ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Integrazione di Approcci Data-Driven e Knowledge-Based in Sistemi di Supporto alle Decisioni

Michela Milano, Michele Lombardi, Andrea Borghesi, Allegra De Filippo, Fabrizio Detassis, Federico Baldo, Mattia Silvestri

Declarative Optimization

Declarative Optimization

- A family of methods for automatically tackling decision making problems
- In a nutshell:

Declarative Optimization = Model + Solve

Example: what to put in a knapsack?

Declarative Optimization

ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA

Declarative Optimization

- A family of methods for automatically tackling decision making problems
- In a nutshell:

Declarative Optimization = Model + Solve

Example: what to put in a knapsack?

x_i = 1 if the object is taken

pool of objects

$$x_i \in \{0, 1\} \qquad \forall i \in \{1..n\}$$

Declarative Optimization

Declarative Optimization

- A family of methods for automatically tackling decision making problems
- In a nutshell:

Declarative Optimization = Model + Solve

Example: what to put in a knapsack?

weight of object i

$$s.t. \quad \sum_{i=1}^{n} w_i x_i \leq C$$

capacity

$$x_i \in \{0, 1\} \qquad \forall i \in \{1..n\}$$

Declarative Optimization

Declarative Optimization

- A family of methods for automatically tackling decision making problems
- In a nutshell:

Declarative Optimization = Model + Solve

Example: what to put in a knapsack?

profit of object i

$$\max \sum_{i=1}^{n} p_i x_i$$

$$s.t. \sum_{i=1}^{n} w_i x_i \leq C$$

$$x_i \in \{0, 1\} \qquad \forall i \in \{1..n\}$$

Declarative Optimization

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Declarative Optimization

- A family of methods for automatically tackling decision making problems
- In a nutshell:

Declarative Optimization = Model + Solve

Example: what to put in a knapsack?

$$\max \sum_{i=1}^{n} p_i x_i$$

$$s.t. \sum_{i=1}^{n} w_i x_i \leq C$$

$$x_i \in \{0, 1\} \qquad \forall i \in \{1..n\}$$

→ Solution Algorithm →

Declarative Optimization

Declarative Optimization

- A family of methods for automatically tackling decision making problems
- In a nutshell:

Declarative Optimization = Model + Solve

Example: what to put in a knapsack?

$$
\begin{aligned}
max \quad & \sum_{i=1}^{n} p_i x_i \\
s.t. \quad & \sum_{i=1}^{n} w_i x_i \leq C \\
& x_i \in \{0, 1\} \qquad \forall i \ \in \{1..n\}
\end{aligned}
$$

$\rightarrow$ Solution Algorithm $\rightarrow$

ML for Optimization

Traditionally

- Models are designed by experts via trial and error
- ...And the same goes for algorithms

How can data-driven methods help?

- By allowing one to take into account data
- By dealing with uncertainty via statistical approaches

In this talk

- Methods that use ML to Model
- Methods that use ML to Solve
- Methods that use ML to Model + Solve

Only *very* few representative approaches for each class

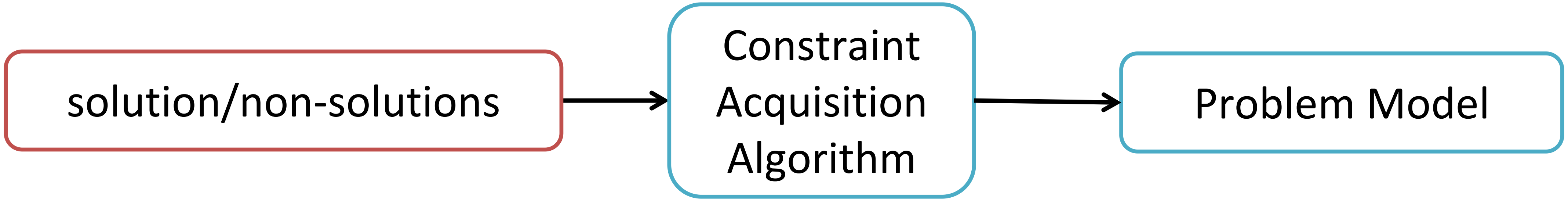
ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

ML for Modeling

Constraint Acquisition

A first class of approaches: Constraint Acquisition

- Start from a collection of solutions (optionally non-solutions)
- Use an algorithm to obtain a declarative model



Constraint Acquisition: an example

Input: solutions + non-solutions

1	2	3
3	1	2
2	3	1

2	3	1
1	2	3
3	1	2

3	1	2
2	3	1
1	2	3

2	3	2
2	1	1
3	2	1

Ouput: declarative model

$$\text{alldiff}\{x_{ij}\ \forall i = 1..3\}\ \forall j \in \{1..3\}$$

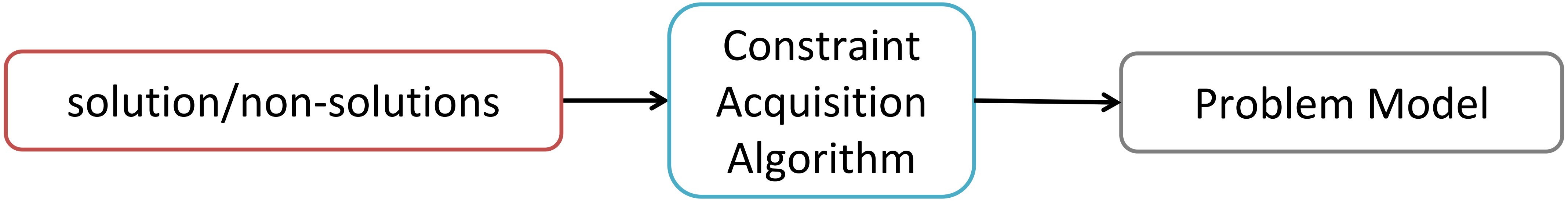
$$\text{alldiff}\{x_{ij}\ \forall j = 1..3\}\ \forall i \in \{1..3\}$$

In other words: never twice the same number on each row & column

Constraint Acquisition

A first class of approaches: Constraint Acquisition

- Start from a collection of solutions (optionally non-solutions)
- Use an algorithm to obtain a declarative model



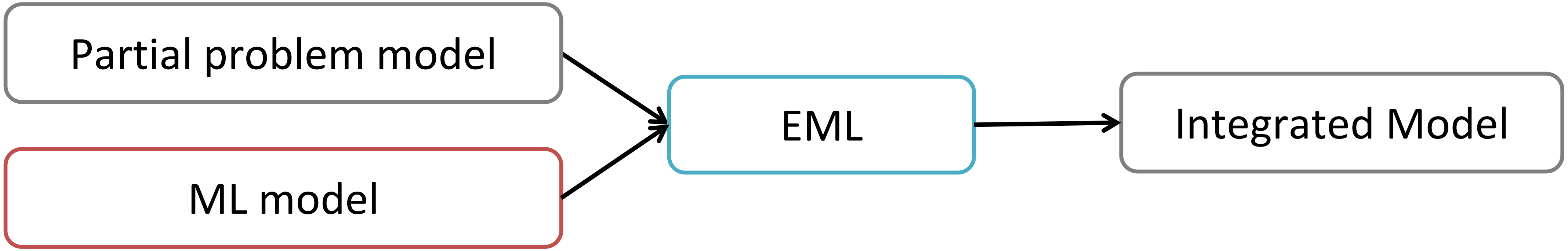
Some comments

- Scalability is still an issue
- Need to choose the pool of avaialable constraints
- Some data and proble constraints can be provided by a user

Empirical Model Learning

A second class of approaches: Empirical Model Learning

- Start from: an incomplete declarative model + a ML model
- Embed the ML model in the declarative model



Empirical Model Learning

Empirical Model Learning: an Example

Input: partial problem model

$$s.t. \quad \sum_{i=1}^{n} w_i x_i \leq C$$

$$x_i \in \{0, 1\} \qquad \forall i \in \{1..n\}$$

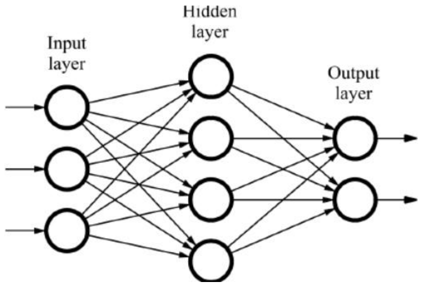
$\longrightarrow$

Ouput: integrated model

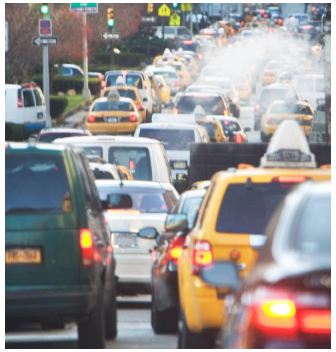
$$max \quad f(x)$$

$$s.t. \quad \sum_{i=1}^{n} w_i x_i \leq C$$

$$x_i \in \{0, 1\} \qquad \forall i \in \{1..n\}$$



$$y = f(x)$$

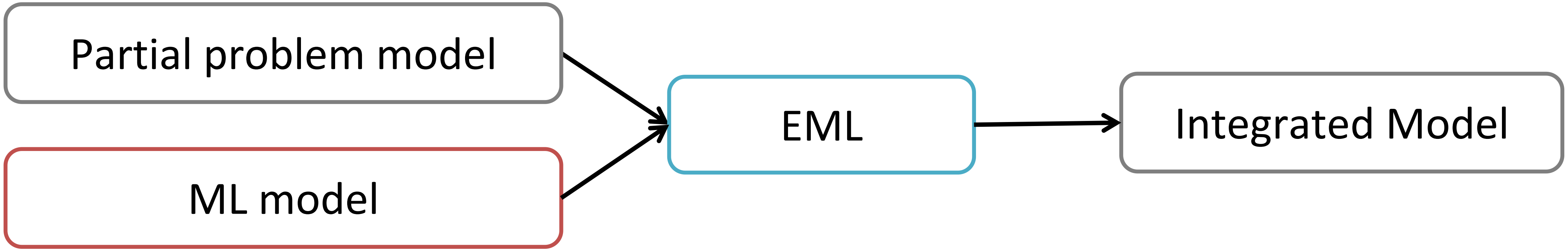


Input: ML model to estimate impact on traffic

Empirical Model Learning

A second class of approaches: Empirical Model Learning

- Start from: an incomplete declarative model + a ML model
- Embed the ML model in the declarative model



Some comments

- Scales much better, but large models still an issue
- Enables *reasoning* on the ML model
- Can be used for verification of ML models

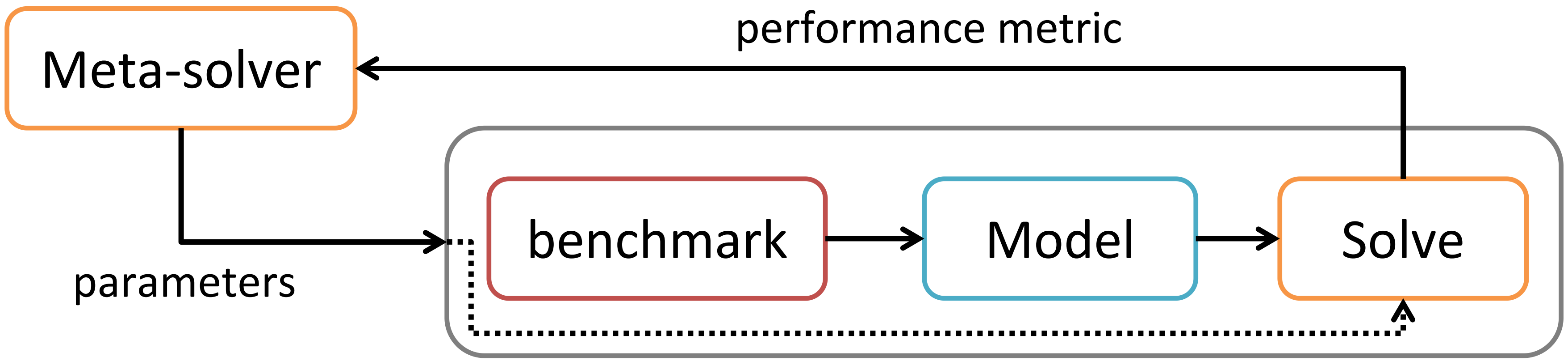
ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

ML for Solving

Algorithm Configuration

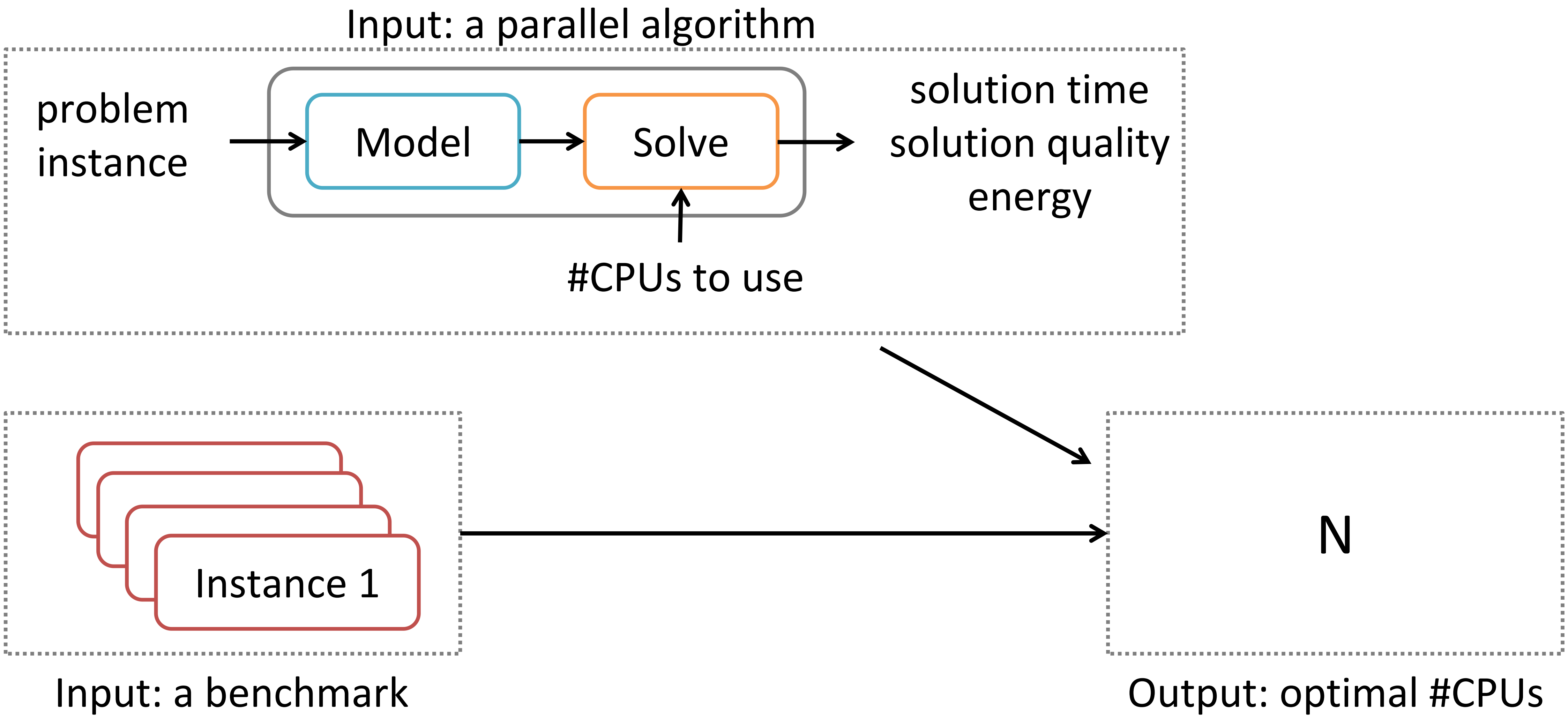
A first class of approaches: Parameter tuning/algorithm configuration

- Find the best solver parameters for a problem class
- Typically: optimize over solver parameters, for a set of benchmark instances



Algorithm Configuration

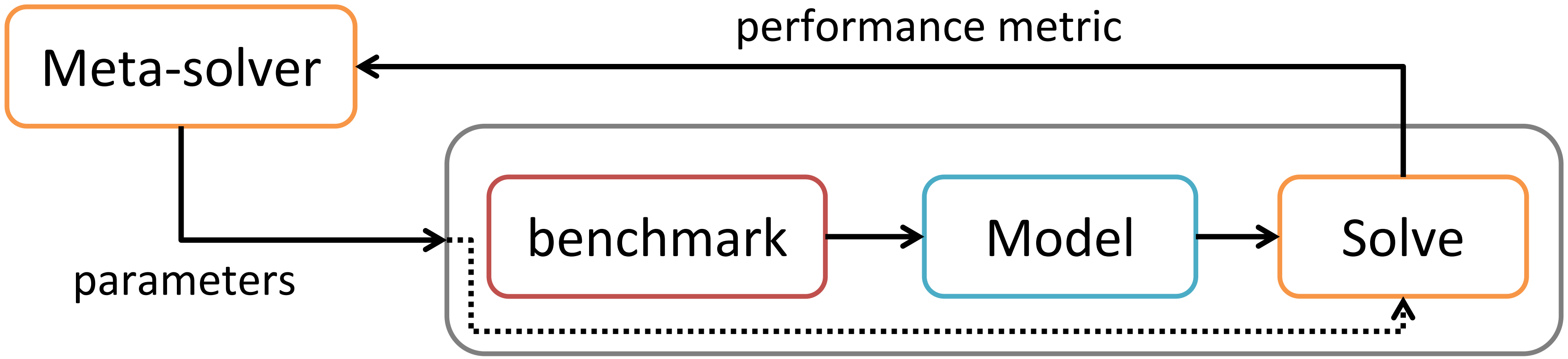
Parameter tuning/algorithm configuration: an example



Algorithm Configuration

A first class of approaches: Parameter tuning/algorithm configuration

- Find the best solver parameters for a problem class
- Typically: optimize over solver parameters, for a set of benchmark instances



Some comments

- Mature tools available
- Basis for Auto-ML
- Connections to similar technique stil underexplored

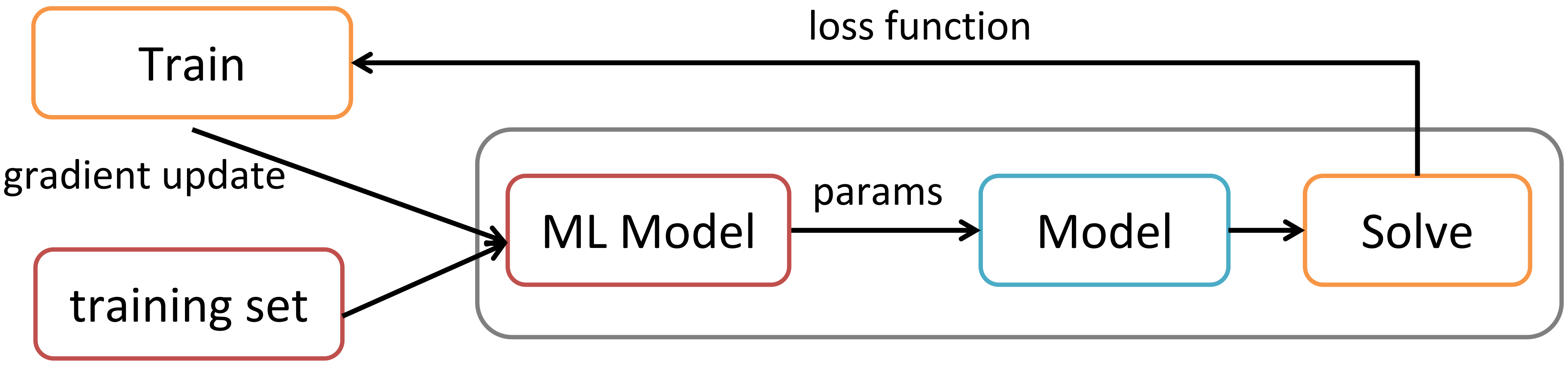
ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

End-to-End Approaches

Smart "Predict, then Optimize"

A first class of approaches: Smart "Predict, then Optimize"

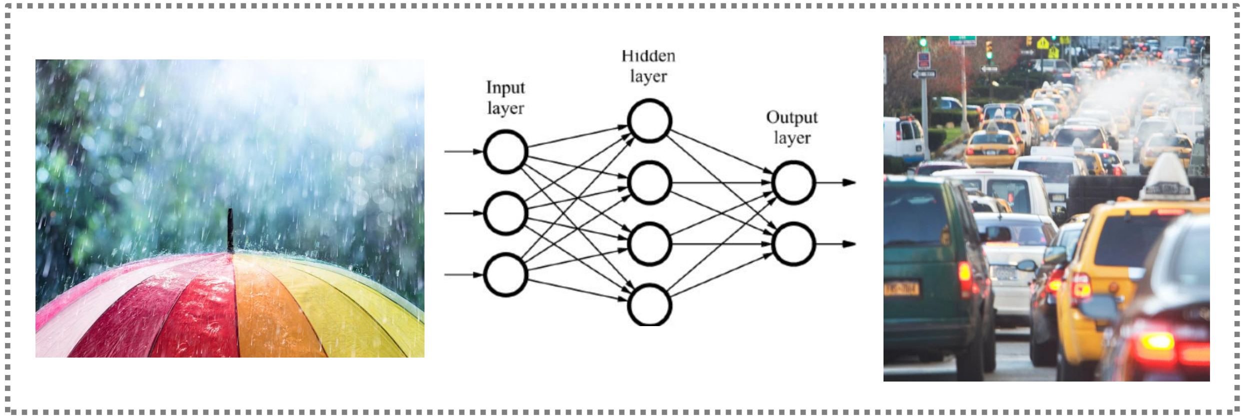
- HP: a ML model estimates model parameters
- Typically: optimize over solver parameters, for a set of benchmark instances

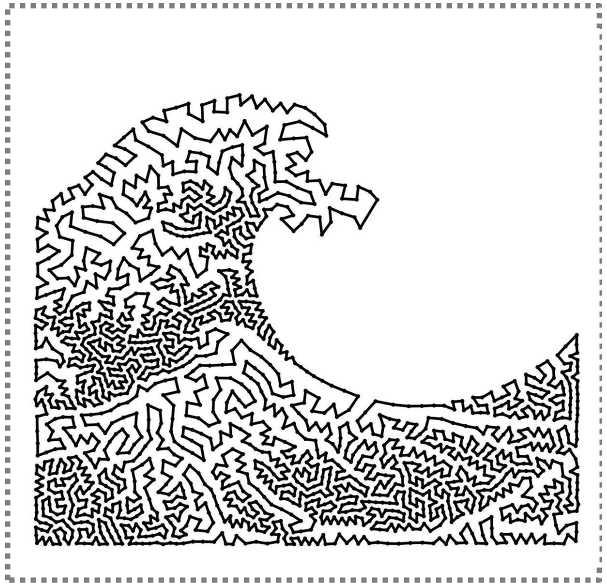


Smart “Predict, then Optimize”

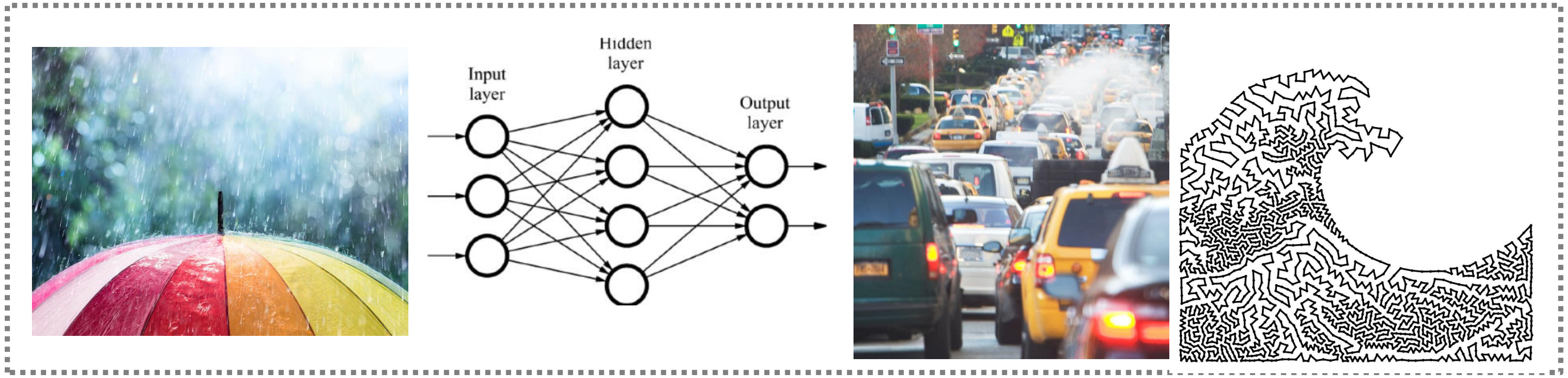
Smart “Predict, then Optimize”: an example

Input: ML model to predict traffic





Input: a TSP solver

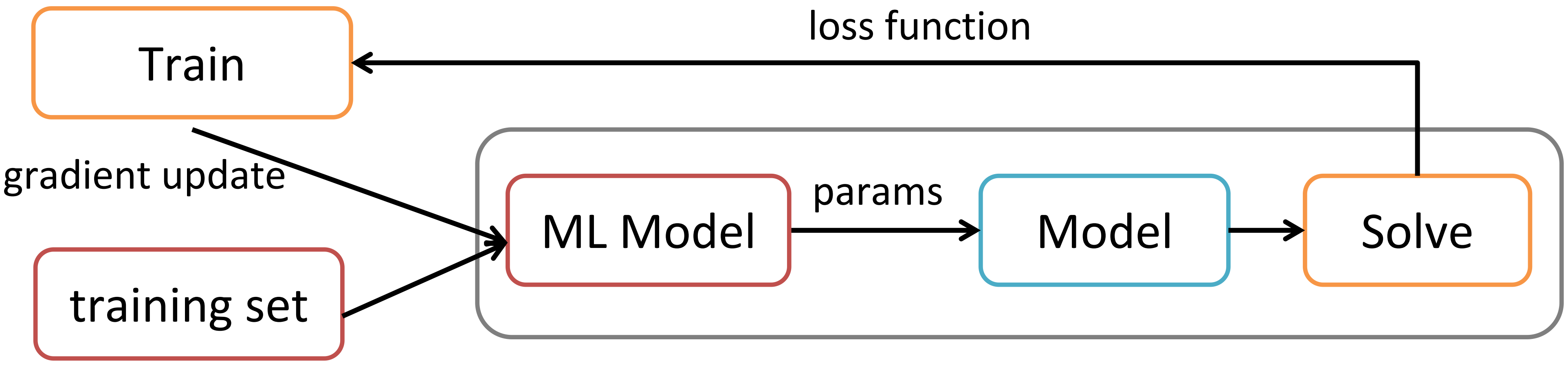


Output: an ML model optimized for the TSP solver

Smart "Predict, then Optimize"

A first class of approaches: Smart "Predict, then Optimize"

- HP: a ML model estimates model parameters
- Typically: optimize over solver parameters, for a set of benchmark instances



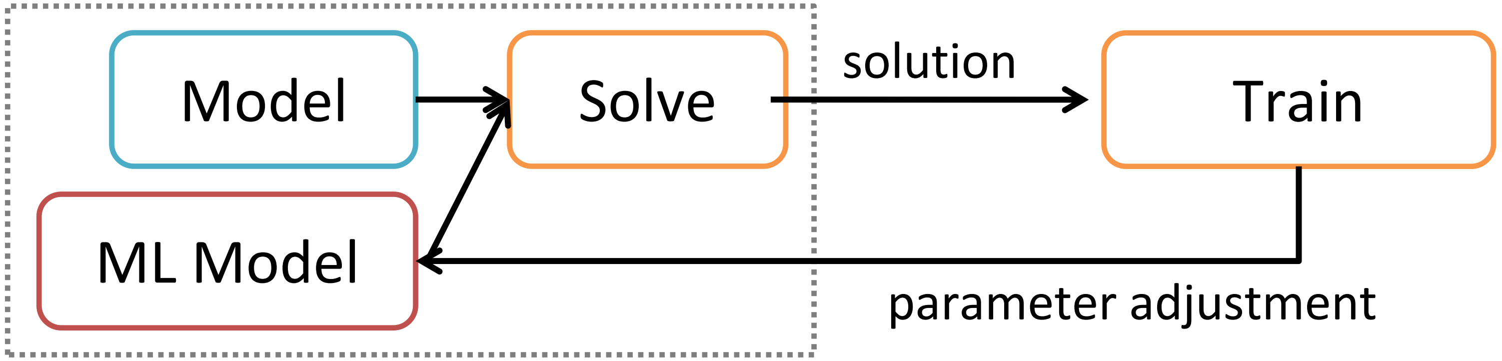
Some comments

- Recent and active area of research
- Decent scalability, but still a bottleneck

“Optimize, then Predict”

A second class of approaches: “Optimize, then Predict”

- Basis: Empirical Model Learning
- Addition: retrain after evaluation (requires experimentation)



Some comments

- Main motivation: adapt to mistakes, be accurate only where needed
- Main challenge: meaningful ML model changes based on few examples
- Preliminary results, but still open

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Refs and Notable Omissis

Some References

ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA

Data Driven Methods to obtain declarative optimization models

Christian Bessiere, Frédéric Koriche, Nadjib Lazaar, Barry O'Sullivan: Constraint acquisition. Artif. Intell. 244: 315-342 (2017)

Christian Bessière, Remi Coletta, Emmanuel Hébrard, George Katsirelos, Nadjib Lazaar, Nina Narodytska, Claude-Guy Quimper, Toby Walsh, "Constraint Acquisition via Partial Queries",IJCAI 2013

Robin Arcangioli, Christian Bessiere, Nadjib Lazaar: Multiple Constraint Acquisition. IJCAI 2016: 698-704

Arnaud Lallouet, Matthieu Lopez, Lionel Martin, Christel Vrain: On Learning Constraint Problems. ICTAI (1) 2010: 45-52

Nicolas Beldiceanu, Helmut Simonis: A Model Seeker: Extracting Global Constraint Models from Positive Examples. CP 2012

Chan, Timothy CY, Taewoo Lee, and Daria Terekhov. "Inverse optimization: Closed-form solutions, geometry, and goodness of fit." Management Science 65.3 (2019): 1115-1135.

Some References

ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA

Optimizing declarative model than include ML models

Laguna, Manuel. "Optimization of complex systems with OptQuest." A White Paper from OptTek Systems, Inc (1997).

Michele Lombardi, Michela Milano, Andrea Bartolini: Empirical decision model learning. Artif. Intell. 244: 343-367 (2017)

Michele Lombardi, Stefano Gualandi: A lagrangian propagator for artificial neural networks in constraint programming. Constraints An Int. J. 21(4): 435-462 (2016)

Michele Lombardi, Michela Milano, Andrea Bartolini: Empirical decision model learning. Artif. Intell. 244: 343-367 (2017)

Matteo Fischetti, Jason Jo: Deep Neural Networks as 0-1 Mixed Integer Linear Programs: A Feasibility Study

Katz, Guy, et al. "Reluplex: An efficient SMT solver for verifying deep neural networks", CAV 2017

Andrei Legtchenko, Arnaud Lallouet: Consistency for Partially Defined Constraints. CP 2005

Some References

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Optimizing declarative model than include ML models

Sicco Verwer, Yingqian Zhang, Qing Chuan Ye: Auction optimization using regression trees and linear models as integer programs. Artif. Intell. 244: 368-395 (2017)

Alexander Thebelt, Jan Kronqvist, Miten Mistry, Robert M. Lee, Nathan Sudermann-Merx, Ruth Misener: ENTMOOT: A Framework for Optimization over Ensemble Tree Models. CoRR abs/2003.04774 (2020)

Algorithm selection and configuration

Lars Kotthoff: Algorithm Selection for Combinatorial Search Problems: A Survey.

Gent, Ian P., Christopher A. Jefferson, Lars Kotthoff, Ian Miguel, Neil Moore, Peter Nightingale, and Karen E. Petrie. “Learning When to Use Lazy Learning in Constraint Solving.” In ECAI2010.

Kadioglu, Serdar, Yuri Malitsky, Meinolf Sellmann, and Kevin Tierney. “ISAC – Instance-Specific Algorithm Configuration.” In ECAI2010.

Xu, Lin, Frank Hutter, Holger H. Hoos, and Kevin Leyton-Brown. “SATzilla: Portfolio-Based Algorithm Selection for SAT.” J. Artif. Intell. Res. (JAIR) 32 (2008)

Some References

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Algorithm selection and configuration (continues)

Kotthoff, Lars. “Hybrid Regression-Classification Models for Algorithm Selection.” In 20th European Conference on Artificial Intelligence, 480–85, 2012.

Kotthoff, Lars. “Ranking Algorithms by Performance.” In LION 8, 2014.

Kadioglu, Serdar, Yuri Malitsky, Ashish Sabharwal, Horst Samulowitz, and Meinolf Sellmann. “Algorithm Selection and Scheduling.” In CP2011.

Stergiou, Kostas. “Heuristics for Dynamically Adapting Propagation in Constraint Satisfaction Problems.” AI Commun. 22, no. 3 (2009): 125–41.

Frank Hutter and Marius Lindauer, “Algorithm Configuration: A Hands on Tutorial”, AAAI 2016

Xu, Lin, Holger H. Hoos, and Kevin Leyton-Brown. “Hydra: Automatically Configuring Algorithms for Portfolio-Based Selection.” In AAAI2010.

Hutter, Frank, Holger H. Hoos, and Kevin Leyton-Brown. "Sequential model-based optimization for general algorithm configuration", LION 2011

Some References

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

ML for boosting Optimization Algorithms

Alejandro Marcos Alvarez, Quentin Louveaux, and Louis Wehenkel. A machine learning-based approximation ofstrong branching.INFORMS Journal on Computing, 2017

A. Sabharwal, H. Samulowitz, H. Reddy, “Guiding combinatorial optimization with UCT”. CPAIOR 2012

Bengio, Yoshua, Andrea Lodi, and Antoine Prouvost. "Machine learning for combinatorial optimization: a methodological tour d’horizon." European Journal of Operational Research (2020).

Predict, the Optimize

Elmachtoub, Adam N., and Paul Grigas. "Smart" predict, then optimize"." arXiv preprint arXiv:1710.08005 (2017).

Donti, Priya, Brandon Amos, and J. Zico Kolter. "Task-based end-to-end model learning in stochastic optimization." Advances in Neural Information Processing Systems. 2017

Bryan Wilder, Bistra Dilkina, Milind Tambe: Melding the Data-Decisions Pipeline: Decision-Focused Learning for Combinatorial Optimization. AAAI 2019: 1658-1665

Aaron Ferber, Bryan Wilder, Bistra Dilkina, Milind Tambe: MIPaaL: Mixed Integer Program as a Layer. AAAI 2020: 1504-1511

Notable Omissis

Black box optimization (connected to “Optimize, then Predict”)

Ky Khac Vu, Claudia D'Ambrosio, Youssef Hamadi, Leo Liberti: Surrogate-based methods for black-box optimization. Int. Trans. Oper. Res. 24(3): 393-424 (2017)

Solving combinatorial optimization models via ML

Bello, Irwan, et al. "Neural combinatorial optimization with reinforcement learning." arXiv preprint arXiv:1611.09940 (2016).

Kool, W., Hoof, H., Welling, M.: Attention solves your tsp, approximately. Statis- tics 1050, 22 (2018)

Elias B. Khalil, Hanjun Dai, Yuyu Zhang, Bistra Dilkina, Le Song: Learning Combinatorial Optimization Algorithms over Graphs. NIPS 2017: 6348-6358

Andrea Galassi, Michele Lombardi, Paola Mello, Michela Milano: Model Agnostic Solution of CSPs via Deep Learning: A Preliminary Study. CPAIOR 2018: 254-262

Andrea Galassi, Michele Lombardi, Paola Mello, Michela Milano: Model Agnostic Solution of CSPs via Deep Learning: A Preliminary Study. CPAIOR 2018: 254-262

Xu, H., Koenig, S., Kumar, T.S.: Towards effective deep learning for constraint satisfaction problems. In: Proc. of CPAIOR. pp. 588–597. Springer (2018)

Notable Omissis

ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA

Constraints in ML models

Michelangelo Diligenti, Marco Gori, Claudio Saccà: Semantic-based regularization for learning and inference. Artif. Intell, 2017

Artur S. d'Avila Garcez, Marco Gori, Luís C. Lamb, Luciano Serafini, Michael Spranger, Son N. Tran: Neural-symbolic Computing: An Effective Methodology for Principled Integration of Machine Learning and Reasoning. FLAP, 2019

Manhaeve, Robin, et al. "Deepproblog: Neural probabilistic logic programming." Advances in Neural Information Processing Systems. 2018.

Ferdinando Fioretto, Pascal Van Hentenryck, Terrence W. K. Mak, Cuong Tran, Federico Baldo, Michele Lombardi: Lagrangian Duality for Constrained Deep Learning. ECML/PKDD (5) 2020: 118-135

Fabrizio Detassis, Michele Lombardi, Michela Milano: Teaching the Old Dog New Tricks: Supervised Learning with Constraints. CoRR abs/2002.10766 (2020)

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Thanks!

Questions?