Anytime computation algorithms for approach-evasion differential games

Erich Mueller\textsuperscript{1,2}, Minghui Zhu\textsuperscript{1,3}, Sertac Karaman\textsuperscript{2}, Emilio Frazzoli\textsuperscript{2}

Abstract—This paper studies a class of approach-evasion differential games, in which one player aims to steer the state of a dynamic system to the given target set in minimum time, while avoiding some set of disallowed states, and the other player desires to achieve the opposite. We propose a class of novel anytime computation algorithms, analyze their convergence properties and verify their performance via a number of numerical simulations. Our algorithms significantly outperform the multi-grid method for the approach-evasion differential games both theoretically and numerically. Our technical approach leverages incremental sampling in robotic motion planning and viability theory.

I. INTRODUCTION

Recent advances in embedded computation and communication have boosted the emergence of cyber-physical systems. Cyber-physical systems are characterized by the strong coupling between the cyber space and the physical world. A number of cyber-physical systems; e.g., autonomous vehicles and the power grid, operate with time-varying computing resources in dynamically changing and uncertain operating environments. This feature demands that the controllers are synthesized in an anytime and online fashion where a feasible controller is quickly returned and its optimality is continuously improved as more processing time is available.

A fundamental problem of cyber-physical systems is safety control; i.e., controlling dynamic systems to achieve the given objectives and simultaneously stay in the given safety sets. A possible formulation is the approach-evasion differential games where one player, say the angle, desires to steer the system states to the goal set as soon as possible and maintain the system states inside the safety set and the objective of the other player, say the demon, is completely opposite. There are mainly two classes of numerical schemes for the approach-evasion differential games: one is based on viscosity solutions of partial differential equations (PDEs); e.g., in [3], [4], [24], and the other is built on viability theory; e.g., in [1], [2], [7]. Both methods are built on the multi-grid successive approximation (or coarse-to-refined) approach in; i.e., [9] and the asymptotic optimality is provable. On the other hand, one can always replace the multi-grid method by the fixed-grid method in the above papers where a grid with fixed resolution is chosen before implementing the algorithms. For this method, the approximation error is determined by the fixed grid resolution, and their relation could be very challenging to characterize, especially for the constrained nonlinear system of interest. In addition, a single predefined gird is not capable of dealing with different scenarios in dynamically changing environments.

In the robotics society, a relevant problem of robotic motion planning has been receiving considerable attention. In robotic motion planning, a feasible, collision-free and open-loop path through a cluttered environment is found, connecting an initial configuration or state to a target region. It is well-known that the problem is computationally challenging [21]. The Rapidly-exploring Random Tree (RRT) algorithm and its variants; e.g., in [15], [16], are able to find a feasible path quickly. Recently, two novel algorithms, PRM\textsuperscript{*} and RRT\textsuperscript{*}, have been developed in [13], and shown to be computationally efficient and asymptotically optimal. The methodology of incremental sampling in [13], [15], [16] has been extended to stochastic optimal control in [11] and stochastic filtering in [8].

Contributions. The paper proposes a class of novel anytime computation algorithms to solve the approach-evasion differential games, and our algorithms significantly outperform the multi-grid method both theoretically and numerically. In particular, we first propose the iGame Algorithm by leveraging incremental sampling and viability theory in [7]. On the basis of random samples, a sequence of games discrete in the time, states and inputs are constructed to approximate the original game continuous in the time, states and inputs. At each iteration, the values of the state samples are updated only once in an asynchronous way. The asymptotic convergence of the iGame Algorithm is formally ensured for all the initial states in contrast to a single initial state in [7]. We refer the readers to Section III-C for a further detailed theoretic comparison with [7].

The new feature asynchronism of iGame offers a greater freedom for the online section of sample updates, and opens up a number of opportunities to improve the computational efficiency. We then propose the variation of the iGame\textsuperscript{*} Algorithm which explicitly exploits the asynchronism. In particular, iGame\textsuperscript{*} maintains a set of directed trees at each iteration and utilizes a novel cascade update rule, only updating a subset of state samples whose child changes its state in [7]. We refer the readers to Section III-C for a further detailed theoretic comparison with [7].

The new feature asynchronism of iGame offers a greater freedom for the online section of sample updates, and opens up a number of opportunities to improve the computational efficiency. We then propose the variation of the iGame\textsuperscript{*} Algorithm which explicitly exploits the asynchronism. In particular, iGame\textsuperscript{*} maintains a set of directed trees at each iteration and utilizes a novel cascade update rule, only updating a subset of state samples whose child changes its state in the previous iteration. The iGame\textsuperscript{*} maintains the same convergence property as iGame.

Most importantly, through the homicidal-chauffeur differential game, we numerically demonstrate that iGame is faster than the multi-grid method in [7] and iGame\textsuperscript{*} is significantly

\textsuperscript{1}Equal contribution from both authors.
\textsuperscript{2}The authors are with Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, MA 02139, \{emueller,sertac,frazzoli\}@mit.edu.
\textsuperscript{3}M. Zhu is with the Department of Electrical Engineering, Pennsylvania State University, 201 Old Main, University Park, PA 16802, mzh1@psu.edu.

This research was supported in part by ONR Grant #N00014-09-1-0751 and the Michigan/AFRL Collaborative Center on Control Sciences, AFOSR grant #FA 8650-07-2-3744.
faster than iGame. The anytime property of iGame* is also shown in the numerical simulations.

Due to the space limitation, the analysis is omitted in the current paper and provided in the enlarged version [20].

**Literature review.** Among the limited number of differential games for which closed-form solutions have been derived are the homicidal-chauffeur and the lady-in-the-lake games [4], [12] which are played in unobstructed environments. For more complicated games, numerical methods must be used to determine solutions. The PDE-based and viability-based methods are two main approaches to solve the approach-evasion differential games. For the PDE-based method in [3], [4], [24], the optimal value functions are characterized as the viscosity solutions of Hamilton-Jacobi-Isaacs equations, and existing numerical schemes for PDEs are applied to approximate the viscosity solutions. This method demands the continuity of the optimal value functions. Unfortunately, this assumption is restrictive, and does not hold for many such games of interest. The more recent viability-based approach in [1], [2], [7] instead characterizes the optimal value functions via discriminating kernels of the Hamiltonian, and further solve the games through approximating discriminating kernels. This method relaxes the continuity assumption on the optimal value functions.

Another relevant set of references is about the reachability problem in the framework of hybrid systems. Another relevant set of references is about the reachability problem in the framework of hybrid systems. The following properties hold:

\[ \mathcal{X} \subseteq \mathbb{R}^N \]

The function \( f \) is continuous in \((x, u, w)\) and Lipschitz continuous in \( x \) for any \((u, w) \in U \times W\).

(A3) For any pair of \( x \in \mathcal{X} \) and \( u \in U \), \( F(x, u) \) is convex where the set-valued map \( F(x, u) \triangleq \bigcup_{w \in W} f(x, u, w) \).

**Remark 2.1:** A sufficient condition for (A3) in Assumption 2.1 is that the set of \( W \) is convex and the function \( f \) is affine with respect to \( w \).

We now proceed to describe the game of interest where the objectives of the angel and the demon are completely opposite. In particular, the angel desires to steer \( x(t) \) to his open goal set \( \mathcal{X}_{\text{goal}} \subseteq \mathbb{R}^N \) as soon as possible and simultaneously keep \( x(t) \) inside the closed constraint set \( \mathcal{X}_{\text{free}} \subseteq \mathbb{R}^N \). In contrast, the demon aims to ensure that the system states never reach \( \mathcal{X}_{\text{goal}} \) and instead leave \( \mathcal{X}_{\text{free}} \) quickly. As [14], we will refer the game to as the time-optimal approach-evasion (TO-AE, for short) differential game. The aforementioned conflicting objectives are formalized as follows. Define by \( t(x, u, w) \) the first time when the trajectory \( \phi(t; x, u, w) \) hits \( \mathcal{X}_{\text{goal}} \) while staying in \( \mathcal{X}_{\text{free}} \) before \( t(x, u, w) \). More precisely, given the trajectory \( \phi(t; x, u, w) \), the quantity \( t(x, u, w) \) is defined as follows:

\[ t(x, u, w) \triangleq \inf\{ t \geq 0 \mid \phi(t; x, u, w) \in \mathcal{X}_{\text{goal}}, \phi(s; x, u, w) \in \mathcal{X}_{\text{free}}, \forall s \in [0, t] \}. \]

If \( \phi(t; x, u, w) \) leaves \( \mathcal{X}_{\text{free}} \) before reaching \( \mathcal{X}_{\text{goal}} \) or never reaches \( \mathcal{X}_{\text{goal}} \), then \( t(x, u, w) = +\infty \). So the angel aims to minimize the hitting time \( t(x, u, w) \), and the demon instead wants to maximize the same cost functional.

To define the value of the TO-AE differential game, we need the notion of nonanticipating or causal strategy in the sense of Varaiya, Roxin, Elliot and Kalton in [25]. The set \( \Gamma^u \) of such strategies for the angel is such that \( \gamma^u : \mathcal{W} \rightarrow \mathcal{U} \) satisfies for any \( T \geq 0, \gamma^u(w(t)) = \gamma^u(w(\cdot)) \) for \( t \in [0, T] \) if \( w(t) = w(\cdot) \) for \( t \in [0, T] \). The lower value of the TO-AE differential game is given by:

\[ T^*(x) = \inf_{\gamma^u \in \Gamma^u} \sup_{w(\cdot) \in \mathcal{W}} t(x, \gamma^u(w(\cdot)), w(\cdot)). \]

The function \( T^* \) will be referred to as the minimum time function. Note that \( t(x, u, w) \) could be infinity and this may cause numerical issues. To deal with this, we normalize the hitting time by the Kružkov transform of \( \Psi(r) = 1 - e^{-r} \). With this nonlinear transform, we further define the discounted cost functional \( J(x, u, w) = \Psi \circ t(x, u, w) \), and the discounted lower value \( v^* \) as follows:

\[ v^*(x) = \inf_{\gamma^u \in \Gamma^u} \sup_{w(\cdot) \in \mathcal{W}} J(x, \gamma^u(w(\cdot)), w(\cdot)). \]

It is easy to see that \( v^*(x) = \Psi \circ T^*(x) \) for \( \forall x \in \mathcal{X} \). We will refer \( v^* \) to as the optimal value function.

\[ \dag \]

The papers [4], [7], [10] refer to the game we consider as the pursuit-evasion game. As [5], we will refer the pursuit-evasion differential games to as those where the dynamics of two players are decoupled and one player aims to capture the other. The homicidal chauffeur game in Section 5 is an example of the pursuit-evasion game. The games of interest can also be applied to robust optimal control where two players act on a single dynamic system.
The objective of the paper is to design and analyze anytime algorithms to approximate $v^*$ (or equivalently, $T^*$) and further synthesize feedback control policies for the angle.

B. Notations and primitive procedures

Because of (A1) and (A2) in Assumption 2.1, $M \triangleq \sup_{x \in \mathcal{X}, u \in \mathbb{U}, w \in \mathbb{W}} f(x, u, w)$ is well-defined. Let $\ell$ be the Lipschitz constant of $f$ with respect to $x$ for any $(u, w) \in \mathbb{U} \times \mathbb{W}$. Let $\mu(S)$ be the Lebesgue measure of the set $S$.

Let $C_N \triangleq \frac{2}{(2n)!}$ be the volume of the unit ball in $\mathbb{R}^N$. Let $B(x, r)$ be the closed ball centered at $x$ with radius $r$.

We will use the short hand $B = B((0,1))$ for the unit ball.

Define the norms $\|v\|_S \triangleq \sup_{x \in S} \|v(x)\|$ and $\|v - \bar{v}\|_S \triangleq \sup_{x \in S} \|v(x) - \bar{v}(x)\|$ for $v, \bar{v} : S \to \mathbb{R}$. Denote $\gamma \triangleq \frac{C_N D_n}{\mu(\mathcal{X}_{\text{free}})}$ where $D_n > 0$ is chosen such that $\gamma > 2$.

The procedure Samp1Lo$(S,n)$: return $n$ states which are uniformly and independently sampled from the set $S$.

C. Preliminary on viability theory

First of all, let us define the set-valued map $\Phi : \mathcal{X} \times \mathbb{R} \times \mathbb{R} \to \mathcal{X} \times \mathbb{R}$ by: for $x \notin \mathcal{X}_{\text{goal}}$, $\Phi(x, \varpi, u) = F(x, u) \times \{-1\}$; otherwise, $\Phi(x, \varpi, u) = \mathcal{C}_O[F(x, u) \times \{-1\}] \cup \{(0,0)\}$ where $F(x, u) \triangleq \bigcup_{w \in \mathbb{W}} f(x, u, w)$ and $\mathcal{C}_O[A]$ is the convex hull of set $A$. The map $\Phi$ is the set-valued version of $\mathbb{I}$ by taking into account all the controls of the demon. In addition, $\Phi$ is augmented by the scalar variable $\varpi$ representing the time instant. If the state $x$ does not reach $\mathcal{X}_{\text{goal}}$, then the time $\varpi$ decreases at a rate 1; otherwise, it would stop. The combination of Theorem 5.2 and Lemma 5.4 in [7] shows that the epigraph of $T^*$ is identical to the viability domain for the set-valued map $\Phi$. That is, if $\Phi$ starts from the initial state $(x(0), \varpi(0))$ with $T^*(x(0)) \leq \varpi(0)$, then there is $u(\cdot)$ which is able to steer the state $x(t)$ to the goal set $\mathcal{X}_{\text{goal}}$ no later than $T^*(x(0))$ and hence $\varpi(T^*(x(t))) - T^*(x(t)) \geq 0$. By utilizing this characterization, the authors in [7] propose a multi-grid successive approximation algorithm on Page 232 to asymptotically reconstruct $T^*$ via a sequence of fully discretized dynamics of $\Phi$. The readers are referred to [1], [2], [7] for a detailed discussion on viability theory for optimal control and differential games.

III. The iGAME Algorithm

In this section, we will develop the iGAME Algorithm which integrates incremental sampling in [13], [15], [16] and viability theory in [7]. We will also characterize its asymptotic convergence properties. The main notations used in this section are summarized in Table I.

| $y_n \in \mathcal{X}$ | the new sample obtained at iteration $n$ |
| $S_n \subseteq \mathcal{X}$ | the set of samples obtained up to iteration $n$ |
| $d_n > 0$ | the space discretization size of $S_n$ |
| $h_n > 0$ | the time discretization size at iteration $n$ |
| $\kappa_n \triangleq h_n - d_n$ | the approximate time discretization |
| $\alpha_n$ | the dilation size at iteration $n$ |
| $w_n : S_n \to \mathbb{U}$ | the controls for the states in $S_n$ |
| $v_n : S_n \to [0,1]$ | the discrete value function for the states in $S_n$ |
| $\bar{v}_n : \mathcal{X} \to [0,1]$ | the interpolated value of $v_n$ |

The initial state $\nu_0(x)$ could be any value in $[0,1]$. It is in contrast to $v_0(x)$ must be equal 0 in [7]. For the simplicity, we generate the initial values by random sampling in the iGAME Algorithm.

set: i.e., $S_n = S_{n-1} \cup \{y_n\}$. Then the algorithm computes the dispersion $d_n$ such that for any $x \in \mathcal{X}_{\text{free}}$, there exists $x' \in S_n$ such that $\|x - x'\| \leq d_n$. The quantity $d_n$ can be viewed as the resolution of the finite grid generated by $S_n$. Based on $d_n$, the algorithm sets the time discretization size $h_n$ such that $d_n = \alpha h_n$. Here we choose $h_n = \frac{d_n}{\alpha}$ where $\alpha$ could be any positive scalar. Denote $\kappa_n \triangleq h_n - d_n$, and $R_n$ be an integer lattice on $\mathbb{R}$ consisting of segments of length $d_n$.

With the above set of parameters, we construct the discretization $\Phi_n$ of $\Phi$ as [7]: for $x \in S_n \setminus B(\mathcal{X}_{\text{goal}}, Mh_n + d_n)$, $\Phi_n(x, y, u) = \{x + h_n F(x, u) + \alpha_n B\} \cap \{y = h_n + [-d_n, d_n]\} \cup \{S_n \times R_n\}$; otherwise, $\Phi_n(x, y, u) = \mathcal{C}_O\{x + h_n F(x, u) + \alpha_n B\} \times \{y = h_n + [-d_n, d_n]\} \cap \{S_n \times R_n\}$, where the dilation size $\alpha_n = 2d_n + \epsilon h_n d_n + M h_n^2$. As shown in [7], $\Phi_n$ is a good approximation of the continuous counterpart $\Phi$ in Section II-C and the choice of $\alpha_n$.

We then solve the discretized game associated with $\Phi_n$ for only one step. Towards this end, let $\nu_{n-1} : S_{n-1} \to [0,1]$ be the estimate of the value function at iteration $n-1$. After obtaining $y_n$, we need to initialize the estimate, say $\bar{v}_{n-1}$, on the new grid $S_n$. The point $y_n$ does not bring any new information about $v^*$, and the estimates on $S_{n-1}$ should not be affected by $y_n$. Thus, we choose $\bar{v}_{n-1}(x) = \nu_{n-1}(x)$ for $x \in S_{n-1}$ and $\bar{v}_{n-1}(y_n) = 1$.

After initializing $\nu_{n-1}$, we execute a single value iteration on each state in $K_n$, a subset of $S_n$, by performing the discrete operator $F_n$ on $\bar{v}_{n-1}$ and get the estimate $v_n$. The discrete operator $F_n$ is defined as follows: for $x \in S_n \setminus B(\mathcal{X}_{\text{goal}}, Mh_n + d_n)$,

$$
F_n \circ \bar{v}_{n-1}(x) = 1 - e^{-\kappa_n} + e^{-\kappa_n} \max_{u \in \mathbb{U}, y \in B(x, h_n f(x, u, w), \alpha_n) \cap S_n} \bar{v}_{n-1}(y),
$$

and, for $x \in S_n \cap B(\mathcal{X}_{\text{goal}}, Mh_n + d_n)$, $F_n \circ \bar{v}_{n-1}(x) = \bar{v}_{n-1}(x)$.

The discrete operator $F_n$ is derived from (33) in [7] by performing the Kružkov transform. It will be shown that there is a unique fixed point $\nu_n^* : S_n \to [0,1]$, of $F_n$ where $v_n^*(x) = \nu_{n-1}(x)$ for $S_n \cap B(\mathcal{X}_{\text{goal}}, Mh_n + d_n)$.

The iGAME Algorithm is formally stated in Algorithm 4. Here, we would like to mention that due to the Kružkov transform, all the computations $v^*$ can be equivalently performed in terms of the estimates of $T^*$. However, the unboundedness of $T^*$ may incur computational issues.
Algorithm 1: The iGame Algorithm

1. for \( x \in S_0 \subset X \) do
2. \( v_0(x) = \text{Sample}([0, 1]); \)
3. \( n \leftarrow 1; \)
4. while \( n < K \) do
5. \( y_n \leftarrow \text{Sample}(X, 1); \)
6. \( S_n \leftarrow S_{n-1} \cup \{y_n\}; \)
7. \( h_n \leftarrow d_n \leftarrow \frac{\alpha_n}{\alpha_n + 2d_n}; \)
8. \( \alpha_n \leftarrow 2d_n + \ell h_n d_n + M h_n^2; \)
9. for \( x \in S_{n-1} \) do
10. \( \tilde{v}_{n-1}(x) = v_{n-1}(x); \)
11. for \( x \in K_n \subset S_n \setminus B(X_{\text{goal}}, M h_n + d_n) \) do
12. \( (v_n(x), u_n(x)) \leftarrow \text{VI}(S_n, \tilde{v}_{n-1}); \)
13. for \( x \in S_n \setminus (K_n \cup B(X_{\text{goal}}, M h_n + d_n)) \) do
14. \( v_n(x) = \min_{y \in B(x, \alpha_n) \cap S_{n-1}} v_{n-1}(y); \)
15. \( v_n(x) \leftarrow \tilde{v}_{n-1}(x); \)
16. end for
17. end for
end while

Algorithm 2: VI\((S_n, \tilde{v}_{n-1})\)

1. \( U_n \leftarrow U_{n-1} \cup \text{Sample}(U, 1); \)
2. \( v_n(x) \leftarrow 1 - e^{-\alpha_n} + e^{-\alpha_n} \max_{w \in W} \min_{y \in B(x, \alpha_n) \cap \gamma_{n-1}} \tilde{v}_{n-1}(y); \)
3. \( u_n(x) \leftarrow \text{the solution to } u \text{ in the above step}; \)

B. Performance analysis

In this section, we analyze the asymptotic convergence properties of the iGame algorithm.

For the discrete fixed point \( v_n^\ast \), we define the interpolated fixed point \( \tilde{v}_{n-1}^\ast : S_n \rightarrow [0, 1] \) as follows:

\[
\tilde{v}_{n-1}^\ast(x) = v_{n-1}^\ast(x), \quad x \in S_{n-1}, \quad \tilde{v}_{n-1}^\ast(y_n) = 1.
\]

The following lemma shows the interpolation is consistent.

Lemma 3.1: The following holds with probability one:

\[
\min_{y \in B(x, \alpha_n) \cap S_{n-1}} \tilde{v}_{n-1}^\ast(y) - \min_{z \in B(x, \alpha_n) \cap S_n} v_n^\ast(z) \|x\| \rightarrow 0.
\]

It follows from Lemma 3.1 and the boundedness of \( D \) that

\[
\sum_{\tau=1}^{D+1} \| y \in B(x, \alpha_n) \cap S_{n-\tau} \| \tilde{v}_{n-\tau}^\ast(y) - \min_{z \in B(x, \alpha_n) \cap S_{n-\tau}} v_n^\ast(z) \|x\| \rightarrow 0.
\]

So there always exist a non-negative and non-increasing sequence \( \{\gamma_n\} \) and a constant \( C > 0 \) such that

\[
\sum_{\tau=0}^{D} \| y \in B(x, \alpha_n) \cap S_{n-\tau} \| \tilde{v}_{n-\tau}^\ast(y) - \min_{z \in B(x, \alpha_n) \cap S_{n-\tau}} v_n^\ast(z) \|x\| \leq \gamma_n, \quad \gamma_n \leq C \gamma_{n+1}.
\]

This following assumption requires that each state sample invokes the procedure VI at least once every \( D + 1 \) iterations.

Assumption 3.1: There is an integer \( D \geq 0 \) such that for any \( n \geq 1 \), it holds that \( S_n \subseteq \cup_{s=0}^{D} K_{n+s} \).

The following theorem is the core of this section and summarizes the convergence properties of the iGame algorithm.

Theorem 3.1: Suppose that Assumptions 2.1 and 3.1 hold. Then the following properties hold for the iGame algorithm:

1. \( \lim_{n \rightarrow +\infty} \frac{1}{\gamma_n} \| v_n(x) - v_n^\ast(x) \|_{S_n} = 0, \)
2. \( \lim_{n \rightarrow +\infty} \frac{1}{\gamma_n} \| \min_{y \in B(x, \alpha_n) \cap S_n} v_n(y) - \min_{z \in B(x, \alpha_n) \cap S_n} v_n^\ast(z) \|_{x} = 0. \)

In addition, it holds that \( \| v_n - v^\ast \|_{S_n} \rightarrow 0. \)

C. Discussion

We would like to make a couple remarks on Theorem 3.1. During the first paragraph of this section, we use \( D = 0 \) corresponding to synchronous updates. In (P1) of Theorem 3.1 \( \| \min_{y \in B(x, \alpha_n) \cap S_n} v_n(y) - \min_{z \in B(x, \alpha_n) \cap S_n} v_n^\ast(z) \|_{x} = 0 \) is the distance to \( v^\ast \) of iGame, and the quantity \( \gamma_n \) represents an over-approximation of the distance to \( v^\ast \) of the multi-grid algorithm in [7]. The relation of \( \gamma_n \leq C \gamma_{n+1} \) means that \( \gamma_{n+1} \) cannot be arbitrarily small in comparison with \( \gamma_n \). One example is that \( \{\gamma_n\} \) decreases in an exponential rate \( a > 0 \); i.e., \( \gamma_n = e^{-a \gamma_0} \). For this case, \( \gamma_{n+1} = e^{-a \gamma_n} \). Another example is \( \gamma_n = \frac{\log n}{n\log n} \), and then \( \gamma_{n+1} = \frac{\log n}{n\log n}, \gamma_{n+1} \leq \frac{n+1}{n} \leq 2 \). For the cases with \( \gamma_n \leq C \gamma_{n+1} \), (P1) of Theorem 3.1 demonstrates that, in the asymptotic sense, iGame is as optimal as the multi-grid method in [7]. Here, we would like to emphasize that iGame only executes once the VI procedure on each grid. In contrast, the algorithm in [7] has to completely determine the fixed point on each grid before the grid refinement, and thus, it may be difficult to determine when to refine the grids. Thus, the aggregate computational complexity of synchronous iGame; i.e., \( D = 0 \), is lower than that of that in [7]. We will numerically demonstrate this fact in Section 3.B.

The iGame algorithm is built on the multi-grid method on Page 232 in [7]. Here we would like to point out several distinctions. Firstly, as [13], [15], [16], iGame utilizes random sampling which scales well in high-dimension spaces in comparison with a priori discretization in [7]. Secondly, iGame only performs the discrete operator \( F_n \) only once on each grid. This helps reduce the aggregate computational complexity of iGame over that in [7]. This issue has been elaborated. Thirdly, the procedure VI can be performed on a subset of \( S_n \) in contrast to the whole set \( S_n \) in [7]. This asynchronism feature itself is new and, more importantly, will allow for a significant improvement in the convergence.
rate in the next section. Fourthly, iGame is shown to be globally converge; that is, that algorithm converges from any initial values of $v_0(x) \in [0,1]$ for $x \in S_0$. This is in contrast to the local convergence in [7] where $v_0(x) = 0$ for $x \in S_0$.

IV. THE iGAME$^*$ ALGORITHM: CASCADE UPDATES

The new feature asynchronism of iGame offers a greater freedom for the online section of sample updates and opens up a number of opportunities to improve the computational efficiency. In this section, we will develop the variation of iGame$^*$ which explicitly exploits the asynchronism.

To motivate iGame$^*$, we first examine the behavior of iGame. For each iteration, we define the directed graph $G_n \triangleq \{S_n, \mathcal{E}_n\}$ such that $(x, y) \in \mathcal{E}_n$ if $y \in B(x, \alpha_n) \cap S_n$. That is, $v^*_n(x)$ is determined by the values of all $y$ such that $(x, y) \in \mathcal{E}_n$.

In iGame, the sample numbers are small for the initial iterations. So $v_n$ could approach $v^*_n(x)$ quickly. Let us assume this is the case at iteration $n$; i.e., $v_n(x) \approx v^*_n(x)$ for $x \in S_n$. Due to the backward propagation of value iterations on $G_n$, the addition of a single point $y_n$ only affects the values of $v^*_{n+1}(x)$ for a subset of $S_n$. When $n$ is large, the value function $||v^*_{n+1}(x) - v^*_n(x)||$ could be small. If all the samples in $S_{n+1}$ are updated at iteration $n + 1$, the decrease of the approximation error could be small. In contrast, a large amount of computation is required. The computational efficiency is particularly low when $n$ is large. This reveals that synchronous updates in iGame could waste a large amount of computation.

The above observation motivates the efficient utilization of asynchronism and thus iGame$^*$ formally stated in Algorithm 3. In particular, iGame$^*$ maintains a set of directed trees where the child $y$ of the sample $x$ is the one realizing the value of $x$ in $VI^*$ at the last iteration. Note that the relation of parent-child is obtained as a byproduct of the procedure $VI^*$. iGame$^*$ employs the cascade update scheme where the updates of the state samples are triggered if their children changes their estimates or they have not updated for $D$ consecutive iterations. That is, each new sample triggers a new round of value updates and the updates are performed backward along the paths of $G_n$. Instead of $G_n$, we use the subgraph $G_n^* \triangleq (S_n, E_n^*)$ of $G_n$ to define the parent-children relation since the generation of $G_n^*$ does not demand any additional computation.

iGame$^*$ shares the same asymptotic convergence as iGame, and the property is summarized as follows:

**Theorem 4.1:** Suppose that Assumption 2.1 holds. Then the properties of (P1) and (P2) in Theorem 3.1 hold for the iGame$^*$ Algorithm.

**Proof:** Note that Assumption 3.1 in Theorem 3.1 is enforced. Thus the proof is identical to that of Theorem 3.1.

**Remark 4.1:** The cascade update rule is not unique. The one we choose in iGame$^*$ is completely built on the computation results of the previous iteration and does not incur any additional computation effort. Intuitively, iGame$^*$ is at least as fast as iGame. It turns out that iGame$^*$ is significantly faster than iGame, shown in Section V.

V. SIMULATION RESULTS

This section describes simulation experiments conducted to assess the performance of iGame and iGame$^*$.

A. Fence Escape Differential Game

The first set of results focus on a simple zero-sum differential game with a known analytic solution and a discontinuous value function, referred to here as fence escape. In this game, two agents, a pursuer and an evader, move along opposite sides of a straight fence segment extending from $x = 0$ to $x = 10$. The positions of the pursuer and evader are respectively denoted by $x_p$ and $x_e$. The agents can directly command their velocities, $u_p = \dot{x}_p$ and $u_e = \dot{x}_e$, which are both bounded in magnitude by 1. The game terminates when the evader passes either end of the fence, but the evader is blocked by the pursuer and may not exit the fence region if $|x_e - x_p| \leq 1$. Formally, the goal set for the evader is:

$$X_{goal} : ((x_e < 0) \lor (x_e > 10)) \land (|x_e - x_p| > 1)$$

The objective of the evader is to reach this set in minimum time. The optimal value function for this game has a simple analytic solution shown in Figure 2.

The evolution of the iGame value function approximation for the fence escape game is shown in Figure 3. Here we see the initial crude approximation to the value function become refined over time, as well as convergence to the discontinuous value function.

B. Homicidal Chauffeur

Now we consider a variation of the classic homicidal chauffeur problem [5], in which a fast but less agile pursuer seeks to bring a slower, more manoeuvrable evader within some capture distance in minimum time, while the evader tries to reach some greater escape distance. For the pursuer we use a simplistic model of vehicular dynamics, in which heading rate is commanded directly, while the evader is
kinematically unconstrained except for a maximum velocity. This results in a game in $\mathbb{R}^5$ with dynamics:

$$
\begin{align*}
\dot{x}_p &= v_p \cos \theta, & \dot{y}_p &= v_p \sin \theta, & \dot{\theta} &= u_p, \\
\dot{x}_e &= v_e \cos u_e, & \dot{y}_e &= v_e \sin u_e,
\end{align*}
$$

Where $(x_p, y_p, \theta)$ defines the state of the pursuer, who commands angular rate, $u_p$, $|u_p| < \omega$, and $(x_e, y_e)$ defines the state of the evader, who commands angle, $u_e$, directly. By expressing the dynamics in a coordinate system fixed to the pursuer, this problem may be reduced to 2 dimensions, in which the dynamics are:

$$
\begin{align*}
\dot{x} &= u_p y + v_e \cos u_e - v_p, \\
\dot{y} &= -u_p x - v_e \sin u_e,
\end{align*}
$$

Algorithm 3: The iGame* Algorithm

1. for $x \in S_0 \subset X$ do
2.  $v_0(x) \leftarrow \text{Sample([0, 1])};$
3.  Flag$(x) \leftarrow 0;$
4.  $n \leftarrow 1;$
5.  while $n < K$ do
6.  $y_n \leftarrow \text{Sample}(X, 1);$ 
7.  $S_n \leftarrow S_{n-1} \cup \{y_n\};$
8.  $h_n \leftarrow d_n^{-1};$
9.  $\alpha_n \leftarrow 2d_n + th_n d_n + M \theta h_n^2;$
10. for $x \in S_{n-1}$ do
11.   $\tilde{v}_n(x) \leftarrow v_{n-1}(x);$ 
12.   $\tilde{v}_{n-1}(y_n) \leftarrow 1;$
13.   $K_n \leftarrow \{y_n\};$
14. for $x \in S_{n-1}$ do
15.   if Flag$(x) \equiv D$ then
16.     $K_n \leftarrow K_n \cup \{x\};$
17.     Continue;
18.   for Flag(Child$(x)) \equiv 0$ do
19.     $K_n \leftarrow K_n \cup \{x\};$
20.   for $x \in K_n \setminus B(x_{\text{goal}}, 2h_n + d_n)$ do
21.     $(v_n(x), u_n(x), \text{Child}(x)) \leftarrow \text{VI}^t(S_n, v_{n-1});$
22.   for $x \in S_n \setminus (K_n \cup B(x_{\text{goal}}, 2h_n + d_n))$ do
23.     $z \leftarrow \arg\min_{y \in B(x, \alpha_n) \cap S_{n-1}} \tilde{v}_{n-1}(y);$ 
24.     $v_n(x) \leftarrow v_{n-1}(z);$ 
25.     $\text{Child}(x) \leftarrow z;$ 
26.     Flag$(x) \leftarrow \text{Flag}(x) + 1;$
27.   for $x \in S_n \cap B(x_{\text{goal}}, 2h_n + d_n)$ do
28.     $v_n(x) \leftarrow \text{Flag}(x) + 1;$
29.   Flag$(x) \leftarrow \text{Flag}(x) + 1;$

Algorithm 4: VI$^t(S_n, v_{n-1})$

1. $U_n \leftarrow U_{n-1} \cup \text{Sample}(U, 1);$ 
2. $v_n(x) \leftarrow 1 - e^{-\alpha_n} + e^{-\alpha_n} \max_{y \in U_n} \min_{w \in U_n} \min_{\omega \in [0, 1]} \max_{x \in B(x, \alpha_n) \cap S_n} \tilde{v}_{n-1}(y);$ 
3. $(v_n(x), u_n(x)) \leftarrow$ the solution to $(w, u)$ in the above step;
4. $\text{Child}(x) \leftarrow \arg\min_{y \in B(x + h_n f(x, u_n(x)), \omega_n) \cap S_n} \tilde{v}_{n-1}(y);$ 

Fig. 2: Progression of the value function approximation computed by iGame, for the fence escape game, indicating the number of samples, and the elapsed computation time. Progressive refinement of the value function and convergence to the analytic solution, shown in figure 2f, is seen.

Where $q = (x, y)$ represents the location of the evader in a coordinate system fixed to the pursuer - translated by $(x_p, y_p)$ and rotated by angle $\theta$. In this formulation of the game, the pursuer wins if $|q|_\infty < r_p$, which represents intersection of the evader with a square pursuer, and the evader wins if $|q| > r$. The parameters used in the simulation tests are:

$$
\begin{array}{c|c|c|c|c|c|c|c|c|c}
\omega & 5 & v_p & 0.5 & v_e & 1 & r & 1 & r_p & 0.05 \end{array}
$$

Firstly, we examine the performance of synchronous iGame ($D = 0$) and iGame* on the 2-dimensional problem. Figure 3 shows the temporal evolutions of the approximation errors generated by the multi-grid, iGame and iGame* Algorithms. Each curve corresponds to the average of a number of trials for a specific algorithm. The trial number of iGame* is 100. Due to their relatively slow convergence, we only simulate the multi-grid method and iGame Algorithm for 5
and 10 times, respectively. The standard deviations of each algorithm is also shown in Figure 3. In order to compute the approximation errors, we use the solution computed by a $200 \times 200$ grid as the benchmark. It is noted that logarithmic scaling is used in the figure.

Figure 3 clearly shows that the convergence of iGame is faster than that of the multi-grid method, and iGame* is significantly faster than iGame. In Figure 4, we compare the average computation times three algorithms take to reach the approximation errors of 0.1, 0.08, 0.06 and 0.04. Figure 4 confirms the superior convergence rates of iGame and iGame* over that of the multi-grid method. The evolution of the iGame value function approximation for the fence escape game is shown in Figure 5.

Fig. 3: The average errors and standard deviations of the multi-grid, iGame and iGame* Algorithms

Secondly, we examine the performance of iGame* when the pursuer uses iGame* and the evader uses a fixed-grid approximation based on a $50 \times 50$ grid iterated to convergence. Figure 6 shows the outcome of the game from initial states on a $50 \times 50$ grid. For practical purposes, a maximum time approximately 10 times larger than a reasonable capture time was used for the duration of the simulated games, with the outcome of games reaching this threshold referred to as “timeout”. Qualitatively, games so terminating generally exhibit oscillatory behaviour, with the pursuer attempting to capture the evader, who narrowly escapes, the pursuer turning around to try again, the evader escaping again, and so on. Thus, while some games labeled as timeout may in fact have terminated at some later time, it is thought that most will behave this way and so timeout can be thought of also in this sense as a stalemate.

These tests demonstrate the performance of iGame* against a fixed grid opponent improving as the number of points increases, to the point where the evader is captured from nearly all states where capture is possible, and some states from which escape is possible. The latter case is not entirely obvious in the figures, mostly due to the fact that the 50x50 fixed grid still yields a fairly good approximation. Importantly, these results also show the benefit of using iGame* in an online setting - early on, when few points are used, games starting from many states terminate in timeout, however, this outcome is not truly a loss for the pursuer, but allows more time to incrementally improve the policy. As the policy is improved, the pursuer is better able to capture the evader, and states which initially would have resulted in timeout result in capture. In other words, so long as the pursuer is able to prevent the evader from escaping, the policy can be improved, and capture is inevitable.

C. Implementation Notes

All simulation tests were written in Java version 1.7.0 and run on a machine running Windows 7, with 8GB RAM and an Intel(R) Core(TM) i7 processor running at 2.80 GHz.

For all simulation tests described in this document, the optimal feedback control policies were determined based on the value function approximation, for all the algorithms, by solving a single-step min-max problem over a discrete set of inputs, as in the value update.

VI. CONCLUSIONS

We have investigated two anytime computation algorithms, iGame and iGame*, for the approach-evasion differential game. Their asymptotic convergence properties have been formally analyzed. Through a number of numerical simulations, we have demonstrated their anytime property and superior computational efficiency over the state of the art. Future directions include stochastic differential games and multi-player differential games.
Fig. 5: Progression of the value function approximation computed by iGame∗, for the homicidal chauffeur game, indicating the number of points in the set, and the elapsed computation time.

Fig. 6: Progression of capture and escape regions for the homicidal chauffeur problem, with the evader using a 50x50 fixed grid approximation, and the pursuer using iGame∗.

REFERENCES
[1] J.P. Aubin. Viability theory. Springer, 2009.
[2] J.P. Aubin, A. Bayen, and P. Saint-Pierre. Viability theory: New directions. Springer-Verlag, New York, 2011.
[3] M. Bardi and I. Capuzzo-Dolcetta. Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations. Birkhäuser, 1997.
[4] M. Bardi, M. Falcone, and P. Soravia. Numerical methods for pursuit-evasion games via viscosity solutions. Annals of the International Society of Dynamic Games, 4:105 – 175, 1999.
[5] T. Basar and G. Olsder. Dynamic noncooperative game theory. SIAM Classics in Applied Mathematics, 1999.
[6] D.P. Bertsekas. Dynamic Programming and Optimal Control. Athena Scientific, 1995.
[7] P. Cardaliaguet, M. Quincampoix, and P. Saint-Pierre. Set-valued numerical analysis for optimal control and differential games. Annals of the International Society of Dynamic Games, 4(1):177 – 247, 1999.
[8] F. Chalushti, S. Karaman, and E. Frazzoli. An incremental sampling-based algorithm for optimal filtering. In IEEE Conference on Robotics and Automation, 2012.
[9] C.S. Chow and J.N. Tsitsiklis. An optimal one-way multigrid algorithm for discrete-time stochastic control. IEEE Transactions on Automatic Control, 36(8):988 – 914, 1991.
[10] M. Falcone. Numerical methods for differential games based on partial differential equations. International Game Theory Review, 8:231 – 272, 2006.
[11] V. Huynh, S. Karaman, and E. Frazzoli. Incremental sampling-based algorithm for stochastic optimal control. In IEEE Conference on Robotics and Automation, 2012.
[12] R. Isaacs. Differential Games: A Mathematical Theory with Applications to Warfare and Pursuit, Control and Optimization. Dover, 1999.
[13] S. Karaman and E. Frazzoli. Sampling-based algorithms for optimal motion planning. International Journal of Robotics Research, 30(7):846 – 894, 2011.
[14] N.N. Krasovskii and A.I. Subbotin. Game-theoretical control problems. Springer-Verlag, New York, 1987.
[15] J.J. Kuffner and S.M. LaValle. RRT-connect: An efficient approach to single-query path planning. In IEEE Conference on Robotics and Automation, pages 995–1001, 2000.
[16] S.M. LaValle and J.J. Kuffner. Randomized kinodynamic planning. The International Journal of Robotics Research, 20(5):378 – 400, 2001.
[17] J. Lygeros. On reachability and minimum cost optimal control. Automatica, 40(6):917 – 927, 2004.
[18] I. Mitchell, A. Bayen, and C. Tomlin. Validating a Hamilton-Jacobi approximation to hybrid reachable sets. In Hybrid Systems: Computation and Control, pages 917 – 927, 2004.
[19] I. Mitchell, A. Bayen, and C. Tomlin. A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games. IEEE Transactions on Automatic Control, 50(7):947 – 957, 2005.
[20] E. Mueller, M. Zhu, S. Karaman, and E. Frazzoli. Anytime computation algorithms for approach-evasion differential games. http://arxiv.org/abs/1308.1174.
[21] J.H. Reif. Complexity of the mover's problem and generalizations. In The 20th Annual IEEE Conference on Foundations of Computer Science, pages 421–427, 1979.
[22] S. Resnick. A Probability Path. Birkhäuser, 1998.
[23] J.A. Sethian. Level Set Methods: Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision and Materials Science. Cambridge University Press, 1996.
[24] P.E. Souganidis. Two-player, zero-sum differential games and viscosity
solutions. *Annales of the International Society of Dynamic Games*, 4(1):69 – 104, 1999.

[25] P. Varaiya, R.J. Elliott, E. Roxin, and N.J. Kalton. The existence of value in differential games. *American Mathematical Society*, (126), 1972.

**VII. APPENDIX**

In this section, we will provide the complete proof for Lemma 3.1 and Theorem 3.1.

**A. Preliminary results**

Let us start with a set of generic preliminary results. We consider finite grids $X_d \subset \mathcal{X}$, $U_d \subset U$ and $W_d \subset W$ such that

\[
\forall x \in \mathcal{X}, \ \exists x' \in X_d, \text{ s.t. } ||x - x'|| \leq d,
\]

\[
\forall u \in U, \ \exists u' \in U_d, \text{ s.t. } ||u - u'|| \leq d,
\]

\[
\forall w \in W, \ \exists w' \in W_d, \text{ s.t. } ||w - w'|| \leq d. \tag{3}
\]

Note that $X_d, U_d$ and $W_d$ are not necessarily regular lattices. The following lemma show the state discretization is non-expansive.

**Lemma 7.1 (Non-expansiveness of state discretization):** Consider any two functions $v, \tilde{v} : X_d \to [0, 1]$. (T1) The following holds for any finite set $X_d' \subseteq X_d$ and any scalar $d' \geq d$:

\[
\|v(y) - \tilde{v}(z)\|_X \leq \|v - \tilde{v}\|_{X_d}, \tag{4}
\]

(T2) The following holds for any $\alpha \geq d \geq d'$:

\[
\|v(y) - \tilde{v}(z)\|_X \leq \|v - \tilde{v}\|_{X_d}, \tag{5}
\]

Proof: (T1) Pick any $x \in \mathcal{X}$, and let $\min_{y \in B(x, \alpha) \cap X_d} v(y) = v(y')$ and $\min_{z \in B(x, \alpha) \cap X_d} \tilde{v}(z) = \tilde{v}(z')$ for some $y', z' \in X_d'$.

Since $y', z' \in B(x, \alpha) \cap X_d$ and $w \in X_d$, then we have $v(y') \leq v(z')$, $\tilde{v}(z') \leq \tilde{v}(y')$.

(T2) We now move to show $||v(y') - \tilde{v}(z')|| \leq \max\{||v(z') - \tilde{v}(z')||, ||v(y') - \tilde{v}(y')||\}$. Towards this end, we have $v(y') - \tilde{v}(z') \leq v(z') - \tilde{v}(z') \leq \max\{||v(z') - \tilde{v}(z')||, ||v(y') - \tilde{v}(y')||\}$,

where the first inequality is a direct result of (7). Analogously, we have $\tilde{v}(z') - v(y') \leq \tilde{v}(z') - v(y') \leq \max\{||v(z') - \tilde{v}(z')||, ||v(y') - \tilde{v}(y')||\}$.

Combine the above two cases, and then we have

\[
\|v(y) - \tilde{v}(z)\|_X \leq \|v(y') - \tilde{v}(z')|| \leq \max\{||v(z') - \tilde{v}(z')||, ||v(y') - \tilde{v}(y')||\} \leq \|v - \tilde{v}\|_{X_d}, \tag{8}
\]

where the last inequality uses the fact that $y', z' \in X_d' \subseteq X_d$.

**B. Under Assumption 2.1: The function $T^*$ is lower semi-continuous by Theorem 5.2 and Corollary 5.6 in [7] with $X_{free} = K$ and $X_{target} = C$. Recall $v^* = \Psi(T^*)$, and**
Ψ is continuous and bijective. This in conjunction with the lower semi-continuity of $T^*$ implies that of $v^*$.

Now we proceed to verify the contraction property of $F_\rho$. Choose any $x \in X_d \setminus B(\lambda_{\text{goal}}, Mh + d)$. Let $(u(x), w(x))$ (resp. $(\bar{u}(x), \bar{w}(x))$) be a solution to attain the minimum and the maximum in $F_\rho \circ v(x)$ (resp. $F_\rho \circ v_{h,d}(x)$) respectively. Let $Y(x) \triangleq x + h f(x, u(x), w(x))$ and $\bar{Y}(x) \triangleq x + h f(x, \bar{u}(x), \bar{w}(x))$. Then we have

$$
\begin{aligned}
\min_{y \in B(Y(x), \alpha) \cap X_d} v(y) &\leq \min_{y \in B(Y(x), \alpha) \cap X_d} v(x), \\
\min_{z \in B(\bar{Y}(x), \alpha) \cap X_d} v^*(z) &\leq \min_{z \in B(\bar{Y}(x), \alpha) \cap X_d} v^*(z).
\end{aligned}
$$

(9)

By following the same lines in (T1) of Lemma 7.1 one can show that

$$
\begin{aligned}
\| F_\rho \circ v(x) - F_\rho \circ v^*(x) \|_{X_d} &\leq e^{-(h-d)} \times \max \{ \| v(y) - \min_{y \in B(\bar{Y}(x), \alpha) \cap X_d} v^*(z) \|, \\
\| \min_{y \in B(\bar{Y}(x), \alpha) \cap X_d} v(y) - v^*(z) \| \}. 
\end{aligned}
$$

(10)

The relation (10) implies the following:

$$
\begin{aligned}
\| F_\rho \circ v(x) - F_\rho \circ v^*(x) \|_{X_d} &\leq e^{-(h-d)} \times \min_{y \in B(\bar{Y}(x), \alpha) \cap X_d} v(x). 
\end{aligned}
$$

(11)

On the other hand, one can see the following:

$$
\| F_\rho \circ v - F_\rho \circ v_{h,d} \|_{B(\lambda_{\text{goal}}, Mh + d)} = 0.
$$

(12)

Combine (11) and (12), and it renders the following:

$$
\begin{aligned}
\| F_\rho \circ v \|_{X_d} - \| F_\rho \circ v_{h,d} \|_{X_d} &\leq e^{-(h-d)} \times \min_{y \in B(\bar{Y}(x), \alpha) \cap X_d} v^*(y) \|_{X_d} \\
\| v - v_{h,d} \|_{X_d} &\leq e^{-(h-d)} \| v - v_{h,d} \|_{X_d}.
\end{aligned}
$$

(13)

where the first and third inequalities are direct results of [4] in Lemma 7.1. We establish the result by noting the fixed point property of $F_\rho \circ v_{h,d}(x) = v_{h,d}(x)$.

It is noticed that $F_\rho$ being a contraction mapping will play an important role in our subsequent analysis. Since $T^*(x)$ is potentially infinite, then $G_\rho$ in [7] may not be a contraction mapping.

B. Preliminary results for Lemma 3.1 and Theorem 3.1

We turn our attention to the iGame Algorithm in this section. Before showing Theorem 3.1 we provide a set of instrumental results.

1) Sample densities: The following lemma characterizes the decreasing rate of $d_n$ under the uniform sampling through a pair of lower and upper bounds for $d_n$.

Lemma 7.2: The following properties hold with probability one:

- (S1) There is $N_\alpha \geq 1$ such that $d_n \leq D_{\alpha}(\log \frac{n}{\alpha})^{-\frac{1}{\nu}}$ for all $n \geq N_\alpha$ with probability one.
- (S2) There is a pair of $n_\alpha \geq 1$ and $d_\alpha > 0$ such that $d_n \geq d_\alpha(\frac{1}{\alpha})^{-\frac{1}{\nu}}$ surely for all $n \geq n_\alpha$.

Proof: Define the event $E_n$ that $d_n > D_{\alpha}(\log \frac{n}{\alpha})^{-\frac{1}{\nu}}$; i.e., there is $\bar{z}_n \in X_{\text{free}}$ such that $B(\bar{z}_n, D_{\alpha}(\log \frac{n}{\alpha})^{-\frac{1}{\nu}}) \cap S_n = \emptyset$.

Note that the Lebesgue measure of the ball $B(\bar{z}_n, D_{\alpha}(\log \frac{n}{\alpha})^{-\frac{1}{\nu}})$ is $\mu(B(\bar{z}_n, D_{\alpha}(\log \frac{n}{\alpha})^{-\frac{1}{\nu}})) = C_N(D_{\alpha}(\log \frac{n}{\alpha})^{-\frac{1}{\nu}})^N$. With this, we can estimate $P(E_n)$ as follows:

$$
P(E_n) = \left(1 - \frac{\mu(B(\bar{z}_n, D_{\alpha}(\log \frac{n}{\alpha})^{-\frac{1}{\nu}}))}{\mu(X_{\text{free}})}\right)^{-n} \\
= \left(1 - C_N D_{\alpha}^{-\frac{N}{\nu}} \log \frac{n}{\alpha}\right)^{-n} \leq e^{-\gamma \frac{n-1}{\alpha} + \frac{n}{\alpha}} \leq n^{-\gamma + \frac{1}{\nu}}.
$$

Since $\gamma > 2$, then $\sum_{n=1}^{\infty} P(E_n) < +\infty$. By the Borel-Cantelli lemma in [22], we have $P(\lim \sup_{n \to \infty} E_n) = 0$. It establishes (S1).

We now move to show (S2) by contradiction. Assume that for any pair of $n_\alpha \geq 1$ and $d_\alpha > 0$, there is $K \geq n_\alpha$ such that $d_K < d_\alpha(\frac{1}{K})^{-\frac{1}{\nu}}$. By the definition of $d_K$, we know that $\cup_{\ell=1}^{K} B(x_{\ell}, d_{\ell}) \supseteq X$. It is noticed that

$$
\mu(\cup_{\ell=1}^{K} B(x_{\ell}, d_{\ell})) = \sum_{\ell=1}^{K} \mu(B(x_{\ell}, d_{\ell}(\frac{1}{K})^{-\frac{1}{\nu}})) = K C_N^\alpha(\frac{1}{K}^{-\frac{1}{\nu}})^N = C_N d_{\alpha}^N.
$$

The above holds for any $d_n > 0$. Choose $d_\alpha$ such that $C_N d_{\alpha}^N < \mu(X)$. It contradicts that $\cup_{\ell=1}^{K} B(x_{\ell}, d_{\ell}) \supseteq X$. We then reach the property of (S2).

With the aid of Lemma 7.2 the following lemma characterizes the non-summability of the sequence $\{\kappa_n\}$ by recalling $\kappa_n = h_n - d_n$.

Lemma 7.3: Suppose Assumption 3.1 holds, then the sequence $\{\kappa_n\}$ is not summable.

Proof: Recall that Lemma 7.2 proves that $d_n(\frac{1}{n})^{-\frac{1}{\nu}} \leq n \leq D_{\alpha}(\log \frac{n}{\alpha})^{-\frac{1}{\nu}}$ for all $n \geq \max\{n_\alpha, N\}$. Since $\frac{d_\alpha}{h_n} \to 0$, there is $n_\alpha \geq \max\{n_\alpha, N\}$ such that the following holds for $n \geq n_\alpha$:

$$
\kappa_n = h_n - d_n = h_n(1 - \frac{d_n}{h_n}) \geq \frac{h_n}{2} \geq \frac{1}{2}(d_\alpha(\frac{1}{n})^{-\frac{1}{\nu}})^{\frac{1}{\nu}}.
$$

Let $\phi_n = \frac{1}{2}(d_\alpha(\frac{1}{n})^{-\frac{1}{\nu}})^{\frac{1}{\nu}}$, then $\{\phi_n\}$ serves as a lower bound of $\{\kappa_n\}$ after a finite time; i.e., there is $N' \geq 1$ such that $\kappa_n \leq \phi_n$. $\forall n \geq N'$.

Let $\tau$ to be the smallest integer such that $\tau D \geq N'$. It is not difficult to see that $\{\phi_n\}_{n \in \Phi, n \geq \tau D}$ and thus $\{\kappa_n\}_{n \in \Phi}$ are not summable. We then reach the desired result.

The quantity $d_n$ in Lemma 7.2 serves as a measure of the sample density of $S_n$. The following lemma provides another characterization.

Lemma 7.4: Consider any pair of $x, y \in X$ with $\|x - y\| \leq \alpha_n$ and $x \neq y$. Then it holds that

$$
B(x, \alpha_n) \cap B(y, \alpha_n) \cap (S_n \setminus \{x, y\}) = \emptyset.
$$

(14)

Proof: We now distinguish three cases.
Case 1: $\|x - y\| \leq d_n$. There is $z \in B(y, d_n) \cap (S_n \setminus \{x, y\})$. Since $\alpha_n > 2d_n$, then $z \in B(x, \alpha_n)$ and (14) holds.

Case 2: $d_n < \|x - y\| \leq 2d_n$. There is $z \in B(x, d_n) \cap (S_n \setminus \{x, y\})$. Since $\alpha_n > 2d_n$, then $z \in B(y, \alpha_n)$ and (14) holds.

Case 3: $2d_n < \|x - y\| \leq \alpha_n$. Let $p = x + d_n(x - x)$. Since $\alpha_n \geq 2d_n$, then $B(p, d_n) \subseteq B(x, \alpha_n) \cap B(y, \alpha_n)$. Since $B(p, d_n) \cap S_n \neq \emptyset$ and $\{x, y\} \cap B(p, d_n) = \emptyset$, so (14) holds.

The combination of the above three cases establishes the lemma.

2) Convergence of fixed points: Based on (7), the following lemma shows that there is a unique fixed point $v_n^*$ of $F_n$. More importantly, $v_n^*$ provides a consistent approximation of $v^*$, and converges to $v^*$ pointwise.

**Lemma 7.5: Suppose that Assumption 2.1 holds.** The following properties hold:

(Q1) For each $n \geq 1$, there is a fixed point $v_n^*$ of $F_n$; i.e., $v_n^* = F_n \circ v_n^*$, where $v_n^* : S_n \to [0, 1]$;

(Q2) The sequence $v_n^*$ converges to $v^*$ pointwise; i.e., for any $x \in X_{\text{free}}$, it holds that

$$v^*(x) = \lim_{n \to +\infty} \min_{y \in B(x, d_n) \cap S_n} v_n^*(y).$$

**Proof:** It is easy to see that $h_n \to 0^+$ and $\frac{d_n}{\alpha_n} \to 0^+$. By following the same lines towards Corollary 5.6 in (7), one can show:

(R1) For each $n \geq 1$, there is a fixed point $T_n^* : G_n \to [0, 1]$;

(R2) The sequence $T_n^*$ converges to $T^*$ pointwise; i.e., for any $x \in X_{\text{free}}$, it holds that

$$T^*(x) = \lim_{n \to +\infty} \min_{y \in B(x, d_n) \cap S_n} T_n^*(y).$$

Given the fixed point $T_n^*$, let $v_n^* = \Psi \circ T_n^*$. The property (Q1) holds via verifying the following:

$$F_n \circ v_n^* = F_n \circ \Psi \circ T_n^* = F_n \circ \Psi \circ \Psi^{-1} \circ v_n^* = \Psi \circ G_n \circ \Psi^{-1} \circ v_n^* = \Psi \circ G_n \circ \Psi^{-1} \circ v_n^* = \Psi \circ G_n \circ \Psi^{-1} \circ v_n^* = \Psi \circ T_n^* = v_n^*,$$

where in the third equality we use $F_n = \Psi \circ G_n \circ \Psi^{-1}$, and in the fifth equality we use the fixed point property of $G_n \circ T_n^* = T_n^*$. One can easily verify (Q2) from (R2) by using the relation $v = \Psi \circ T$. ■

For the discrete fixed point $v_n^*-1$, we define the interpolation fixed point $\tilde{v}_{n-1}$: $S_{n-1} \to [0, 1]$ as follows:

$$\tilde{v}_{n-1}(x) = v_{n-1}(x), \quad x \in S_{n-1}, \quad \tilde{v}_{n-1}(y_n) = 1.$$

The following lemma shows the interpolation of fixed points is consistent.

**Lemma 7.6:** for any $x \in X$, it holds that

$$\min_{y \in B(x, \alpha_n) \cap S_n} \tilde{v}_{n-1}(y) = \min_{y \in B(x, \alpha_n) \cap S_{n-1}} v_n^*(y).$$

**Proof:** When $y_n \notin B(x, \alpha_n)$, then (15) holds. Now consider the case of $y_n \in B(x, \alpha_n)$. By Lemma 7.4 with $y = y_n$, we have $B(x, \alpha_n) \cap B(y_n, \alpha_n) \cap S_{n-1} \neq \emptyset$.

Recall $\tilde{v}_{n-1}(y_n) = 1 \geq \max_{x \in B(y_n, \alpha_n) \cap S_{n-1}} v_n^*(x)$. This implies that

$$\tilde{v}_{n-1}(y_n) \geq \min_{y \in B(x, \alpha_n) \cap S_n} \tilde{v}_{n-1}(y),$$

and thus

$$\tilde{v}_{n-1}(y_n) \geq \min_{y \in B(x, \alpha_n) \cap S_{n-1}} v_n^*(y) = \min_{y \in B(x, \alpha_n) \cap S_{n-1}} v_n^*(y).$$

(16)

The relation (16) implies (15).

In Lemma 7.5 we show the pointwise convergence of $v_n^*$ to $v^*$. The following lemma then shows that such convergence holds for the whole $X$.

**C. Proof for Lemma 3.7:**

By using the above intermediate lemmas, we now proceed to show Lemma 3.1

**Proof:** Since (15) holds for any $x \in X$, then we have

$$\| \min_{y \in B(x, \alpha_n) \cap S_n} \tilde{v}_{n-1}(y) - \min_{y \in B(x, \alpha_n) \cap S_{n-1}} v_n^*(y) \|_X = 0.$$

With this, we have

$$\| \min_{y \in B(x, \alpha_n) \cap S_n} \tilde{v}_{n-1}(y) - \min_{z \in B(x, \alpha_n) \cap S_n} v_n^*(z) \|_X = \| \min_{z \in B(x, \alpha_n) \cap S_n} \tilde{v}_{n-1}(y) - \min_{z \in B(x, \alpha_n) \cap S_n} v_n^*(z) \|_X.$$

(17)

By Lemma 7.3, we know that $\alpha_n \geq d_n - 1$. Then it follows from (17) and (T2) Lemma 7.1 that

$$\| \min_{y \in B(x, \alpha_n) \cap S_n} \tilde{v}_{n-1}(y) - \min_{z \in B(x, \alpha_n) \cap S_n} v_n^*(y) \|_X \leq \| \min_{z \in B(x, \alpha_n) \cap S_n} v_n^*(z) - \min_{z \in B(x, \alpha_n) \cap S_n} v_n^*(z) \|_X.$$

(18)

Combine (18), (Q2) in Lemmas 7.5 and we reach the desired result. ■

D. Proof for Theorem 3.7

We are now in the position to show Theorem 3.1

**Proof:** We first show (P1). By Theorem 7.1 we have:

$$\| v_n - v_n^* \|_K = \| F_n \circ \tilde{v}_{n-1} - F_n \circ v_n^* \|_K \leq \| F_n \circ \tilde{v}_{n-1} - F_n \circ v_n^* \|_S \leq e^{-\kappa_n} \| \min_{y \in B(x, \alpha_n) \cap S_n} \tilde{v}_{n-1}(y) - \min_{z \in B(x, \alpha_n) \cap S_n} v_n^*(z) \|_X \leq e^{-\kappa_n} \left( \| \min_{y \in B(x, \alpha_n) \cap S_n} \tilde{v}_{n-1}(y) - \min_{z \in B(x, \alpha_n) \cap S_n} v_n^*(z) \|_X + \| \min_{y \in B(x, \alpha_n) \cap S_n} \tilde{v}_{n-1}(y) - \min_{z \in B(x, \alpha_n) \cap S_n} v_n^*(z) \|_X \right).$$

(19)

Consider the first term on the right-hand side of (19). We have the following estimates:

$$\| \min_{y \in B(x, \alpha_n) \cap S_n} \tilde{v}_{n-1}(y) - \min_{z \in B(x, \alpha_n) \cap S_n} v_n^*(z) \|_X \leq \| \tilde{v}_{n-1} - v_n^* \|_S \leq \max \{ \| v_n - v_n^* \|_{S_{n-1}}, \| \tilde{v}_{n-1}(y_n) - \tilde{v}_{n-1}(y_n) \| \} \leq \| v_n - v_n^* \|_{S_{n-1}},$$

(20)
where the last inequality is an application of (5).

Substitute the relation of (20) into (19), and it renders that

\[
\|v_n - v_n^*\|_{K_n} \leq e^{-\kappa_n} \left(\|v_n-1 - v_{n-1}^*\|_{S_{n-1}} + \sum_{y \in B(x, \alpha_n) \cap S_n} \min_{z \in B(x, \alpha_n) \cap S_n} v_n^*(z) \right). \tag{21}
\]

For each \(x \in S_n \setminus \left( K_n \cup B(X_{\text{goal}}, Mh_n + d_n) \right) \), we then have the following:

\[
\|v_n(x) - v_n^*(x)\| = \left\| \min_{y \in B(x, \alpha_n) \cap S_{n-1}} v_{n-1}(y) - \min_{z \in B(x, \alpha_n) \cap S_n} v_n^*(z) \right\| \\
\leq \left\| \min_{y \in B(x, \alpha_n) \cap S_{n-1}} v_{n-1}(y) - v_{n-1}(z) \right\| \\
+ \left\| v_{n-1}(z) - \min_{z \in B(x, \alpha_n) \cap S_n} v_n^*(z) \right\| \\
\leq \|v_{n-1} - v_{n-1}^*\|_{S_{n-1}} \\
+ \left\| \min_{y \in B(x, \alpha_n) \cap S_{n-1}} v_{n-1}(y) - v_{n-1}(z) \right\| \\
+ \left\| v_{n-1}(z) - \min_{z \in B(x, \alpha_n) \cap S_n} v_n^*(z) \right\|, \tag{22}
\]

where we apply Lemma 7.1 in the second inequality and Lemma 7.6 in the last inequality.

For each \(x \in S_n \cap B(X_{\text{goal}}, Mh_n + d_n) \), we have

\[
v_n(x) = v_n^*(x). \tag{23}
\]

Recall Assumption 3.1 that \(S_n \subseteq \cup_{s=0}^{D} K_{n+s} \). With this property, we combine (21), (22), and (23), and reach the following:

\[
\|v_n(x) - v_n^*(x)\|_{S_n} \leq e^{-\kappa_n} \left(\|v_{n-D} - v_{n-D}^*\|_{S_{n-D-1}} + \gamma_{n-1} \right), \tag{24}
\]

where the sequence \(\{\gamma_n\}\) is defined in (2).

Denote \(\beta_n \triangleq \|v_n - v_n^*\|_{S_n}\). It follows from (24) that

\[
\beta_{n+1} \leq e^{-\kappa_{n+1}} (\beta_{n-D} + \gamma_n). \tag{25}
\]

Denote by \(\xi_n \triangleq \frac{\beta_n}{\gamma_n}\). The following is derived from (25) for \(n \geq D\):

\[
\xi_{n+1} = \frac{\beta_{n+1}}{\gamma_{n+1}} \leq \left(\frac{\beta_{n-D}}{\gamma_{n-D}}\right)^{1-\epsilon} e^{-\kappa_{n+1}} \left(\frac{\beta_{n-D}}{\gamma_{n-D}} + \frac{\gamma_n}{\gamma_{n-D}} \right) \\
\leq C(D+2)(1-\epsilon) e^{-\kappa_{n+1}} (\xi_{n-D} + \gamma_{n-D}^\epsilon). \tag{26}
\]

In Lemma 7.2, it is shown that \(d_s \left(\frac{1}{s}\right)^{\frac{1}{\alpha}} \leq d_n \leq D_s \left(\log \frac{n}{\alpha}\right)^{\frac{1}{\alpha}} \) for \(n \geq \max\{n_s, N_s\}\). Since \(\frac{d_s}{h_n} \to 0\), there is \(n_1 \geq \max\{n_s, N_s\}\) such that the following holds for \(n \geq n_1\):

\[
\kappa_n = h_n - d_n = h_n(1 - \frac{d_n}{h_n}) \geq \frac{h_n}{2} \geq \frac{\eta}{n^{(1+\epsilon)/\alpha}}, \tag{27}
\]

where \(\eta \triangleq \frac{1}{2} (d_s) \frac{1}{\alpha}\). Then, it follows from (26) and (27) that for \(n \geq n_1\):

\[
\xi_{n+1} \leq C(D+2)(1-\epsilon) e^{-\eta \frac{1}{n^{(1+\epsilon)/\alpha}}} (\xi_{n-D} + \gamma_{n-D}^\epsilon) \\
\leq C(D+2)(1-\epsilon) e^{-\eta \frac{1}{n^{\frac{1}{(1+\epsilon)/\alpha}}}} \xi_1 \leq C(D+2)(1-\epsilon) \sum_{\tau = \tau_1}^{n+1} e^{-\eta \frac{1}{n^{\frac{1}{(1+\epsilon)/\alpha}}} \gamma_{\tau}^\epsilon}. \tag{28}
\]

We now proceed to show that \(\xi_n \to 0\) via (28).

Let \(\delta_n \triangleq \gamma_n^\epsilon\). Since \(\frac{1}{n^{(1+\epsilon)/\alpha}} \in (0, 1)\), the harmonic sequence of \(\{\frac{1}{n^{(1+\epsilon)/\alpha}}\}_{s \geq 1}\) is not summable. So \(e^{-\eta \sum_{s=1}^{n+1} s^{-\frac{1}{n^{(1+\epsilon)/\alpha}}}} \to 0\) as \(n \to +\infty\). We now consider the term of \(\sum_{\tau = \tau_1}^{n+1} e^{-\eta \sum_{s=1}^{n+1} s^{-\frac{1}{n^{(1+\epsilon)/\alpha}}} \delta_s}\). Recall that \(\delta_s\) diminishes. Pick any \(\epsilon > 0\). There is \(n_2 \geq n_1\) such that \(\|\delta_s\| \leq \epsilon\) for all \(n \geq n_2\). So we have

\[
\sum_{\tau = \tau_1}^{n+1} e^{-\eta \sum_{s=1}^{n+1} s^{-\frac{1}{n^{(1+\epsilon)/\alpha}}}} \delta_s \\
\leq \sum_{\tau = \tau_1}^{n_2} e^{-\eta \sum_{s=1}^{n+1} s^{-\frac{1}{n^{(1+\epsilon)/\alpha}}}} \delta_s + \varepsilon \sum_{\tau = n_2+1}^{n+1} e^{-\eta \sum_{s=1}^{n+1} s^{-\frac{1}{n^{(1+\epsilon)/\alpha}}}}. \tag{29}
\]

Since \(\{\delta_s\}\) converges, so is uniformly bounded. Then we have the following for the first term on the right-hand side of (29) by taking \(n \to +\infty\):

\[
\sum_{\tau = \tau_1}^{n_2} e^{-\eta \sum_{s=1}^{n_2+1} s^{-\frac{1}{n^{(1+\epsilon)/\alpha}}}} \|\delta_s\| \to 0. \tag{30}
\]

We now consider the second term on the right-hand side of (29). Since \(\{\frac{1}{n^{(1+\epsilon)/\alpha}}\}\) is monotonically decreasing, then we have

\[
\sum_{s=\tau}^{n+1} s^{-\frac{1}{n^{(1+\epsilon)/\alpha}}} \geq \int_{\tau}^{n+1} x^{-\frac{1}{n^{(1+\epsilon)/\alpha}}} dx \\
= \frac{1}{1 - \frac{1}{N(1+\alpha)} - \tau^{1 - \frac{1}{N(1+\alpha)}}} + \frac{\tau^{1 - \frac{1}{N(1+\alpha)}}}{1 - \frac{1}{N(1+\alpha)}} - 1 \\
= \frac{1}{1 - \frac{1}{N(1+\alpha)}} (n + 1 - \tau)^{1 - \frac{1}{N(1+\alpha)}}. \tag{31}
\]
By (31), we have

\[
\sum_{\tau=n_{2}+1}^{n+1} e^{-\frac{\eta}{n+1} s - \frac{\eta}{n+1} s} \leq \sum_{\tau=n_{2}+1}^{n+1} e^{-\frac{\eta}{n+1} s - \frac{\eta}{n+1} s} \leq \sum_{\tau=0}^{+\infty} e^{-\frac{\eta}{n+1} s - \frac{\eta}{n+1} s} < +\infty.
\]

(32)

Substitute (30) and (32) into (28), and we have the following for all \( n \geq 1 \):

\[ \xi_{n+1} \leq O(\varepsilon). \]

Take the limit on both sides of the above relation, and it renders that

\[ \limsup_{n \to +\infty} \xi_{n} \leq O(\varepsilon), \]

and then

\[ \lim_{n \to +\infty} \xi_{n} = O(\varepsilon). \]

(33)

Since (33) holds for any \( \varepsilon > 0 \), we conclude that \( \xi_{n} = \frac{\beta}{\gamma_{n} - \epsilon} \to 0 \) and then the second property of (P1) holds. This further implies that

\[
\frac{1}{\gamma_{n}} \left\| \min_{y \in B(x, \alpha_{n}) \cap S_{n}} v_{n}(y) - \min_{z \in B(x, \alpha_{n}) \cap S_{n}} v_{n}^{*}(z) \right\|_{X} \leq \frac{\|v_{n} - v_{n}^{*}\|_{S_{n}}}{\gamma_{n} - \epsilon} \to 0,
\]

where Lemma 7.2 is invoked. We then establish the second property of (P1). The property (P2) is a direct result of the combination of (Q2) in Lemma 7.5 and (P1).