Featured Games

Uli Fahrenberg[1]   Axel Legay[2]

École Polytechnique, Palaiseau, France

Université Catholique de Louvain, Belgium

TASE 2021

Elevator Statement

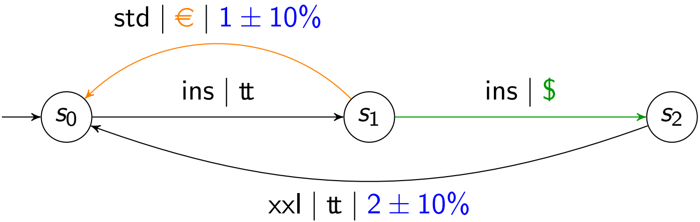
- Model checking is useful
- ... and often done by solving two-player games
- ... but hampered by state-space explosion

- For software product lines: featured model checking
- ... replacing the boolean codomain by the lattice of products
- ... but scant literature about featured games

$\Rightarrow$ Let's do this!

Introduction, by Way of Example

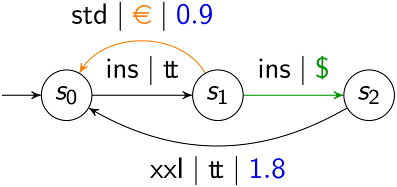
A coffee machine SPL, as a featured transition system (FTS)

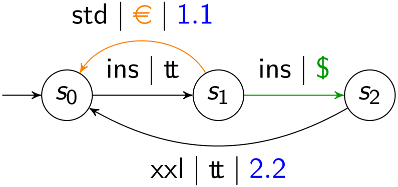


- two features, € and \$ $\Rightarrow$ four products: $\emptyset$, {€}, {\$}, {€, \$}
- energy annotations:
  - brewing std coffee uses $1 \pm 10\%$ enery units
  - brewing xxl coffee uses $2 \pm 10\%$ enery units
- Our concern: how robust is this, depending on the product?

More Coffee

Minimal and maximal energy consumption:





Example:

- infinite run $(\text{ins}, \text{std}, \text{ins}, \text{std}, \dots)$
- consumption difference $(0, 0.2, 0, 0.2, \dots)$
- standard trick: apply discounting; here $\lambda = 0.99$
- accumulated difference: $0 + \lambda \cdot 0.2 + \lambda^2 \cdot 0 + \lambda^3 \cdot 0.2 + \cdots = 9.95$

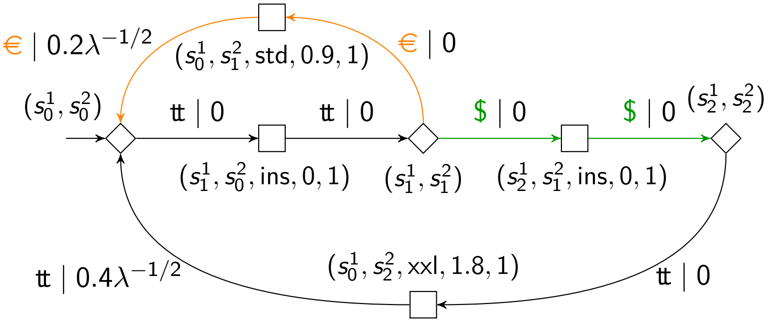
Robust Featured Coffee

- to describe general robustness of a model under the discounted semantics, use discounted bisimulation distance:

$$d(s_1, s_2) = \max \begin{cases} \max\limits_{s_1 \xrightarrow{a,x} t_1} \; \min\limits_{s_2 \xrightarrow{a,y} t_2} |x - y| + \lambda \cdot d(t_1, t_2) \\ \max\limits_{s_2 \xrightarrow{a,y} t_2} \; \min\limits_{s_1 \xrightarrow{a,x} t_1} |x - y| + \lambda \cdot d(t_1, t_2) \end{cases}$$

- to compute discounted bisimulation distance, use discounted games
- and now extend this to FTS

Discounted Coffee Games



Contributions & Related Work

- We introduce featured extensions of
  - reachability games
  - minimum reachability games
  - discounted games
  - energy games
  - parity games
- For each type, we show how to solve games by computing featured attractors
- For each type, we show how to compute optimal featured strategies
- Related work: ter Beek, van Loo, de Vink, Willemse: Family-based SPL model checking using parity games with variability, FASE 2020: featured μ-calculus for FTS; translation to featured parity games; but different algorithm for solving these

Featured Discounted Games

Showing one of our contributions in detail.

- Remember there are others ⌣̈
- Developments are quite similar for the different game types (reachability; minimum reachability; discounted; energy; parity)
- but no uniform setting exists

$\Rightarrow$ Future work

- See Gimbert, Zielonka: Games where you can play optimally without any memory, CONCUR 2005 and Bouyer, Le Roux, Oualhadj, Randour, Vandenhove: Games where you can play optimally with arena-independent finite memory, CONCUR 2020 for first steps in that direction

Discounted Games

- a weighted game structure: $G = (S_1, S_2, i, T)$
  - S_1, S_2 states of players 1 and 2
  - $S_1 \cap S_2 = \emptyset$; $S := S_1 \cup S_2$
  - $i \in S$ initial
  - $T \subseteq S \times \mathbb{Z} \times S$ transitions
- $\lambda \in]0, 1[$ discounting factor
- intuition:
  - players cooperate to create infinite path in G starting in i
  - when $s \in S_i$, player i determines next transition
  - value of infinite path $\pi = s_1 \xrightarrow{x_1} s_2 \xrightarrow{x_2} s_3 \xrightarrow{x_3} \cdots$:
    $$\text{val}(\pi) = x_1 + \lambda x_2 + \lambda^2 x_3 + \cdots$$
  - goal of player 1 is to create an infinite path with maximal value; player 2: minimal

Discounted Games, contd.

- $G = (S_1, S_2, i, T)$ weighted game structure, $\lambda \in]0,1[$ disc. factor
- players cooperate to create infinite path in G starting in i
- when $s \in S_i$, player i determines next transition
- value of infinite path $\pi = s_1 \xrightarrow{x_1} s_2 \xrightarrow{x_2} s_3 \xrightarrow{x_3} \cdots$:
$$\mathrm{val}(\pi) = x_1 + \lambda x_2 + \lambda^2 x_3 + \cdots$$
- goal of player 1: create an infinite path with maximal value
- configurations for player i: $\mathrm{Conf}_i = \{\pi \text{ finite path} \mid \mathrm{end}(\pi) \in S_i\}$
- strategies for player i: $\Theta_i = \{\theta : \mathrm{Conf}_i \to \mathbb{Z} \times S \mid$
$$\forall \pi \in \mathrm{Conf}_i : (\mathrm{end}(\pi), \theta(\pi)_1, \theta(\pi)_2) \in T\}$$
- outcome of strategy pair $\theta_1 \in \Theta_1$, $\theta_2 \in \Theta_2$: infinite path $\mathrm{out}(\theta_1, \theta_2) = s_1 \xrightarrow{x_1} s_2 \xrightarrow{x_2} s_3 \xrightarrow{x_3} \cdots$ given by $s_1 = i$ and
$$(x_k, s_{k+1}) = \begin{cases} \theta_1(s_1, \dots, s_k) & \text{if } s_k \in S_1 \\ \theta_2(s_1, \dots, s_k) & \text{if } s_k \in S_2 \end{cases}$$
- value of G: $\mathrm{val}(G) = \sup_{\theta_1 \in \Theta_1} \inf_{\theta_2 \in \Theta_2} \mathrm{val}(\mathrm{out}(\theta_1, \theta_2))$

Solving Discounted Games

Zwick, Paterson: The complexity of mean payoff games on graphs, TCS 1996:

- For $G = (S_1, S_2, i, T)$ a weighted game structure, define $\text{attr} : (S \to \mathbb{R}) \to (S \to \mathbb{R})$ by
$$\text{attr}(U)(s) = \begin{cases} \max_{s \xrightarrow{x} s'} x + \lambda U(s') & \text{if } s \in S_1 \\ \min_{s \xrightarrow{x} s'} x + \lambda U(s') & \text{if } s \in S_2 \end{cases}$$
- Theorem: The equation system $V = \text{attr}(V)$ has a unique solution $\text{attr}^* : S \to \mathbb{R}$, and $\text{val}(G) = \text{attr}^*(i)$
- Say that a strategy θ_1 is locally optimal if
$$\text{attr}^*(s) = \theta_1(s)_1 + \lambda \text{attr}^*(\theta_1(s)_2) \text{ for all } s \in S_1$$
- Theorem: Any locally optimal strategy is optimal, and any discounted game admits a locally optimal strategy. Also, locally optimal strategies are easy to compute.

Featured Discounted Games

- a featured weighted game structure: $G = (S_1, S_2, i, T, \gamma)$
  - $G = (S_1, S_2, i, T)$ a weighted game structure; $\gamma : T \to \mathbb{B}(N)$
  - N finite set of features; $\mathbb{B}(N)$ boolean expressions over N
- the projection of G to product $p \subseteq N$: the weighted game structure $\text{proj}_p(G) = (S_1, S_2, i, T')$ with $T' = \{t \in T \mid p \models \gamma(t)\}$
- Goal: solve discounted games $\text{proj}_p(G) = (S_1, S_2, i, T')$ for all products p simultaneously

Featured Discounted Games: Our Results

- for $G = (S_1, S_2, i, T, \gamma)$ a featured weighted game structure, define $\text{fattr} : (S \to (\mathbb{B}(N) \to \mathbb{R})) \to (S \to (\mathbb{B}(N) \to \mathbb{R}))$ by

$$\text{fattr}(U)(s)(\phi) = \begin{cases} \max\limits_{s \xrightarrow{x} s'} x + \lambda U(s')(\gamma((s,x,s')) \wedge \phi) & \text{if } s \in S_1 \\ \min\limits_{s \xrightarrow{x} s'} x + \lambda U(s')(\gamma((s,x,s')) \wedge \phi) & \text{if } s \in S_2 \end{cases}$$

- Theorem: The equation system $V = \text{fattr}(V)$ has a unique solution $\text{fattr}^* : S \to (\mathbb{B}(N) \to \mathbb{R})$, and for any $p \subseteq N$, $\text{val}(\text{proj}_p(G)) = \text{fattr}^*(i)(\gamma_p)$
- Theorem: Any featured discounted game admits a locally optimal featured strategy. If ξ_1 is locally optimal, then $\xi_1(\gamma_p)$ is optimal in $\text{proj}_p(G)$ for every $p \subseteq N$.

Conclusion

- We have introduced featured extensions of
  - reachability games
  - minimum reachability games
  - discounted games
  - energy games
  - parity games
- We have shown how to solve games of each type:
  - featured attractors
  - optimal featured strategies

- Extend to mean-payoff games by reduction to energy and discounted games
- Algorithms using guard partitions, late splitting and BDDs
- Featured timed games; featured stochastic games