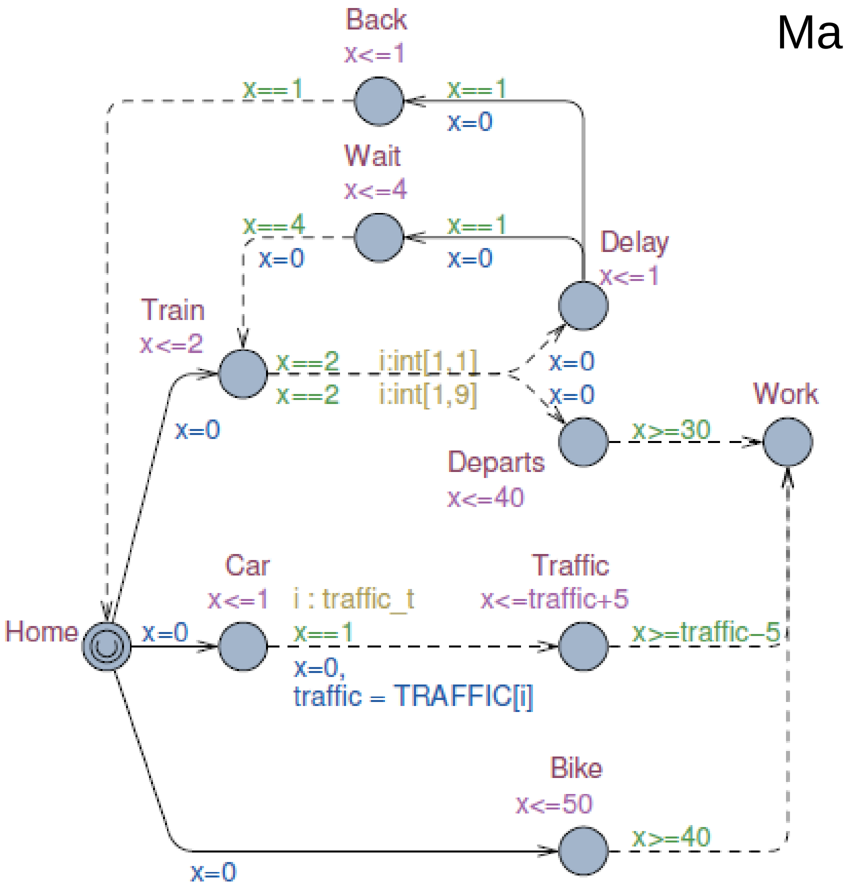
On Time with Minimal Expected Cost !

Alexandre David, Peter G Jensen,
Kim G Larsen, Axel Legay, Didier Lime,
Mathias G Sørensen, Jakob H Taankvist

CISS – Aalborg University
DENMARK

Motivation

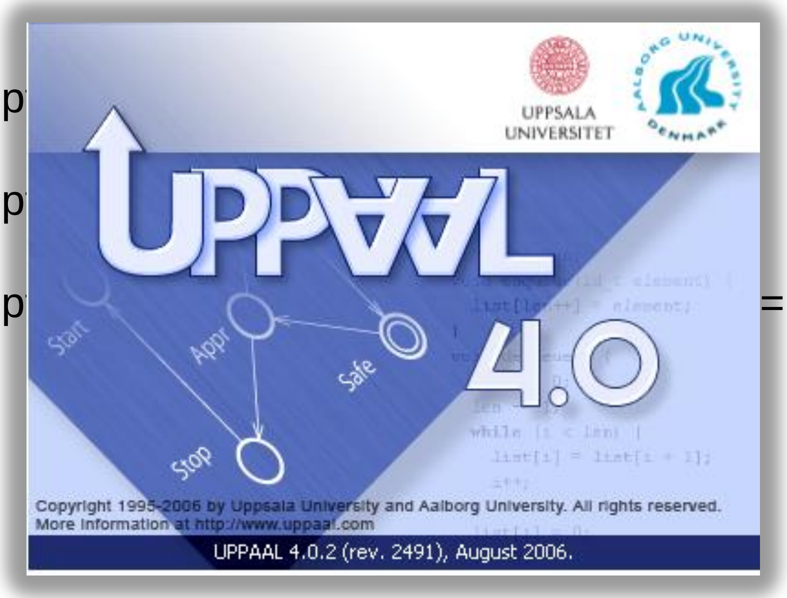
2-Player Game (Antagonistic opponent)
Markov Decision Process (probabilistic opponent)



Op

Op

Op =60:



Copyright 1995-2006 by Uppsala University and Aalborg University. All rights reserved.
More information at http://www.uppaal.com

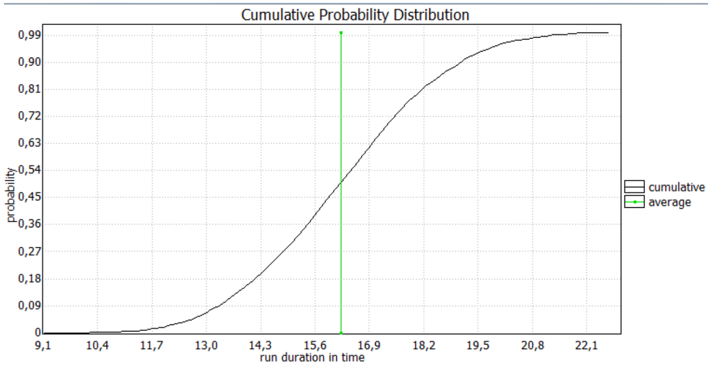
UPPAAL 4.0.2 (rev. 2491), August 2006.

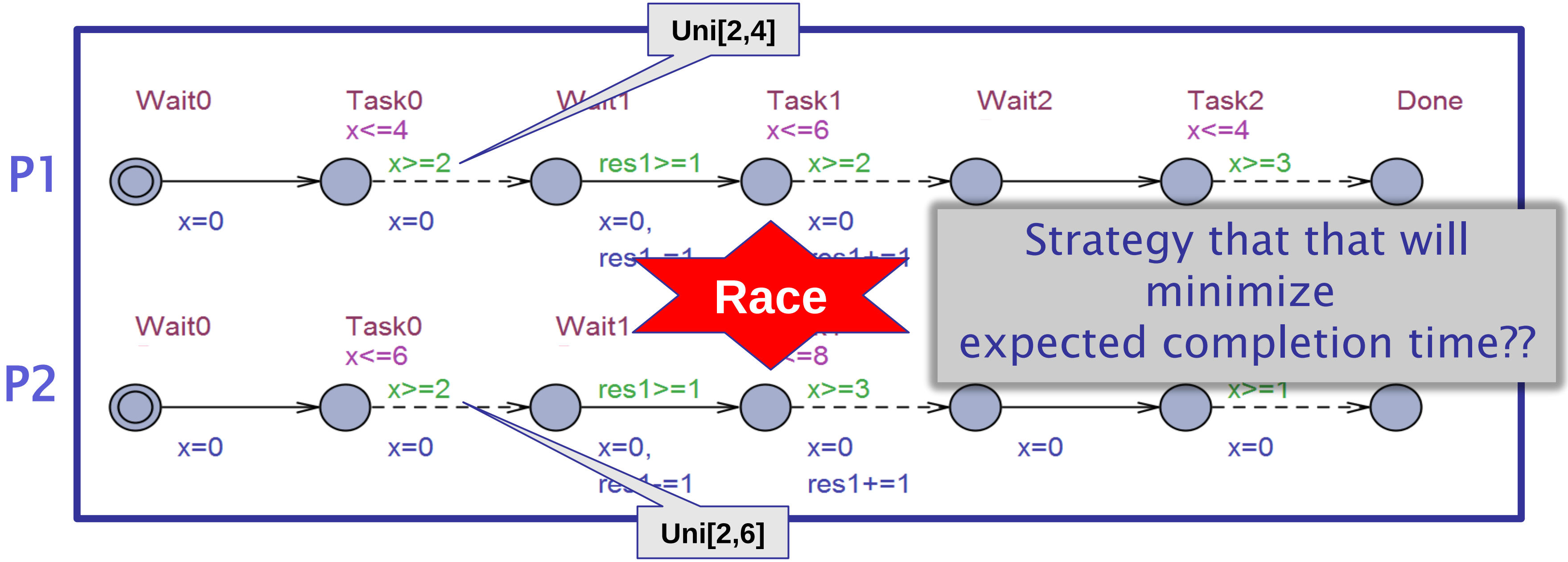
Bruyere, V., Filiot, E., Randour, M., Raskin, J.F.: Meet your expectations with guarantees: Beyond worst-case synthesis in quantitative games. STACS14

Duration Probabilistic Automata

```
/* Resources */  res1:1;
/* Processes */  P1:  [2,4].<res1:1>[2,6].[3,4];
                 P2:  [2,6].<res1:1>[3,8].[1,5];
```

Pr[<=1000](<> P1.Done && P2.Done)





Uni[2,4]

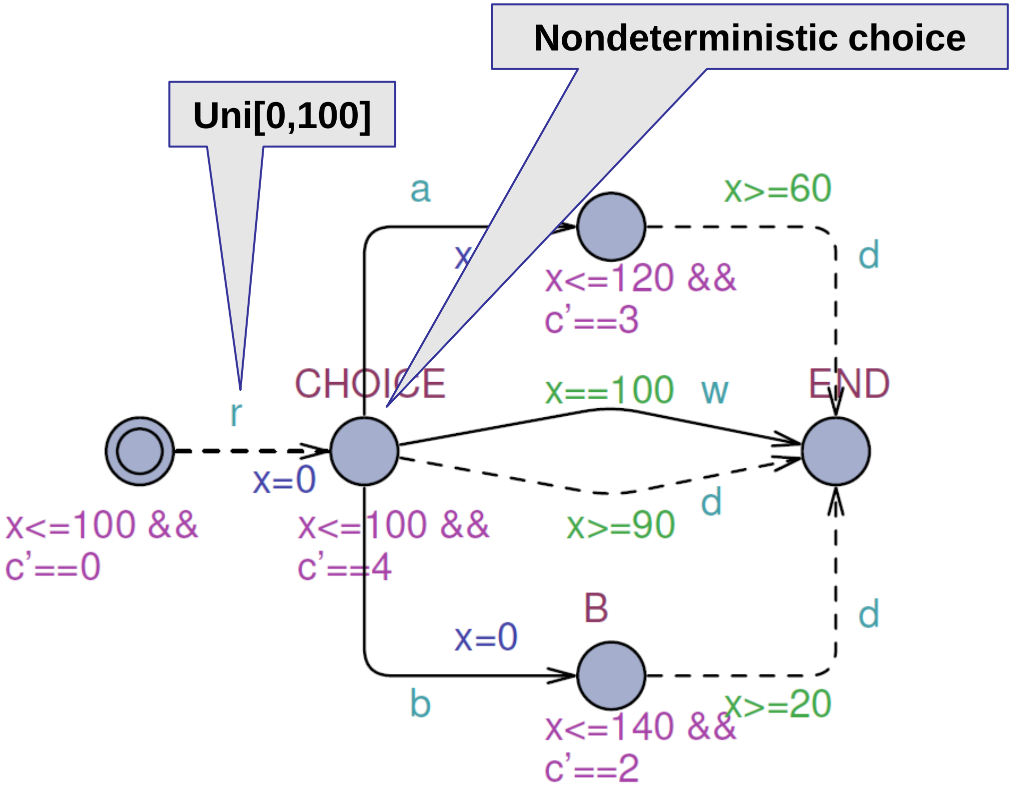
Uni[2,6]

Race

Strategy that that will minimize expected completion time??

Kempf, J.F., Bozga, M., Maler, O.: As soon as probable: Optimal scheduling under stochastic uncertainty. In: TACAS. pp. 385{400 (2013)

Motivation



× Priced Timed Game

✓ Timed Game TIGA

✓ Timed Automata UPPAAL

✓ Priced Timed Automata CORA

✓ Stochastic (P)TA SMC

× **Priced Timed MDP** TIGA/SMC

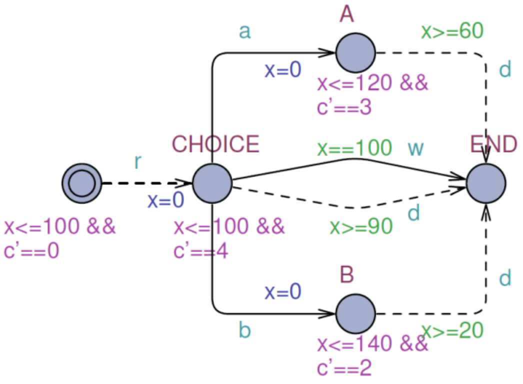
~ **Decision Stochastic Priced Timed Automata**

Minimize **expected cost**
subject to **guaranteed time-bound**

Overview

- Priced Timed Games
  - Time bounded reachability strategies
- Priced Timed Markov Decision Processes
  - Minimal expected cost reachability strategy
- Optimal Strategy Synthesis Using Reinforcement Learning
- Representation of Stochastic Strategies
- Experimental Results

PTA and PTG



$$\mathcal{A} = (L, \ell_0, X, \Sigma, E, P, Inv)$$

is a tuple where

- L is a finite set of locations,
- $\ell_0 \in L$ is the initial location,
- X is a finite set of non-negative real-valued clocks,
- Σ is a finite set of actions,
- $E \subseteq L \times \mathcal{B}(X) \times \Sigma \times 2^X \times L$ is a finite set of edges,
- $P : L \to \mathbb{N}$ assigns a price-rate to each location, and
- $Inv : L \to \mathcal{B}(X)$ sets an invariant for each location.

Priced Timed Game $\Sigma = \Sigma_c \uplus \Sigma_u$

PTA Semantics

$$S_{\mathcal{A}} = (Q, q_0, \Sigma, \rightarrow)$$

- $(\ell, v) \in Q$ for $\ell \in L$ and $v \in \mathbb{R}^{X}_{\geq 0}$ st $v \models Inv(\ell)\}$,
- $q_0 = (\ell_0, 0)$ is the in

and[1]

- $(\ell, v) \xrightarrow{a}_0 (\ell', v')$ if $(\ell \xrightarrow{g,a,r} \ell') \in E$ st.
- $(\ell, v) \xrightarrow{d}_p (\ell, v + d$ $p = P(\ell) \cdot d$, $v \models Inv(\ell)$ and $v + d \models Inv(\ell)$.

- Set of runs of: $Exec_{\mathcal{A}}$.
- Set of finite (maximal) runs: $Exec^f_{\mathcal{A}}$ ($Exec^m_{\mathcal{A}}$).
- $\pi[i]$ the state q_i,
- $\pi|_i$ ($\pi|^i$) the prefix (suffix) of π ending (starting) at q_i.
- $C(\pi)$ ($T(\pi)$) denotes total accumulated cost (time).

Run π:

$$q_0 \xrightarrow{d_0}_{p_0} q'_0 \xrightarrow{a_0}_0 q_1 \xrightarrow{d_1}_{p_1} q'_1 \xrightarrow{a_1}_0 \cdots \xrightarrow{d_{n-1}}_{p_{n-1}} q'_{n-1} \xrightarrow{a_{n-1}}_0 q_n \cdots$$

$a_i \in \Sigma$, $d_i, p_i \in \mathbb{R}_{\geq 0}$, and q_i is a state (ℓ_{q_i}, v_{q_i}).

Strategies & Outcome

Priced Timed Game $\Sigma = \Sigma_c \uplus \Sigma_u$

$$\sigma : Exec^f_{\mathcal{G}} \rightharpoonup \mathcal{P}\left(\Sigma_c \cup \{\lambda\}\right) \setminus \{\emptyset\}$$

such that for any finite run π, if $q = last(\pi)$ and $a \in \sigma(\pi) \cap \Sigma_c$, then $q \xrightarrow{a} q'$ fs q'.

$Out(\sigma) \subseteq Exec_{\mathcal{G}}$

- $q_0 \in Out(\sigma)$
- If $\pi \in Out(\sigma)$ then $\pi' = \pi \xrightarrow{e} q' \in Out(\sigma)$ if $\pi' = Exec_{\mathcal{G}}$ and either one of the following three conditions hold:
  1. $e \in \Sigma_u$, or
  2. $e \in \Sigma_c$ and $e \in \sigma(\pi)$, or
  3. $e \in \mathbb{R}_{\geq 0}$ and for all $e' < e$, $last(\pi) \xrightarrow{e'} q'$ for some q' st $\sigma(\pi \xrightarrow{e'} q') \ni \lambda$.

Cost Bounded Reachability Strategies

For $G \subseteq L$, $B \in \mathbb{R}_{\geq 0}$: (G, B) is a cost-bounded reachability objective.

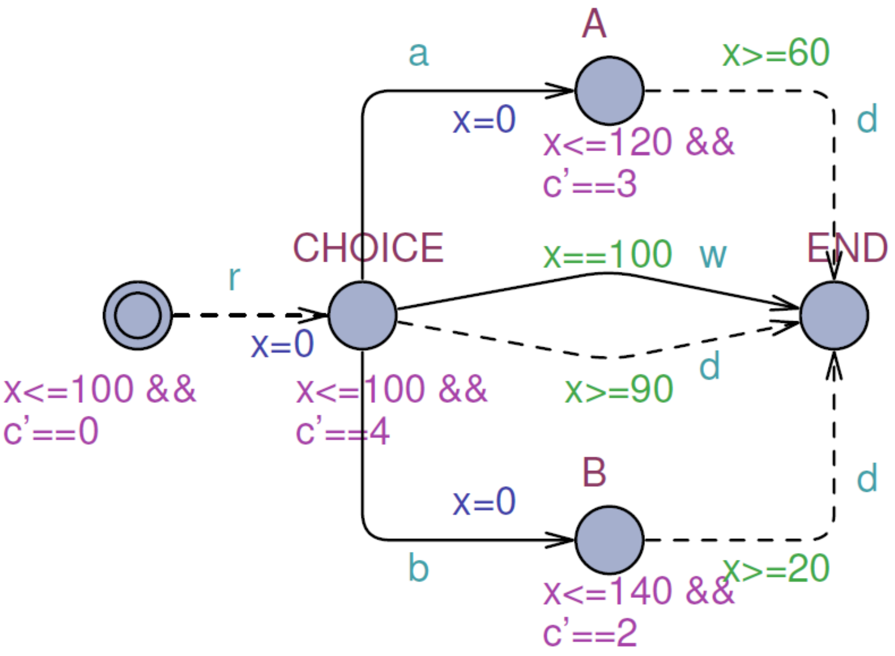
π is winning w.r.t. (G, B), if $\mathit{last}(\pi) \in G \times \mathbb{R}^{X}_{\geq 0}$ and $C(\pi) \leq B$.
A strategy σ over $\mathcal{G}$ is a winning strategy if all runs in $\mathit{Out}(\sigma)$ are winning.

Theorem (Memoryless, Most Permissive Strategies)

Let $\mathcal{G}$ be a non-Zeno, clocked TG. If a time-bounded reachability objective (G, T) has a winning strategy, then it has

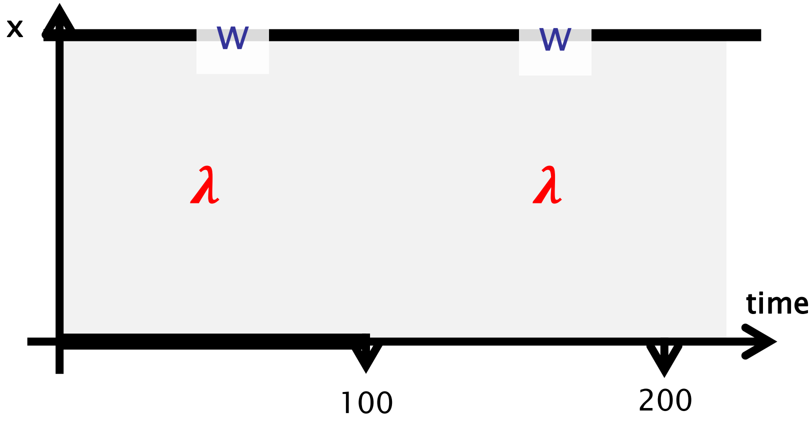
1. *a deterministic, memoryless winning strategies, and*
2. *a (unique) most permissive, memoryless winning strategy $\sigma^{p}_{\mathcal{G}}(G, T)$.*

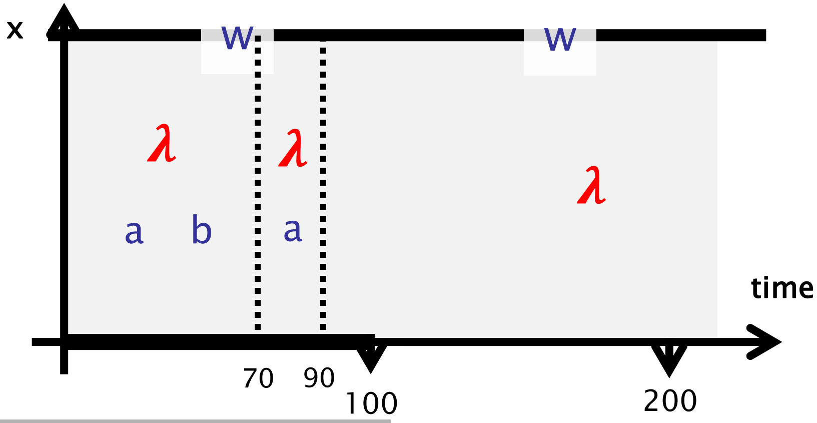
Motivation



Objective: $A\langle\rangle$ **(END $\wedge$ time ≤ 210)**

Deterministic, memoryless strategy:





Most permissive, memoryless strategy

Priced Timed MDPs

$$\mathcal{M} = \langle \mathcal{G}, \mu^u \rangle$$

where

- $\mathcal{G} = (L, \ell_0, X, \Sigma_c, \Sigma_u, E, P, Inv)$ is a PTG, and
- μ^u is a family of density-functions, $\{\mu^u_q : \exists \ell \exists v . q = (\ell, v)\}$, with $\mu^u_q(d, u) \in \mathbb{R}_{\geq 0}$ assigning the *density* of the environment aiming at taking the uncontrollable action $u \in \Sigma_u$ after a delay of d from state q.

Assumptions:

1. $\mu^u_q(d, u) > 0$ only if $q \xrightarrow{d,u}$ in $\mathcal{G}$.
2. $\sum_u (\int_{t \geq 0} \mu^u_q(t, u) dt) = 1$

Stochastic Strategies

μ^c for a PTMDP $\mathcal{M} = \langle \mathcal{G}, \mu^u \rangle$

is a family of density-functions, $\mu^c = \{\mu^c_q : \exists \ell \exists v . q = (\ell, v)\}$, where $\mu^c_q(d, c) \in \mathbb{R}_{\geq 0}$ assigns the *density* of the controller aiming at taking the controllable action $c \in \Sigma_c$ after a delay of d from state q.

- Repeated races between μ^u and μ^c,
- Induced probability measure $\mathbb{P}_{\langle \mathcal{G}, \mu^u \rangle, \mu^c}$ on (certain) sets of runs.

Induced Probability Measure

Cylinder set $\mathcal{C}(q, I_0\ell_0 I_1 \cdots I_n\ell_n)$ with $\ell_i \in L$ and $I_i = [l_i, u_i]$ with $l_i, u_i \in \mathbb{Q}$, $i = 0..n$, consists of all maximal runs having a prefix of the form:

$$q \xrightarrow{d_0}\xrightarrow{a_0} (\ell_0, v_0) \xrightarrow{d_1}\xrightarrow{a_1} \cdots \xrightarrow{d_n}\xrightarrow{a_n} (\ell_n, v_n)$$

where $d_i \in I_i$ for all $i < n$.

Probability Measure

$$\mathbb{P}_{\langle\mathcal{G},\mu^u\rangle,\mu^c}(\mathcal{C}(q, I_0\ell_0 I_1\ell_1 \cdots I_{n-1}\ell_n)) =$$

$$\sum_{p\in\{u,c\}} \sum_{\substack{a\in\Sigma_p \\ \ell_q \xrightarrow{a} \ell_1}} \int_{t\in I_0} \mu_q^p(t,a) \cdot \Big(\int_{\tau>t} \mu_q^{\overline{p}}(\tau) d\tau \Big) \cdot$$

$$\mathbb{P}_{\langle\mathcal{G},\mu^u\rangle,\mu^c}(\mathcal{C}((q^t)^a, \mathcal{C}(I_1 \cdots I_{n-1}\ell_n)) \, dt$$

where $\mu_q^p(\tau) = \sum_{a\in\Sigma_p} \mu_q^p(\tau, a)$.

Minimum Expected Cost

Let $\pi \in Exec^m$ and let G be as set of goal locations.

$$C_G(\pi) = min\{C(\pi|_i) \;:\; \pi[i] \in G\}$$

denotes the accumulated cost before π reaches G.

Expected Value of C_G given μ^c:

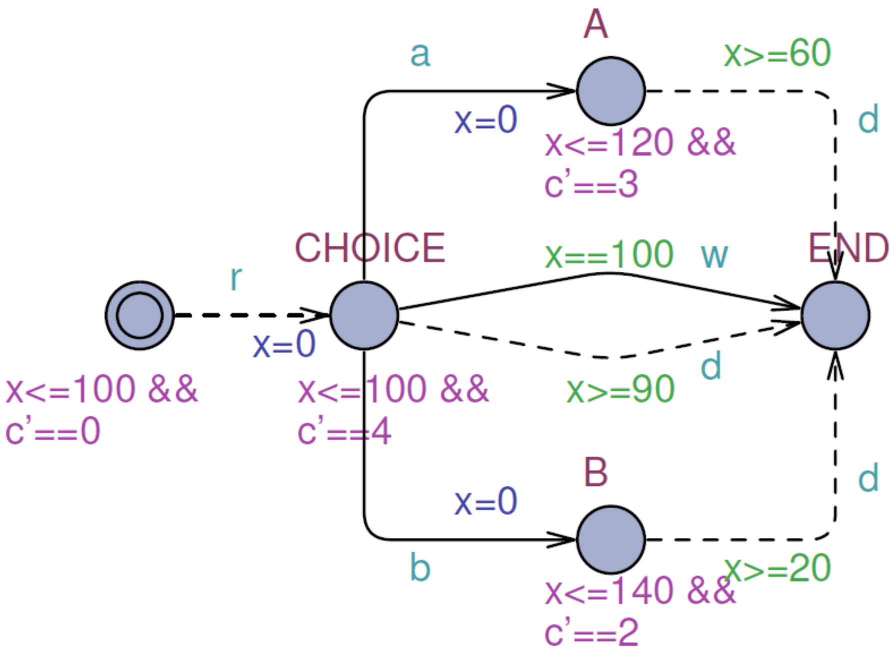
$$\mathbb{E}_{\mu^c}^{\langle \mathcal{G},\mu^u\rangle}(C_G) = \int_{\pi \in Exec^m} C_G(\pi)\mathbb{P}_{\langle \mathcal{G},\mu^u\rangle,\mu^c}(d\pi)$$

Optimal strategy μ^o

$$\mathbb{E}_{\mu^o}^{\langle \mathcal{G},\mu^u\rangle}(C_G) = \inf\left\{\, \mathbb{E}_{\mu^c}^{\langle \mathcal{G},\mu^u\rangle}(C_G) \mid \mu^c \prec \sigma^p(G,T) \,\right\}$$

where $\sigma^p(G,T)$ is the most permissive T time-bounded reachability strategy.

Motivation



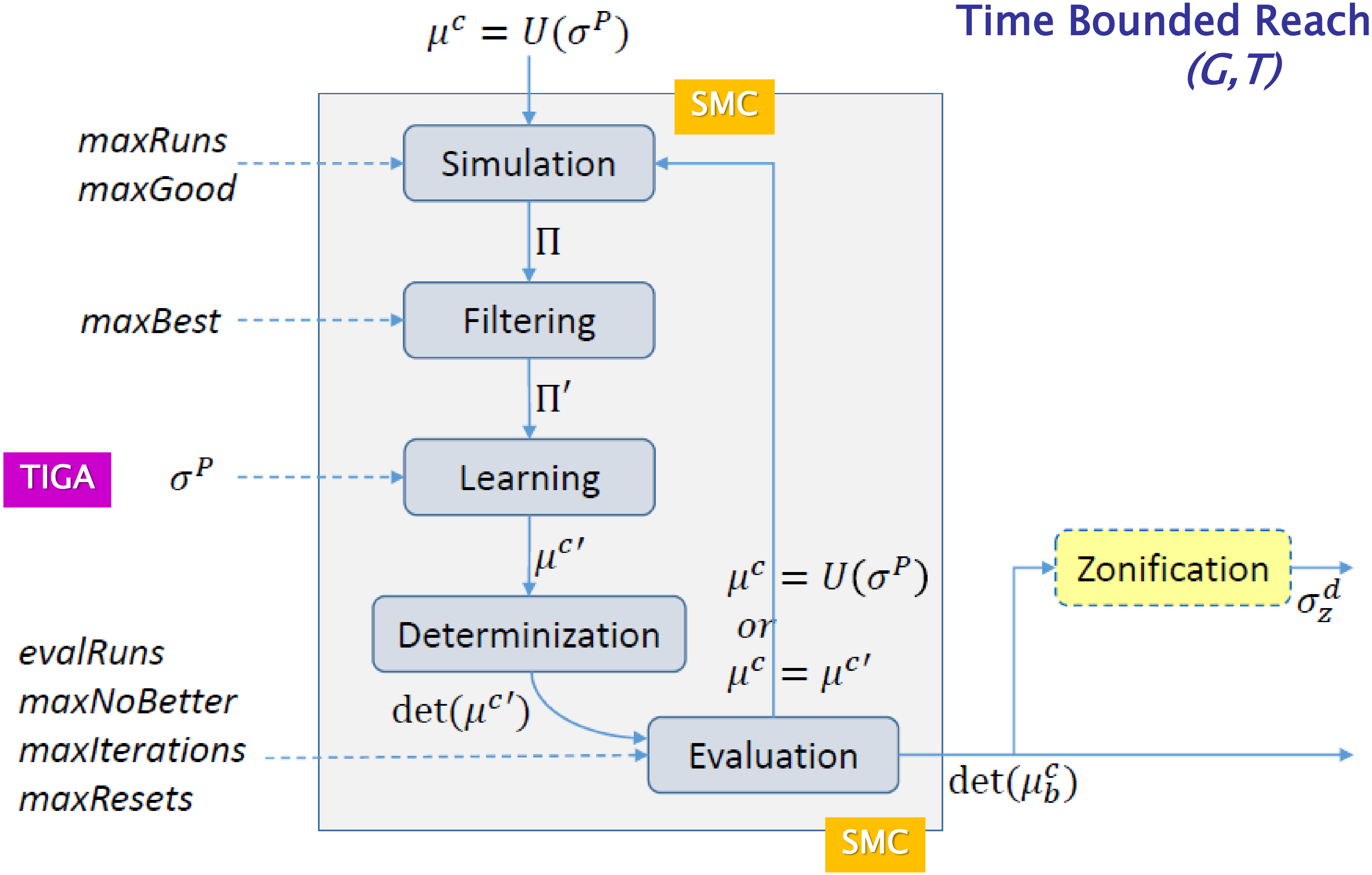
Minimal Expected Cost
Strategy (0,b) **2*80=160**

Expected Cost for TIGA
Strategy (100,w) **4*95=380**

Minimal Expected Cost while guaranteeing END is reached within time 210:

Strat.: t>90→ (100,w)
t>70→ (0,b)
ow → (0,a)
=
204

Reinforcement Learning



Time Bounded Reachability
(G,T)

Strategies

Nondeterministic Strategies (UPPAAL TIGA)

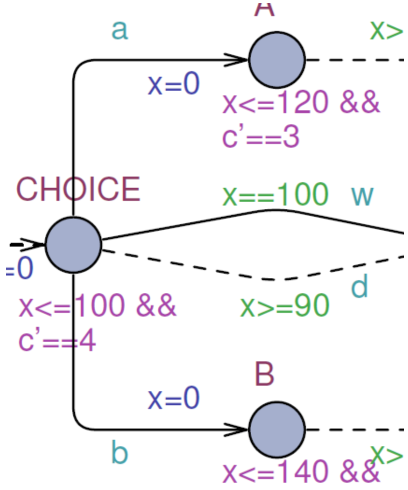
$R_\ell = \{(Z_1, a_1), \ldots, (Z_k, a_k)\}$, where $a_i \in \Sigma_c \cup \{\lambda\}$. Now R represents the strategy σ_R where $\sigma_R((\ell, v)) \ni a$ iff $(Z, a) \in R_\ell$ for some Z with $v \in Z$.

Stochastic Strategies (non-lazy *)

- Urgent: $\mu^c_{(\ell,v)}(d, a) = 0$ if $d > 0$, or
- Wait: $\mu^c_{(\ell,v)}(d, a) = 0$ whenever $\sigma^p(\ell, v + d) \ni \lambda$.

$$\mu^c_{(\ell,v)} : (\Sigma_c \cup \{\mathtt{w}\}) \rightarrow [0, 1].$$

Classes allowing for efficient representation and learning

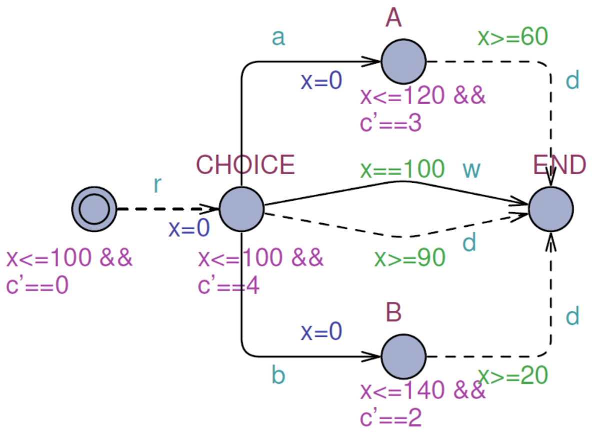


* Non-lazy strategies suffices for DPAs

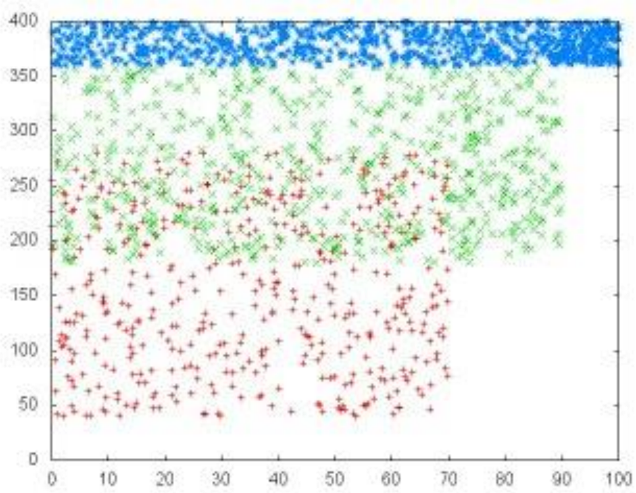
Learning

Given a set of runs Π the relevant information for the sub-strategy μ_ℓ^c is given as In_ℓ:

$$In_\ell = \{(s_n, v) \in (\Sigma_c \cup \mathbb{R}) \times \mathbb{R}^X_{\geq 0} \mid (q_0 \xrightarrow{s_0}_{p_0} \dots \xrightarrow{s_{n-1}}_{p_{n-1}} (\ell, v) \xrightarrow{s_n}_{p_n} \dots) \in \Pi\}$$



C(π)



wait

a

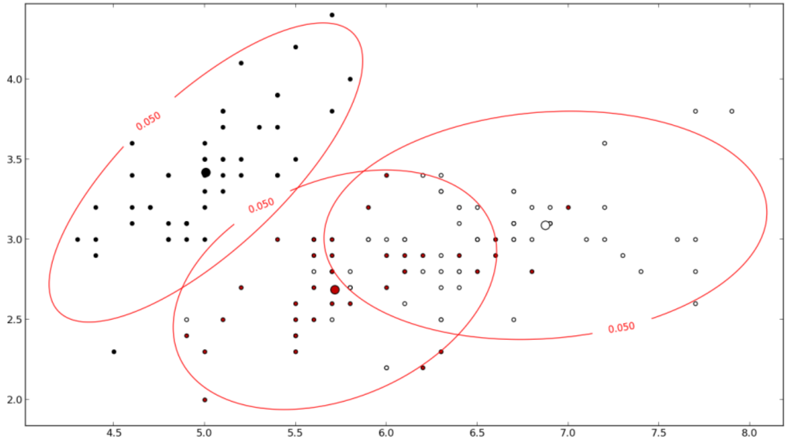
b

time(π)@Choice

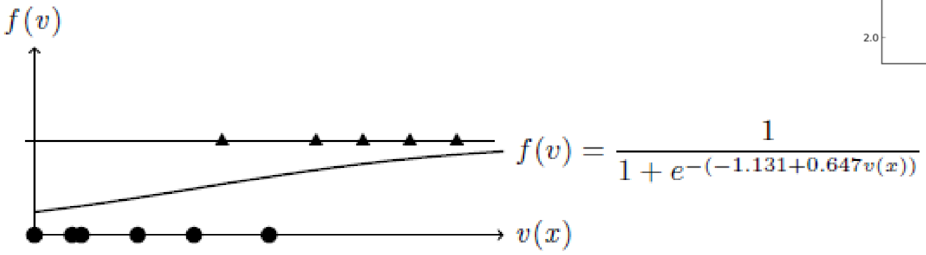
Simulation of $\boldsymbol{Uni}(\boldsymbol{\sigma^p})$ for $A\langle\rangle$ **(END** ∧ **time** ≤ 210)

Strategies – Representation & Manipulation

- Covariance Matrices

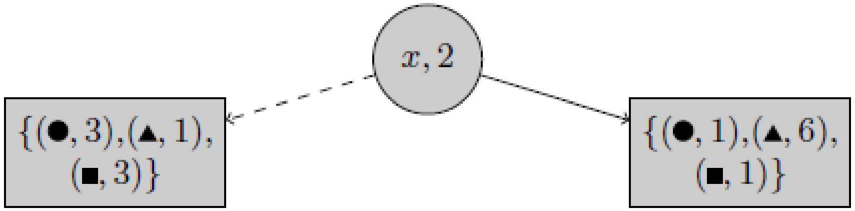


- Logistic Regression



$$f(v) = \frac{1}{1 + e^{-(-1.131+0.647v(x))}}$$

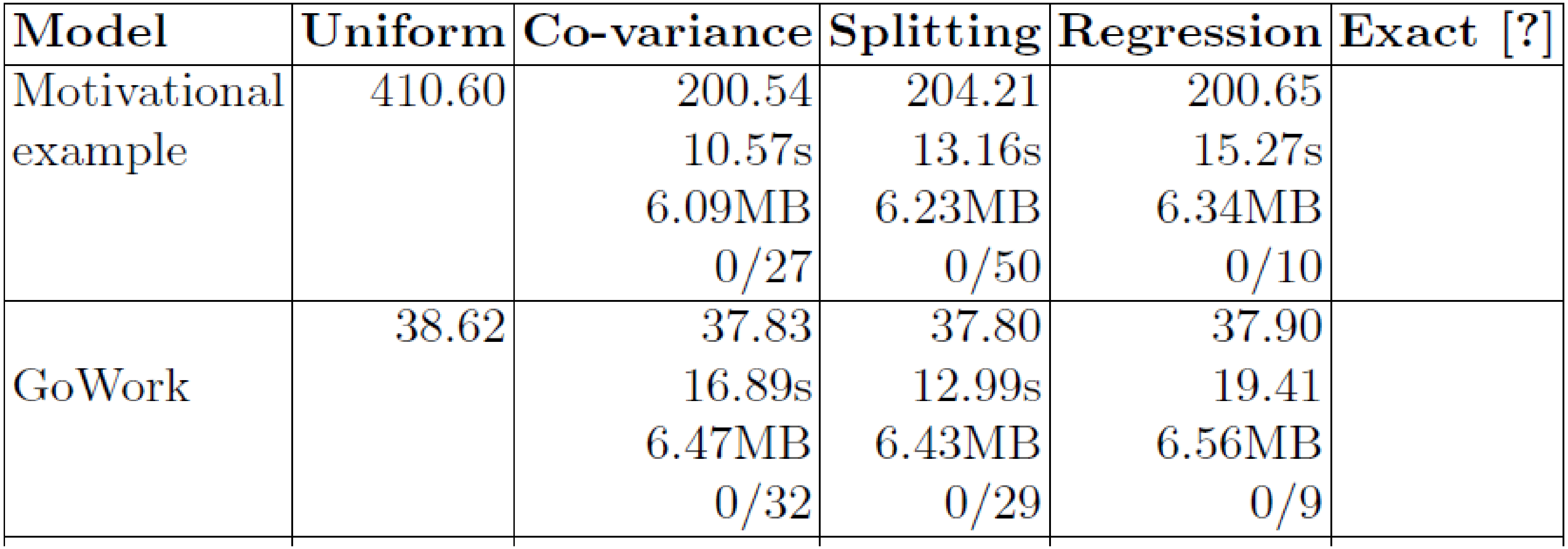
- Splitting



Using
Learning
Determinization

$$\mu^{c}_{(\ell,v)} : (\Sigma_c \cup \{\mathtt{w}\}) \to [0,1].$$

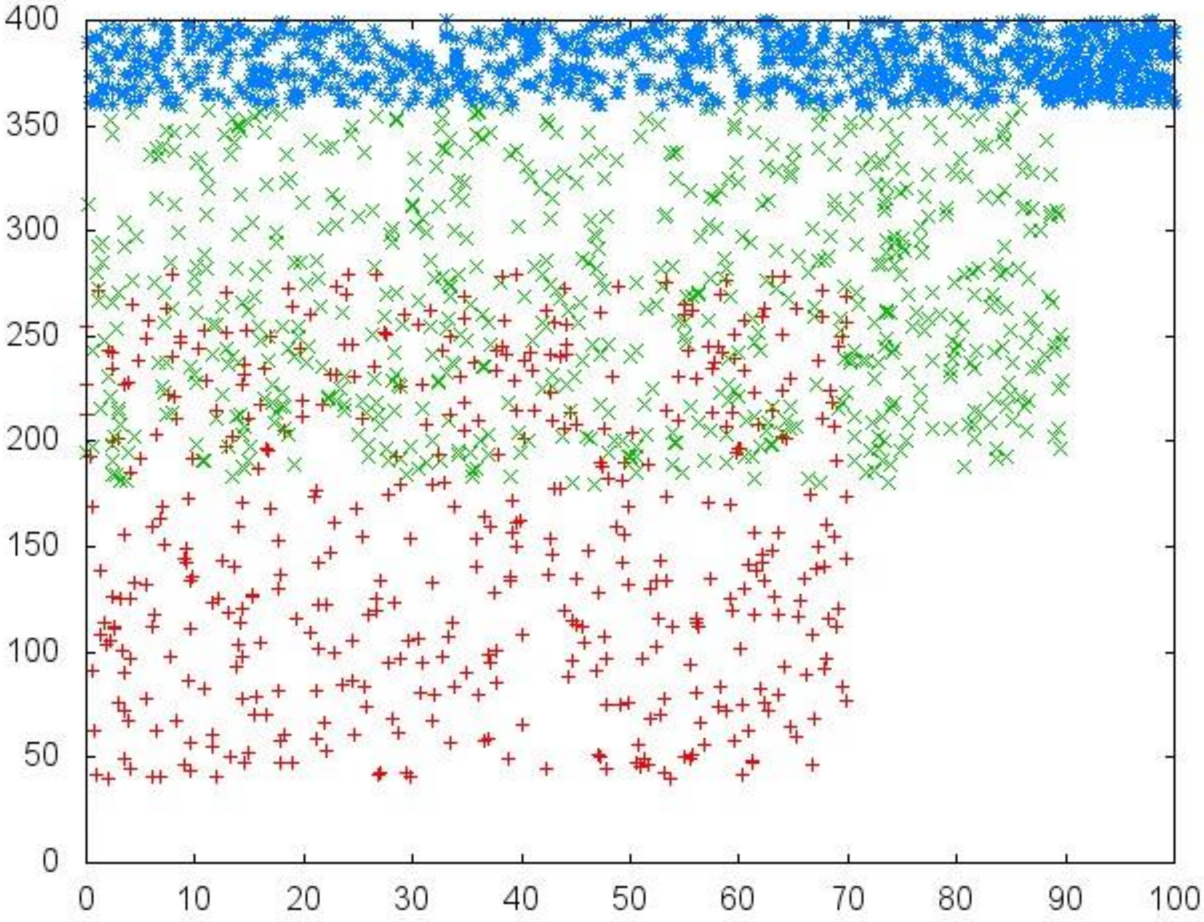
Experiments



Model	Uniform	Co-variance	Splitting	Regression	Exact [?]
Motivational example	410.60	200.54 10.57s 6.09MB 0/27	204.21 13.16s 6.23MB 0/50	200.65 15.27s 6.34MB 0/10	
GoWork	38.62	37.83 16.89s 6.47MB 0/32	37.80 12.99s 6.43MB 0/29	37.90 19.41 6.56MB 0/9	

Learned Strategies

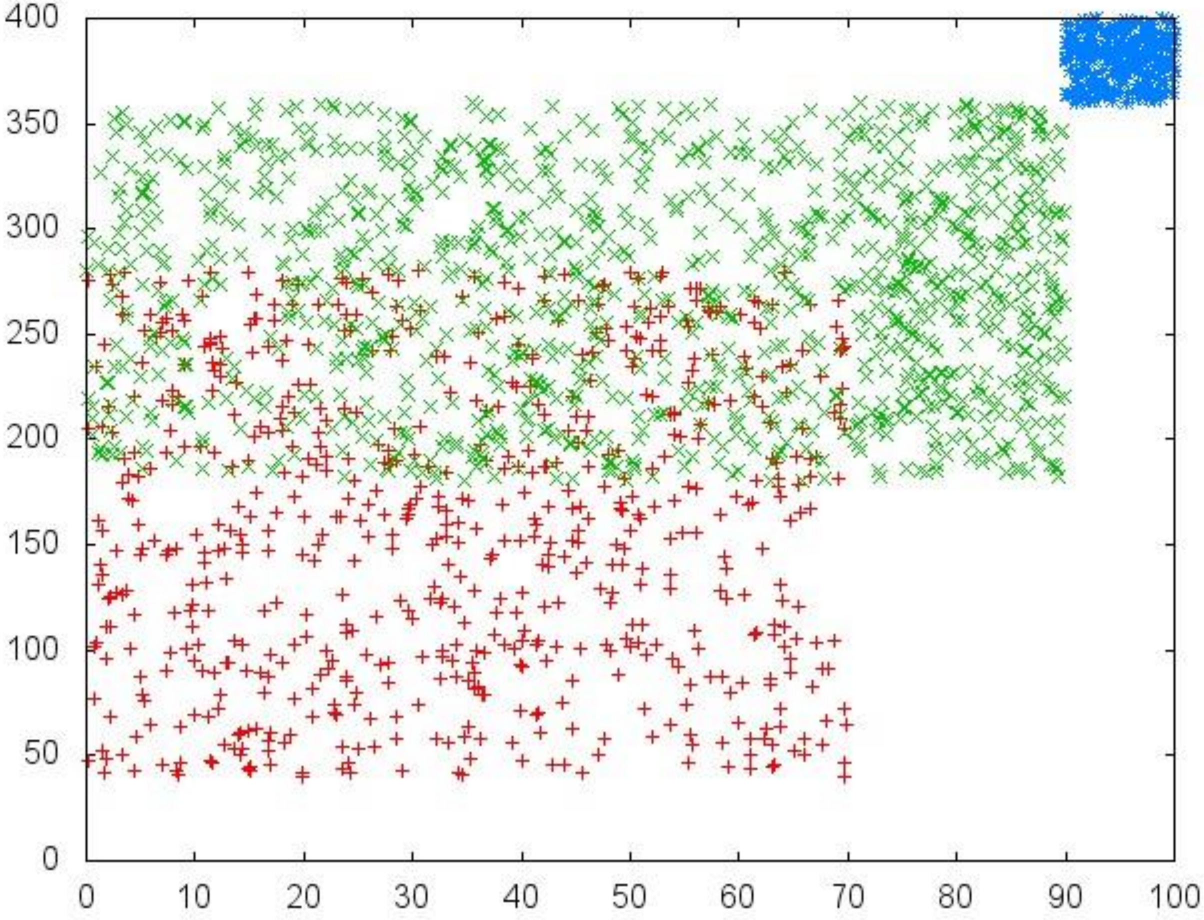
Covariance



e.

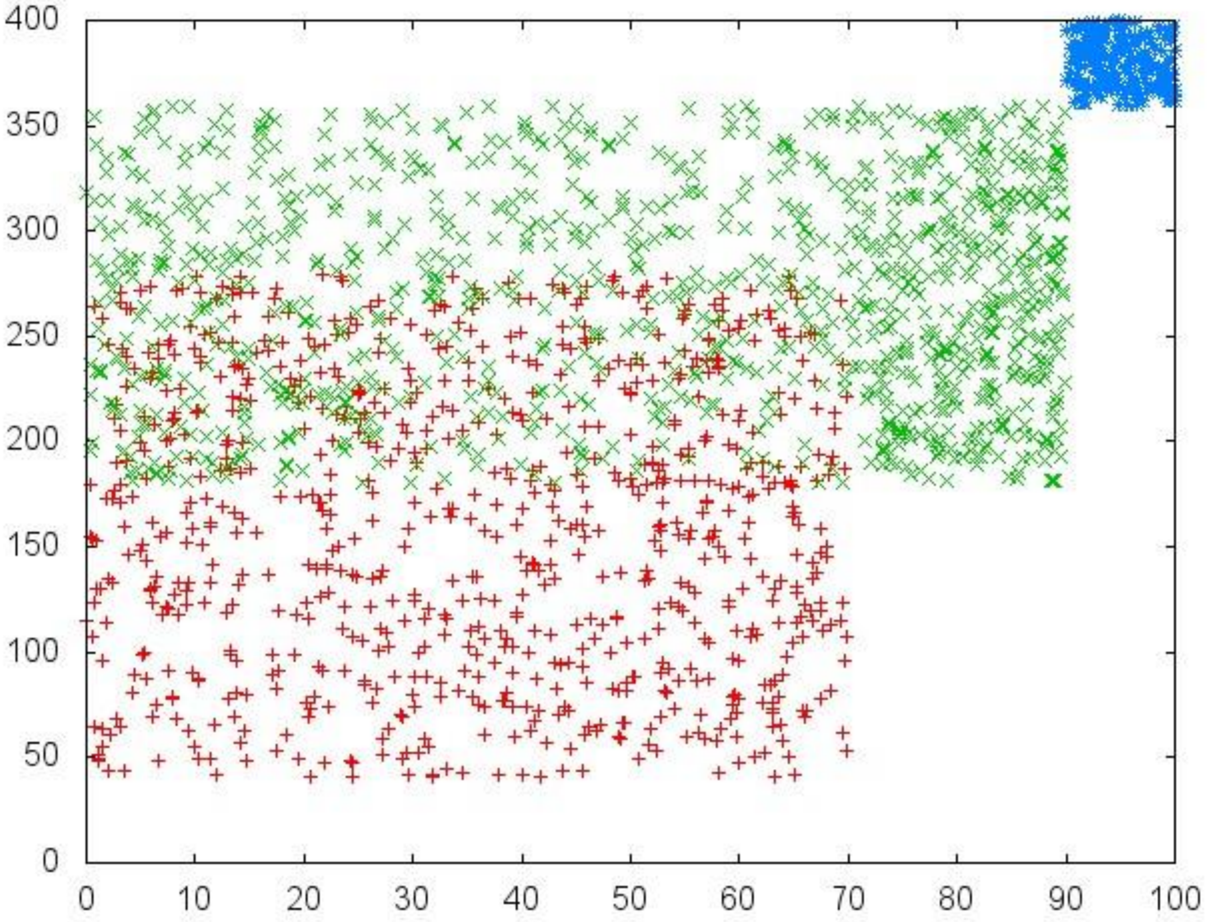
Learned Strategies

Covariance



e.

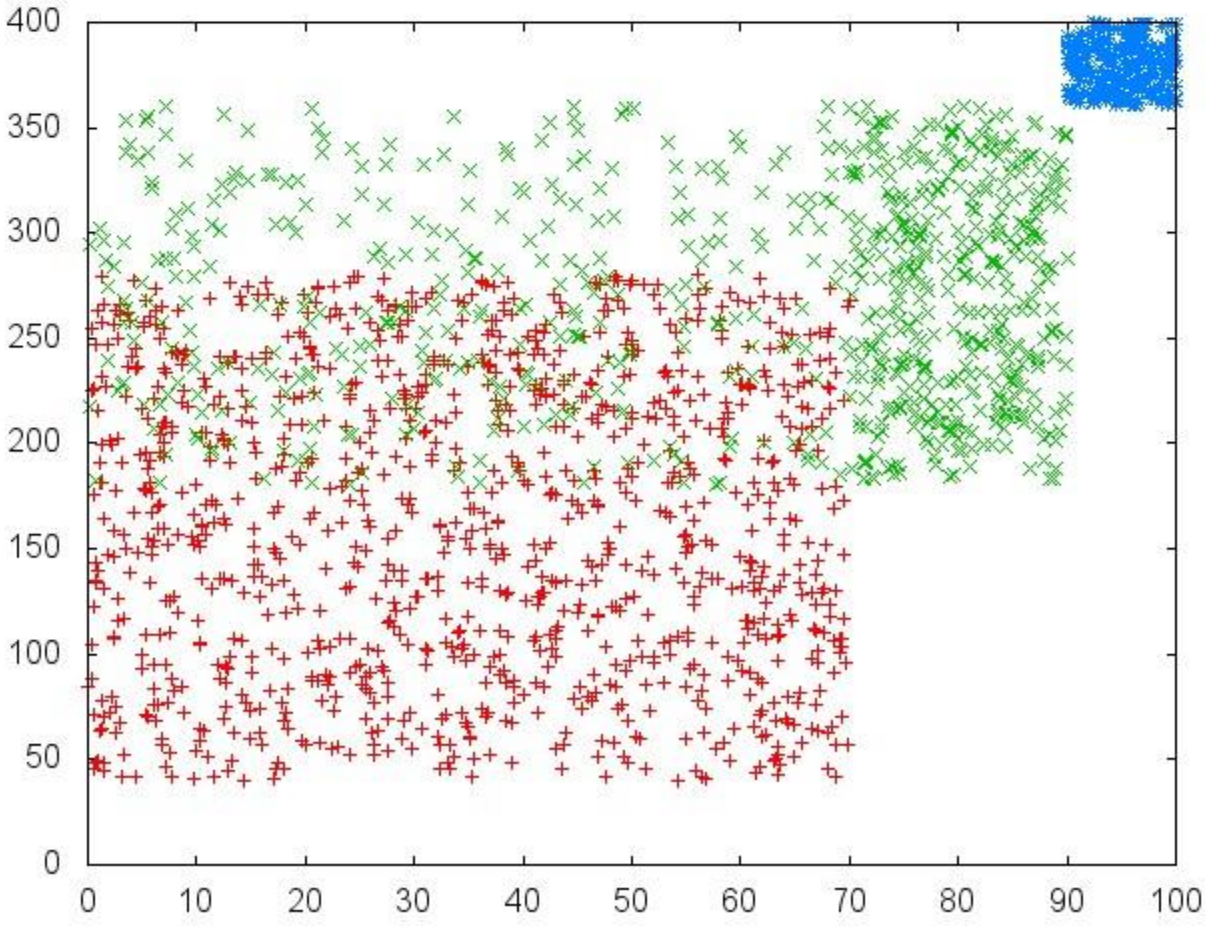
Learned Strategies



e.

Learned Strategies

Covariance

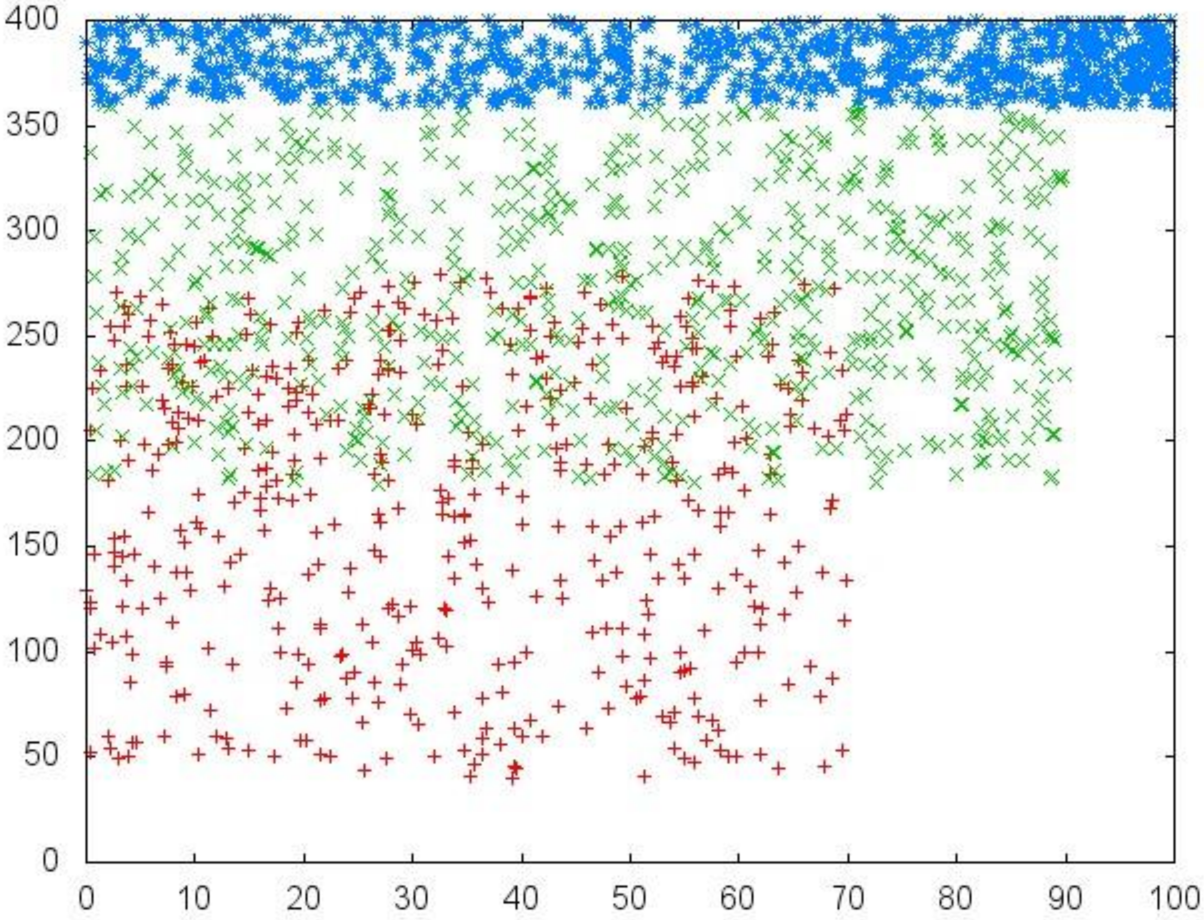


e.

Learned Strategies

Learned Strategies

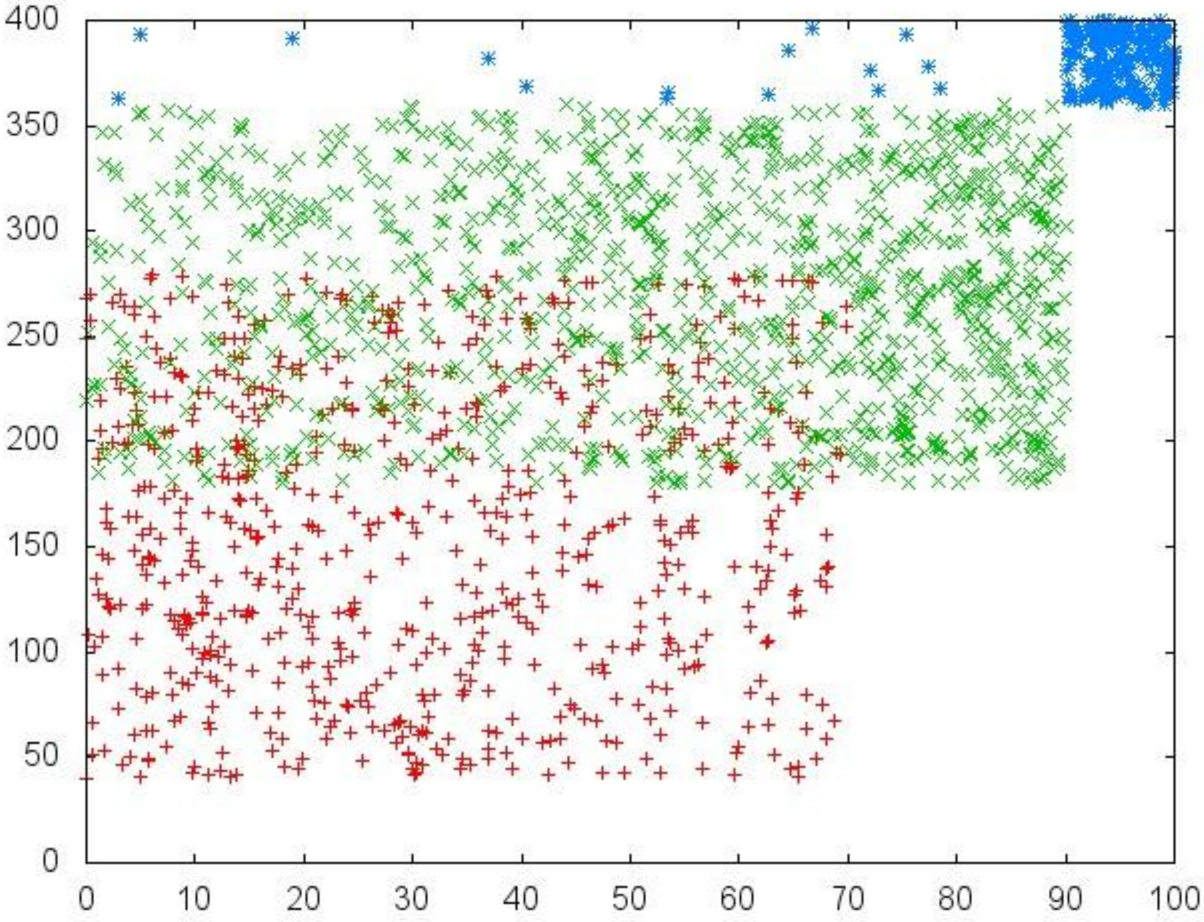
Regression



e.

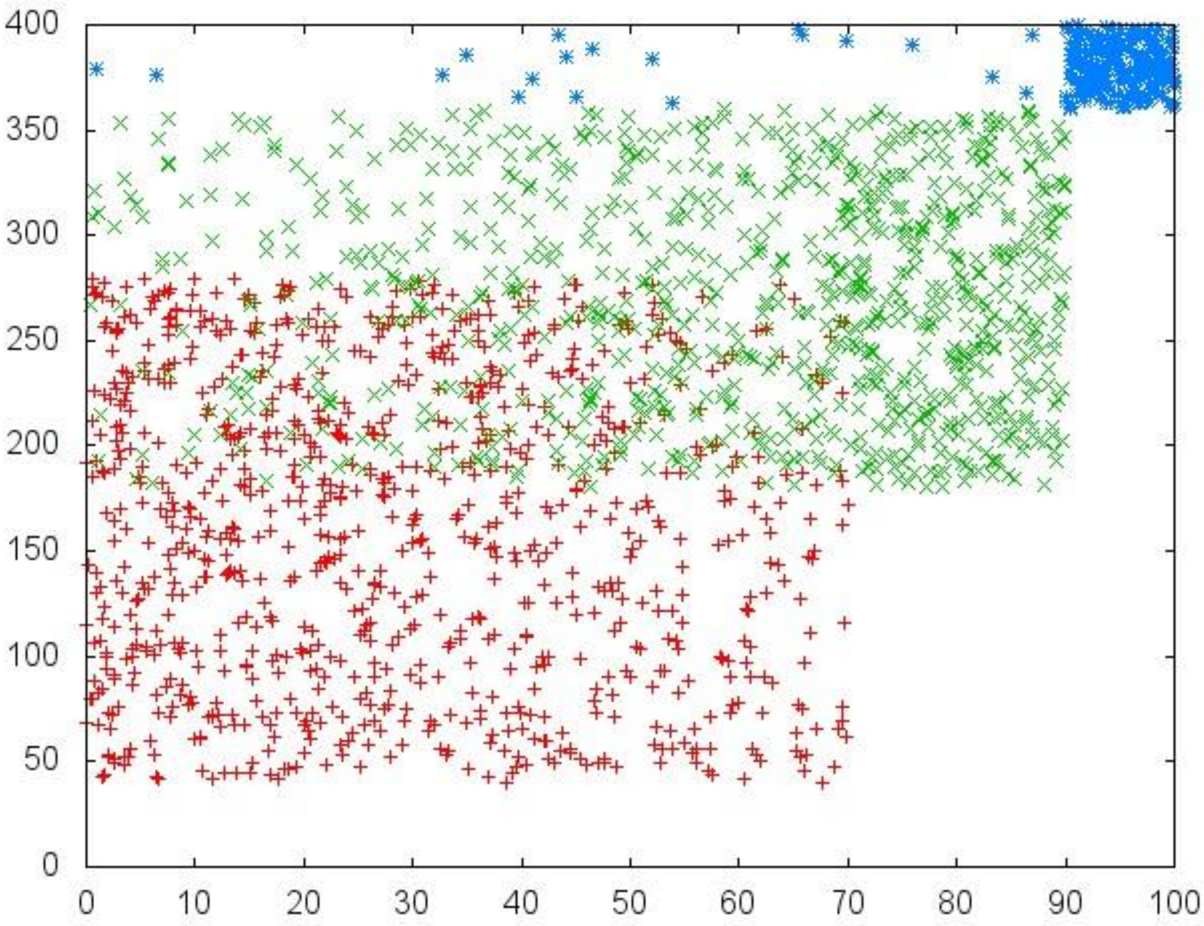
Learned Strategies

Regression



e.

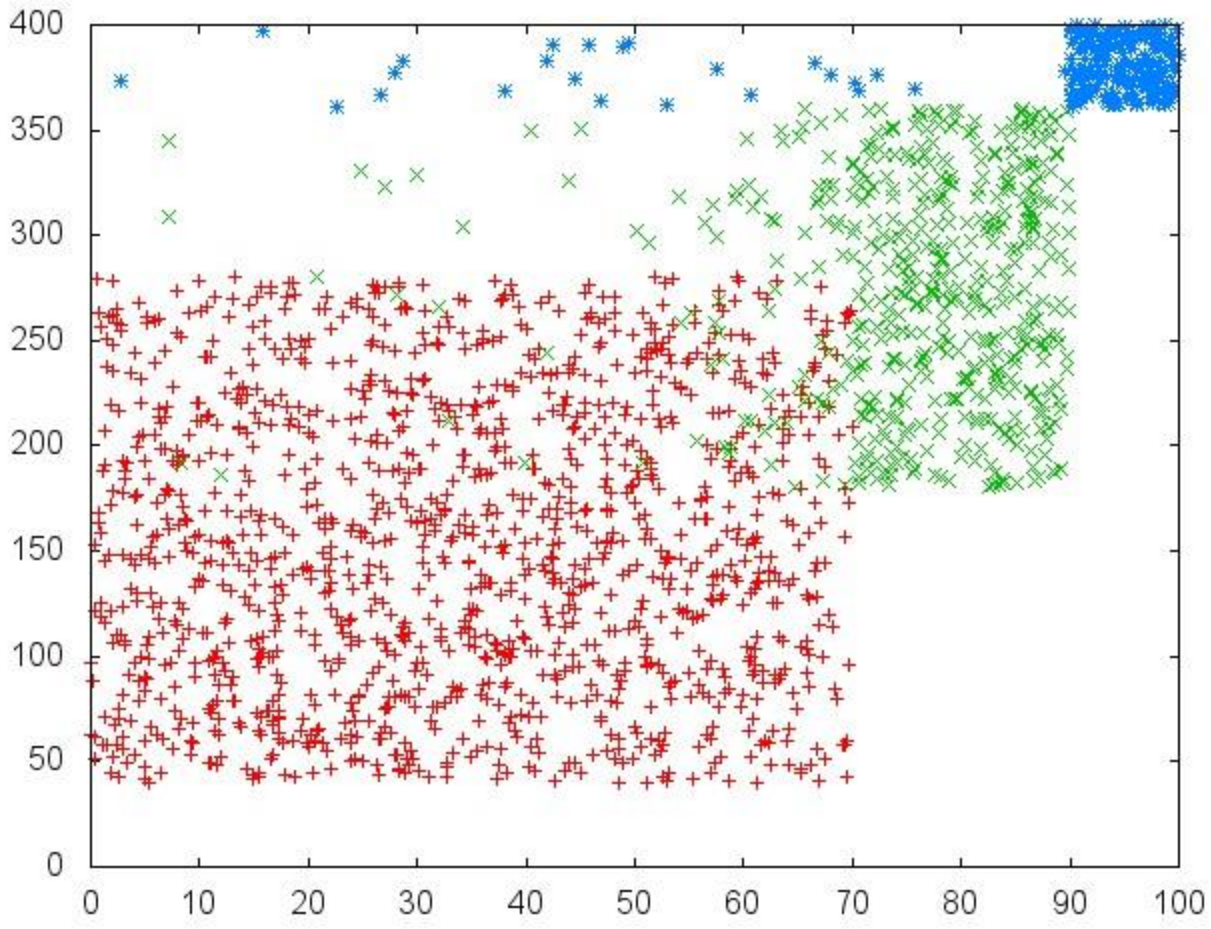
Learned Strategies



e.

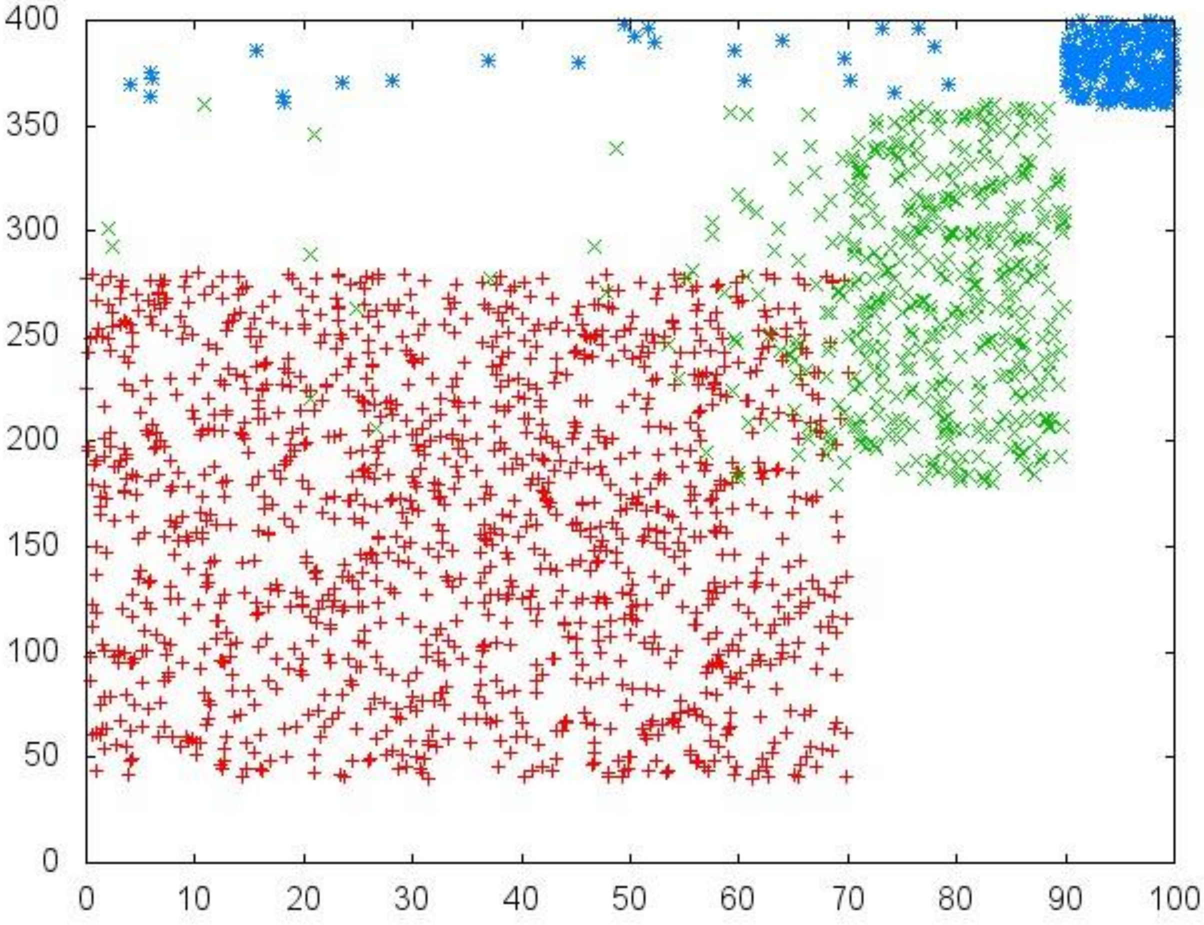
Learned Strategies

Regression



e.

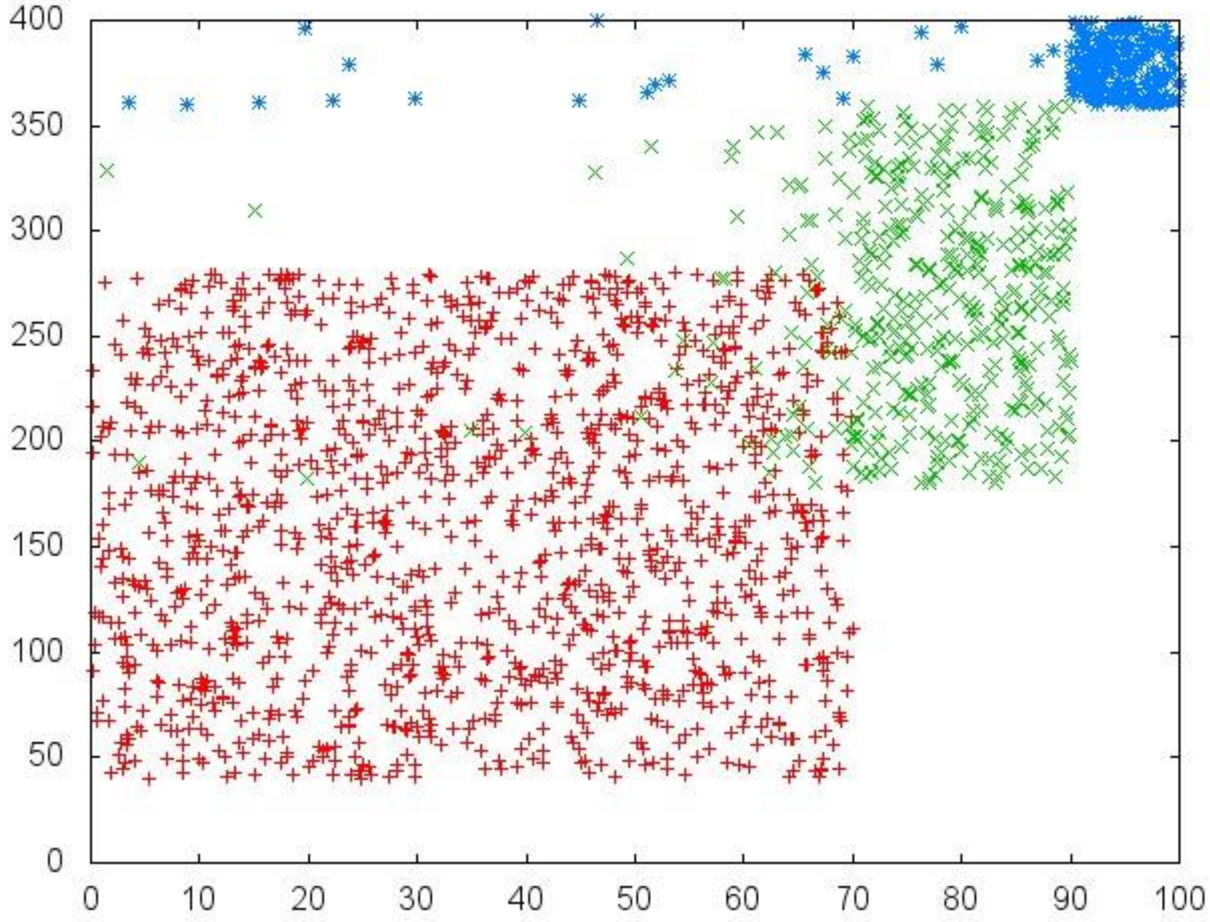
Learned Strategies



e.

Learned Strategies

Regression

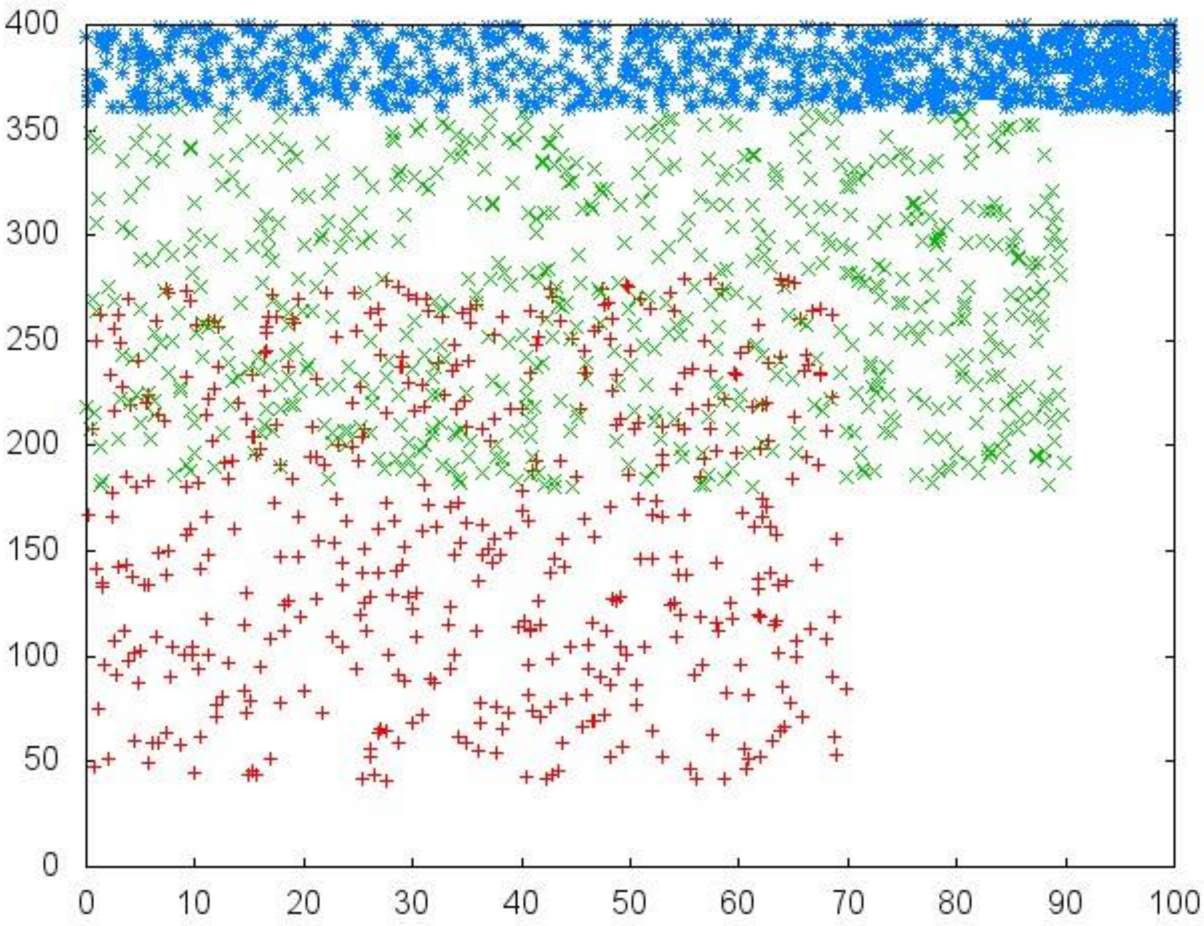


e.

Learned Strategies

Learned Strategies

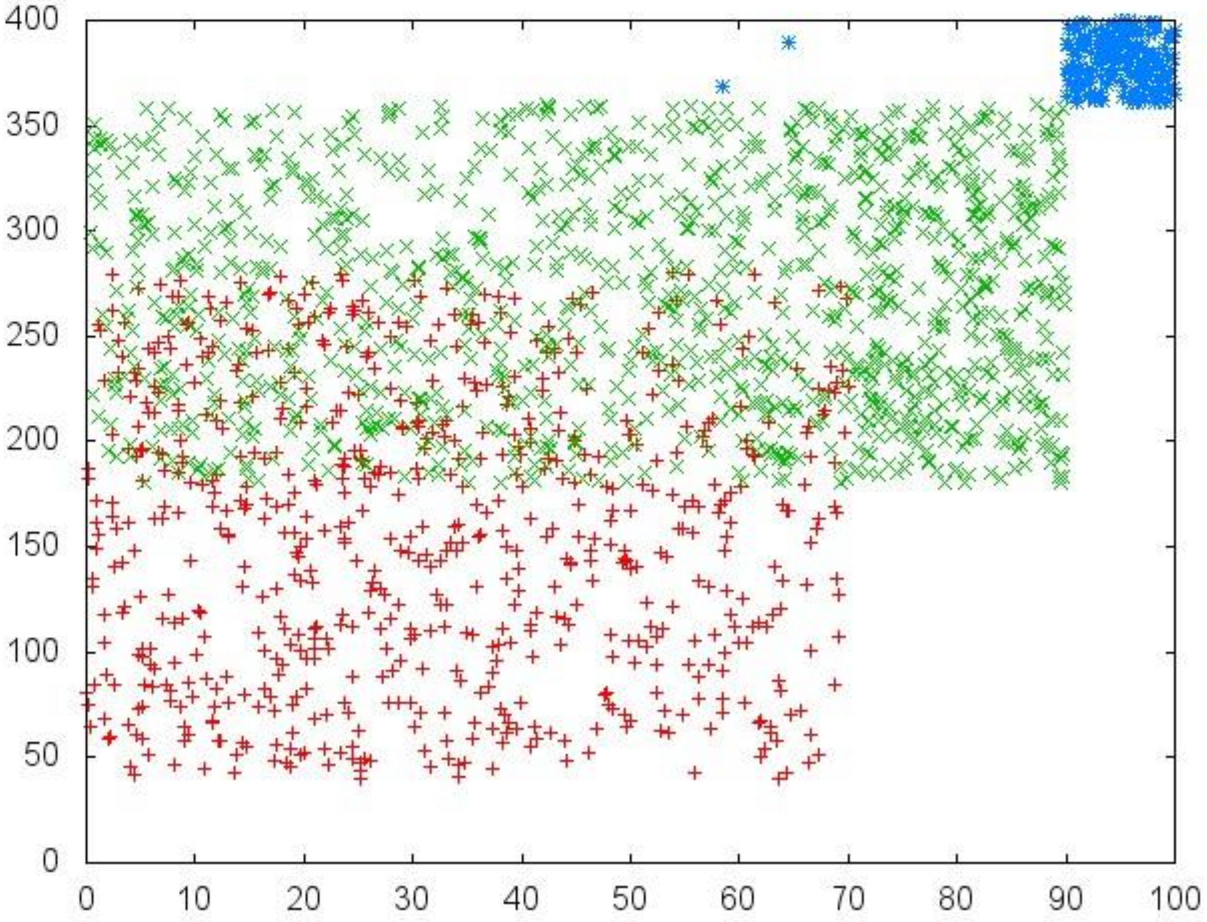
Splitting



e.

Learned Strategies

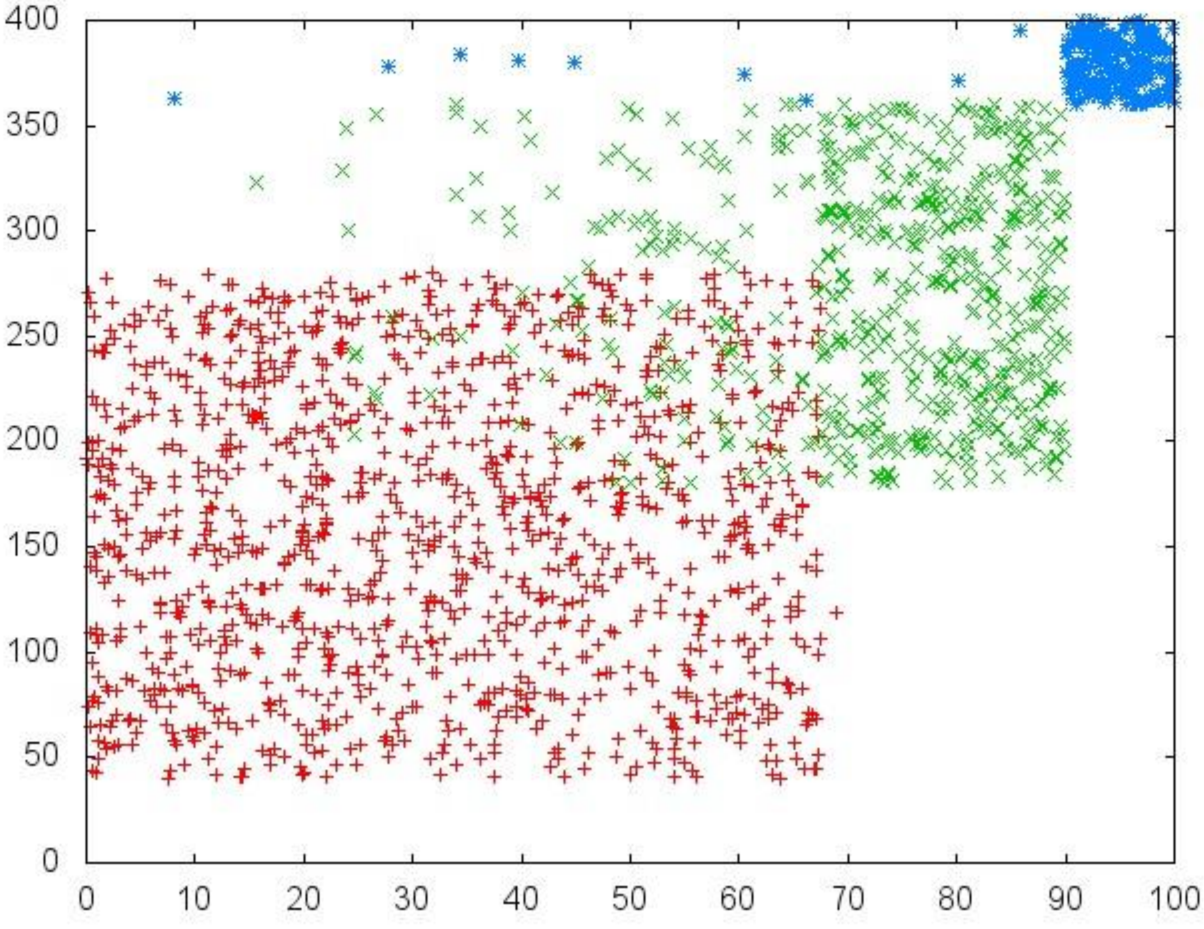
Splitting



e.

Learned Strategies

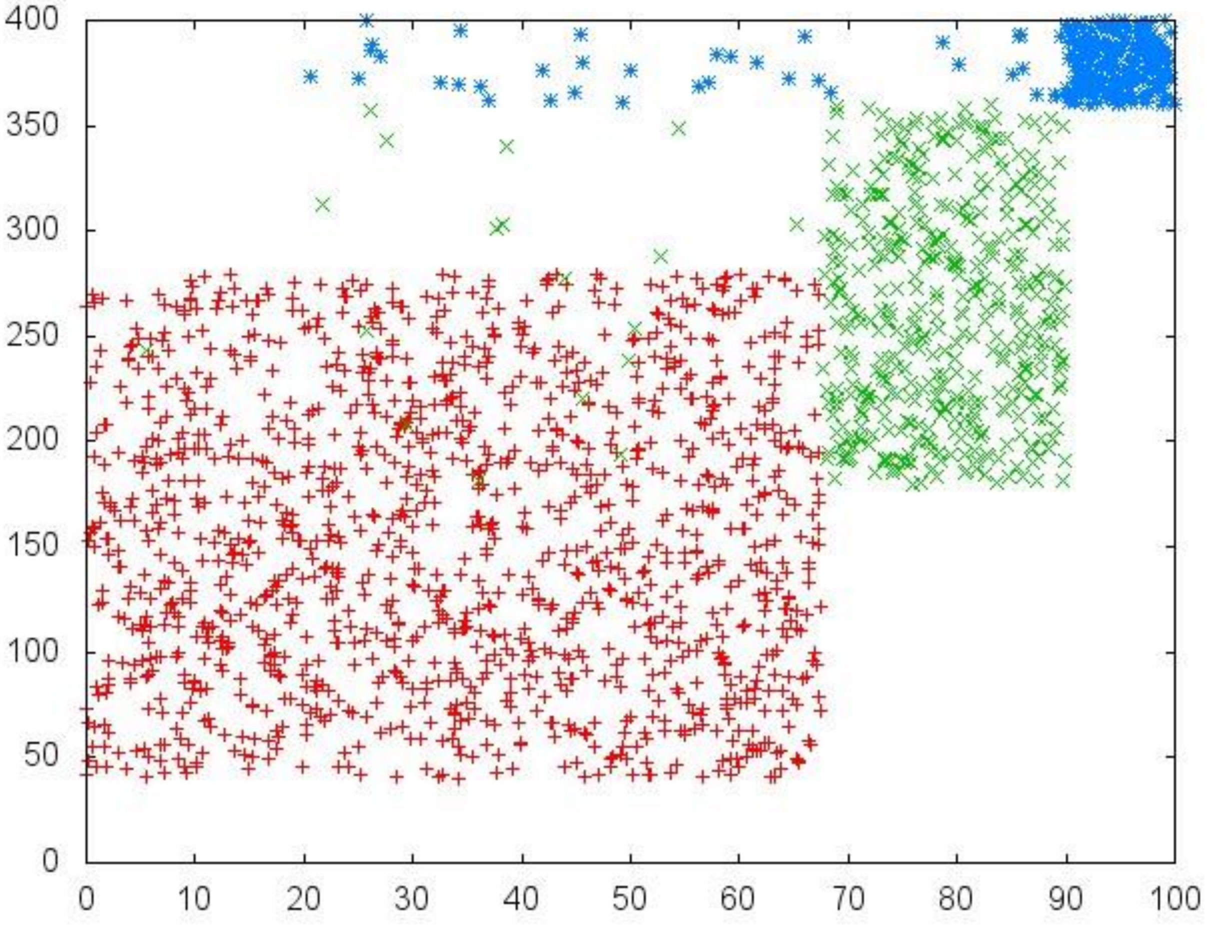
Splitting



e.

Learned Strategies

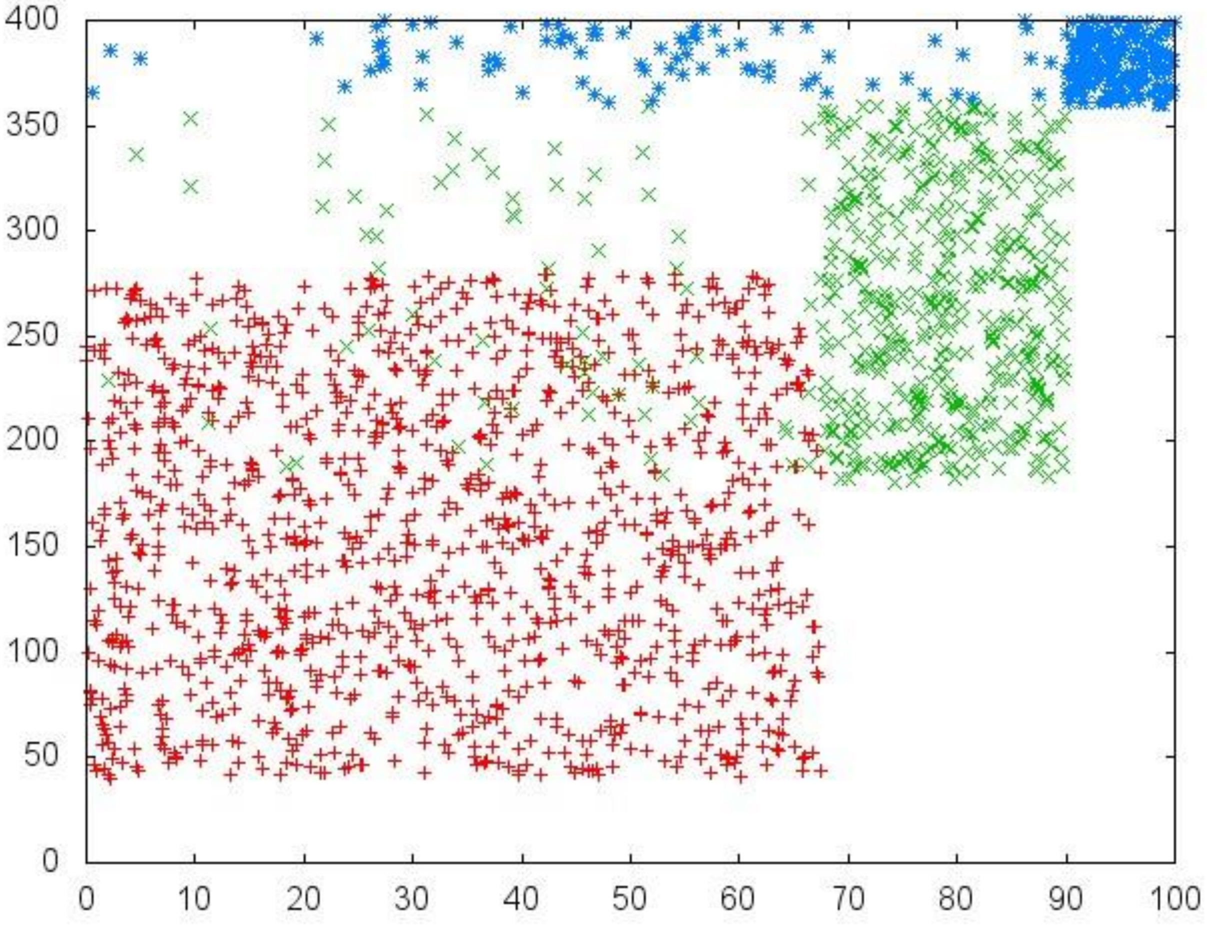
Splitting



e.

Learned Strategies

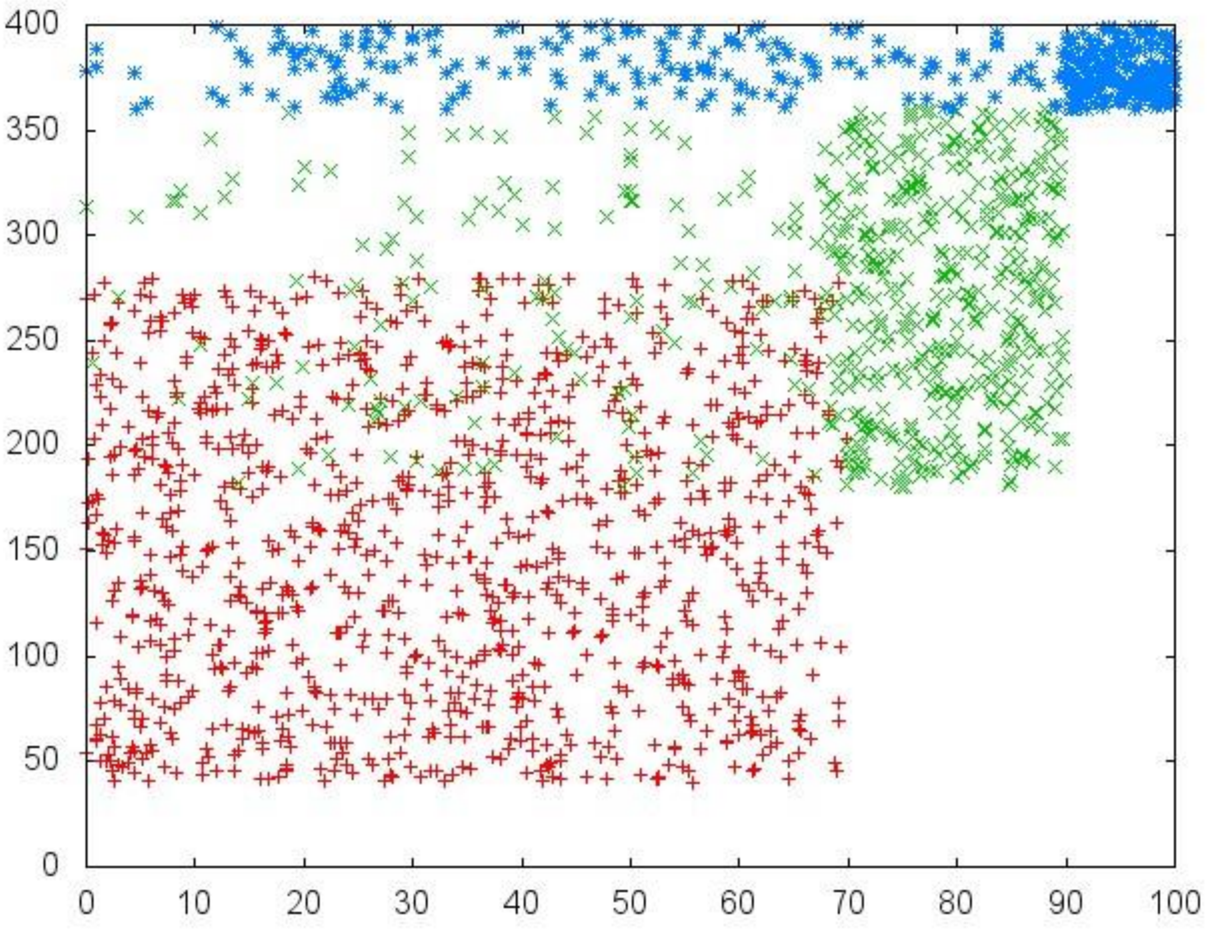
Splitting



e.

Learned Strategies

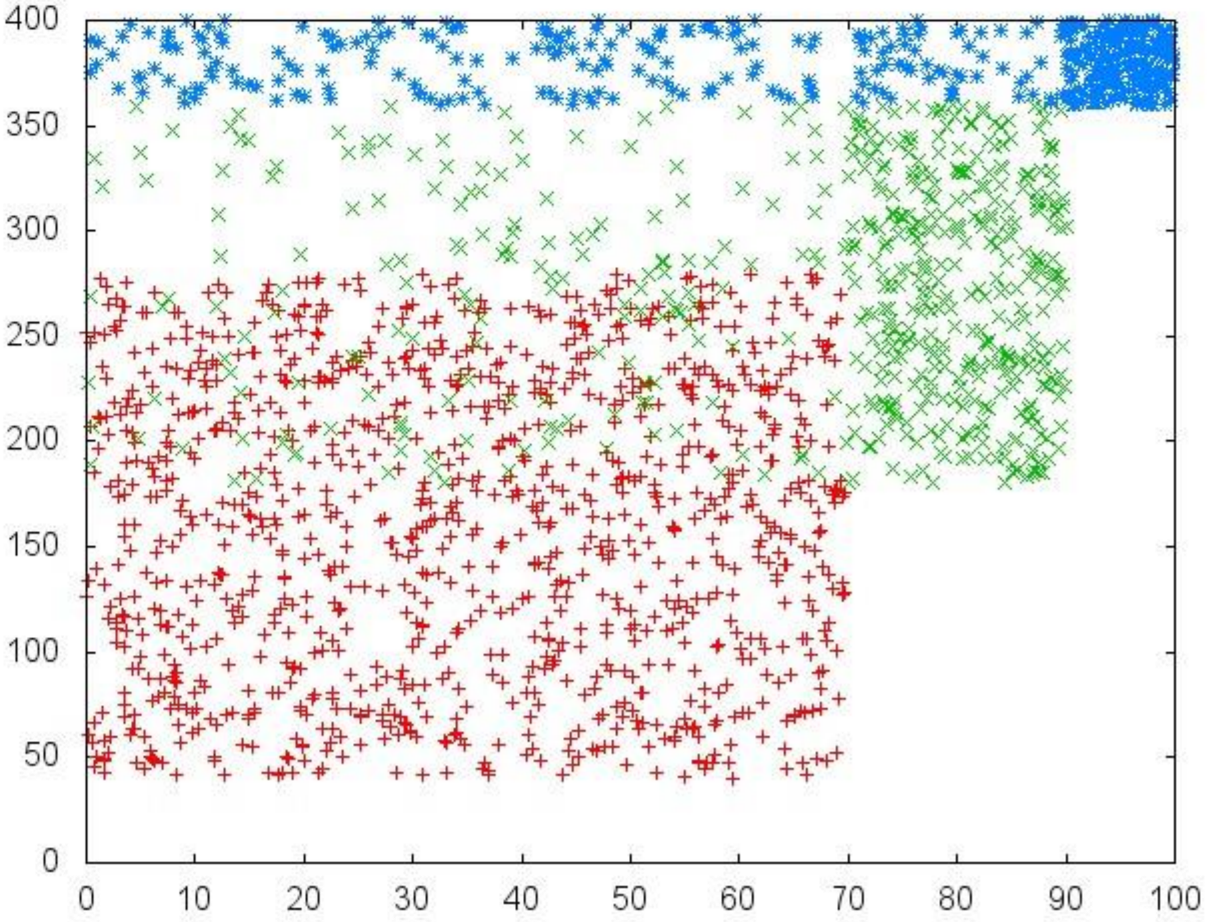
Splitting



e.

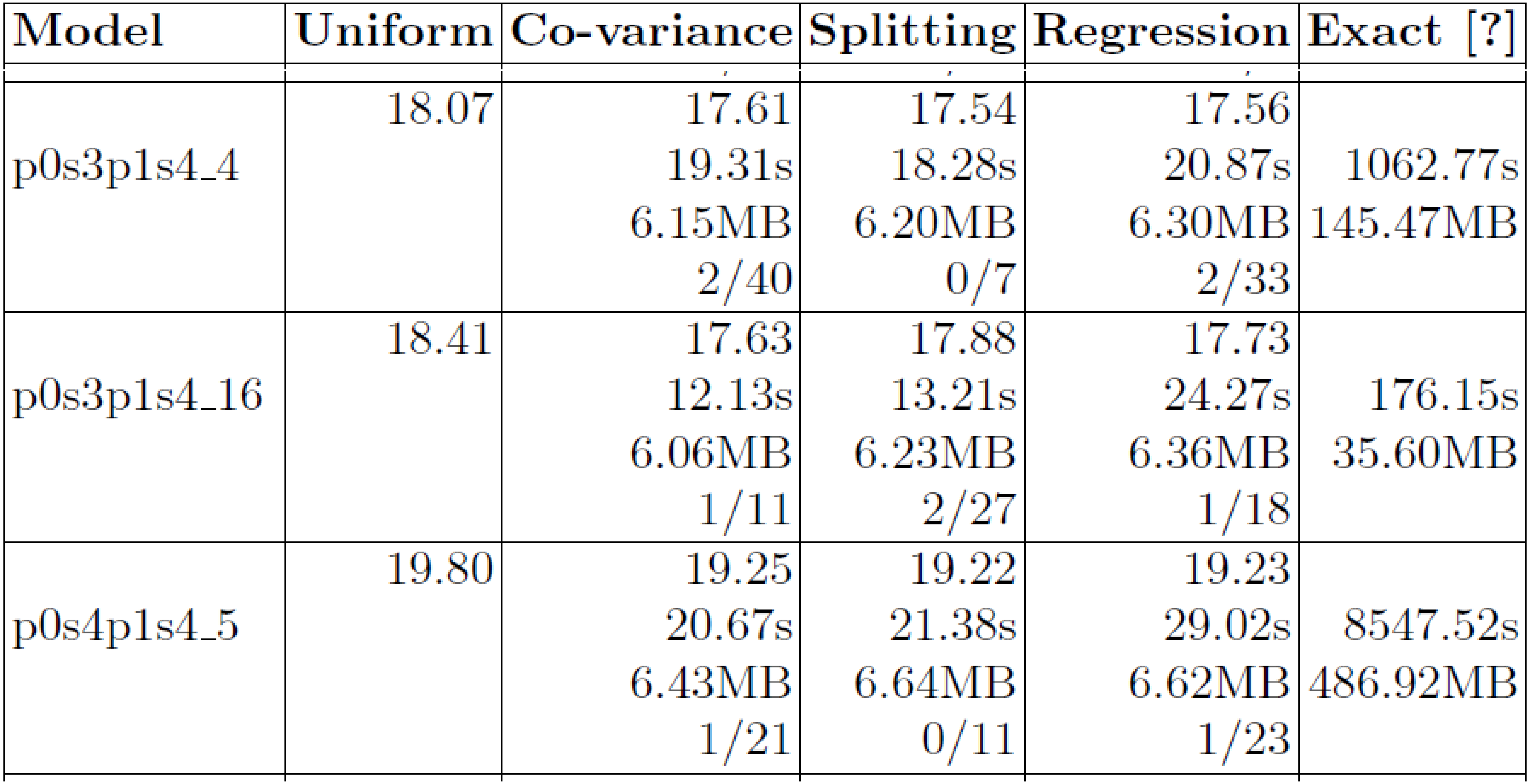
Learned Strategies

Splitting



e.

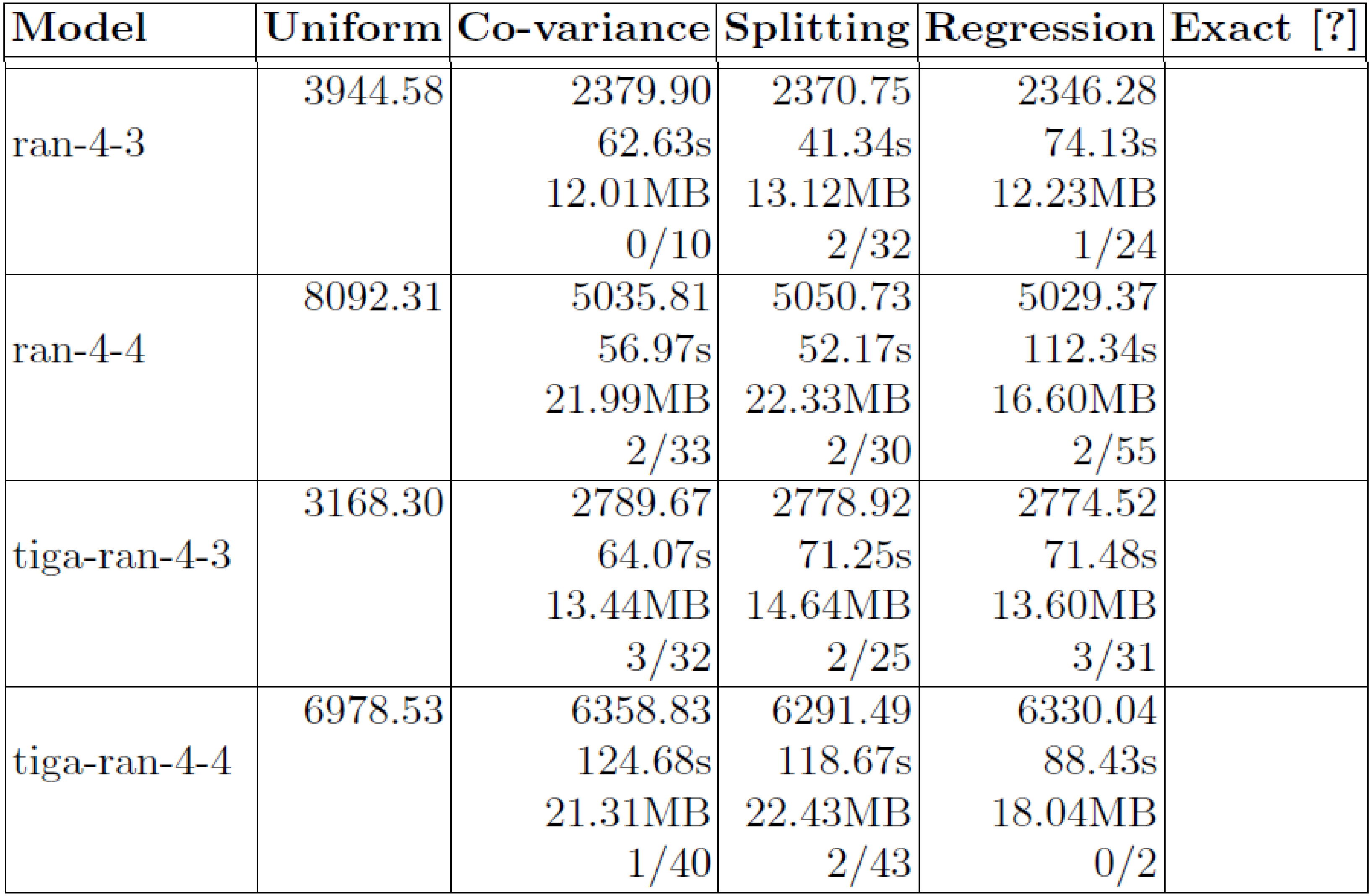
Experiments /DPA



Model	Uniform	Co-variance	Splitting	Regression	Exact [?]
p0s3p1s4_4	18.07	17.61 19.31s 6.15MB 2/40	17.54 18.28s 6.20MB 0/7	17.56 20.87s 6.30MB 2/33	1062.77s 145.47MB
p0s3p1s4_16	18.41	17.63 12.13s 6.06MB 1/11	17.88 13.21s 6.23MB 2/27	17.73 24.27s 6.36MB 1/18	176.15s 35.60MB
p0s4p1s4_5	19.80	19.25 20.67s 6.43MB 1/21	19.22 21.38s 6.64MB 0/11	19.23 29.02s 6.62MB 1/23	8547.52s 486.92MB

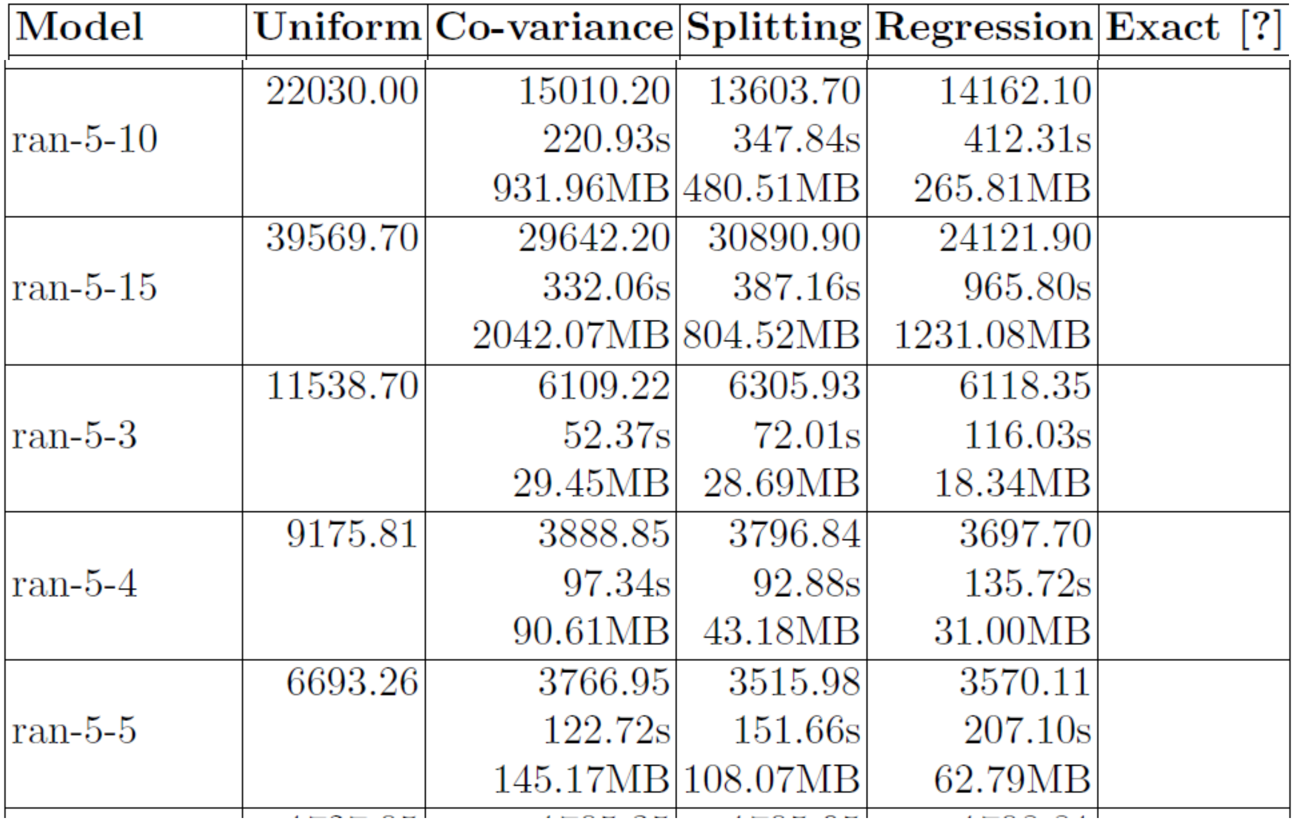
Kempf, J.F., Bozga, M., Maler, O.: As soon as probable: Optimal scheduling under stochastic uncertainty. In: TACAS. pp. 385{400 (2013)
http://www-verimag.imag.fr/PROJECTS/TEMPO/DATA/201304_dpa/

Experiments /DPA Random



Model	Uniform	Co-variance	Splitting	Regression	Exact [?]
ran-4-3	3944.58	2379.90 62.63s 12.01MB 0/10	2370.75 41.34s 13.12MB 2/32	2346.28 74.13s 12.23MB 1/24	
ran-4-4	8092.31	5035.81 56.97s 21.99MB 2/33	5050.73 52.17s 22.33MB 2/30	5029.37 112.34s 16.60MB 2/55	
tiga-ran-4-3	3168.30	2789.67 64.07s 13.44MB 3/32	2778.92 71.25s 14.64MB 2/25	2774.52 71.48s 13.60MB 3/31	
tiga-ran-4-4	6978.53	6358.83 124.68s 21.31MB 1/40	6291.49 118.67s 22.43MB 2/43	6330.04 88.43s 18.04MB 0/2	

Experiments /DPA Random



Model	Uniform	Co-variance	Splitting	Regression	Exact [?]
ran-5-10	22030.00	15010.20 220.93s 931.96MB	13603.70 347.84s 480.51MB	14162.10 412.31s 265.81MB	
ran-5-15	39569.70	29642.20 332.06s 2042.07MB	30890.90 387.16s 804.52MB	24121.90 965.80s 1231.08MB	
ran-5-3	11538.70	6109.22 52.37s 29.45MB	6305.93 72.01s 28.69MB	6118.35 116.03s 18.34MB	
ran-5-4	9175.81	3888.85 97.34s 90.61MB	3796.84 92.88s 43.18MB	3697.70 135.72s 31.00MB	
ran-5-5	6693.26	3766.95 122.72s 145.17MB	3515.98 151.66s 108.07MB	3570.11 207.10s 62.79MB	

Conclusion & Future Work

- Efficient synthesis of strategies for PTMDP ensuring time-bounds and minimizing expected cost.
- If not time-bound needed we can omit the UPPAAL TIGA synthesis
- Extension to Hybrid MDPs utilizing UPPAAL SMCs support for SHAs.

- Make TIGA/SMC available to you!
- Datastructures supporting general stochastic strategies – not just non-lazy ones.
- More clever filtrations of runs.