Approximating Fixpoints of Approximated Functions

Barbara König
Universität Duisburg-Essen

Joint work with Paolo Baldan, Sebastian Gurke, Tommaso Padoan, Florian Wittbold

Motivation

Wanted

Given a function $f\colon L \to L$ over a partially ordered set $(L, \sqsubseteq)$, we want to compute its least fixpoint x, i.e., the least $x \in L$ such that

$$f(x) = x$$

Motivation

Wanted

Given a function $f\colon L \to L$ over a partially ordered set $(L, \sqsubseteq)$, we want to compute its least fixpoint x, i.e., the least $x \in L$ such that

$$f(x) = x$$

- There are many results concerning fixpoints: Banach, Knaster-Tarski, Kleene, ...

Motivation

Wanted

Given a function $f \colon L \to L$ over a partially ordered set $(L, \sqsubseteq)$, we want to compute its least fixpoint x, i.e., the least $x \in L$ such that

$$f(x) = x$$

- There are many results concerning fixpoints: Banach, Knaster-Tarski, Kleene, ...
- But: what if f is a function that can only be approximated?
  For instance: a function over the reals, involving probabilities that can only be estimated.
  Which fixpoint iterations still work? For which types of functions? Do we need new techniques?

Fixpoint Theory

We are interested in techniques for solving fixpoint equations.

Least and greatest fixpoints

Solve the equation given as $f(x) = x$
where

- $f: L \to L$ is a monotone function over a complete lattice $(L, \sqsubseteq)$
- μf denotes the least and νf the greatest fixpoint of f

Applications in concurrency theory (behavioural equivalences and metrics), model checking (μ-calculus), program analysis (dataflow analysis), games (computation of value vectors and strategies), ...

Fixpoint Theory

Solution techniques

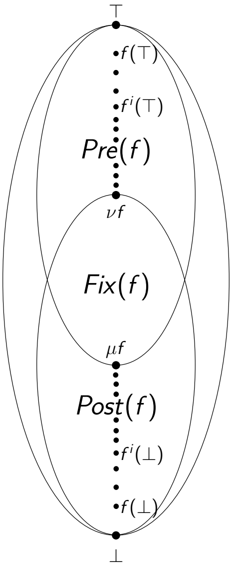
- The Knaster-Tarski theorem guarantees the existence of least and greatest fixpoints for monotone functions

Fixpoint Theory

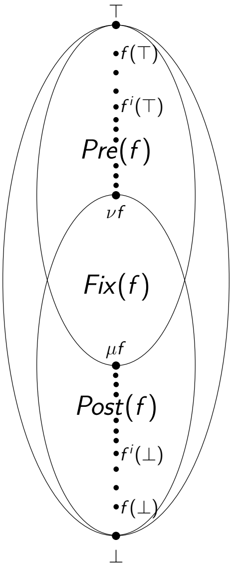
Solution techniques

- The Knaster-Tarski theorem guarantees the existence of least and greatest fixpoints for monotone functions
- Kleene iteration: whenever f is (co-)continuous
  - Least fixpoint: $\mu f = \bigsqcup_{i \in \mathbb{N}} f^i(\bot)$
  - Greatest fixpoint: $\nu f = \bigsqcap_{i \in \mathbb{N}} f^i(\top)$

Fixpoint Theory

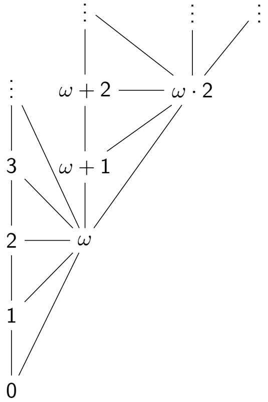


Fixpoint Theory



If f is *not* (co-)continuous:

$\rightsquigarrow$ Kleene iteration over the ordinals (beyond ω)



Fixpoint Theory

(Power) contractions and non-expansive maps

Let (X, d) be a metric space (with metric $d\colon X \times X \to \mathbb{R}_0^+$). Then $f\colon X \to X$ is a contraction, whenever there exists $0 \le q < 1$ such that:

$$d(f(x), f(y)) \le q \cdot d(x, y) \qquad \text{for all } x, y \in X$$

The function f is called non-expansive if this holds for $q = 1$.
It is a power contraction if there exists $n \in \mathbb{N}$ such that f^n is a contraction.

Fixpoint Theory

(Power) contractions and non-expansive maps

Let (X, d) be a metric space (with metric $d: X \times X \to \mathbb{R}_0^+$). Then $f: X \to X$ is a contraction, whenever there exists $0 \le q < 1$ such that:

$$d(f(x), f(y)) \le q \cdot d(x, y) \qquad \text{for all } x, y \in X$$

The function f is called non-expansive if this holds for $q = 1$.
It is a power contraction if there exists $n \in \mathbb{N}$ such that f^n is a contraction.

Example: $X = [0, c]^d$ (for $c > 0$) is a partially ordered complete metric space (and a complete lattice). We use the supremum distance:

$$d((x_1, \dots, x_d), (y_1, \dots, y_d)) = \max_i |x_i - y_i|$$

Fixpoint Theory

Banach Fixpoint Theorem

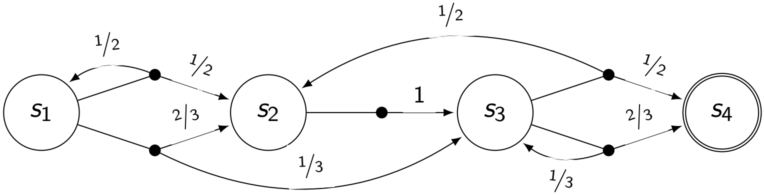
Let (X, d) be a non-empty complete metric space and let $f\colon X \to X$ be a (power) contraction. Then f has a unique fixpoint $x^* = \mu f = \nu f$. From any starting point fixpoint iteration converges to x^*.

Introducing Approximation: Applications

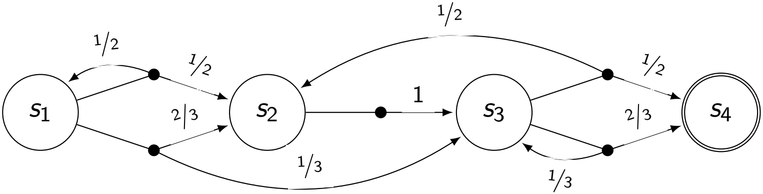
We discuss two applications where approximated functions play a role:

- Model-based reinforcement learning for Markov decision processes (MDPs)
- Model-checking quantitative μ-calculi

Markov Decision Processes (MDPs)



Markov Decision Processes (MDPs)



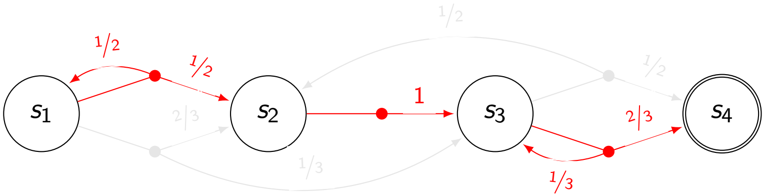
Markov decision process

A Markov decision process (MDP) is a tuple (S, T) where

- S is a finite set of *states* and
- $T : S \to \mathcal{P}_f(\mathcal{D}(S))$ is a *transition* function.

We let $T(s)$ be indexed over (pairwise disjoint) sets $A(s)$ of *actions*, writing $F = \{s \in S \mid A(s) = \emptyset\}, A = \bigcup_{s \in S} A(s)$, and $T(s' \mid s, a)$ is the probability of going from s to s' when a is chosen.

Markov Decision Processes (MDPs)

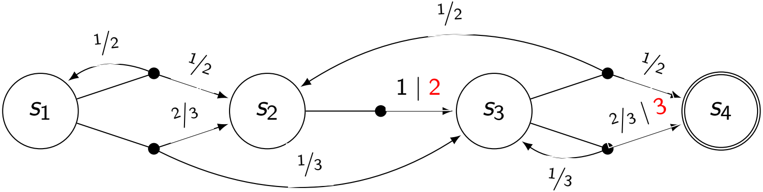


Definition (Policy)

Given an MDP $M = (S, T)$, a policy is a function $\pi : S \setminus F \to A$ with $\pi(s) \in A(s)$.

A policy defines the Markov chain $M^\pi = (S, T^\pi)$ where $T^\pi(s' \mid s) = T(s' \mid s, \pi(s))$.

Markov Decision Processes (MDPs)



Possible Objectives for the agent:

- Reachability / avoidance objectives
- Objectives given in (temporal) logic
- Collect reward given by (step-wise) reward function

$$R : S \times A \times S \to \mathbb{R}$$

Markov Decision Processes (MDPs)

Fix an MDP with states $S = \{1, \dots, d\}$. The least fixpoint of the following function $f : [0, c]^d \to [0, c]^d$ gives the expected reward for each state (Bellman optimality operator):

$$f(v)(s) = \max_{a \in A(s)} \sum_{s' \in S} T(s' \mid s, a) \cdot \big(R(s, a, s') + \gamma v(s')\big)$$

where $\gamma \in (0, 1]$ is a discount factor. Note that $\gamma < 1$ makes f contractive.

Markov Decision Processes (MDPs)

Fix an MDP with states $S = \{1, \ldots, d\}$. The least fixpoint of the following function $f\colon [0,c]^d \to [0,c]^d$ gives the expected reward for each state (Bellman optimality operator):

$$f(v)(s) = \max_{a \in A(s)} \sum_{s' \in S} T(s' \mid s, a) \cdot \big(R(s, a, s') + \gamma v(s')\big)$$

where $\gamma \in (0,1]$ is a discount factor. Note that $\gamma < 1$ makes f contractive.

We assume that the expected reward is bounded by c (more about this later).

Markov Decision Processes (MDPs)

Question: How to determine μf if the probabilities (given by T) and possibly the rewards (given by R) are not known precisely, but can only be approximated by sampling via interacting with the MDP?

Markov Decision Processes (MDPs)

Question: How to determine μf if the probabilities (given by T) and possibly the rewards (given by R) are not known precisely, but can only be approximated by sampling via interacting with the MDP?

This is exactly the question in reinforcement learning that synthesizes strategies for an MDP with unknown parameters while exploring it (Q-Learning, SARSA, Dyna, . . .).

But: Reinforcement learning typically concentrates on the contractive case ($\gamma < 1$), where fixpoints are unique and errors made in early stages are removed by the contraction.

Markov Decision Processes (MDPs)

Question: How to determine μf if the probabilities (given by T) and possibly the rewards (given by R) are not known precisely, but can only be approximated by sampling via interacting with the MDP?

This is exactly the question in reinforcement learning that synthesizes strategies for an MDP with unknown parameters while exploring it (Q-Learning, SARSA, Dyna, ...).

But: Reinforcement learning typically concentrates on the contractive case ($\gamma < 1$), where fixpoints are unique and errors made in early stages are removed by the contraction.

Here: We address the non-contractive case ($\gamma = 1$) and concentrate on model-based reinforcement learning (we construct a model of the MDP while exploring it).

Quantitative μ-Calculi

Quantitative μ-calculus (Huth/Kwiatkowska, Mio/Simpson)

$$\begin{aligned}\varphi \quad ::= \quad & \mathbf{1} \mid \mathbf{0} \mid x \mid p \mid r \cdot \varphi \mid \max\{\varphi, \varphi'\} \mid \min\{\varphi, \varphi'\} \mid \\ & \Diamond\varphi \mid \Box\varphi \mid \mu x.\varphi \mid \nu x.\varphi\end{aligned}$$

where $x \in PVar$ is a propositional variable, $p \in Prop$ is a propositional symbol.

Quantitative μ-Calculi

Quantitative μ-calculus (Huth/Kwiatkowska, Mio/Simpson)

$$\varphi \quad ::= \quad \mathbf{1} \mid \mathbf{0} \mid x \mid p \mid r \cdot \varphi \mid \max\{\varphi, \varphi'\} \mid \min\{\varphi, \varphi'\} \mid \Diamond\varphi \mid \Box\varphi \mid \mu x.\varphi \mid \nu x.\varphi$$

where $x \in PVar$ is a propositional variable, $p \in Prop$ is a propositional symbol.

Such formulas φ can be evaluated on MDPs, given an environment $\rho\colon Prop \cup PVar \to [0,1]^S$, resulting in $[\![\varphi]\!]_\rho\colon S \to [0,1]$.

- $[\![\Diamond\varphi]\!]_\rho(s) = \max_{a \in A(s)} \sum_{s' \in S} T(s' \mid s, a) \cdot [\![\varphi]\!]_\rho(s')$
- $[\![\mu x.\varphi]\!]_\rho = \mu(\lambda v.[\![\varphi]\!]_{\rho \cup [x \mapsto v]})$
  Least and greatest fixpoints can be arbitrarily nested.

Quantitative μ-Calculi

Questions:

- How to model-check when the MDP can only be approximated?

Quantitative μ-Calculi

Questions:

- How to model-check when the MDP can only be approximated?
- Even if the MDP is known exactly: if we allow arbitrary (non-expansive) operators, fixpoints can only be approximated. Hence computations of outer fixpoints have to deal with approximated functions.

  (For Łukasiewicz μ-calculi resulting in piecewise linear functions there is an exact technique by Petković/Simpson.)

Introducing Approximation

We do not want to resort to (power) contractions, where approximation is (relatively) harmless. Instead, we concentrate on the non-expansive case:

Task

Given a sequence of monotone and non-expansive (wrt. supremum distance) functions $f_1, f_2, f_3, \ldots : [0, c]^d \to [0, c]^d$ that (uniformly) converges to $f : [0, c]^d \to [0, c]^d$.
Compute a sequence $x_0, x_1, x_2, \ldots$ that converges to μf.

Introducing Approximation

We do not want to resort to (power) contractions, where approximation is (relatively) harmless. Instead, we concentrate on the non-expansive case:

Task

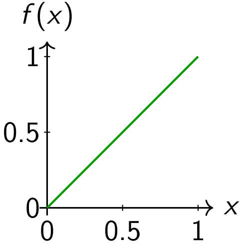
Given a sequence of monotone and non-expansive (wrt. supremum distance) functions $f_1, f_2, f_3, \ldots : [0, c]^d \to [0, c]^d$ that (uniformly) converges to $f : [0, c]^d \to [0, c]^d$.
Compute a sequence $x_0, x_1, x_2, \ldots$ that converges to μf.

Remarks:

- In a compact space, uniform convergence for non-expansive functions follows from pointwise convergence.
- Non-expansiveness wrt. supremum distance covers many interesting cases: termination probabilities in Markov chains, MDPs, stochastic games, behavioural metrics, ...

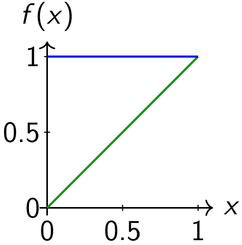
Introducing Approximation

$f, f_n \colon [0,1] \to [0,1]$ with $f_n(x) = 1/n + (1 - 1/n) \cdot x$, $f(x) = x$



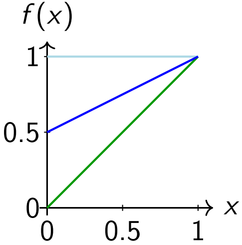
Introducing Approximation

$f, f_n\colon [0,1] \to [0,1]$ with $f_n(x) = 1/n + (1 - 1/n) \cdot x,\ f(x) = x$



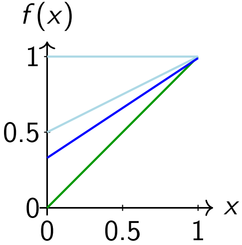
Introducing Approximation

$f, f_n \colon [0,1] \to [0,1]$ with $f_n(x) = 1/n + (1 - 1/n) \cdot x$, $f(x) = x$



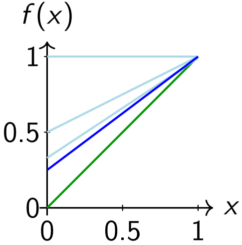
Introducing Approximation

$f, f_n \colon [0,1] \to [0,1]$ with $f_n(x) = 1/n + (1 - 1/n) \cdot x$, $f(x) = x$



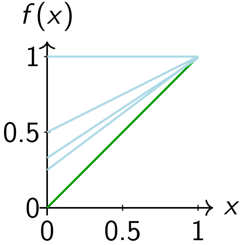
Introducing Approximation

$f, f_n \colon [0,1] \to [0,1]$ with $f_n(x) = 1/n + (1 - 1/n) \cdot x$, $f(x) = x$



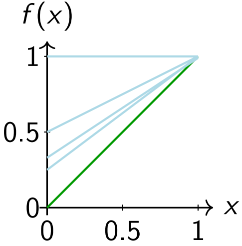
Introducing Approximation

$f, f_n \colon [0,1] \to [0,1]$ with $f_n(x) = 1/n + (1 - 1/n) \cdot x$, $f(x) = x$



Introducing Approximation

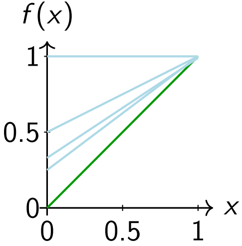
$f, f_n \colon [0,1] \to [0,1]$ with $f_n(x) = 1/n + (1 - 1/n) \cdot x$, $f(x) = x$



$\mu f_n = 1$ (for all n), while $\mu f = 0$.
Hence $\lim_{n\to\infty} \mu f_n = 1 \neq 0 = \mu f$.

Introducing Approximation

$f, f_n \colon [0,1] \to [0,1]$ with $f_n(x) = 1/n + (1 - 1/n) \cdot x$, $f(x) = x$

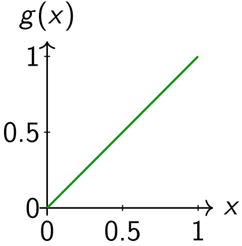


$\mu f_n = 1$ (for all n), while $\mu f = 0$.
Hence $\lim_{n\to\infty} \mu f_n = 1 \neq 0 = \mu f$.

The fixpoints of the approximations need not converge to the fixpoint of f.

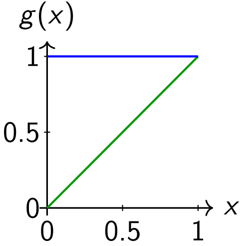
Introducing Approximation

$g, g_n\colon [0,1] \to [0,1]$ with $g_n(x) = {}^1\!/_n$ if $x \leq {}^1\!/_n$, $g_n(x) = x$ otherwise, $g(x) = x$



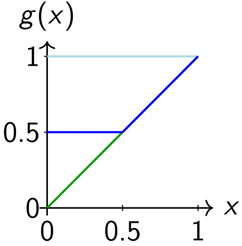
Introducing Approximation

$g, g_n\colon [0,1] \to [0,1]$ with $g_n(x) = 1/n$ if $x \le 1/n$, $g_n(x) = x$ otherwise, $g(x) = x$



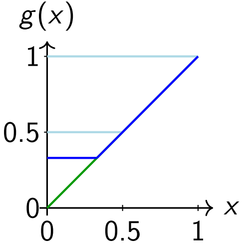
Introducing Approximation

$g, g_n \colon [0,1] \to [0,1]$ with $g_n(x) = {}^1\!/\!_n$ if $x \leq {}^1\!/\!_n$, $g_n(x) = x$ otherwise, $g(x) = x$



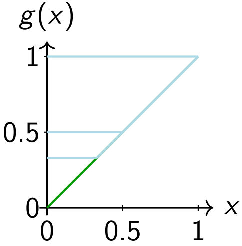
Introducing Approximation

$g, g_n \colon [0,1] \to [0,1]$ with $g_n(x) = 1/n$ if $x \le 1/n$, $g_n(x) = x$ otherwise, $g(x) = x$



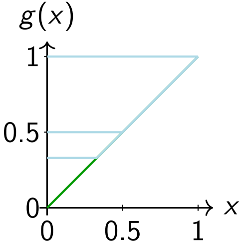
Introducing Approximation

$g, g_n \colon [0,1] \to [0,1]$ with $g_n(x) = 1/n$ if $x \leq 1/n$, $g_n(x) = x$ otherwise, $g(x) = x$



Introducing Approximation

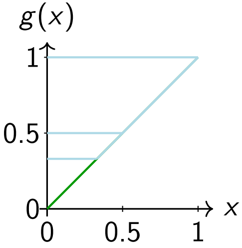
$g, g_n \colon [0,1] \to [0,1]$ with $g_n(x) = 1/n$ if $x \leq 1/n$, $g_n(x) = x$ otherwise, $g(x) = x$



In this case $\lim_{n\to\infty} \mu f_n = 0 = \mu f$.

Introducing Approximation

$g, g_n \colon [0,1] \to [0,1]$ with $g_n(x) = 1/n$ if $x \leq 1/n$, $g_n(x) = x$ otherwise, $g(x) = x$



In this case $\lim_{n\to\infty} \mu f_n = 0 = \mu f$.

But: Starting a Kleene iteration with some f_n will over-estimate the least fixpoint and further iterations with functions f_m ($m > 0$) will never decrease it.

Dampened Mann Iteration

Inspired by a paper by Kim/Xu we now consider the following form of iteration:

$$x_{n+1} = (1 - \beta_n) \cdot \big(\alpha_n \cdot x_n + (1 - \alpha_n) \cdot f_n(x_n)\big)$$

which is

- a Mann iteration $\alpha_n \cdot x_n + (1 - \alpha_n) \cdot f_n(x_n)$
- with a dampening factor $1 - \beta_n$.

Dampened Mann Iteration

Inspired by a paper by Kim/Xu we now consider the following form of iteration:

$$x_{n+1} = (1-\beta_n)\cdot\big(\alpha_n \cdot x_n + (1-\alpha_n)\cdot f_n(x_n)\big)$$

which is

- a Mann iteration $\alpha_n \cdot x_n + (1-\alpha_n)\cdot f_n(x_n)$
- with a dampening factor $1-\beta_n$.

We assume:

1. $\lim_{n\to\infty} \alpha_n < 1$,
2. $\lim_{n\to\infty} \beta_n = 0$ and $\sum_{n=1}^{\infty} \beta_n = \infty$ (equivalently: $\prod_{i=1}^{n}(1-\beta_n) = 0$).
3. $\sum_{n=1}^{\infty} |\alpha_n - \alpha_{n+1}| < \infty$ and $\sum_{n=1}^{\infty} |\beta_n - \beta_{n+1}| < \infty$.

Canonical choices: $\beta_n = 1/n$ and $\alpha_n = 1/n$ or $\alpha_n = 0$.

Dampened Mann Iteration

$$x_{n+1} = (1-\beta_n) \cdot \big(\alpha_n \cdot x_n + (1-\alpha_n) \cdot f_n(x_n)\big)$$

Intuition:

- a dampening factor $1-\beta_n$: this converges to 1, but there is always enough "power" left to decrease a current over-estimation to the true least fixpoint.
- linear combination with α_n: provides extra flexibility and gives the option to generalize the results.

Dampened Mann Iteration

The sequence (x_n) converges to the correct solution μf from any starting point in the following cases:

Dampened Mann Iteration

The sequence (x_n) converges to the correct solution μf from any starting point in the following cases:

1. When iterating with the correct function, i.e. when $f_n = f$ for all n,

Dampened Mann Iteration

The sequence (x_n) converges to the correct solution μf from any starting point in the following cases:

1. When iterating with the correct function, i.e. when $f_n = f$ for all n,
2. when f is a power contraction and $f_n \to f$,

Dampened Mann Iteration

The sequence (x_n) converges to the correct solution μf from any starting point in the following cases:

1. When iterating with the correct function, i.e. when $f_n = f$ for all n,
2. when f is a power contraction and $f_n \to f$,
3. when $\mu f_n \to \mu f$ and $f_n \to f$ monotonically,

Dampened Mann Iteration

The sequence (x_n) converges to the correct solution μf from any starting point in the following cases:

1. When iterating with the correct function, i.e. when $f_n = f$ for all n,
2. when f is a power contraction and $f_n \to f$,
3. when $\mu f_n \to \mu f$ and $f_n \to f$ monotonically,
4. when $f_n \to f$ normally, i.e.

$$\sum \|f_n - f\|_\infty < \infty$$

Dampened Mann Iteration

The sequence (x_n) converges to the correct solution μf from any starting point in the following cases:

1. When iterating with the correct function, i.e. when $f_n = f$ for all n,
2. when f is a power contraction and $f_n \to f$,
3. when $\mu f_n \to \mu f$ and $f_n \to f$ monotonically,
4. when $f_n \to f$ normally, i.e.

$$\sum \|f_n - f\|_\infty < \infty$$

These conditions are satisfied for the second example above, but not for the first example and also not for MDPs.

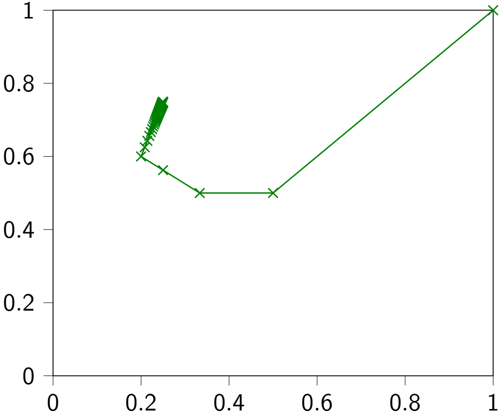
We also have a counterexample for the case $\mu f_n \to \mu f$.

Dampened Mann Iteration

Dampened Mann iteration (with $\alpha_n = 0$) for $f_n \colon [0,1]^2 \to [0,1]^2$

$$f_n(x_1, x_2) = (\max\{x_1, (1-y_1)x_1^n + y_1\}, \max\{x_2, (1-y_2)x_2^n + y_2\})$$

where $(y_1, y_2) = (1/4, 3/4)$.



Dampened Mann Iteration for MDPs

MDPs do not satisfy one of the above criteria, however the technique still works under some conditions. We first fix some terminology:

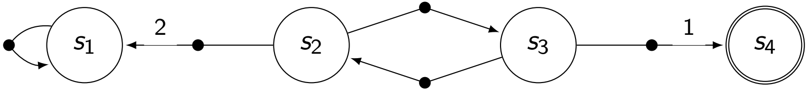
End component of an MDP

Let $M = (S, T)$ be an MDP. Then $E \subseteq S$ is an end component if

- the graph induced by E is strongly connected
- and for each $s \in E$ there exists $a \in A(s)$:

$$\{s' \mid T(s' \mid s, a) > 0\} \subseteq E$$

There exists a strategy that stays in the end component.



Dampened Mann Iteration for MDPs

A Markov chain is terminating if for any starting state its probability of eventually reaching a terminal state is 1. An MDP M is terminating if for all policies π, the induced Markov chain M^π is terminating.

For an MDP M, the following are equivalent:

- M is terminating.
- M has no end components.
- The corresponding function f_M is a power-contraction.

Dampened Mann Iteration for MDPs

In an MDP with end components it holds that:

- The expected reward is bounded if no reward is given when staying in an end component.
- Then all states in an end component have the same expected reward.
- The expected reward can be computed by merging all states in a maximal end component, obtaining a terminating MDP.

Dampened Mann Iteration for MDPs

In an MDP with end components it holds that:

- The expected reward is bounded if no reward is given when staying in an end component.
- Then all states in an end component have the same expected reward.
- The expected reward can be computed by merging all states in a maximal end component, obtaining a terminating MDP.

We can now deduce that dampened Mann iteration works for MDP sampling, assuming that rewards are only given outside of end components.

Dampened Mann Iteration for MDPs

In an MDP with end components it holds that:

- The expected reward is bounded if no reward is given when staying in an end component.
- Then all states in an end component have the same expected reward.
- The expected reward can be computed by merging all states in a maximal end component, obtaining a terminating MDP.

We can now deduce that dampened Mann iteration works for MDP sampling, assuming that rewards are only given outside of end components.

When sampling, the sequence of approximating MDPs will almost surely converge to the correct MDP. We can also assume that if for a transition probability $T(s' \mid s, a) = 0$, it will never be non-zero in an approximation.

Dampened Mann Iteration for MDPs

Lower bound

The expected reward for an MDP M is the supremum of the expected rewards for Markov chains M^π over all policies π.
In a Markov chain all states in an end component have reward 0.
Fixing the value to 0 in the states of an end component gives us a power contraction with the same fixpoint.
From this one can deduce that the true expected reward (μf) is always a lower bound for the outcome of the dampened Mann iteration.

Dampened Mann Iteration for MDPs

Upper bound

- Eventually, the approximating MDPs have the same maximal end components than the exact MDP.
- Each maximal end component can be merged to a singleton end component (with loop) to obtain an over-approximation.
- In a singleton end component we only make an error due to approximated probabilities if we leave the end component.
- The errors that are made behave similarly to rewards and hence the total error is bounded and vanishes with better approximations.

This argument can be extended to more general "MDP-like" functions.

Dampened Mann Iteration for Other Cases

There are still interesting cases, where neither of the sufficient conditions applies. For instance: stochastic games (MDPs enriched with a Min player, in addition to the usual Max player).

What can we do in these cases?

Dampened Mann Iteration for Other Cases

Idea: fix a subsequence of functions $f_{n_1}, f_{n_2}, f_{n_3}, \ldots$ that convergences normally (sum of the errors is bounded).
Intuition: perform enough sampling steps before the next iteration.

We can estimate how close we are to the exact function:

Hoeffding's inequality

$$\mathbb{P}\big[|T^n(s' \mid s, a) - T(s' \mid s, a)| > \varepsilon\big] \leq 2e^{-2\varepsilon^2/n}$$

Dampened Mann Iteration for Other Cases

Choose n_i such that

$$\mathbb{P}\left[\|f_{n_i} - f\| > \gamma_i\right] \leq \delta_i,$$

where $\sum_i \gamma_i < \infty$ and $\sum_i \delta_i < \infty$ (for instance $\gamma_i = \delta_i = 1/i^2$).

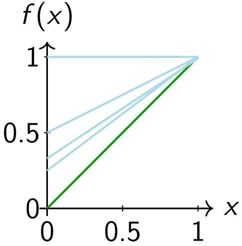
By the Borel-Cantelli Lemma we get that

$$\mathbb{P}\left[\|f_{n_i} - f\| > \gamma_i \text{ for infinitely many } i\right] = 0$$

Hence, almost surely we have $\|f_{n_i} - f\| \leq \gamma_i$ eventually and by our fixpoint results (normal convergence) the sequence produced by the algorithm converges to the solution vector of its input in that case.

Dampened Mann Iteration for Other Cases

Example: $f, f_n\colon [0,1] \to [0,1]$ with $f_n(x) = 1/n + (1 - 1/n) \cdot x$, $f(x) = x$



Perform dampened Mann iteration with the sequence $(f_{n^2})_n$.

This works, although the sequence of least fixpoints of these functions does not converge to the least fixpoint of f!

Conclusion

Implementation

We have implemented this form of iteration and obtained encouraging results. The runtime and accuracy after n steps are similar to computing μf_n by Kleene iteration.

Conclusion

Future Work

- Apply this to quantitative μ-calculi
  - Show that the sufficient critera are met
  - or estimate how close we are to the exact function.

Conclusion

Future Work

- Apply this to quantitative μ-calculi
  - Show that the sufficient critera are met
  - or estimate how close we are to the exact function.
- Approximating coalgebras

Conclusion

Future Work

- Apply this to quantitative μ-calculi
  - Show that the sufficient critera are met
  - or estimate how close we are to the exact function.
- Approximating coalgebras
- Chaotic iteration

Conclusion

Future Work

- Apply this to quantitative μ-calculi
  - Show that the sufficient critera are met
  - or estimate how close we are to the exact function.
- Approximating coalgebras
- Chaotic iteration
- What if the coefficients α_n converge to 1? (Mann iteration converging to the identity)

Conclusion

Future Work

- Apply this to quantitative μ-calculi
  - Show that the sufficient critera are met
  - or estimate how close we are to the exact function.
- Approximating coalgebras
- Chaotic iteration
- What if the coefficients α_n converge to 1? (Mann iteration converging to the identity)
- Model-free learning
  Idea: Sequence $f_1, f_2, \dots$ of functions approximates f in the limit-average: $\frac{1}{n}\sum_{i=1}^{n} f_i \to f$
  Aim: obtain model-free reinforcement learning algorithms as special cases