings of the Twelfth International Conference on Machine Learning*. Morgan Kaufmann, 1995, pp. 30–37.

[15] N. K. Jong and P. Stone, "Model-based function approximation in reinforcement learning," in *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*. ACM, 2007, p. 95.

[16] A. Shapiro, D. Dentcheva *et al.*, *Lectures on stochastic programming: modeling and theory*. Siam, 2014, vol. 16.

[17] A. Korostelev, "Stochastic recurrent procedures: Local properties," *Nauka: Moscow (in Russian)*, 1984.

[18] V. R. Konda and J. N. Tsitsiklis, "Convergence rate of linear two-time-scale stochastic approximation," *Annals of applied probability*, pp. 796–819, 2004.

[19] Y. Ermoliev, "Stochastic quasigradient methods and their application to system optimization," *Stochastics: An International Journal of Probability and Stochastic Processes*, vol. 9, no. 1-2, pp. 1–36, 1983.

[20] R. S. Sutton, H. R. Maei, and C. Szepesvári, "A convergent $o(n)$ temporal-difference algorithm for off-policy learning with linear function approximation," in *Advances in neural information processing systems*, 2009, pp. 1609–1616.

[21] G. Kimeldorf and G. Wahba, "Some results on tchebycheffian spline functions," *Journal of mathematical analysis and applications*, vol. 33, no. 1, pp. 82–95, 1971.

[22] K. Slavakis, P. Bouboulis, and S. Theodoridis, "Online learning in reproducing kernel hilbert spaces," *Signal Processing Theory and Machine Learning*, pp. 883–987, 2013.

[23] D. Ormoneit and Ś. Sen, "Kernel-based reinforcement learning," *Machine learning*, vol. 49, no. 2-3, pp. 161–178, 2002.

[24] G. Taylor and R. Parr, "Kernelized value function approximation for reinforcement learning," in *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM, 2009, pp. 1017–1024.

[25] S. Grünewälder, G. Lever, L. Baldassarre, M. Pontil, and A. Gretton, "Modelling transition dynamics in mdps with rkhs embeddings," in *International Conference on Machine Learning*, vol. 1, 2012, pp. 535–542.

[26] A.-m. Farahmand, C. Ghavamzadeh, Mohammadand Szepesvári, and S. Mannor, "Regularized policy iteration with nonparametric function spaces," *Journal of Machine Learning Research*, vol. 17, no. 139, pp. 1–66, 2016. [Online]. Available: http://jmlr.org/papers/v17/13-016.html

[27] B. Dai, N. He, Y. Pan, B. Boots, and L. Song, "Learning from conditional distributions via dual kernel embeddings," *arXiv preprint arXiv:1607.04579*, 2016.

[28] Y. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal Matching Pursuit: Recursive Function Approximation with Applications to Wavelet Decomposition," in *Proceedings of the Asilomar Conference on Signals, Systems and Computers*, 1993.

[29] G. Lever, J. Shawe-Taylor, R. Stafford, and C. Szepesvari, "Compressed conditional mean embeddings for model-based reinforcement learning," in *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[30] E. J. Candes, "The restricted isometry property and its implications for compressed sensing," *Comptes Rendus Mathematique*, vol. 346, no. 9, pp. 589–592, 2008.

[31] Y. Engel, S. Mannor, and R. Meir, "Bayes meets bellman: The gaussian process approach to temporal difference learning," in *Proc. of the 20th International Conference on Machine Learning*, 2003.

[32] A. Koppel, G. Warnell, E. Stump, and A. Ribeiro, "Parsimonious online learning with kernels via sparse projections in function space," *arXiv preprint arXiv:1612.04111*, 2016.

[33] M. Wang, E. X. Fang, and H. Liu, "Stochastic compositional gradient descent: Algorithms for minimizing compositions of expected-value functions," *Mathematical Programming*, vol. 161, no. 1-2, pp. 419–449, 2017.

[34] B. Schölkopf, R. Herbrich, and A. J. Smola, “A generalized representer theorem,” in *International Conference on Computational Learning Theory*. Springer, 2001, pp. 416–426.

[35] V. Norkin and M. Keyzer, “On stochastic optimization and statistical learning in reproducing kernel hilbert spaces by support vector machines (svm),” *Informatica*, vol. 20, no. 2, pp. 273–292, 2009.

[36] R. Wheeden, R. Wheeden, and A. Zygmund, *Measure and Integral: An Introduction to Real Analysis*, ser. Chapman & Hall/CRC Pure and Applied Mathematics. Taylor & Francis, 1977. [Online]. Available: https://books.google.com/books?id=YDkDmQ_hdmcC

[37] C. A. Micchelli, Y. Xu, and H. Zhang, “Universal kernels,” *Journal of Machine Learning Research*, vol. 7, no. Dec, pp. 2651–2667, 2006.

[38] J. Kivinen, A. J. Smola, and R. C. Williamson, “Online Learning with Kernels,” *IEEE Transactions on Signal Processing*, vol. 52, pp. 2165–2176, August 2004.

[39] D. Needell, J. Tropp, and R. Vershynin, “Greedy signal recovery review,” in *Signals, Systems and Computers, 2008 42nd Asilomar Conference on*. IEEE, 2008, pp. 1048–1050.

[40] P. Vincent and Y. Bengio, “Kernel matching pursuit,” *Machine Learning*, vol. 48, no. 1, pp. 165–187, 2002.

[41] H. Robbins and S. Monro, “A stochastic approximation method,” *Ann. Math. Statist.*, vol. 22, no. 3, pp. 400–407, 09 1951.

[42] J. Kiefer, J. Wolfowitz *et al.*, “Stochastic estimation of the maximum of a regression function,” *The Annals of Mathematical Statistics*, vol. 23, no. 3, pp. 462–466, 1952.

[43] J. N. Tsitsiklis, “Asynchronous stochastic approximation and q-learning,” *Machine Learning*, vol. 16, no. 3, pp. 185–202, 1994.

[44] T. Jaakkola, M. I. Jordan, and S. P. Singh, “On the convergence of stochastic iterative dynamic programming algorithms,” *Neural computation*, vol. 6, no. 6, pp. 1185–1201, 1994.

[45] V. S. Borkar and S. P. Meyn, “The ode method for convergence of stochastic approximation and reinforcement learning,” *SIAM Journal on Control and Optimization*, vol. 38, no. 2, pp. 447–469, 2000.

[46] M. Wang and D. P. Bertsekas, “Incremental constraint projection-proximal methods for nonsmooth convex optimization,” *SIAM Journal on Optimization (to appear)*, 2014.

[47] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, “Trust region policy optimization,” in *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, 2015, pp. 1889–1897.

[48] H. Brezis, *Functional analysis, Sobolev spaces and partial differential equations*. Springer Science & Business Media, 2010.

[49] M. Anthony and P. L. Bartlett, *Neural network learning: Theoretical foundations*. cambridge university press, 2009.

[50] Y. Engel, S. Mannor, and R. Meir, “The kernel recursive least-squares algorithm,” *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2275–2285, Aug 2004.

## Appendix

### A. Auxiliary Results and Technical Lemmas

Next we turn to establishing some technical results which are necessary precursor to the proofs of the main stability results.

**Proposition 1.** *Given independent identical realizations* $(\mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)$ *of the random triple* $(\mathbf{x}, \pi(\mathbf{x}), \mathbf{y})$*, the difference between the projected stochastic functional quasi-gradient and the stochastic functional quasi-gradient of the instantaneous cost instantaneous risk defined by* (22) *and* (23)*, respectively, is bounded for all* $t$ *as*

$$\begin{aligned}\|\tilde{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t) - \hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)\|_{\mathcal{H}} \\ \leq \frac{\epsilon_t}{\alpha_t} \end{aligned} \tag{36}$$

*where* $\alpha_t > 0$ *denotes the algorithm step-size and* $\epsilon_t > 0$ *is the compression budget parameter of Algorithm 2.*

**Proof :** As in Proposition 6 of [32], consider the square-Hilbert-norm difference of $\tilde{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)$ and $\hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)$ defined in (22) and (23), respectively,

$$\begin{aligned}&\|\tilde{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t) - \hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)\|_{\mathcal{H}} \\ &= \Big\| \frac{1}{\alpha} \Big( V_t - \mathcal{P}_{\mathcal{H}_{\mathbf{D}_{t+1}}} \Big[ V_t - \alpha_t \hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t) \Big] \Big) \\ &\qquad - \hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t) \Big\|_{\mathcal{H}}^2 \end{aligned} \tag{37}$$

Multiply and divide $\hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)$, the last term, by $\alpha_t$, and reorder terms to write

$$\begin{aligned}&\Bigg\| \frac{\Big(V_t - \alpha_t \hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)\Big)}{\alpha_t} \\ &\qquad - \frac{\mathcal{P}_{\mathcal{H}_{\mathbf{D}_{t+1}}} \Big[ V_t - \alpha_t \hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t) \Big] \Big)}{\alpha_t} \Bigg\|_{\mathcal{H}}^2 \\ &= \frac{1}{\alpha_t^2} \Big\| \Big( V_t - \alpha_t \hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t) \\ &\qquad - \mathcal{P}_{\mathcal{H}_{\mathbf{D}_{t+1}}} \Big[ V_t - \alpha_t \hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t) \Big] \Big) \Big\|_{\mathcal{H}}^2 \\ &= \frac{1}{\alpha_t^2} \|\tilde{V}_{t+1} - V_{t+1}\|_{\mathcal{H}}^2 \leq \frac{\epsilon_t^2}{\alpha_t^2} \end{aligned} \tag{38}$$

where we have pulled the nonnegative scalar $\alpha_t$ outside the norm on the second line and substituted the definition of $\tilde{V}_{t+1}$ and $V_{t+1}$ in (12) and (15), respectively, in the last one. These facts combined with the KOMP residual stopping criterion in Algorithm 2 is $\|\tilde{V}_{t+1} - V_{t+1}\|_{\mathcal{H}} \leq \epsilon_t$ applied to the last term on the right-hand side of (38) yields (36).

■

**Lemma 1.** *Denote the filtration* $\mathcal{F}_t$ *as the time-dependent sigma-algebra containing the algorithm history* $\mathcal{F}_t \supset (\{V_u, z_u\}_{u=0}^t \cup \{\mathbf{x}_s, \pi(\mathbf{x}_s), \mathbf{y}_s\}_{s=0}^{t-1})$ *Let Assumptions 1 - 3 hold true and consider the sequence of iterates defined by Algorithm 1. Then:*

i) *The conditional expectation of the Hilbert-norm difference of value functions at the next and current iteration satisfies the relationship*

$$\mathbb{E}\big[\|V_{t+1} - V_t\|_{\mathcal{H}}^2 \,\big|\, \mathcal{F}_t\big] \leq 2\alpha_t^2 (G_\delta^2 G_V^2 + \lambda^2 K^2) + 2\epsilon_t^2 \tag{39}$$

ii) *The conditional expectation of the Hilbert-norm difference of value functions at the next and current iteration satisfies the relationship*

$$\begin{aligned}\mathbb{E}\big[\|V_{t+1} - V^*\|_{\mathcal{H}}^2 \big| \mathcal{F}_t\big] \leq &\Big(1 + \frac{\alpha_t^2}{\beta_t} G_V^2\Big) \|V_t - V^*\|_{\mathcal{H}}^2 + 2\epsilon_t \|V_t - V^*\|_{\mathcal{H}} \\ &- 2\alpha_t [J(V_t) - J(V^*)] + \alpha_t^2 \sigma_V^2 \\ &+ \beta_t \mathbb{E}\left[(z_{t+1} - \bar{\delta}_t)^2 \,\middle|\, \mathcal{F}_t\right] \ . \end{aligned} \tag{40}$$

iii) *Define the expected value of the temporal difference given the state variable* $\mathbf{x}$ *and policy* $\pi$ *as* $\bar{\delta}_t = \mathbb{E}[\delta_t \,|\, \mathbf{x}_t, \pi(\mathbf{x}_t)]$*. Then the evolution of the auxiliary sequence* $z_t$ *with respect to* $\bar{\delta}_t$ *satisfies*

$$\begin{aligned}\mathbb{E}\big[(z_{t+1} - \bar{\delta}_t)^2 \big| \mathcal{F}_t\big] \leq &(1 - \beta_t)(z_t - \bar{\delta}_{t-1})^2 + \frac{L_V}{\beta_t} \|V_t - V_{t-1}\|_{\mathcal{H}}^2 \\ &+ 2\beta_t^2 \sigma_\delta^2 \end{aligned} \tag{41}$$

**Proof of Lemma 1**(i): Consider the Hilbert-norm difference of value functions at the next and current iteration, and use the definition of $V_{t+1}$ in (24), i.e.,

$$\begin{aligned}\|V_{t+1} - V_t\|_{\mathcal{H}}^2 &= \alpha_t^2 \|\tilde{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)\|_{\mathcal{H}}^2 \\ &\leq 2\alpha_t^2 \|\hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)\|_{\mathcal{H}}^2 \\ &\quad + 2\alpha_t^2 \|\hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t) \\ &\quad - \tilde{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)\|_{\mathcal{H}}^2 \ , \end{aligned} \tag{42}$$

where we add and subtract the functional stochastic quasi-gradient $\hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)$ on the first line of (42) and apply the triangle inequality $(a+b)^2 \leq 2a^2 + 2b^2$ which holds for any $a, b$. Now, we may apply Proposition 1 to the second term. Doing so and computing the expectation conditional on the filtration $\mathcal{F}_t$ yields

$$\begin{aligned}&\mathbb{E}[\|V_{t+1} - V_t\|_{\mathcal{H}}^2 \,|\, \mathcal{F}_t] \\ &\quad = 2\alpha_t^2 \mathbb{E}[\|\hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)\|_{\mathcal{H}}^2 \,|\, \mathcal{F}_t] + 2\epsilon_t^2 \ . \end{aligned} \tag{43}$$

Use the Cauchy-Schwartz inequality together with Law of Total Expectation and the definition of the functional stochastic quasi-gradient (22) to upper-estimate the right-hand side of (43) as

$$\begin{aligned}&\mathbb{E}[\|V_{t+1} - V_t\|_{\mathcal{H}}^2 \,|\, \mathcal{F}_t] \\ &\quad \leq 2\alpha_t^2 \mathbb{E}\Big\{\|\gamma\kappa(\mathbf{y}_t, \cdot) - \kappa(\mathbf{x}_t, \cdot))\|_{\mathcal{H}}^2 \mathbb{E}[z_{t+1}^2 \,|\, \mathbf{x}_t, \pi(\mathbf{x}_t)] \,|\, \mathcal{F}_t\Big\} \\ &\quad\quad + 2\alpha_t^2 \lambda \|V_t\|_{\mathcal{H}}^2 + 2\epsilon_t^2 \ , \end{aligned} \tag{44}$$

which together with Assumption 26 regarding fact that $z_{t+1}$ has a finite second conditional moment, yields

$$\begin{aligned}\mathbb{E}[\|V_{t+1} - V_t\|_{\mathcal{H}}^2 | \mathcal{F}_t] &\leq 2\alpha_t^2 G_\delta^2 \mathbb{E}\Big[\|\gamma\kappa(\mathbf{y}_t, \cdot) - \kappa(\mathbf{x}_t, \cdot))\|_{\mathcal{H}}^2 \Big| \mathcal{F}_t\Big] \\ &\quad + 2\alpha_t^2 \lambda \|V_t\| + 2\epsilon_t^2 \\ &\leq 2\alpha_t^2 (G_\delta^2 G_V^2 + \lambda^2 K^2) + 2\epsilon_t^2 \ , \end{aligned} \tag{45}$$

where we have also applied the fact that the functional gradient of the temporal difference $\gamma\kappa(\mathbf{y}_t, \cdot) - \kappa(\mathbf{x}_t, \cdot))$ has a finite second conditional moment and the bound on the function sequence [cf. (30)], allowing us to conclude (39). ■

**Proof of Lemma 1**(ii): This proof is a generalization of Lemma 3 in Appendix G.2 in the Supplementary Material of [33] to a function-valued stochastic quasi-gradient step combined with bias induced by the sparse subspace projections $\mathcal{P}_{\mathcal{H}_{\mathbf{D}_{t+1}}}[\cdot]$ in (15). Begin by considering the square-Hilbert norm sub-optimality of $V_{t+1}$, i.e.,

$$\begin{aligned}&\|V_{t+1} - V^*\|_{\mathcal{H}}^2 \\ &= \|V_t - \alpha_t \tilde{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t) - V^*\|_{\mathcal{H}}^2 \\ &= \|V_t - V^*\|_{\mathcal{H}}^2 - 2\alpha_t \langle \tilde{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t), V_t - V^* \rangle_{\mathcal{H}} \\ &\quad + \alpha_t^2 \|\tilde{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)\|_{\mathcal{H}}^2 \ , \end{aligned} \tag{46}$$

where we use the reformulation of the projected functional stochastic quasi-gradient step defined in (24) for the first equality, and expand the square in the second. Now, adding and subtracting $\hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)$ the (un-projected) functional stochastic quasi-gradient (22) yields

$$\begin{aligned}&\|V_{t+1} - V^*\|_{\mathcal{H}}^2 \\ &\quad = \|V_t - V^*\|_{\mathcal{H}}^2 - 2\alpha_t \langle \hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t), V_t - V^* \rangle_{\mathcal{H}} \\ &\quad\quad + 2\alpha_t \langle \hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t) \\ &\quad\quad - \tilde{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t), V_t - V^* \rangle_{\mathcal{H}} \\ &\quad\quad + \alpha_t^2 \|\tilde{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)\|_{\mathcal{H}}^2 \ . \end{aligned} \tag{47}$$

Apply the Cauchy-Schwartz inequality to the third term on the right-hand side of (47) together with the bound on the difference between unprojected and projected stochastic quasi-gradients in Proposition 1 to obtain

$$\begin{aligned}&\|V_{t+1} - V^*\|_{\mathcal{H}}^2 \\ &\; = \|V_t - V^*\|_{\mathcal{H}}^2 - 2\alpha_t \langle \hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t), V_t - V^* \rangle_{\mathcal{H}} \\ &\;\; + 2\epsilon_t \|V_t - V^*\|_{\mathcal{H}} + \alpha_t^2 \|\tilde{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)\|_{\mathcal{H}}^2 \ . \end{aligned} \tag{48}$$

Now, with $\bar{\delta}_t = \mathbb{E}[\delta_t \,|\, \mathbf{x}_t, \pi(\mathbf{x}_t)]$, add and subtract $\hat{\nabla}_V J(V_t, \bar{\delta}_t; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)$, the stochastic quasi-gradient evaluated at $(V_t, \bar{\delta}_t)$ rather than $(V_t, z_{t+1})$, inside the inner-product term on the right-hand side of (48), to write

$$\begin{aligned}&\|V_{t+1} - V^*\|_{\mathcal{H}}^2 \\ &\quad = \|V_t - V^*\|_{\mathcal{H}}^2 - 2\alpha_t \langle \hat{\nabla}_V J(V_t, \delta_t; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t), \\ &\quad\quad\quad\quad\quad\quad V_t - V^* \rangle_{\mathcal{H}} \\ &\quad\quad + 2\epsilon_t \|V_t - V^*\|_{\mathcal{H}} \\ &\quad\quad + 2\alpha_t \langle (\gamma\kappa(\mathbf{y}_t, \cdot) - \kappa(\mathbf{x}_t, \cdot))(\bar{\delta}_t - z_{t+1}), V_t - V^* \rangle_{\mathcal{H}} \\ &\quad\quad + \alpha_t^2 \|\tilde{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)\|_{\mathcal{H}}^2 \ , \end{aligned} \tag{49}$$

where we substitute in the definitions of $\hat{\nabla}_V J(V_t, \bar{\delta}_t; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)$ and $\hat{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)$ [cf. (10), (22), respectively] in (49), and cancel out the common regularization term $\lambda V_t$. We define the directional error associated with difference between the stochastic quasi-gradient and the stochastic gradient as

$$v_t = 2\alpha_t \langle (\gamma\kappa(\mathbf{y}_t, \cdot) - \kappa(\mathbf{x}_t, \cdot))(\bar{\delta}_t - z_{t+1}), V_t - V^* \rangle_{\mathcal{H}} \tag{50}$$

From here, compute the expectation conditional on the algorithm history $\mathcal{F}_t$:

$$\begin{aligned}&\mathbb{E}\left[\|V_{t+1} - V^*\|_{\mathcal{H}}^2 \,|\, \mathcal{F}_t\right] \\ &= \|V_t - V^*\|_{\mathcal{H}}^2 - 2\alpha_t \left\langle \mathbb{E}\left[\hat{\nabla}_V J(V_t, \bar{\delta}_t; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t) \,\middle|\, \mathcal{F}_t\right], V_t - V^* \right\rangle_{\mathcal{H}} \\ &\quad + 2\epsilon_t \|V_t - V^*\|_{\mathcal{H}} + \mathbb{E}\left[v_t \,|\, \mathcal{F}_t\right] \\ &\quad + \alpha_t^2 \mathbb{E}\left[\|\tilde{\nabla}_V J(V_t, z_{t+1}; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t)\|_{\mathcal{H}}^2 \,\middle|\, \mathcal{F}_t\right] \ . \end{aligned} \tag{51}$$

Note that the compositional objective $J(V)$ is convex with respect to $V$, which allows us to write

$$\left\langle \mathbb{E}\left[\hat{\nabla}_V J(V_t, \bar{\delta}_t; \mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t) \middle| \mathcal{F}_t\right], V_t - V^* \right\rangle_{\mathcal{H}} \geq J(V_t) - J(V^*). \tag{52}$$

Now, we may use Assumption 2 [cf. (28)] regarding the finite conditional moments of the projected stochastic quasi-gradient to the last term in (51) so that it may be replaced by its upper-estimate, which together with (52) simplifies to

$$
\begin{aligned}
&\mathbb{E}\left[\|V_{t+1}-V^*\|_{\mathcal{H}}^2 \,\middle|\, \mathcal{F}_t\right] \\
&\quad= \|V_t - V^*\|_{\mathcal{H}}^2 - 2\alpha_t\left[J(V_t)-J(V^*)\right] \\
&\qquad + 2\epsilon_t\|V_t-V^*\|_{\mathcal{H}} + \alpha_t^2\sigma_V^2 + \mathbb{E}\left[v_t \,\middle|\, \mathcal{F}_t\right] \ . \qquad (53)
\end{aligned}
$$

It remains to analyze $v_t$, the directional error associated with using stochastic quasi-gradients rather than stochastic gradients. In doing so, we derive the fact that the sub-optimality $\|V_t - V^*\|$ is intrinsically coupled to the auxiliary sequence $(z_{t+1}-\bar{\delta}_t)$, which is the focus of Lemma 1(iii). Proceed by applying the Cauchy-Schwartz inequality to (50), which allows us to write

$$
v_t \le 2\alpha_t\|\gamma\kappa(\mathbf{y}_t,\cdot)-\kappa(\mathbf{x}_t,\cdot)\|_{\mathcal{H}}^2|z_{t+1}-\bar{\delta}_t|\|V_t-V^*\|_{\mathcal{H}} \qquad (54)
$$

Note that $2ab \le \rho a^2 + b^2/\rho$ for $\rho, a, b > 0$, which we apply to (54) with $a = |z_{t+1}-\bar{\delta}_t|$, $b = \alpha_t\|\gamma\kappa(\mathbf{y}_t,\cdot)-\kappa(\mathbf{x}_t,\cdot)\|_{\mathcal{H}}\|V_t - V^*\|_{\mathcal{H}}$, and $\rho = \beta_t$ so that (54) becomes

$$
v_t \le \beta_t(z_{t+1}-\bar{\delta}_t)^2 + \frac{\alpha_t^2}{\beta_t}\|\gamma\kappa(\mathbf{y}_t,\cdot)-\kappa(\mathbf{x}_t,\cdot)\|_{\mathcal{H}}^2\|V_t-V^*\|_{\mathcal{H}}^2 \ . \qquad (55)
$$

The conditional mean of $v_t$ [cf. (50)], using (55), is then

$$
\begin{aligned}
\mathbb{E}\left[v_t \,\middle|\, \mathcal{F}_t\right] &\le \beta_t\mathbb{E}\left[(z_{t+1}-\bar{\delta}_t)^2 \,\middle|\, \mathcal{F}_t\right] \qquad (56) \\
&\quad + \frac{\alpha_t^2}{\beta_t}\mathbb{E}\left[\|\gamma\kappa(\mathbf{y}_t,\cdot)-\kappa(\mathbf{x}_t,\cdot)\|_{\mathcal{H}}^2\middle|\mathcal{F}_t\right]\|V_t-V^*\|_{\mathcal{H}}^2 \\
&\le \beta_t\mathbb{E}\left[(z_{t+1}-\bar{\delta}_t)^2\middle|\mathcal{F}_t\right] + \frac{\alpha_t^2}{\beta_t}G_V^2\|V_t-V^*\|_{\mathcal{H}}^2 \ ,
\end{aligned}
$$

where we apply the finite variance property of the functional component of the stochastic gradient [cf. (27)] for the final inequality (56). Now, substitute (56) into the right-hand side of (53) and gather like terms:

$$
\begin{aligned}
&\mathbb{E}\left[\|V_{t+1}-V^*\|_{\mathcal{H}}^2 \,\middle|\, \mathcal{F}_t\right] \qquad (57) \\
&\quad\le \left(1+\frac{\alpha_t^2}{\beta_t}G_V^2\right)\|V_t-V^*\|_{\mathcal{H}}^2 + 2\epsilon_t\|V_t-V^*\|_{\mathcal{H}} \\
&\qquad -2\alpha_t[J(V_t)-J(V^*)] + \alpha_t^2\sigma_V^2 + \beta_t\mathbb{E}\left[(z_{t+1}-\bar{\delta}_t)^2\middle|\mathcal{F}_t\right] ,
\end{aligned}
$$

which is as stated in Lemma 1(ii). ■

**Proof of Lemma 1**(iii): This proof is an adaptation of Lemma 2 in Appendix G.1 in the Supplementary Material of [33] to the recursively averaged temporal difference sequence $z_t$ defined in (11). Begin by defining the scalar quantity $e_t$ as the difference of mean temporal differences scaled by the forgetting factor $\beta_t$, i.e. $e_t = (1-\beta_t)(\bar{\delta}_t - \bar{\delta}_{t-1})$. Then we consider the difference of the evolution of the auxiliary variable $z_{t+1}$ with respect to the conditional mean temporal difference $\bar{\delta}_t$, plus the difference of mean temporal differences:

$$
\begin{aligned}
&z_{t+1} - \bar{\delta}_t + e_t \\
&\quad= (1-\beta_t)z_t + \beta_t\delta_t - [(1-\beta_t)\bar{\delta}_t + \beta_t\bar{\delta}_t] + (1-\beta_t)(\bar{\delta}_t - \bar{\delta}_{t-1}) \\
&\quad= (1-\beta_t)\left(z_t - \bar{\delta}_{t-1}\right) + \beta_t(\delta_t - \bar{\delta}_t) \qquad (58)
\end{aligned}
$$

where we make use of the definition of $z_{t+1}$ in (11), the fact that $\bar{\delta}_t = [(1-\beta_t)\bar{\delta}_t + \beta_t\bar{\delta}_t]$, and the definition of $e_t$ on the first line of (58), and in the second we gather terms with respect to coefficients $(1-\beta_t)$ and $\beta_t$, and cancel the redundant $\bar{\delta}_t$ term. Now, consider the square of the expression (58), using it's simplification on the right-hand side of the preceding expression

$$
\begin{aligned}
&(z_{t+1} - \bar{\delta}_t + e_t)^2 \\
&\qquad= \left[(1-\beta_t)\left(z_t - \bar{\delta}_{t-1}\right) + \beta_t(\delta_t - \bar{\delta}_t)\right]^2 \\
&\qquad= (1-\beta_t)^2\left(z_t - \bar{\delta}_{t-1}\right)^2 + \beta_t^2(\delta_t - \bar{\delta}_t)^2 \\
&\qquad\quad + 2(1-\beta_t)\beta_t\left(z_t - \bar{\delta}_{t-1}\right)(\delta_t - \bar{\delta}_t). \qquad (59)
\end{aligned}
$$

where we expand the square to obtain the second line in the previous expression. Now, compute the expectation of (59) conditional on the filtration $\mathcal{F}_t$, which yields

$$
\begin{aligned}
&\mathbb{E}[(z_{t+1} - \bar{\delta}_t + e_t)^2 \,|\, \mathcal{F}_t] \\
&\quad= (1-\beta_t)^2\left(z_t - \bar{\delta}_{t-1}\right)^2 + \beta_t^2\mathbb{E}[(\delta_t - \bar{\delta}_t)^2 \,|\, \mathcal{F}_t] \\
&\qquad + 2(1-\beta_t)\beta_t\left(z_t - \bar{\delta}_{t-1}\right)\mathbb{E}[(\delta_t - \bar{\delta}_t) \,|\, \mathcal{F}_t] \ . \qquad (60)
\end{aligned}
$$

Now we apply the assumption [cf. (26)] that the fact that the temporal difference $\delta_t$ is an unbiased estimator for its conditional mean $\bar{\delta}_t$ (so that the last term in the previous expression is null), with finite variance $\mathbb{E}[(\delta_t - \bar{\delta}_t)^2 \,|\, \mathcal{F}_t] \le \sigma_\delta^2$ (Assumption 2), to write

$$
\mathbb{E}[(z_{t+1}-\bar{\delta}_t + e_t)^2|\mathcal{F}_t] = (1-\beta_t)^2\left(z_t - \bar{\delta}_{t-1}\right)^2 + \beta_t^2\sigma_\delta^2. \qquad (61)
$$

We may use the relationship in (61) to obtain an upper estimate on the conditional mean square of $z_{t+1} - \bar{\delta}_t$ by using the inequality $\|a+b\|^2 \le (1+\rho)\|a\|^2 + (1+1/\rho)\|b\|^2$ which holds for any $\rho > 0$: set $a = z_{t+1} - \bar{\delta}_t + e_t$, $b = -e_t$, and $\rho = \beta_t$. Therefore, we obtain

$$
(z_{t+1}-\bar{\delta}_t)^2 \le (1+\beta_t)(z_{t+1}-\bar{\delta}_t+e_t)^2 + \left(1+\frac{1}{\beta_t}\right)e_t^2 \ . \qquad (62)
$$

Now, we may use the conditionally expected value of (62) in lieu of (61), while gaining a multiplicative factor of $(1+\beta_t)$ on the right-hand side of (61) plus the error term $(1+1/\beta_t)e_t$, yielding

$$
\begin{aligned}
&\mathbb{E}[(z_{t+1} - \bar{\delta}_t)^2 \,|\, \mathcal{F}_t] \qquad (63) \\
&\quad= (1+\beta_t)\Big[(1-\beta_t)^2\left(z_t - \bar{\delta}_{t-1}\right)^2 + \beta_t^2\sigma_\delta^2\Big] + \left(\frac{1+\beta_t}{\beta_t}\right)e_t^2.
\end{aligned}
$$

Use the fact that $(1-\beta_t^2)(1-\beta_t) \le (1-\beta_t)$ to the first term in (63) and $(1+\beta_t)\beta_t^2 \le 2\beta_t^2$ to the second (since $\beta_t \in (0,1)$) to simplify (63) as

$$
\begin{aligned}
&\mathbb{E}[(z_{t+1} - \bar{\delta}_t)^2 \,|\, \mathcal{F}_t] \qquad (64) \\
&\qquad= (1-\beta_t)\left(z_t - \bar{\delta}_{t-1}\right)^2 + 2\beta_t^2\sigma_\delta^2 + \left(\frac{1+\beta_t}{\beta_t}\right)e_t^2.
\end{aligned}
$$

From here, we turn to controlling the term involving $e_t$, which represents the difference of mean temporal differences. By definition, we have

$$
|e_t| = (1-\beta_t)|(\bar{\delta}_t - \bar{\delta}_{t-1})| \le (1-\beta_t)L_V\|V_t - V_{t-1}\|_{\mathcal{H}} \qquad (65)
$$

where we apply the Lipschitz continuity of the conditional average temporal difference $\bar{\delta}_t = \mathbb{E}_{\mathbf{y}_t}[r(\mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t) + \gamma V(\mathbf{y}_t) - V(\mathbf{x}_t) \,|\, \mathbf{x}_t, \pi(\mathbf{x}_t)]$ with respect to the value function [cf. (29)] stated in Assumption 3. Substitute the right-hand side of (65) into (64), and simplify the expression in the last term as $(1-\beta_t^2)/\beta_t \leq 1/\beta_t$ to conclude (41). ■

**Lemma 2.** *(Coupled Supermartingale Theorem [46][Lemma 6]) Let $\{\xi_k\}$, $\{\zeta_k\}$, $\{u_k\}$, $\{\bar{u}_k\}$, $\{\eta_k\}$, $\{\theta_k\}$, $\{\varepsilon_k\}$, $\{\mu_k\}$, $\{\nu_k\}$ be sequences of nonnegative random variables such that*

$$\mathbb{E}[\xi_{k+1} \,|\, \mathcal{G}_k] \leq (1+\eta_k)\xi_k - u_k + c\theta_k\zeta_k + \mu_k \ , \tag{66}$$

$$\mathbb{E}[\zeta_{k+1} \,|\, \mathcal{G}_k] \leq (1-\theta_k)\zeta_k - \bar{u}_k + \varepsilon_k\xi_k + \nu_k \ , \tag{67}$$

*where $\mathcal{G}_k = \{\xi_s, \zeta_s, u_s, \bar{u}_s, \eta_s, \theta_s, \varepsilon_s, \mu_s, \nu_s\}_{s=0}^k$ is the filtration, and $c > 0$ is a scalar. Suppose the following summability conditions hold:*

$$\sum_{k=0}^{\infty} \eta_k < \infty \ , \quad \sum_{k=0}^{\infty} \varepsilon_k < \infty \ ,$$
$$\sum_{k=0}^{\infty} \mu_k < \infty \ , \ \sum_{k=0}^{\infty} \nu_k < \infty \ , \tag{68}$$

*almost surely. Then $\xi_k$ and $\zeta_k$ converge almost surely to two respective nonnegative random variables, and we may conclude that*

$$\sum_{k=0}^{\infty} u_k < \infty \ , \quad \sum_{k=0}^{\infty} \bar{u}_k < \infty \ , \ \sum_{k=0}^{\infty} \theta_k\zeta_k < \infty \ , \tag{69}$$

*almost surely.*

We can use Lemma 2 to establish convergence with probability 1 of Algorithm 1 by considering the expressions derived in Lemma 1.

### *B. Proof of Theorem 1*

We use the relations established in Lemma 1 to construct a coupled supermartingale of the form in Lemma 2 as follows. First, consider the expression (40) for the value function sub-optimality, using approximation budget $\epsilon_t = \alpha_t^2$ and the fact that the value function is bounded in Hilbert norm [cf. (30)] to obtain $\|V_t - V^*\|_{\mathcal{H}} \leq 2K$ :

$$\begin{aligned}
&\mathbb{E}\left[\|V_{t+1} - V^*\|_{\mathcal{H}}^2 \,\middle|\, \mathcal{F}_t\right] \\
&\quad \leq \left(1+\frac{\alpha_t^2}{\beta_t}G_V^2\right)\|V_t - V^*\|_{\mathcal{H}}^2 - 2\alpha_t\left[J(V_t) - J(V^*)\right] \\
&\qquad + \alpha_t^2(\sigma_V^2 + 4K) + \beta_t\mathbb{E}\left[(z_{t+1} - \bar{\delta}_t)^2 \,\middle|\, \mathcal{F}_t\right] \ .
\end{aligned} \tag{70}$$

and then substitute (41) regarding the evolution of $z_t$ with respect to its conditional expectation into (70) to obtain :

$$\begin{aligned}
&\mathbb{E}\left[\|V_{t+1} - V^*\|_{\mathcal{H}}^2 \,\middle|\, \mathcal{F}_t\right] \\
&\quad \leq \left(1+\frac{\alpha_t^2}{\beta_t}G_V^2\right)\|V_t - V^*\|_{\mathcal{H}}^2 - 2\alpha_t\left[J(V_t) - J(V^*)\right] \\
&\qquad + \alpha_t^2(\sigma_V^2 + 4K) + \beta_t(1-\beta_t)(z_t - \bar{\delta}_{t-1})^2 \\
&\qquad + L_V\|V_t - V_{t-1}\|_{\mathcal{H}}^2 + 2\beta_t^3\sigma_\delta^2 \ .
\end{aligned} \tag{71}$$

Assume that $\beta_t \in (0, 1)$ for all $t$, so that the right-hand side of (71) may be simplified to

$$\begin{aligned}
&\mathbb{E}\left[\|V_{t+1} - V^*\|_{\mathcal{H}}^2 \,\middle|\, \mathcal{F}_t\right] \\
&\quad \leq \left(1+\frac{\alpha_t^2}{\beta_t}G_V^2\right)\|V_t - V^*\|_{\mathcal{H}}^2 - 2\alpha_t\left[J(V_t) - J(V^*)\right] \\
&\qquad + \beta_t(z_t - \bar{\delta}_{t-1})^2 + \alpha_t^2(\sigma_V^2 + 4K) \\
&\qquad + L_V\|V_t - V_{t-1}\|_{\mathcal{H}}^2 + 2\beta_t^2\sigma_\delta^2 \ .
\end{aligned} \tag{72}$$

We may identify (72) with the first supermartingale relationship in Lemma 2 [cf. (66)] via the identifications

$$\begin{aligned}
&\xi_t = \|V_t - V^*\|_{\mathcal{H}}^2 \ , \eta_t = \frac{\alpha_t^2}{\beta_t}G_V^2 \ , u_t = 2\alpha_t[J(V_t) - J(V^*)] \ , \\
&c = 1 \ , \qquad \zeta_t = (z_t - \bar{\delta}_{t-1})^2 \ , \quad \theta_t = \beta_t \ , \\
&\mu_t = \alpha_t^2(\sigma_V^2 + 4K) + L_V\|V_t - V_{t-1}\|_{\mathcal{H}}^2 + 2\beta_t^2\sigma_\delta^2 \ ,
\end{aligned} \tag{73}$$

where $u_t \geq 0$ by the definition of the optimal objective $J(V^*)$. The summability of $\mu_t$ may be established as follows: consider summing the expression in Lemma 1(i) for all $t$, which by the fact that $\sum_t \alpha_t^2 < \infty$ [cf. (31)], implies that the conditional mean series is finite. Consequently, $\sum_{t=0}^{\infty}\|V_t - V_{t-1}\|_{\mathcal{H}}^2 < \infty$ with probability 1 using the fact that $\|V_t - V_{t-1}\|_{\mathcal{H}}$ is bounded. Thus $\sum_t \mu_t < \infty$.

Now, let's connect the evolution of the auxiliary temporal difference sequence $z_t$ (11) in Lemma 1(iii). In particular, (41) is related to (67) via the identifications:

$$\bar{u}_t = 0 \ , \varepsilon_t = 0 \ , \nu_t = \frac{L_V}{\beta_t}\|V_t - V_{t-1}\|_{\mathcal{H}}^2 + 2\beta_t^2\sigma_\delta^2 \ , \tag{74}$$

with $\zeta_t = (z_t - \bar{\delta}_{t-1})^2$ and $\theta_t = \beta_t$ as in (73). The summability of $\nu_t$ follows the following logic: consider the expression $\|V_t - V_{t-1}\|_{\mathcal{H}}^2/\beta_t$ which is of order $\mathcal{O}(\alpha_t^2/\beta_t)$ in conditional expectation by Lemma 1(i). Sum the resulting conditional expectation for all $t$, which by the summability of the sequence $\sum_t \alpha_t^2/\beta_t < \infty$ is finite. Therefore, $\sum_t \|V_t - V_{t-1}\|_{\mathcal{H}}^2/\beta_t < \infty$ almost surely.

Together with the conditions on the step-size sequences $\alpha_t$ and $\beta_t$ (31), the summability conditions (68) of Lemma 2, the Coupled Supermartingale Theorem, are satisfied, which allows us to conclude that $\xi_t = \|V_t - V^*\|_{\mathcal{H}}^2$ and $\zeta_t = (z_t - \bar{\delta}_{t-1})^2$ converge to two nonnegative random variables with probability 1, and that:

$$\sum_t \alpha_t[J(V_t) - J(V^*)] < \infty \ , \quad \sum_t \beta_t(z_{t+1} - \bar{\delta}_t)^2 < \infty \ , \tag{75}$$

almost surely. The non-summability of the step-size sequences $\alpha_t$ and $\beta_t$ (31) allows us to conclude that:

$$\liminf_{t\to\infty} J(V_t) = J(V^*) \ , \quad \liminf_{t\to\infty} (z_{t+1} - \bar{\delta}_t)^2 = 0 \ , \tag{76}$$

almost surely. Then, the convergence of the whole sequence $\|V_t - V^*\|_{\mathcal{H}}^2$ implies that this sequence is bounded with probability 1. Then, since $J(V_t) \to J(V^*)$ almost surely along a subsequence, $V_t \to V^*$ almost sure along a subsequence using the continuity of $J(V)$. However, since the whole sequence $\|V_t - V^*\|_{\mathcal{H}}^2$ converges to a unique limit, the whole sequence $\{V_t\}$ converges to $V^*$ with probability 1.

### C. *Proof of Theorem 2*

Before analyzing the mean convergence behavior of the value function, we consider the mean sub-optimality of the auxiliary variable $z_t$ with respect to the conditional mean of the temporal difference $\bar{\delta}_t$. To do so, compute the total expectation of Lemma 1(iii), stated as

$$\begin{aligned}&\mathbb{E}\left[(z_{t+1}-\bar{\delta}_t)^2\right] \qquad (77)\\ &\leq(1-\beta)\mathbb{E}\left[(z_t-\bar{\delta}_{t-1})^2\right]+\frac{L_V}{\beta}\mathbb{E}\left[\|V_t-V_{t-1}\|_{\mathcal{H}}^2\right]+2\beta^2\sigma_\delta^2\ ,\end{aligned}$$

where we have substituted in constant learning rate $\beta_t = \beta$ in (77). The total expectation of Lemma 1(i) regarding $\|V_t - V_{t-1}\|_{\mathcal{H}}^2$, the difference of value functions in Hilbert-norm, may be substituted into (77), with constant step-size $\alpha_t = \alpha$ and compression budgets $\epsilon_t = \epsilon$ to obtain

$$\begin{aligned}&\mathbb{E}\left[(z_{t+1}-\bar{\delta}_t)^2\right]\\ &\quad\leq(1-\beta)\mathbb{E}\left[(z_t-\bar{\delta}_{t-1})^2\right]\\ &\qquad+\frac{2L_V}{\beta}\left[\alpha^2(G_\delta^2G_V^2+\lambda^2K^2)+\epsilon^2\right]+2\beta^2\sigma_\delta^2\ , \qquad (78)\end{aligned}$$

Observe that (80) gives a relationship between the sequence $\mathbb{E}\left[(z_{t+1}-\bar{\delta}_t)^2\right]$ and its value at the previous iterate. We can substitute $t+1$ by $t$ in (80) to write

$$\begin{aligned}&\mathbb{E}\left[(z_t-\bar{\delta}_{t-1})^2\right]\\ &\quad\leq(1-\beta)\mathbb{E}\left[(z_{t-1}-\bar{\delta}_{t-2})^2\right]\\ &\qquad+\frac{2L_V}{\beta}\left[\alpha^2(G_\delta^2G_V^2+\lambda^2K^2)+\epsilon^2\right]+2\beta^2\sigma_\delta^2\ , \qquad (79)\end{aligned}$$

Substituting (79) into the right-hand side of (80) yields

$$\begin{aligned}&\mathbb{E}\left[(z_{t+1}-\bar{\delta}_t)^2\right] \qquad (80)\\ &\quad\leq(1-\beta)^2\mathbb{E}\left[(z_{t-1}-\bar{\delta}_{t-2})^2\right]\\ &\quad+[1+(1-\beta)]\left\{\frac{2L_V}{\beta}\left[\alpha^2(G_\delta^2G_V^2+\lambda^2K^2)+\epsilon^2\right]+2\beta^2\sigma_\delta^2\right\}.\end{aligned}$$

We can recursively apply the previous two steps backwards in time to the initialization to obtain

$$\begin{aligned}&\mathbb{E}\left[(z_{t+1}-\bar{\delta}_t)^2\right] \qquad (81)\\ &\quad\leq(1-\beta)^{t+1}(z_0-\bar{\delta}_{-1})^2\\ &\quad+\sum_{u=0}^{t}(1-\beta)^u\left\{\frac{2L_V}{\beta}\left[\alpha^2(G_\delta^2G_V^2+\lambda^2K^2)+\epsilon^2\right]+2\beta^2\sigma_\delta^2\right\}\ ,\end{aligned}$$

In (81), the first term on the left-hand side vanishes due to the initialization $z_0 = 0$ and the convention $\delta_{-1} = 0$. Moreover, the finite geometric sum may be evaluated, provided $\beta < 1$, as $\sum_{u=0}^{t}(1-\beta)^u = [1-(1-\beta)^t]/\beta$. The numerator in this simplification is strictly less than unit, which means that the right-hand side of (81) simplifies to

$$\begin{aligned}\mathbb{E}\left[(z_{t+1}-\bar{\delta}_t)^2\right]&\leq\frac{2L_V}{\beta^2}\left[\alpha^2(G_\delta^2G_V^2+\lambda^2K^2)+\epsilon^2\right]+2\beta\sigma_\delta^2\\ &=\mathcal{O}\left(\frac{\alpha^2+\epsilon^2}{\beta^2}+\beta\right) \qquad (82)\end{aligned}$$

With this relationship established for the auxiliary sequence $z_t$, we shift gears to addressing the evolution of the value function sub-optimality $\|V_t - V^*\|_{\mathcal{H}}$ in expectation. Begin by using the fact that the Hilbert-norm regularizer $(\lambda/2)\|V\|_{\mathcal{H}}^2$ in (8) implies the objective $J(V)$ is strongly convex, i.e.

$$\frac{\lambda}{2}\|V_t-V^*\|_{\mathcal{H}}^2\leq J(V_t)-V(V^*)\ , \qquad (83)$$

together with the expression in Lemma 1(ii) regarding the evolution of the value function sub-optimality, assuming constant learning rates and compression budget, i.e. $\alpha_t = \alpha, \beta_t = \beta, \epsilon_t = \epsilon$, to write

$$\begin{aligned}&\mathbb{E}\left[\|V_{t+1}-V^*\|_{\mathcal{H}}^2\,\middle|\,\mathcal{F}_t\right]\\ &\quad\leq\left(1+\frac{\alpha^2}{\beta}G_V^2-\alpha\lambda\right)\|V_t-V^*\|_{\mathcal{H}}^2+2\epsilon\|V_t-V^*\|_{\mathcal{H}}\\ &\qquad+\alpha^2\sigma_V^2+\beta\mathbb{E}\left[(z_{t+1}-\bar{\delta}_t)^2\,\middle|\,\mathcal{F}_t\right]\ . \qquad (84)\end{aligned}$$

Consider the total expectation of (84), using choice of compression budget $\epsilon = C\alpha^2$ for some arbitrary constant $C > 0$, the fact that $\|V_t - V^*\|_{\mathcal{H}} \leq 2K$, and applying (82) to the last term on the right-hand side of the preceding expression to obtain:

$$\begin{aligned}&\mathbb{E}\left[\|V_{t+1}-V^*\|_{\mathcal{H}}^2\right]\\ &\quad\leq\left(1+\frac{\alpha^2}{\beta}G_V^2-\alpha\lambda\right)\mathbb{E}\left[\|V_t-V^*\|_{\mathcal{H}}^2\right]\\ &\qquad+\alpha^2(\sigma_V^2+4CK)+2\beta^2\sigma_\delta^2\\ &\qquad+\frac{2L_V}{\beta}\left[\alpha^2(G_\delta^2G_V^2+\lambda^2K^2)+C^2\alpha^4\right]\ . \qquad (85)\end{aligned}$$

From (85), substitute in the regularizer selection $\lambda = G_V^2\alpha/\beta + \lambda_0$ for $\lambda_0 < 1$. We may establish asymptotic convergence to a neighborhood by analyzing the conditions for which we have a decreasing sequence, i.e., the following holds

$$\begin{aligned}&\mathbb{E}\left[\|V_{t+1}-V^*\|_{\mathcal{H}}^2\right]\\ &\quad\leq(1-\lambda_0)\,\mathbb{E}\left[\|V_t-V^*\|_{\mathcal{H}}^2\right]+\alpha^2(\sigma_V^2+4CK)\\ &\quad+2\beta^2\sigma_\delta^2+\frac{2L_V}{\beta}\left[\alpha^2(G_\delta^2G_V^2+(G_V^2\alpha/\beta+\lambda_0)^2K^2)+C^2\alpha^4\right]\\ &\quad\leq\mathbb{E}\left[\|V_t-V^*\|_{\mathcal{H}}^2\right] \qquad (86)\end{aligned}$$

Partition the set of time indices $\{t \geq 0\}$ into two disjoint sets $\{t_k\}$ and $\{t_j\}$, and suppose that (86) holds along subsequence $\mathbb{E}\left[\|V_{t_k} - V^*\|_{\mathcal{H}}^2\right]$ associated with time indices $\{t_k\}$. We may simplify the condition in (86) for this subsequence to

$$\begin{aligned}&\lambda_0^{-1}\Big(\alpha^2(\sigma_V^2+4K)+2\beta^2\sigma_\delta^2\\ &\qquad+\frac{2L_V}{\beta}\left[\alpha^2(G_\delta^2G_V^2+(G_V^2\alpha/\beta+\lambda_0)^2K^2)+C^2\alpha^4\right]\Big)\\ &\quad=\mathcal{O}\left(\alpha^2+\beta^2+\frac{\alpha^2}{\beta}\left[1+\alpha^2+\frac{\alpha}{\beta}+\frac{\alpha^2}{\beta^2}\right]\right)\\ &\quad\leq\mathbb{E}\left[\|V_{t_k}-V^*\|_{\mathcal{H}}^2\right]\ . \qquad (87)\end{aligned}$$

For this subsequence, since (86) holds, $\mathbb{E}\left[\|V_{t_k} - V^*\|_{\mathcal{H}}^2\right]$ is decreasing, and since it is bounded, it thus converges to its infimum by the Monotone Convergence Theorem. The infimum of $\mathbb{E}\left[\|V_{t_k} - V^*\|^2_{\mathcal{H}}\right]$ is the left-hand side of (87), so that we may write

$$\lim_{t\to\infty} \mathbb{E}\left[\|V_{t_k} - V^*\|^2_{\mathcal{H}}\right] = \mathcal{O}\left(\alpha^2 + \beta^2 + \frac{\alpha^2}{\beta}\left[1 + \alpha^2 + \frac{\alpha}{\beta} + \frac{\alpha^2}{\beta^2}\right]\right) \tag{88}$$

For all elements of the sequence $\mathbb{E}\left[\|V_t - V^*\|^2_{\mathcal{H}}\right]$ not part of the subsequence of indices $\{t_k\}$, i.e., those associated with $\{t_j\}$, the condition in (87) fails to hold:

$$\mathbb{E}\left[\|V_{t_j} - V^*\|^2_{\mathcal{H}}\right] < \mathcal{O}\left(\alpha^2 + \beta^2 + \frac{\alpha^2}{\beta}\left[1 + \alpha^2 + \frac{\alpha}{\beta} + \frac{\alpha^2}{\beta^2}\right]\right) . \tag{89}$$

The statements in (88) and (89) taken together imply

$$\limsup_{t\to\infty} \mathbb{E}\left[\|V_t - V^*\|^2_{\mathcal{H}}\right] = \mathcal{O}\left(\alpha^2 + \beta^2 + \frac{\alpha^2}{\beta}\left[1 + \alpha^2 + \frac{\alpha}{\beta} + \frac{\alpha^2}{\beta^2}\right]\right) . \tag{90}$$

When $\alpha = \beta$, the posynomial of the learning rates on the right-hand side of (90) simplifies to be $\mathcal{O}(\alpha + \alpha^2 + \alpha^3)$, which is as stated in (33) (Theorem 2).

### *D. Proof of Corollary 1*

We now prove Corollary 1: In Theorem 3 of [32][Appendix D.1], it is established for a nonparametric stochastic program without any compositional structure that the effect of sparse subspace projections on the functional stochastic gradient sequence in an RKHS is to yield a function sequence of finite model order, provided a constant algorithm step-size and compression budget are used. The proof of Corollary 1 is nearly identical: the same projection operator is used and the same compactness properties of the state and action spaces apply. The only point of departure is that a distinct deterministic bound is needed on the functional stochastic quasi-gradient for all $\{\mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t\}$, i.e., to apply the reasoning following equations (74) in [32][Appendix D.1], we require the existence of a deterministic constant $D$ such that $|[\gamma\kappa(\mathbf{y}_t, \cdot) - \kappa(\mathbf{x}_t, \cdot)]z_{t+1}| \leq D$ for all $\{\mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t\}$. We turn to establishing such an upper-estimate. To do so, we first establish that the auxiliary sequence $z_t$ stated in (11) is bounded, i.e.

**Proposition 2.** *The auxiliary sequence $z_t$ [cf. (11)] satisfies the following upper bound when constant step-size $\beta_t = \beta$ is used :*

$$|z_t| = (\gamma + 1)K + R_{\max} \textit{ for all } t \tag{91}$$

**Proof:** We pursue a proof by induction. First, the base case: with $V_0 = 0$, we have $|z_1| \leq \beta R_{\max} \leq (\gamma + 1)K + R_{\max}$ making use of the bound on $V_t$ for all $t$ in (30) and the fact that the step-size is less than unit. Now we consider the induction step: assume the prior bound holds for $z_u$ for $u \leq t$. Write for $z_{t+1}$

$$|z_{t+1}| = (1 - \beta)|z_t| + \beta|\delta_t| \leq (\gamma + 1)K + R_{\max} \tag{92}$$

where in the last inequality we apply the induction hypothesis together with the upper-estimate on the temporal difference $\delta_t \leq (\gamma + 1)K + R_{\max}$. ■

By making use of Proposition 2 together with the bound on the reproducing kernel map (Assumption 1), we have the following uniform deterministic bound:

$$|[\gamma\kappa(\mathbf{y}_t, \cdot) - \kappa(\mathbf{x}_t, \cdot)]z_{t+1}| \leq X(\gamma + 1)[(\gamma + 1)K + R_{\max}] := D \text{ for all } \{\mathbf{x}_t, \pi(\mathbf{x}_t), \mathbf{y}_t\} \tag{93}$$

Then, we may apply the same reasoning as that of Appendix D.1 of [32] which allows us to conclude that the number of Euclidean balls of radius $d = \epsilon/D$ needed to cover the space $\phi(\mathcal{X}) = \kappa(\mathcal{X}, \cdot)$ is finite, where $\epsilon$ is a constant as in (32). See [49], [50] for further details. Therefore, for Algorithm 1, there exists a finite $M^\infty < \infty$ such that the model order $M_t \leq M^\infty$ for all $t$.