Online appendix for
“Toward breaking the curse of dimensionality: an FPTAS for stochastic dynamic programs with multidimensional actions and scalar states”

Nir Halman ∗ Giacomo Nannicini †
August 22, 2018

1 Additional routines

We report here the pseudocode and running time of the routines taken from [HKL+14, HN16] referenced in the main text of the paper. Given a monotone nondecreasing function ϕ, we define a routine FuncSearchINC(ϕ, D, ℓ, u) that looks for a point x ∈ D such that ℓ ≤ ϕ(x) ≤ u. Implementing this function is straightforward for both discrete domains and real intervals, using binary search.

In Alg. 1 we formally define the routine CompressINC that constructs an oracle for a K-approximation function of a monotone nondecreasing ϕ in the form of a canonical representation.

The main routine used in Algorithm 1 is ApxSetINC. We give its pseudocode in Algorithm 2 for continuous functions that are bounded away from zero; its counterpart for functions over discrete domains is straightforward. It is shown in [HN16] that ApxSetINC determines a K-approximation set with O \( \left( \frac{1}{\epsilon} \log \frac{c_{\text{max}}}{c_{\text{min}}} \right) \) points in O \( \left( 1 + t_{\phi} \left( \frac{1}{\epsilon} \log \frac{c_{\text{max}}}{c_{\text{min}}} \right) \log ((B - A)\kappa) \right) \) time. For a function defined over a discrete domain D, the running time of ApxSetINC becomes O \( \left( (1 + t_{\phi}) \left( \frac{1}{\epsilon} \log \frac{c_{\text{max}}}{c_{\text{min}}} \right) \log |D| \right) \), see [HKL+14].

Algorithm 1: Function CompressINC(ϕ, [A, B], K).

1: W ← ApxSetINC(ϕ, [A, B], K)
2: return \{ (x, ϕ(x)) | x ∈ W \} as an array of tuples sorted by their first coordinate

Algorithm 2: Function ApxSetINC(ϕ, [A, B], K).

1: x ← A, W ← {A, B}
2: while x < B do
3: x ← FuncSearchINC(ϕ, [x, B], \( K + \frac{1}{2} \phi(x), K\phi(x) \))
4: W ← W ∪ \{x\}
5: return W
2 Proof of Prop. 5.2

Proof. For \( i = 0, \ldots, n \), define:

\[
\psi_i(y) := \begin{cases} 
(\Delta_i - \Delta_{i-1})(y - a_i) & \text{if } y \geq a_i, \\
0 & \text{otherwise.}
\end{cases}
\]

Because \( \psi \) is piecewise linear, for all \( y \in [A, B] \) we have \( \psi(y) = \psi(A) + \sum_{i=1}^{n} \psi_i(y) \). We have:

\[
\mathbb{E}_D(\xi(f(x, D))) \leq \mathbb{E}_D(\psi(f(x, D))) \\
= \int_{d_1}^{d_m} \psi(f(x, d))F'(d) \, dd \\
= \psi(A) + \int_{d_1}^{d_m} \sum_{i=1}^{n} \psi_i(f(x, d))F'(d) \, dd \\
= \psi(A) + \sum_{i=1}^{n} \left( \int_{d_1}^{d_m} \psi_i(f(x, d))F'(d) \, dd \right) \\
= \psi(A) + \sum_{i=1}^{n} (\Delta_i - \Delta_{i-1}) \int_{d_1}^{d_m} \max\{0, (bx + e - d - a_i)\} F'(d) \, dd \\
= \psi(A) + \sum_{i=1}^{n} (\Delta_i - \Delta_{i-1}) \int_{d_1}^{d_m} (bx + e - d - a_i) F'(d) \, dd,
\]

where the first inequality is due to \( \psi \) being a \( K_1 \)-approximation function of \( \xi \), and the rest are algebraic manipulations.

We construct a discrete r.v. \( \hat{D} \) that takes values \( d_1, \ldots, d_{m-1} \) with:

\[
\Pr(\hat{D} = d_j) = \begin{cases} 
F(d_2) & \text{if } j = 1 \\
F(d_{j+1}) - F(d_j) & \text{if } j = 2, \ldots, m - 1.
\end{cases}
\]

It follows that the CDF \( \hat{F} \) of \( \hat{D} \) is \( \hat{F}(d) = \max\{F(d_{j+1}) : d_j \leq d, j = 1, \ldots, m - 1\} \). In order to compute expected values of continuous functions of \( \hat{D} \) using the classical integration approach, we define the generalized PDF of \( \hat{D} \) as follows:

\[
\hat{F}'(d) := \delta(d - d_1)F(d_2) + \sum_{j=2}^{m-1} \delta(d - d_j)(F(d_{j+1}) - F(d_j)).
\]

Notice that \( \hat{D} \preceq D \) in the usual stochastic order, because \( \Pr(\hat{D} > d) \leq \Pr(D > d) \) for all \( d \). Since \( \hat{D} \preceq D \) and \( (bx + e - d - a_i) \) is a decreasing function in \( d \), it follows that:

\[
\int_{d_1}^{\max\{d: bx + e - d \geq a_1\}} (bx + e - d - a_i)F'(d) \, dd \leq \int_{d_1}^{\max\{d: bx + e - d \geq a_1\}} (bx + e - d - a_i) \hat{F}'(d) \, dd = \\
(bx + e - d_1 - a_i)F(d_2) + \sum_{j=2}^{m_i(x)} (bx + e - d_j - a_i)(F(d_{j+1}) - F(d_j)) = \\
(bx + e - d_1 - a_i)\hat{F}(d_2) + \sum_{j=2}^{m_i(x)} (bx + e - d_j - a_i)(\hat{F}(d_{j+1}) - \hat{F}(d_j)) = \\
(bx + e - d_{m_i(x)} - a_i)\hat{F}(d_{m_i(x)+1}) + \sum_{k=1}^{m_i(x)-1} (d_{k+1} - d_k)\hat{F}(d_{k+1}).
\]
Putting everything together in (1), we obtain:

\[ \mathbb{E}_D(\xi(f(x, D))) \leq \psi(A) + \sum_{i=1}^{n} (\Delta_i - \Delta_{i-1}) \left( (bx + e - d_{m_i}(x) - a_i)\tilde{F}(d_{m_i}(x) + 1) + \sum_{k=1}^{m_i(x)-1} (d_{k+1} - d_k)\tilde{F}(d_{k+1}) \right). \]

Because \( \psi(x) \leq K_1 \xi(x) \) for all \( x \) and \( F' \) is nonnegative, we can write:

\[ \mathbb{E}_D(\psi(f(x, D))) = \int_{d_1}^{d_m} \psi(f(x, d))F'(d) \, dd \leq K_1 \int_{d_1}^{d_m} \xi(f(x, d))F'(d) \, dd = K_1 \mathbb{E}_D(\xi(f(x, D))). \] (2)

We now construct a discrete r.v. \( \hat{D} \) that takes values \( d_1, \ldots, d_m \) with \( \Pr(\hat{D} = d_j) = F(d_j) - F(d_{j-1}) = \tilde{F}(d_j) - \tilde{F}(d_{j-1}) \). It follows that the CDF \( \tilde{F} \) of \( \hat{D} \) is \( \tilde{F}(d) = \max\{F(d_j) : d_j \leq d, j = 1, \ldots, m\} \). As before, we define the generalized PDF of \( \hat{D} \) as follows:

\[ \tilde{F}'(d) := \sum_{j=1}^{m} \delta(d - d_j)(F(d_j) - F(d_{j-1})). \]

Notice that \( D \leq \hat{D} \) in the usual stochastic order, because \( \Pr(D > d) \leq \Pr(\hat{D} > d) \) for all \( d \). Since \( D \leq \hat{D} \) and \( (bx + e - d - a_i) \) is a decreasing function in \( d \), it follows that:

\[
\int_{d_1}^{\max\{d:bx+e-d\geq a_i\}} (bx + e - d - a_i)\tilde{F}'(d) \, dd \geq \int_{d_1}^{\max\{d:bx+e-d\geq a_i\}} (bx + e - a_i)\tilde{F}'(d) \, dd =
\sum_{j=1}^{m_i(x)} (bx + e - d_j - a_i)(F(d_j) - F(d_{j-1})) =
\sum_{j=1}^{m_i(x)} (bx + e - d_j - a_i)(\tilde{F}(d_j) - \tilde{F}(d_{j-1})) =
(bx + e - d_{m_i(x)} - a_i)\tilde{F}(d_{m_i(x)}) + \sum_{k=1}^{m_i(x)-1} (d_{k+1} - d_k)\tilde{F}(d_k) \geq
\frac{1}{K_2} (bx + e - d_{m_i(x)} - a_i)\tilde{F}(d_{m_i(x)+1}) + \frac{1}{K_2} \sum_{k=1}^{m_i(x)-1} (d_{k+1} - d_k)\tilde{F}(d_{k+1}),
\]

where the last inequality follows from the fact that \( F(d_{j+1}) \leq K_2 F(d_j) \) for \( j = 1, \ldots, m - 1 \) by definition of \( K \)-approximation set for monotone function. Then we can write:

\[
\mathbb{E}_D(\psi(f(x, D))) = \psi(A) + \sum_{i=1}^{n} (\Delta_i - \Delta_{i-1}) \int_{d_1}^{\max\{d:bx+e-d\geq a_i\}} (bx + e - d - a_i)F'(d) \, dd \geq
\psi(A) +
\frac{1}{K_2} \sum_{i=1}^{n} (\Delta_i - \Delta_{i-1}) \left( (bx + e - d_{m_i(x)} - a_i)\tilde{F}(d_{m_i(x)+1}) + \sum_{k=1}^{m_i(x)-1} (d_{k+1} - d_k)\tilde{F}(d_{k+1}) \right) \geq
\frac{\xi(x)}{K_2}. \] (3)

By combining inequalities (2) and (3) we get the desired approximation ratio.
It is easy to verify that \( \tilde{\xi} \) has increasing slopes and is therefore a convex piecewise linear increasing function. To conclude, we discuss how to compute a representation of \( \xi \) in terms of breakpoints and slopes. By examining the expression:

\[
\tilde{\xi}(x) = \psi(A) + \sum_{i=1}^{n} (\Delta_i - \Delta_{i-1}) \left( (bx + e - d_{m_i}(x) - a_i)\hat{F}(d_{m_i}(x)+1) + \sum_{k=1}^{m_i(x)-1} (d_{k+1} - d_k)\hat{F}(d_{k+1}) \right),
\]

we see that the slope of each term of the summation changes whenever \( m_i(x) \) changes. There are at most \( m \) such changes for each term, and their location can be computed in \( O(m) \) time because for term \( i \) the breakpoints are of the form \( d_j - e + a_i \) for \( j = 1, \ldots, m \). We then obtain, in \( O(mn) \) time, \( n \) sorted lists with \( m \) elements each. These lists can be merged in \( O(mn \log n) \) time, yielding a superset of the breakpoints of \( \xi(x) \). To compute the slopes we only need an additional \( O(m) \) time to preprocess the \( m - 1 \) partial sums \( \sum_{k=1}^{m-1} (d_{k+1} - d_k)\hat{F}(d_{k+1}) \), since the value of \( \hat{F} \) at all queried points is known and available in the approximation set that induces \( \hat{F} \). The overall time requirement is therefore \( O(mn \log n) \).

\[\square\]

3 \((\Sigma, \Pi)\)-approximation functions and their calculus

These results are taken from [HN16].

**Proposition 3.1 (Adapted from Prop. 3.7 in [HN16])** Let \( \varphi : [A, B] \to \mathbb{R}^+ \) be a \( \kappa \)-Lipschitz continuous convex function. Then, for every constants \( \Sigma > 0 \) and \( \Pi = 1 + \epsilon > 1 \), one can construct a piecewise-linear convex \((\Sigma, \Pi)\)-approximation function \( \tilde{\varphi} \) for \( \varphi \) with \( p := O\left( \frac{1}{\epsilon} \log \frac{\kappa \varphi_{\max}}{\Sigma} \right) \) pieces in \( O\left( (1 + t_\varphi)(\frac{1}{\epsilon} \log \frac{\epsilon \varphi_{\max}}{\Sigma} \log \frac{\kappa(B - A)}{\Sigma}) \right) \) time, with explicitly computed breakpoints and slopes. Moreover, the value of \( \tilde{\varphi}(\cdot) \) can be determined in \( \log p \) time at any point in \([A, B]\).

**Proposition 3.2 (Calculation of \((\Sigma, \Pi)\)-approximation Functions)** For \( i = 1, 2 \) let \( \Sigma_i \geq 0 \), \( \Pi_i \geq 1 \), let \( \varphi_i : D \to \mathbb{R}^+ \) be an arbitrary function over continuous domain \( D \), and let \( \tilde{\varphi}_i : D \to \mathbb{R}^+ \) be a \((\Sigma_i, \Pi_i)\)-approximation of \( \varphi_i \). Let \( \psi_i : D \to D \), and let \( \alpha_i \in \mathbb{R}^+ \). The following rules hold:

1. \( \varphi_1 \) is a \((0, 1)\)-approximation of itself.
2. (linearity of appr.) \( \alpha_1 \tilde{\varphi}_1 + \alpha_2 \) is a \((\alpha_1 \Sigma_1, \Pi_1)\)-approximation of \( \alpha_1 \varphi_1 + \alpha_2 \).
3. (summation of appr.) \( \tilde{\varphi}_1 + \tilde{\varphi}_2 \) is a \((\Sigma_1 + \Sigma_2, \max\{\Pi_1, \Pi_2\})\)-approximation of \( \varphi_1 + \varphi_2 \).
4. (composition of appr.) \( \tilde{\varphi}_1(\psi) \) is a \((\Sigma_1, \Pi_1)\)-approximation of \( \varphi_1(\psi) \).
5. (minimization of appr.) \( \min\{\tilde{\varphi}_1, \tilde{\varphi}_2\} \) is a \((\max\{\Sigma_1, \Sigma_2\}, \max\{\Pi_1, \Pi_2\})\)-approximation of \( \min\{\varphi_1, \varphi_2\} \).
6. (maximization of appr.) \( \max\{\tilde{\varphi}_1, \tilde{\varphi}_2\} \) is a \((\max\{\Sigma_1, \Sigma_2\}, \max\{\Pi_1, \Pi_2\})\)-approximation of \( \max\{\varphi_1, \varphi_2\} \).
7. (approximation of appr.) If \( \varphi_2 = \varphi_1 \) then \( \tilde{\varphi}_2 \) is a \((\Sigma_2 + \Pi_2 \Sigma_1, \Pi_1 \Pi_2)\)-approximation of \( \varphi_1 \).

References
