Learning Monotone Partitions of Partially-Ordered Domains

Oded Maler

VERIMAG
CNRS and the University of Grenoble (UGA)
France

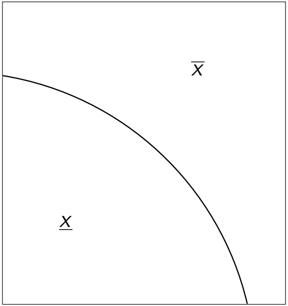
July 2017
Massif de Cahrtreuse

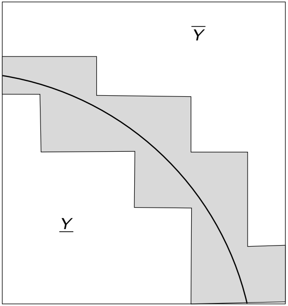
Setting

- Let X be a bounded and partially ordered set, say $[0,1]^n$
- A subset $\overline{X}$ of X is *upward closed* if

$$\forall x, x' \in X \; (x \in \overline{X} \wedge x' \geq x) \rightarrow x' \in \overline{X}$$

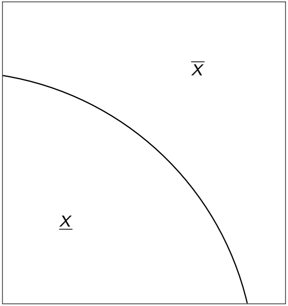
- The complement $\underline{X} = X - \overline{X}$ is downward-closed
- Together they form a *monotone partition* $M = (\underline{X}, \overline{X})$ of X

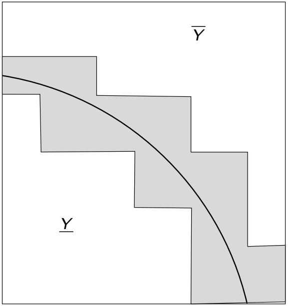




Learning the Partition from Queries

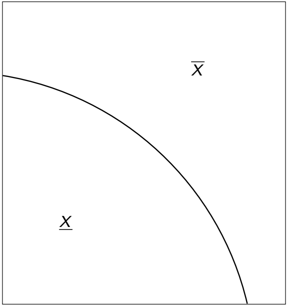
- We do not have an explicit representation of the boundary
- We can pose queries to a membership oracle that can answer whether $x \in \overline{X}$ for any x
- Based on this sampling we build an approximation $M' = (\underline{Y}, \overline{Y})$ of the partition with $\underline{Y} \subseteq \underline{X}$, $\overline{Y} \subseteq \overline{X}$
- There is a remaining gap for which we do not know, it is an over-approximation of the partition boundary

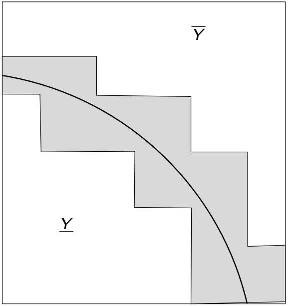




Motivation I: Multi-Criteria Optimization

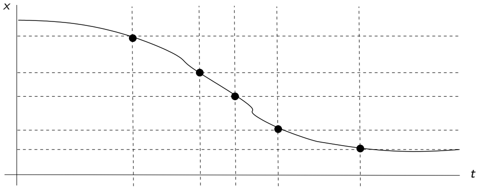
- X is the cost space and $\overline{X}$ are feasible costs (in minimization)
- The boundary is the Pareto front of the problem
- We ask a solver whether some costs are feasible or not and use the information to construct a approximation of the front (thesis of Julien Legriel)





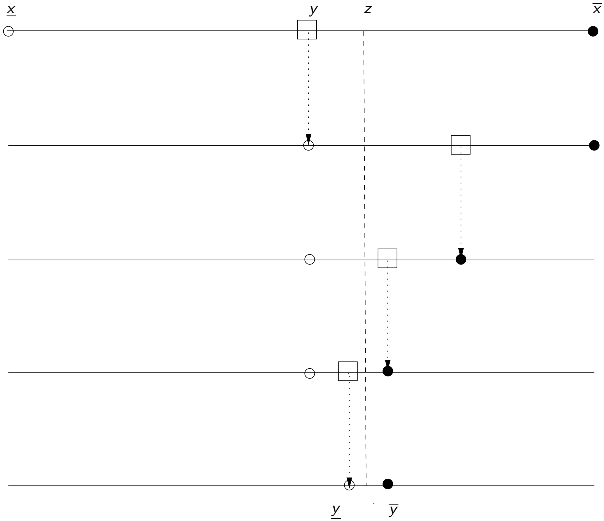
Motivation II: Parametric Identification

- A parameterized family of predicates/constraints $\{\varphi_p\}$ where p is a vector of parameters
- Example: $u(t)$ is real-valued signal that should stabilize below p_2 within p_1 time: $\exists t < p_1 \; u(t) < p_2$ or in STL $F_{[0,p_1]} u < p_2$
- Find the range of parameters that make φ_p satisfied by a given u
- Under certain assumptions (no parameter appear in opposite sides of inequalities) the set can be made upward closed and the boundary gives the set of tightest parameters



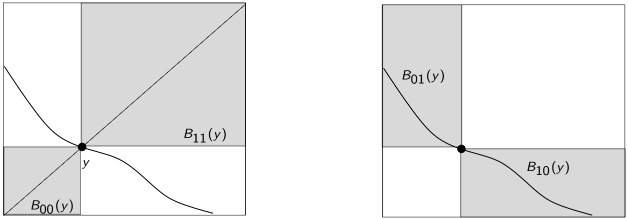
Binary Search in One Dimension

- In one dimension $M = ([0, z), [z, 1])$, for some $0 < z < 1$
- The boundary can be found/approximated by binary search



In Higher Dimension

- The intersection of the diagonal of the rectangle with the boundary can be found by one-dimensional binary search
- Due to monotonicity, the rectangle above y is in $\overline{X}$ and the one below it is in $\underline{X}$
- The boundary approximation is refined into the union of incomparable rectangles



The whole Algorithm

- Maintain a list of rectangles whose union contains the boundary
- Each time pick one rectangle (the fattest), run binary search on its diagonal and refine it
- Problem: number of incomparable rectangles is $2^n - 2$
- Theoretical and empirical complexity under investigation