Dynamic programming using radial basis functions and Shepard approximations

Oliver Junge, Alex Schreiber

Fakultät für Mathematik
Technische Universität München

Workshop on algorithms for dynamical systems and Lyapunov functions

Reykjavik, July 2013

Problem

- discrete-time control system

$$x_{k+1} = f(x_k, u_k), \quad k = 0, 1, 2, \dots,$$

  $f : \Omega \times U \to \Omega$ continuous
- $\Omega \subset \mathbb{R}^d$ and $U \subset \mathbb{R}^m$ compact
- target set $T \subset \Omega$, compact
- goal: construct feedback $F : S \to U$, $S \subset \Omega$, such that for the closed loop system

$$x_{k+1} = f(x_k, F(x_k)), \quad x_k \in S,$$

  the target T is asymptotically stable.

Optimal control

- cost function $c : \Omega \times U \to [0, \infty)$ continuous, $c(x, u) \geq \delta > 0$ for $x \notin T$ and any $u \in U$.
- accumulated cost

$$J(x_0, (u_k)_k) = \sum_{k=0}^{\infty} c(x_k, u_k),$$

  with trajectory $(x_k)_k$ associated to $x_0 \in \Omega$ and $(u_k)_k \in U^{\mathbb{N}}$.
- optimal value function

$$V(x) = \inf_{(u_k)_k} J(x, (u_k)_k)$$

The Bellman equation

- V fulfills the Bellman equation

$$\begin{aligned} V(x) &= \inf_{u \in U} \{c(x,u) + V(f(x,u))\} \\ &=: L[V](x) \end{aligned}$$

  with boundary condition $V(T) = 0$.

- optimal feedback

$$F(x) = \operatorname*{argmin}_{u \in U} \{c(x,u) + V(f(x,u))\}$$

  (whenever the min exists)

Numerical treatment

- assume $V \in \mathcal{F}$
- approximation space $\mathcal{A} \subset \mathcal{F}$, $\dim(\mathcal{A}) < \infty$
- projection $\Pi : \mathcal{F} \to \mathcal{A}$
- discretized Bellman operator

$$\Pi \circ L : \mathcal{A} \to \mathcal{A}$$

- value iteration: choose $V^{(0)} \in \mathcal{A}$ with $V^{(0)}(T) = 0$,

$$V^{(n+1)} := \Pi \circ L[V^{(n)}], \quad n = 0, 1, \ldots$$

- typical $\mathcal{A}$: finite differences, finite elements (order p)
- problem: $\dim(\mathcal{A}) \sim \mathcal{O}(n^d)$ for error $\mathcal{O}(n^{-p})$

Nonlinear approximation

Theorem [Girosi, Anzellotti, '92]

If $f \in H^{s,2}(\mathbb{R}^d)$, $s > d/2$, we can find

- n coefficients $c_i \in \mathbb{R}$,
- n centers $x_i \in \mathbb{R}^d$,
- and n variances $\sigma_i > 0$ such that

$$\left\| f - \sum_{i=1}^{n} c_i e^{-\frac{\|x-x_i\|^2}{2\sigma_i^2}} \right\|_\infty^2 = \mathcal{O}(n^{-1}).$$

Scattered data interpolation

Problem

Given

- sites $X = \{x_1, \dots, x_N\} \subset \Omega \subset \mathbb{R}^d$
- data $f_1, \dots, f_N \in \mathbb{R}$,

find a function $a \in \mathcal{A}$ such that

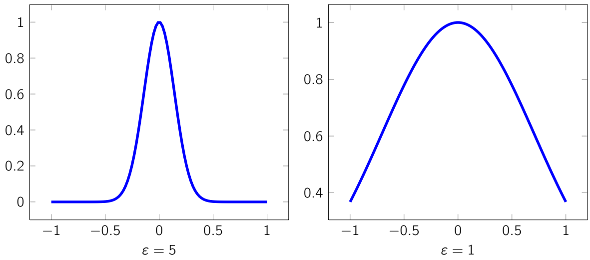
$$a(x_i) = f_i, \quad i = 1, \dots, N.$$

For $\mathcal{A} = span\{a_1, \dots, a_N\}$ we get

$$Ac = f, \quad \text{with } A_{ij} = a_j(x_i).$$

Radial basis functions

- radial basis functions $a(\,\cdot\,, x_j) = \varphi(\|\cdot - x_j\|_2)$
- examples:
  - Gaussian: $\varphi(r) = \exp(-r^2)$,
  - Wendland function: $\varphi(r) = (1-r)_+^4 \cdot (4r+1)$
- scaling: $a_j = a_j^\varepsilon = \varphi(\varepsilon\|\cdot - x_j\|)$



The Kruzkov transform

- problem: $V(x)$ increasing, but $\varphi(x)$ decreasing as $\|x\| \to \infty$
- Kruzkov transform: $V \mapsto \hat{V} = e^{-V(\cdot)}$
- Kruzkov-Bellman equation

  $$\hat{V}(x) = \sup_{u \in U} \{ e^{-c(x,u)} \cdot \hat{V}(f(x,u)) \} =: \hat{L}[V](x), \quad x \in \Omega \setminus T$$

  with boundary condition $\hat{V}(T) = 1$.
- under the assumption $c(x,u) \geq \delta > 0$ for $x \notin T$, the Kruzkov-Bellman operator $\hat{L}$ is a contraction on L^∞.

Dynamic programming using radial basis functions

- approximation space

$$\mathcal{A} = \mathcal{A}_{X,\varepsilon} = span\{\varphi(\varepsilon\| \cdot - x\|_2) : x \in X\}$$

- interpolation operator on X

$$\Pi : \mathcal{F} \to \mathcal{A}$$

- discretized Kruzkov-Bellman operator

$$\Pi \circ \hat{L} : \mathcal{A} \to \mathcal{A}$$

- value iteration: choose $\hat{V}^{(0)} \in \mathcal{A}$ with $\hat{V}^{(0)}(0) = 1$,

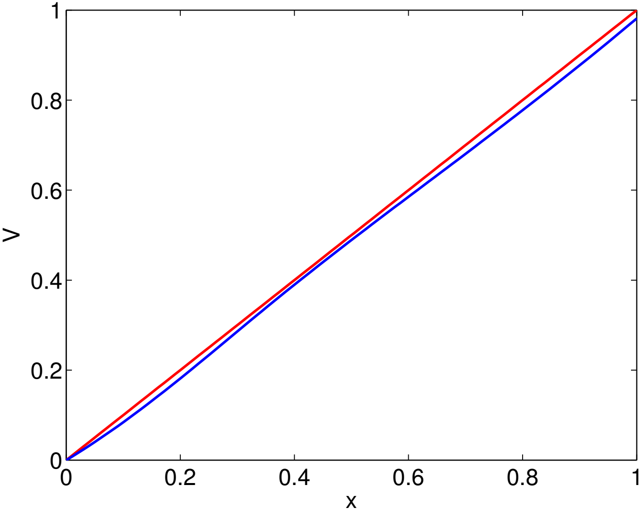
$$\hat{V}^{(n+1)} := \Pi \circ \hat{L}[\hat{V}^{(n)}], \quad n = 0, 1, \ldots$$

Example

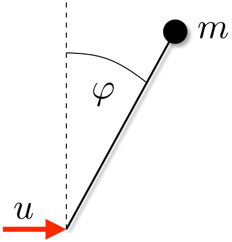
- $$x_{k+1} = (1 + au_k)x_k,$$
  $x_k \in X = [0, 1], u_k \in U = [-1, 1]$ and $a \in (0, 1)$ fixed.
- cost
  $$g(x, u) = ax.$$
- optimal control sequence: $\mathbf{u}(x) = (-1, -1, \ldots)$.
- value function: $V(x) = x$.

Example

$n = 3$



Example: inverted pendulum

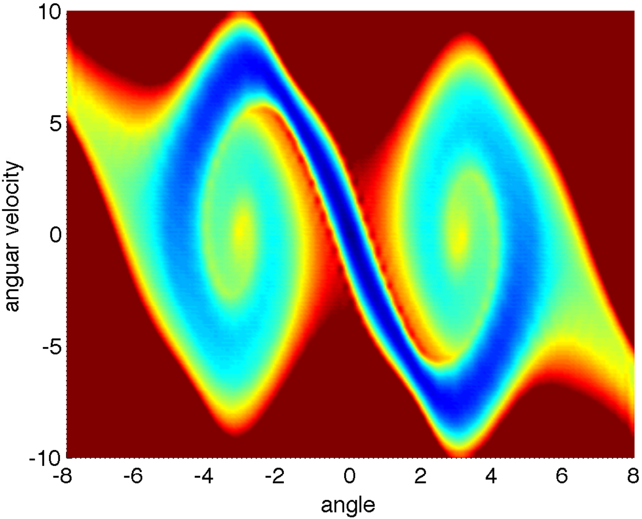


- state: $x = (\varphi, \dot{\varphi})$
- system: $f(x, u) = \Phi^T(x, u)$
- cost

$$g(x, u) = \int_0^T q_1 \varphi^2(t) + q_2 \dot{\varphi}^2(t)\, dt + T q_3 u^2$$

Example

$n = 150^2$



Weighted least squares

Problem

Given

- sites $X = \{x_1, \ldots, x_N\} \subset \Omega \subset \mathbb{R}^d$,
- data $f_1, \ldots, f_N \in \mathbb{R}$,
- approximation space $\mathcal{A} = span\{a_1, \ldots, a_m\}$, $m < N$,
- weight function $w : \Omega \to \mathbb{R}$ with associated scalar product $\langle f, g \rangle_w := \sum_{k=1}^N f(x_k) g(x_k) w(x_k)$ and induced norm

find a function $a \in \mathcal{A}$ such that

$$\|f - a\|_w \stackrel{!}{=} \min$$

Optimal coefficient vector c:

$$Gc = f_{\mathcal{A}}$$

with Gram matrix $G = (\langle a_i, a_j \rangle_w)_{ij}$ and $f_{\mathcal{A}} = (\langle f, a_j \rangle_w)_j$.

Moving least squares

Idea

In computing an approximation to the function $f : \Omega \to \mathbb{R}$ at $x \in \Omega$, only the values at sites $x_j \in X$ close to x should play a role.

- moving weight function $w : \Omega \times \Omega \to \mathbb{R}$
- $w(x, y)$ small for $\|x - y\|_2$ large
- inner product: $\langle f, g \rangle_{w(\cdot,x)} := \sum_{k=1}^{N} f(x_k) g(x_k) w(x_k, x)$
- moving least squares approximation a of data f is

$$a(x) = a^x(x),$$

  where $a^x \in \mathcal{A}$ is minimizing $\|f - a^x\|_{w(\cdot,x)}$.
- given by solving the Gram system $G^x c^x = f_{\mathcal{A}}^x$

Shepard's method

D. Shepard, *A two dimensional interpolation function for irregularly spaced data*, Proc. 23rd Nat. Conf. ACM, 1968.

- simply choose $\mathcal{A} = span\{1\}$
- Gram matrix $G^x = \langle 1, 1 \rangle_{w(\cdot,x)} = \sum_{i=1}^N w(x_i, x)$
- right hand side $f_{\mathcal{A}}^x = \langle f, 1 \rangle_{w(\cdot,x)} = \sum_{i=1}^N f(x_i) w(x_i, x)$
- thus we get

$$c^x = f^x / G^x = \sum_{i=1}^N f(x_i) \underbrace{\frac{w(x_i, x)}{\sum_{i=1}^N w(x_i, x)}}_{=: a_i(x)}$$

- and so the Shepard approximant is

$$Sf(x) = c^x \cdot 1 = \sum_{i=1}^N f(x_i) a_i(x)$$

- advantage: Shepard approximation requires no matrix solve

Shepard discretization of the Bellman equation

- approximation space

$$\mathcal{A} = span\left\{\frac{w(x_i,\cdot)}{\sum_{i=1}^{N} w(x_i,x)}, x_i \in X\right\}$$

- Shepard approximation operator

$$S : \mathcal{F} \to \mathcal{A}$$

- discretized Kruzkov-Bellman operator

$$S \circ \hat{L} : \mathcal{A} \to \mathcal{A}$$

- value iteration as usual

Convergence of the value iteration

- $f \mapsto Sf$ is linear,
- for each $x \in \Omega$, $Sf(x)$ is a convex combination of the values $f(x_1), \ldots, f(x_n)$, therefore
- the Shepard operator $S : (L^\infty, \|\cdot\|_\infty) \to (\mathcal{A}, \|\cdot\|_\infty)$ has norm 1,

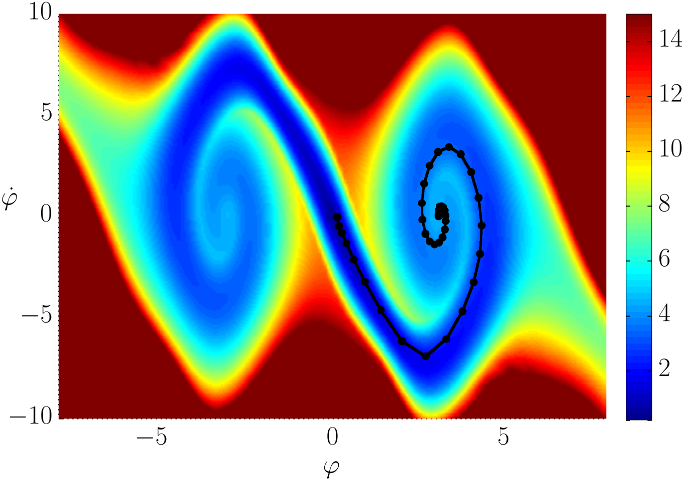
thus we get

Lemma

Value iteration with the discretized Kruzkov-Bellman operator $S \circ \hat{L} : (\mathcal{A}, \|\cdot\|_\infty) \to (\mathcal{A}, \|\cdot\|_\infty)$ converges to the unique fixed point of $S \circ \hat{L}$.

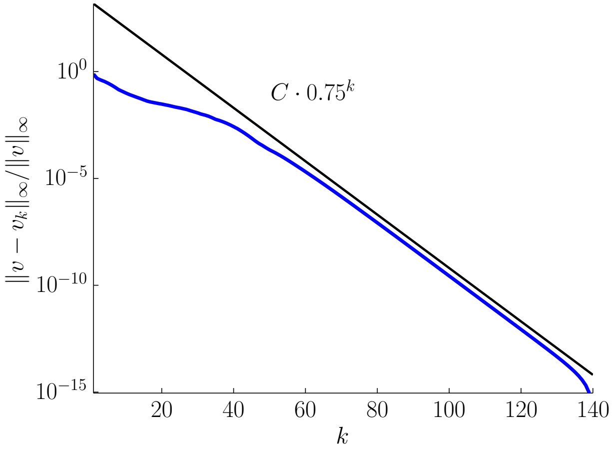
Example

inverted pendulum, $n = 10^4$ nodes



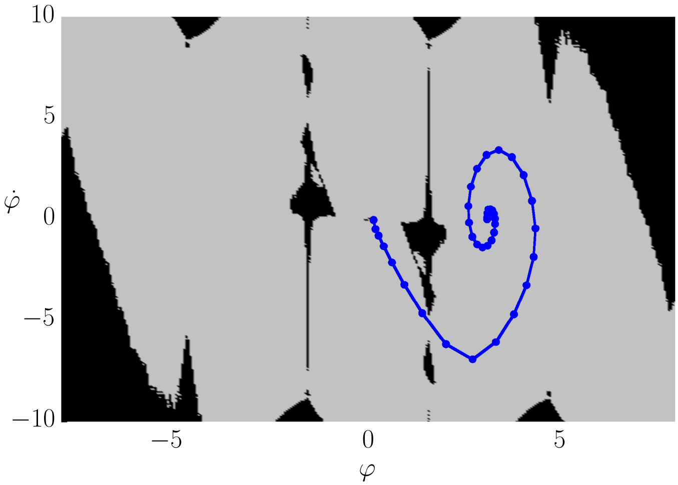
Example

inverted pendulum, value iteration history



Example

inverted pendulum, residual



Convergence for fill distance $\to 0$

fill distance of $X \subset \Omega$

$$h = h(X, \Omega) = \sup_{x \in \Omega} \min_{x_j \in X} \|x - x_j\|_2$$

If $f : \Omega \to \mathbb{R}$ is Lipschitz continuous with constant L then

$$\|f - Sf\|_\infty \leq CLh$$

for some constant $C > 0$.

Convergence for fill distance $\to 0$

- sequence $(X_n)_n$ of nodes sets, $X_n \subset \Omega$, fill distances h_n, Shepard operators S_n,
- $K < 1$ contraction constant of $\hat{L}$,
- $\hat{V}$ fixed point of $\hat{L}$, $\hat{V}_n$ fixed point of $S_n \circ \hat{L}$

Theorem

If $\hat{V}$ is Lipschitz continuous, then

$$\|\hat{V} - \hat{V}_n\|_\infty \leq \frac{CL}{1-K} h$$

Conclusion and future work

- multi-level
- greedy construction of sites
- complexity in dependence of d ("low discrepancy sites")
- approximate/relaxed dynamic programming