Active Learning for Abstract Models of Collectives

Alexander Schiendorfer, Christoph Lassner, Gerrit Anders, Wolfgang Reif, Rainer Lienhart

fertilized forests

Universität Augsburg University

Motivation

Institute for Software & Systems Engineering

- Adaptive systems used in resource-intensive domains
  - Unit commitment in virtual power plants (Schiendorfer et al., 2014)
  - Load balancing
- Efficient algorithms for resource allocation central to business processes
  - Discrete Optimization
  - Constraint/Mathematical Programming
  - ...
- Hierarchical decomposition desired
  - Abstraction based on *sampling* points of a collective
  - Which sampling points to select?

Resource Allocation Problems (RAP)

Institute for Software & Systems Engineering

General resource allocation (Van Zandt, 1995)

- One homogeneous resource – the *demand* x_R (continuous or discrete)
- e.g., Energy, Computing Resources (CPU time required for a job), Technical parts to be produced, in our model ...
- Provided by several agents x_i at different costs

$$\underset{\langle x_1,\ldots,x_n\rangle}{\text{minimize}} \quad \sum_{i=1}^{n} c_i(x_i) \quad \text{subject to} \quad \sum_{i=1}^{n} x_i = x_R$$

How to calculate the solution?

Institute for
Software & Systems
Engineering

Variation of general resource allocation

- Demand is given for a time frame $\mathcal{T}$
- Agents are subject to constraints such as minimal or maximal rates and *inertia*
- Cost function per plant, per production level

$$
\begin{aligned}
\underset{\mathcal{P}}{\text{minimize}} \quad & \sum_{t \in \mathcal{T}} \sum_{a \in \mathcal{A}} \kappa(\mathcal{P}_t^a) \\
\text{subject to} \quad & \forall a \in \mathcal{A}, t \in \mathcal{T} : \\
& \exists [x, y] \in L^a : x \leq \mathcal{P}_t^a \leq y, \\
& \sum_{t \in \mathcal{T}} |\mathcal{D}_t - \sum_{a \in \mathcal{A}} \mathcal{P}_t^a| \\
& [\ldots]
\end{aligned}
$$

How to solve this RAP?

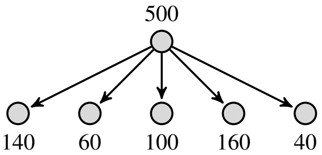
Institute for Software & Systems Engineering

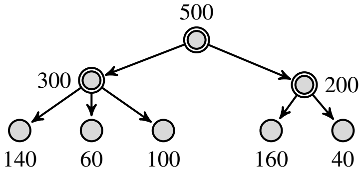
Constraint-Satisfaction-Optimisation-Problem (CSOP)

- Declarative and *versatile* approach for combinatorial problems
- Specify problem in terms of *variables*, *domains* and *constraints*
- Solve with Mixed-Integer-Programming (MIP), Linear Programming (LP), Constraint Programming (CP), etc.

Scalability issues with increasing numbers of agents!

Hierarchical System Structures



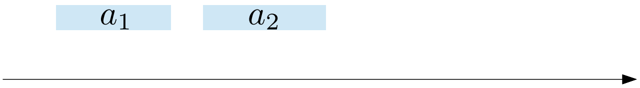


- Example: Unit Commitment in Power Management
- Energy offered by Suppliers
- Providers are organized hierarchically in *virtual power plants*)
  - Scalability benefits
  - Local uncertainty compensation mechanisms (scenarios, robust optimization (Anders et al., 2014))
- Abstraction reduces the complexity of the model of a collective's behavior

Abstraction

- Frame feasible regions: Find the corners and holes of the production space of a whole collective.
  - Done by means of interval arithmetics
  - Propagate to upper levels
- Approach based on sampling
  - Assume functional relationships
  - Approximation by iterated solving of slightly modified optimization problems and collecting input-output pairs *("Sampling")*
  - Using this approximation on higher levels

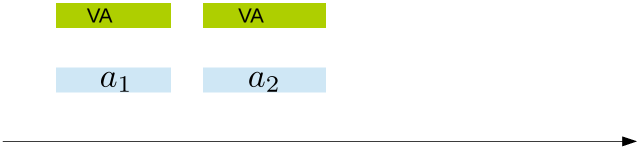
General Abstraction – Intuition



Possible production

$a_1 = \{[0,0],[1,4]\}, a_2 = \{[0,0][7,10]\}$

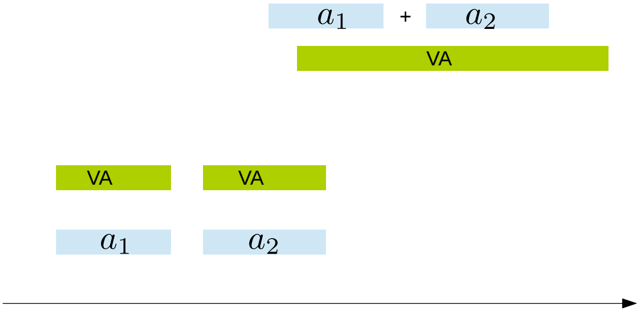
General Abstraction – Take individuals



Possible production

$\text{VA} \supset \{[0,0],[1,4],[7,10]\}$

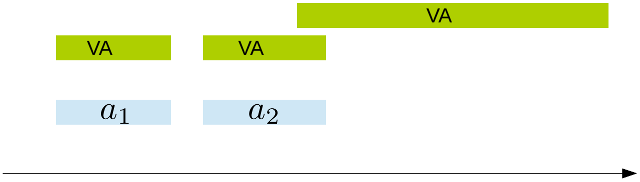
General Abstraction – Combine



Possible production

$[1, 4] \oplus [7, 10] = [8, 14]$

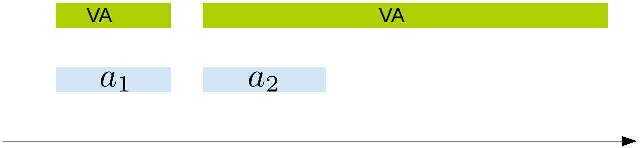
General Abstraction – Possible Overlaps



Possible production

$\mathrm{merge}(\{[0,0],[1,4],[7,10]\} \cup \{[8,14]\})$

General Abstraction – Merge



Possible production

$\text{merge}(\{[0,0],[1,4],[7,10]\} \cup \{[8,14]\}) = \{[0,0],[1,4],[7,14]\}$

Data collection

Institute for Software & Systems Engineering

1. Solve for *minimize* costs, given that the combined production is 0

   obtained by $e_{vp} = \underbrace{e_{bm}}_{0} + \underbrace{e_{hy}}_{0}$, costs: 0

2. Solve for *minimize* costs, given that the combined production is 50

   obtained by $e_{vp} = \underbrace{e_{bm}}_{25} + \underbrace{e_{hy}}_{25}$, costs: 400

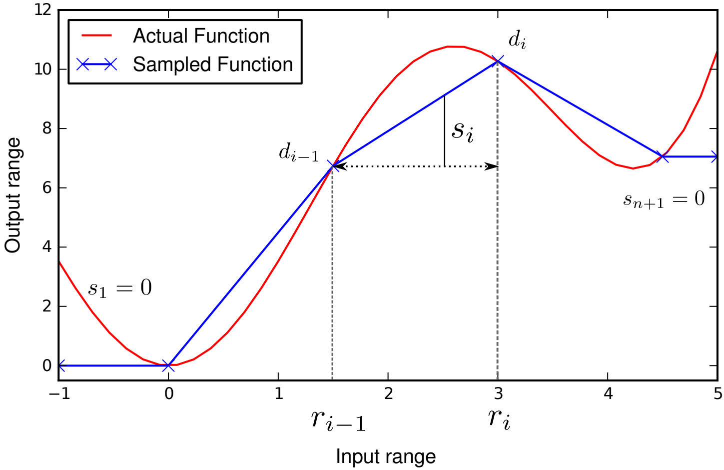
3. Solve for *minimize* costs, given that the combined production is 55

   obtained by $e_{vp} = \underbrace{e_{bm}}_{35} + \underbrace{e_{hy}}_{20}$, costs: 550

4. $\cdots$

Approximation with piecewise linear functions

Institute for Software & Systems Engineering



An example

Institute for Software & Systems Engineering

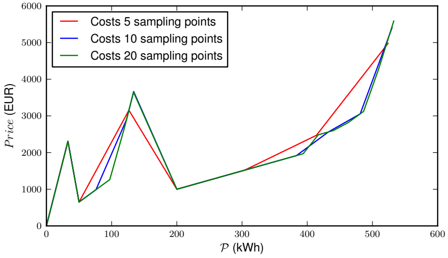
$$L_p = \langle [0,0], [50,100] \rangle, \qquad \kappa^p = 13$$
$$L_q = \langle [0,0], [15,35] \rangle, \qquad \kappa^q = 70$$
$$L_r = \langle [0,0], [200,400] \rangle, \qquad \kappa^r = 5$$

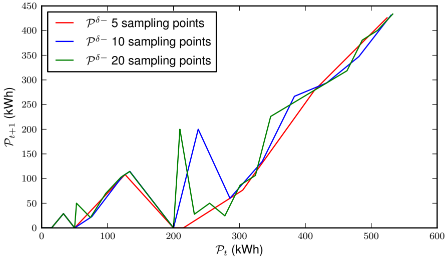
Three power plants p, q, and r controlled by an intermediary; costs are linear coefficients κ^i.

General abstraction yields feasible regions:

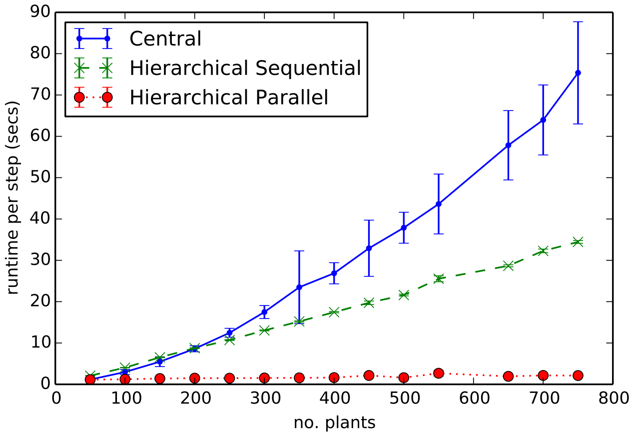
$$\langle [0,0], [15,35], [50,135], [200,535] \rangle$$

Sampling Abstraction – Visualised



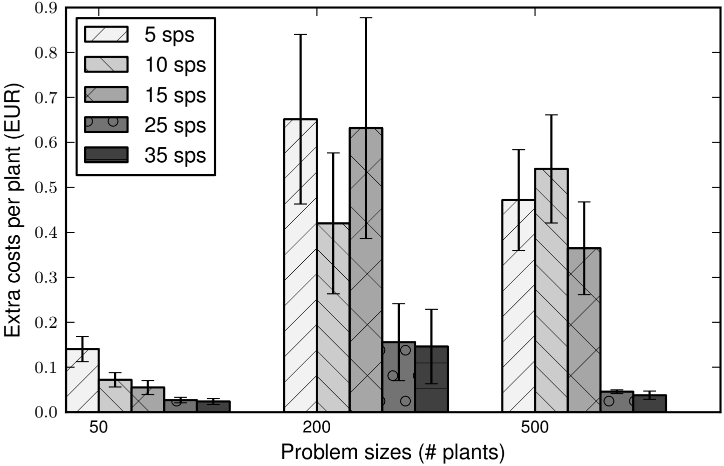


It's all fun and games . . .



Mean runtimes hierarchical vs. central

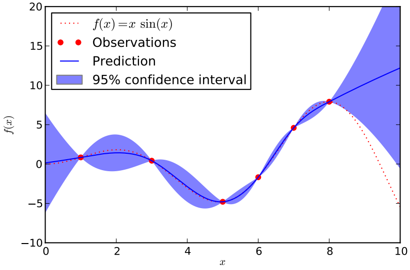
. . . until someone gets hurt



Only using more sampling points does not guarantee to increase accuracy!

Approach

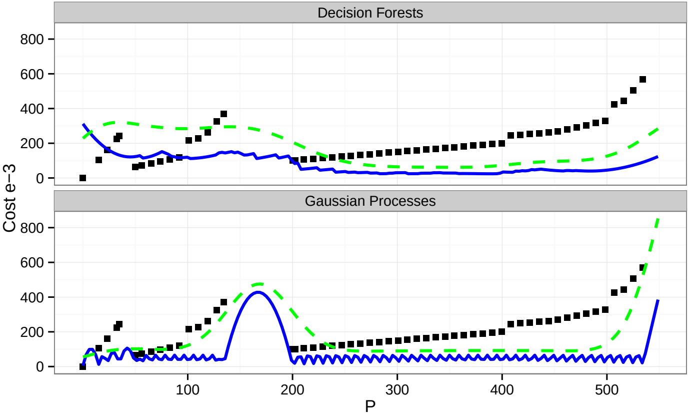
Apply *active learning* techniques (Settles, 2010): A learning algorithm chooses the next input to be labeled.



We investigated two regressors offering probabilistic estimates

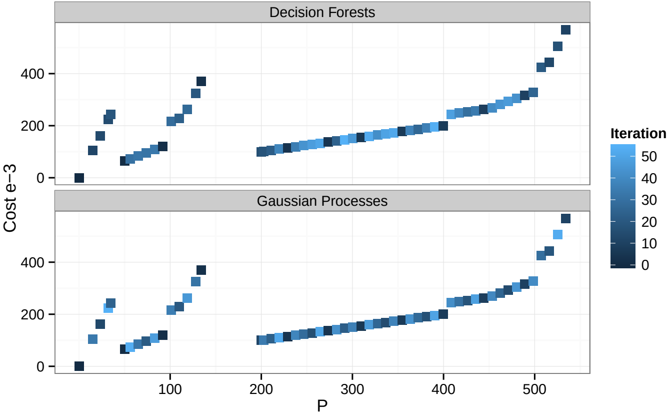
- Gaussian Processes (GPy)
- Decision Forests (Lassner and Lienhart, 2015)

Results I



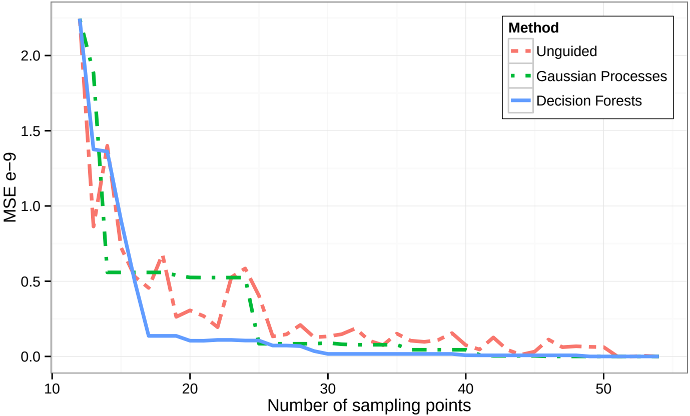
Query point selection by the two evaluated methods.

Results II



Point selection order: The lighter the point color, the later it was selected by the algorithm.

Results III



The mean MSE (scaled with 10^{-9}) for the entire trace is 0.28 (unguided), 0.26 (GP-guided) and 0.18 (DF-guided).

Future and ongoing work

Institute for Software & Systems Engineering

- Apply technique to other sampled functions
- Integrate the informed selector into the overall simulation/optimization algorithm
- Investigate other AL techniques
- Integrate domain-specific knowledge to improve the selection

Bibliography I

Institute for Software & Systems Engineering

Anders, G., Schiendorfer, A., Steghöfer, J.-P., and Reif, W. (2014).

Robust Scheduling in a Self-Organizing Hierarchy of Autonomous Virtual Power Plants.

In Stechele, W. and Wild, T., editors, *Proc. 2nd Int. Wsh. Self-optimisation in Organic and Autonomic Computing Systems (SAOS'14)*, pages 1–8.

Lassner, C. and Lienhart, R. (2015).

Norm-induced entropies for decision forests.

In *Proc. IEEE Winter Conference on Applications of Computer Vision 2015 (WACV'15)*.

Schiendorfer, A., Steghöfer, J.-P., and Reif, W. (2014).

Synthesis and Abstraction of Constraint Models for Hierarchical Resource Allocation Problems.

In *Proc. 6th Int. Conf. Agents and Artificial Intelligence (ICAART'14), Vol. 2*, pages 15–27. SciTePress.

Bibliography II

Settles, B. (2010).
Active learning literature survey.
Computer Sciences Technical Report 1648, University of Wisconsin–Madison.

Van Zandt, T. (1995).
Hierarchical Computation of the Resource Allocation Problem.
European Economic Review, 39(3-4):700–708.