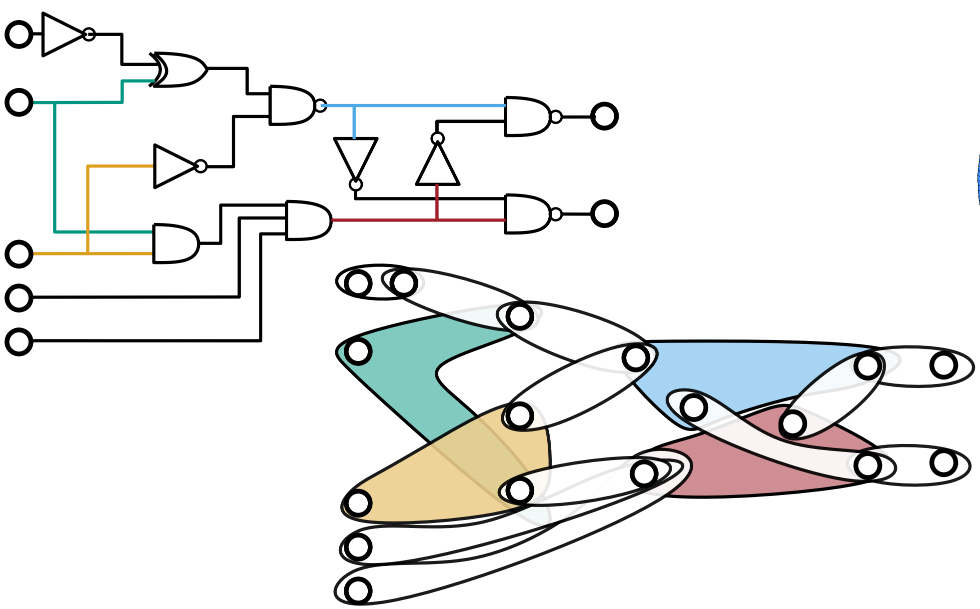
Karlsruhe Institute of Technology

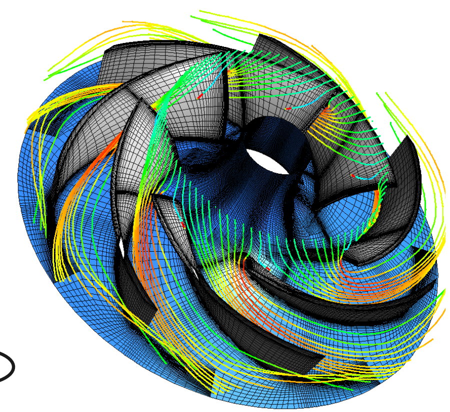
Recent Advances in (Hyper-)Graph Partitioning

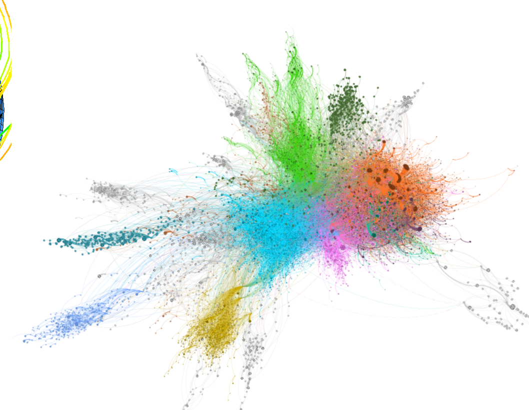
Annual SPP Meeting · September 17, 2019

Peter Sanders and Tobias Heuer

INSTITUTE OF THEORETICAL INFORMATICS · ALGORITHMICS GROUP







News on (Hyper)graph Partitioning

Karlsruhe Institute of Technology

- **Scalable Edge Partitioning**
  ALENEX'19, S. Schlag, C. Schulz, D. Seemaier, D. Strash
- **Faster Support Vector Machines**
  ALENEX'19, S. Schlag, M. Schmitt, C. Schulz
- **Data Distribution for Phylogenetic Inference with Site Repeats via Judicious Hypergraph Partitioning**
  IPDPSW'19, I. Baar, L. Hübner, P. Oettig, A. Zapletal, S. Schlag, A. Stamatakis, B. Morel
- **Network Flow-Based Refinement for Multilevel Hypergraph Partitioning**
  JEA'19, T. Heuer, P. Sanders, S. Schlag
- **Evaluation of a Flow-Based Hypergraph Bipartitioning Algorithm**
  ESA'19, L. Gottesbüren, M. Hamann, D. Wagner
- **WeGotYouCovered**
  PACE Challenge'19, Vertex Cover Track D. Hespe, S. Lamm, C. Schulz, D. Strash

News on (Hyper)graph Partitioning

Karlsruhe Institute of Technology

- **Scalable Edge Partitioning**
  ALENEX'19, S. Schlag, C. Schulz, D. Seemaier, D. Strash
- **Faster Support Vector Machines**
  ALENEX'19, S. Schlag, M. Schmitt, C. Schulz
- **Data Distribution for Phylogenetic Inference with Site Repeats via Judicious Hypergraph Partitioning**
  IPDPSW'19, I. Baar, L. Hübner, P. Oettig, A. Zapletal, S. Schlag, A. Stamatakis, B. Morel
- **Network Flow-Based Refinement for Multilevel Hypergraph Partitioning**
  JEA'19, T. Heuer, P. Sanders, S. Schlag
- **Evaluation of a Flow-Based Hypergraph Bipartitioning Algorithm**
  ESA'19, L. Gottesbüren, M. Hamann, D. Wagner
- **WeGotYouCovered**
  PACE Challenge'19, Vertex Cover Track D. Hespe, S. Lamm, C. Schulz, D. Strash

Scalable Edge Partitioning

S. Schlag, C. Schulz, D. Seemaier, D. Strash

The Edge Partitioning Problem

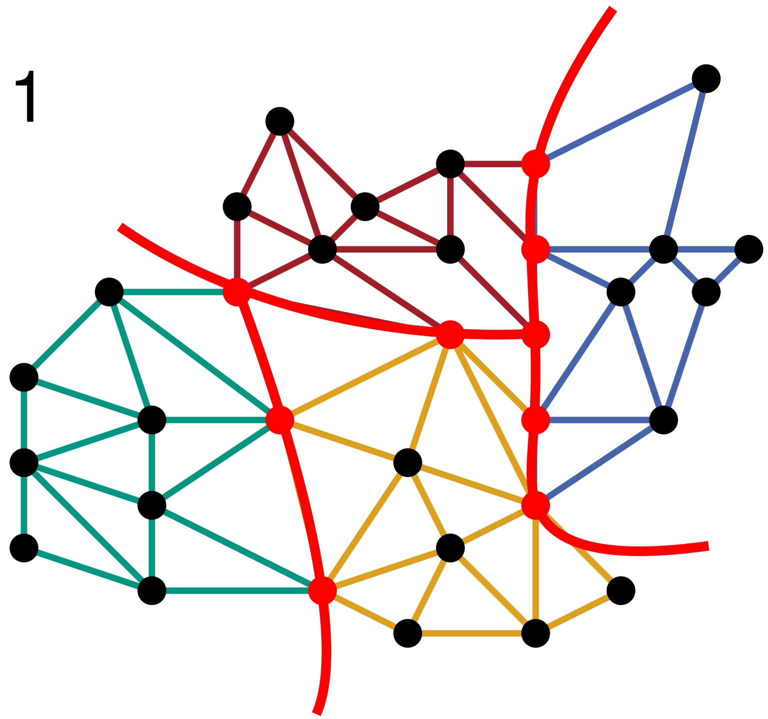
Partition edge set of graph $G = (V, E, c, \omega)$ into **k** disjoint blocks $\Pi = \{E_1, \ldots, E_k\}$ such that

- Blocks E_i are roughly equal-sized:

$$\omega(E_i) \leq (1 + \varepsilon) \left\lceil \frac{\omega(E)}{k} \right\rceil$$

- minimize vertex cut:

$$\sum_{v \in V} |\mathrm{I}(v)| - 1$$



Institute of Theoretical Informatics
Algorithmics Group

The Edge Partitioning Problem

Partition edge set of graph $G = (V, E, c, \omega)$ into **k** disjoint blocks $\Pi = \{E_1, \ldots, E_k\}$ such that

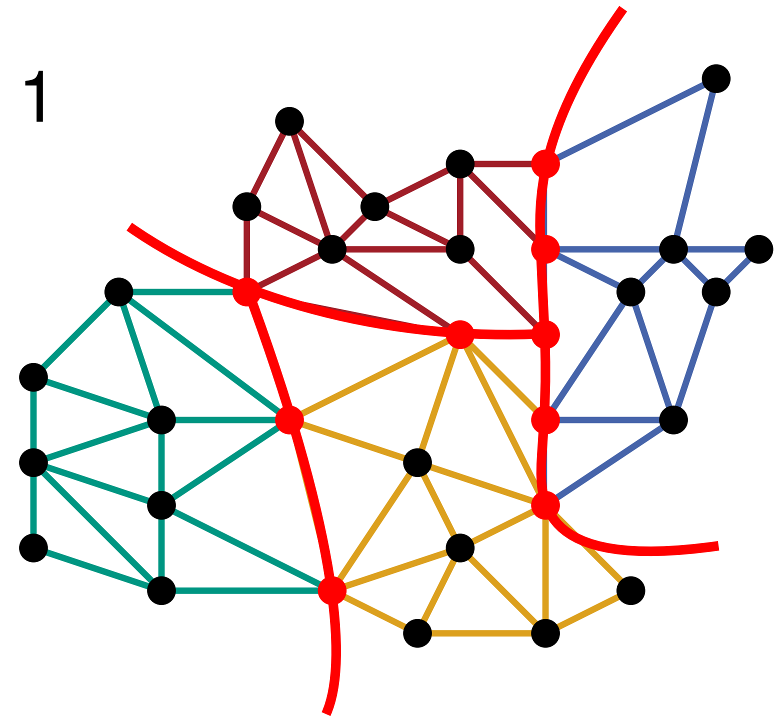
- Blocks E_i are roughly equal-sized:

$$\omega(E_i) \leq (1 + \varepsilon) \left\lceil \frac{\omega(E)}{k} \right\rceil$$

- minimize vertex cut:

$$\sum_{v \in V} |I(v)| - 1$$

 # blocks with edges incident to v



The Edge Partitioning Problem

Partition edge set of graph $G = (V, E, c, \omega)$ into **k** disjoint blocks $\Pi = \{E_1, \ldots, E_k\}$ such that

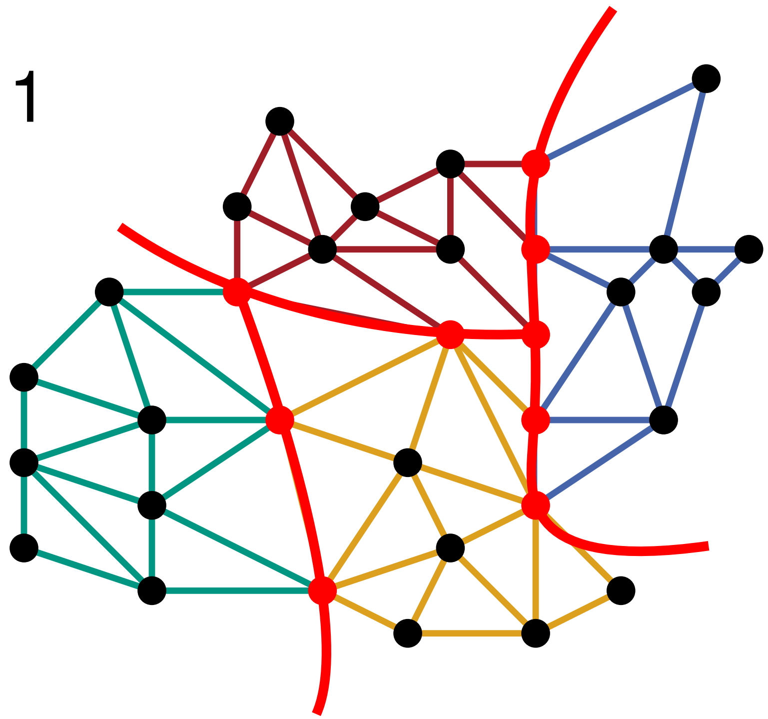
- Blocks E_i are roughly equal-sized:

$$\omega(E_i) \leq (1 + \varepsilon) \left\lceil \frac{\omega(E)}{k} \right\rceil$$

- minimize vertex cut:

$$\sum_{v \in V} |I(v)| - 1$$

 # blocks with edges incident to v



Motivation [Gonzalez et al.'12]:

- edge-centric distributed computations
- combat shortcomings of TLAV approaches
- duplicate node-centric computations

Institute of Theoretical Informatics
Algorithmics Group

The Edge Partitioning Problem

Karlsruhe Institute of Technology

Partition edge set of graph $G = (V, E, c, \omega)$ into **k** disjoint blocks $\Pi = \{E_1, \ldots, E_k\}$ such that

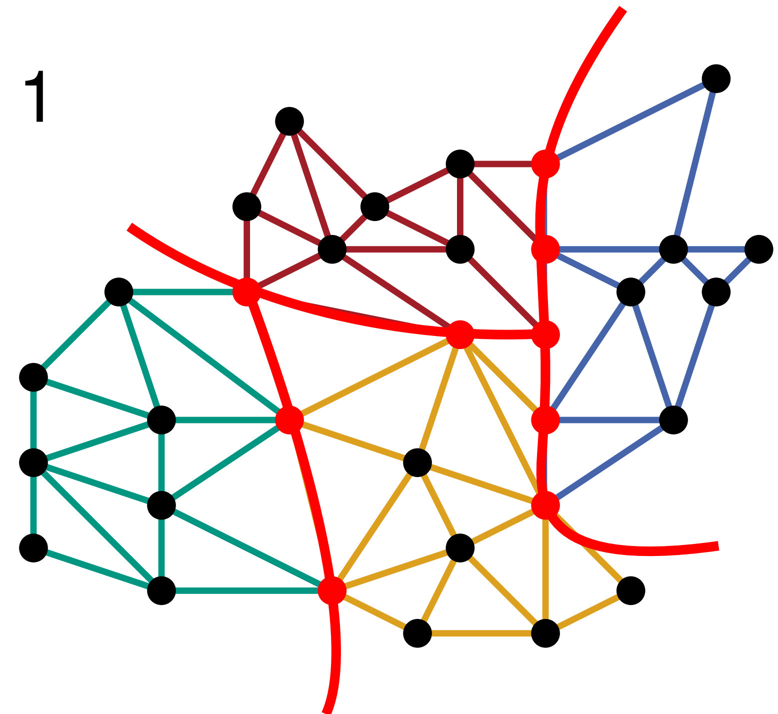
- Blocks E_i are roughly equal-sized:

$$\omega(E_i) \leq (1 + \varepsilon) \left\lceil \frac{\omega(E)}{k} \right\rceil$$

- minimize vertex cut:

$$\sum_{v \in V} |I(v)| - 1$$

 # blocks with edges incident to v



Motivation [Gonzalez et al.'12]:

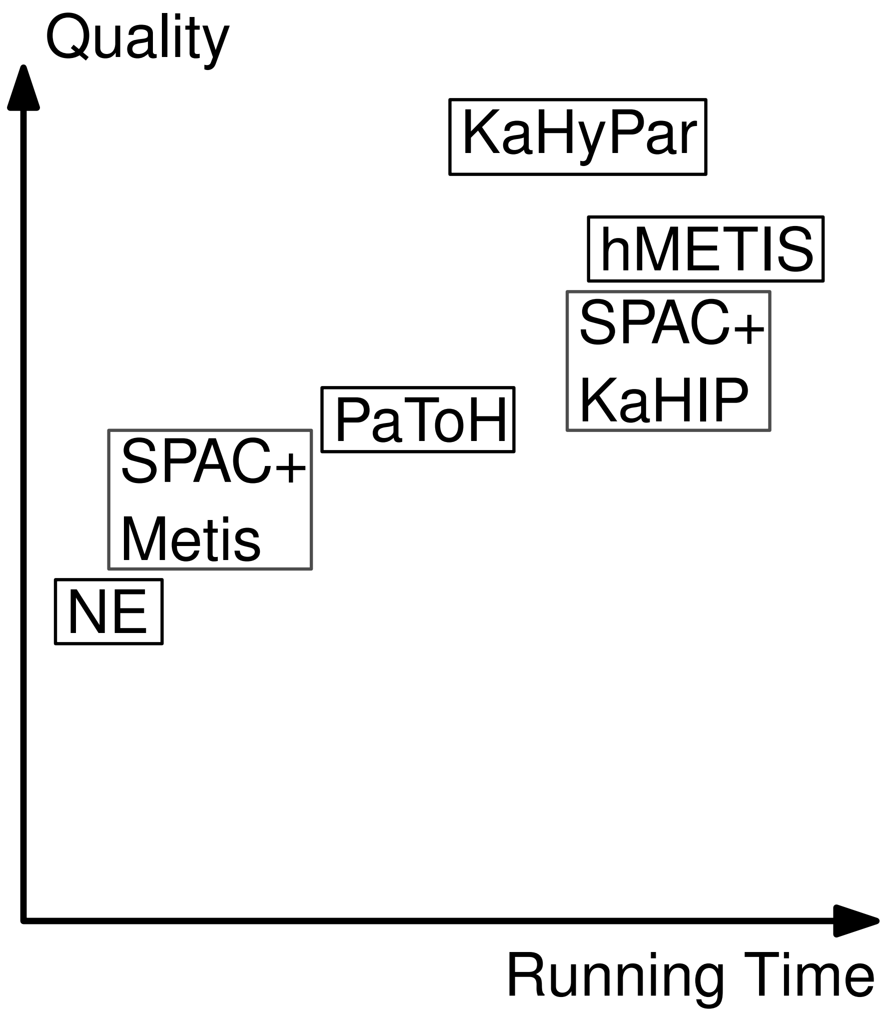
- edge-centric distributed computations
- combat shortcomings of TLAV approaches
- duplicate node-centric computations

Think-Like-A-Vertex

Institute of Theoretical Informatics
Algorithmics Group

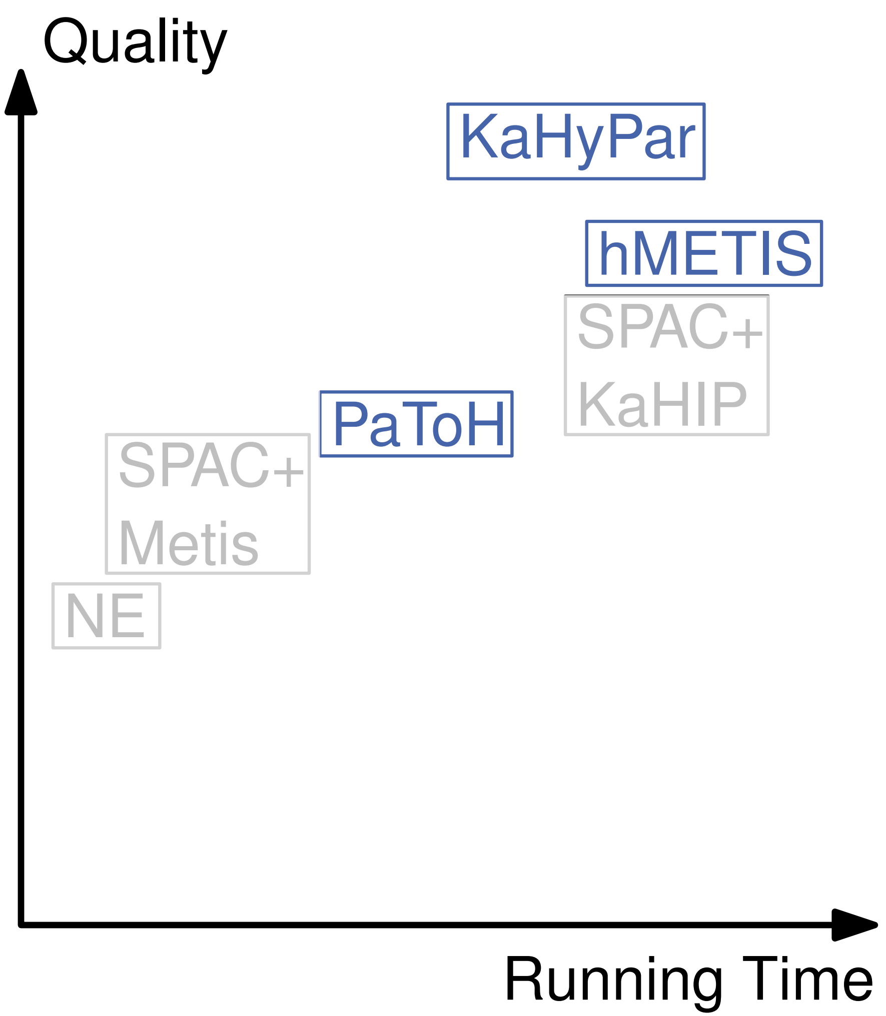
Edge Partitioning Algorithms

Sequential



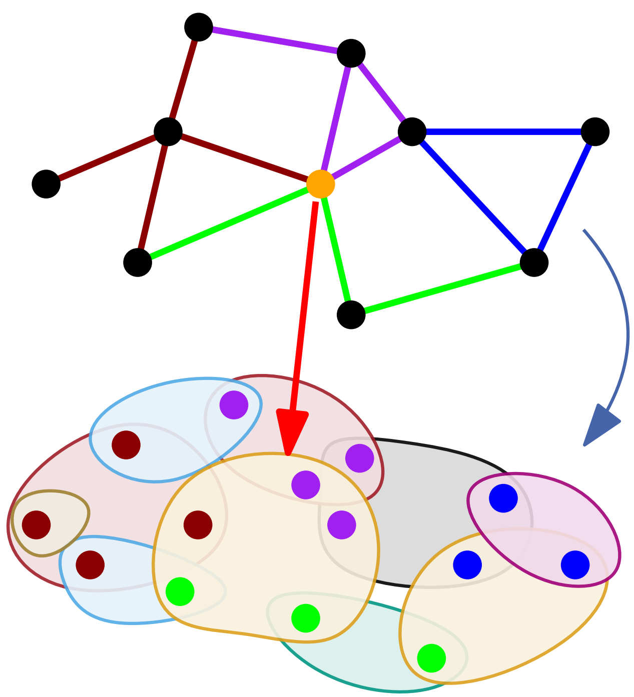
Edge Partitioning Algorithms

Sequential



Hypergraph Model:

- Graph edge ⇝ vertex
- Graph node ⇝ hyperedge
- Optimize connectivity

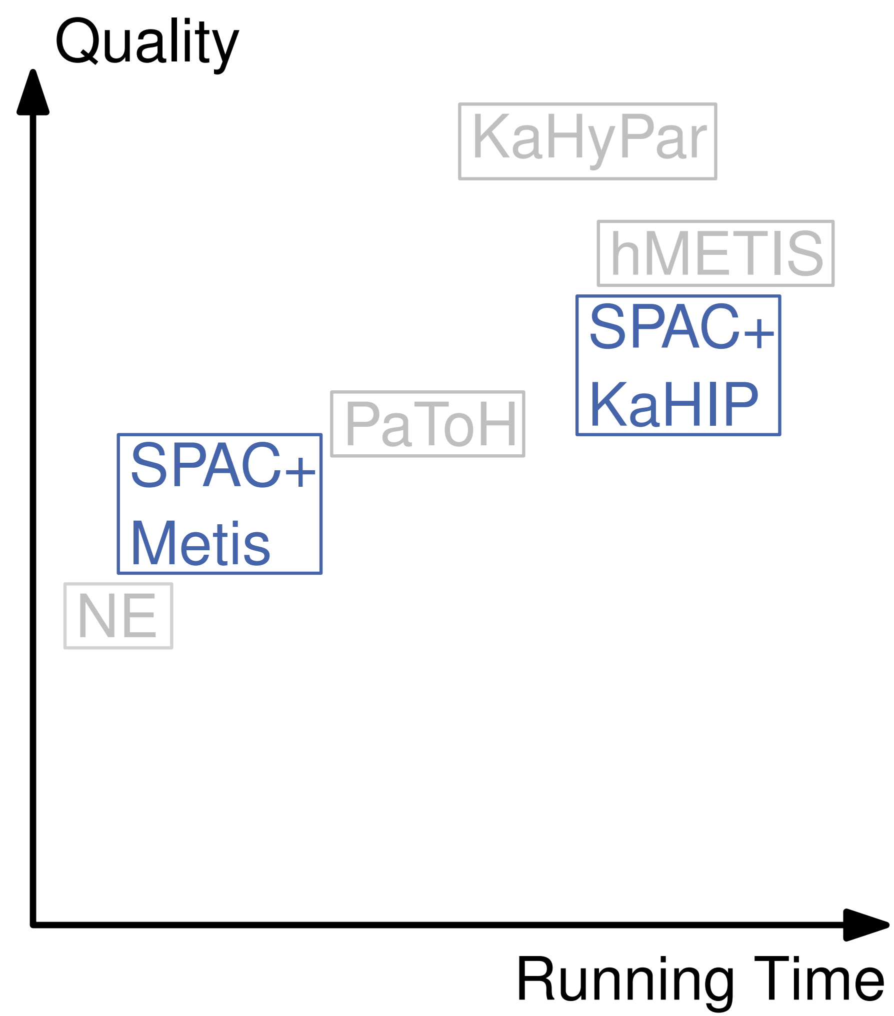


Institute of Theoretical Informatics
Algorithmics Group

Edge Partitioning Algorithms

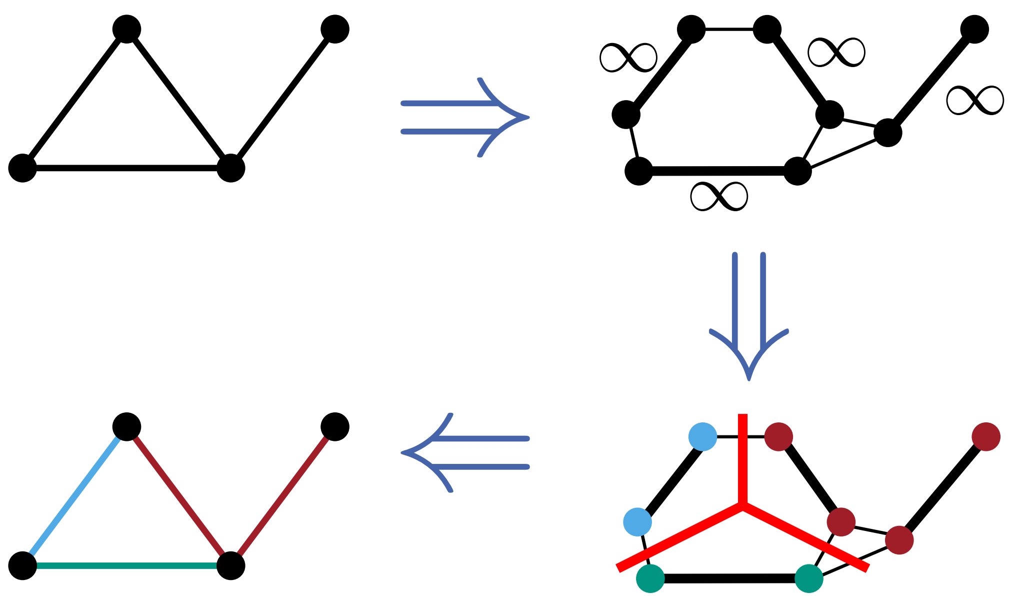
Karlsruhe Institute of Technology

Sequential



Split-And-Connect (SPAC) [Li et al.'17]:

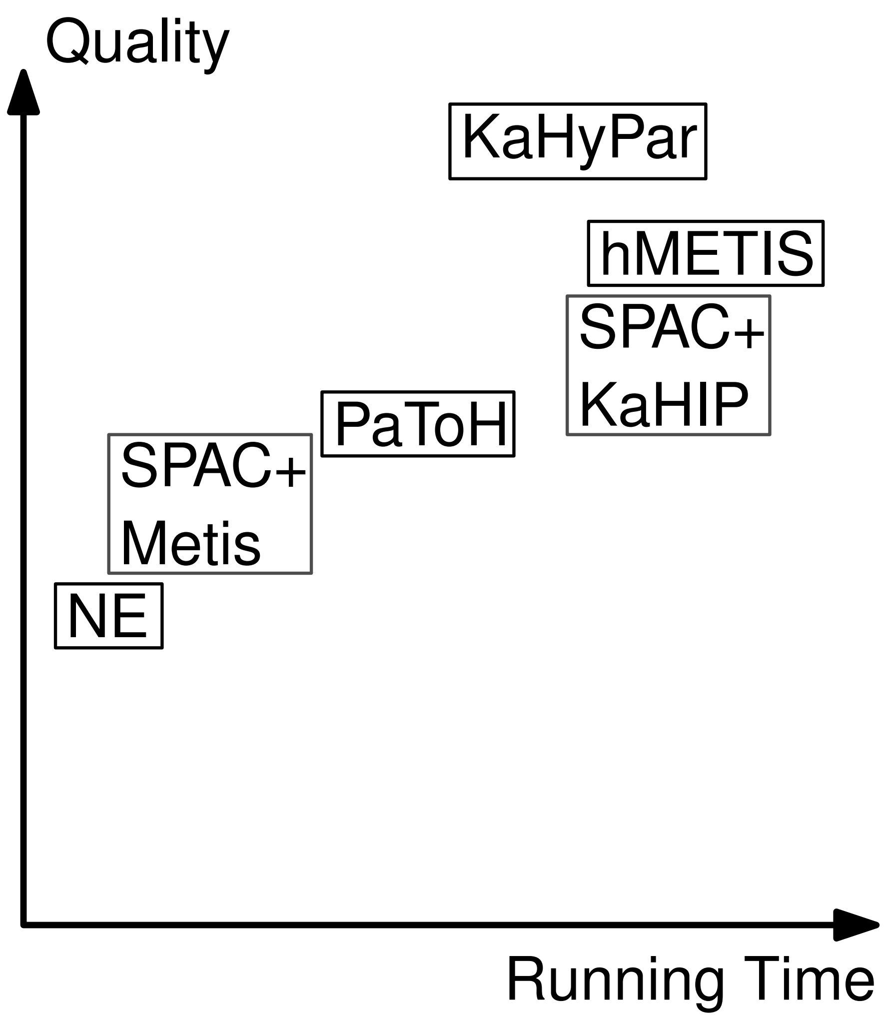
- Build auxilary graph
- Use vertex partitioning algorithm



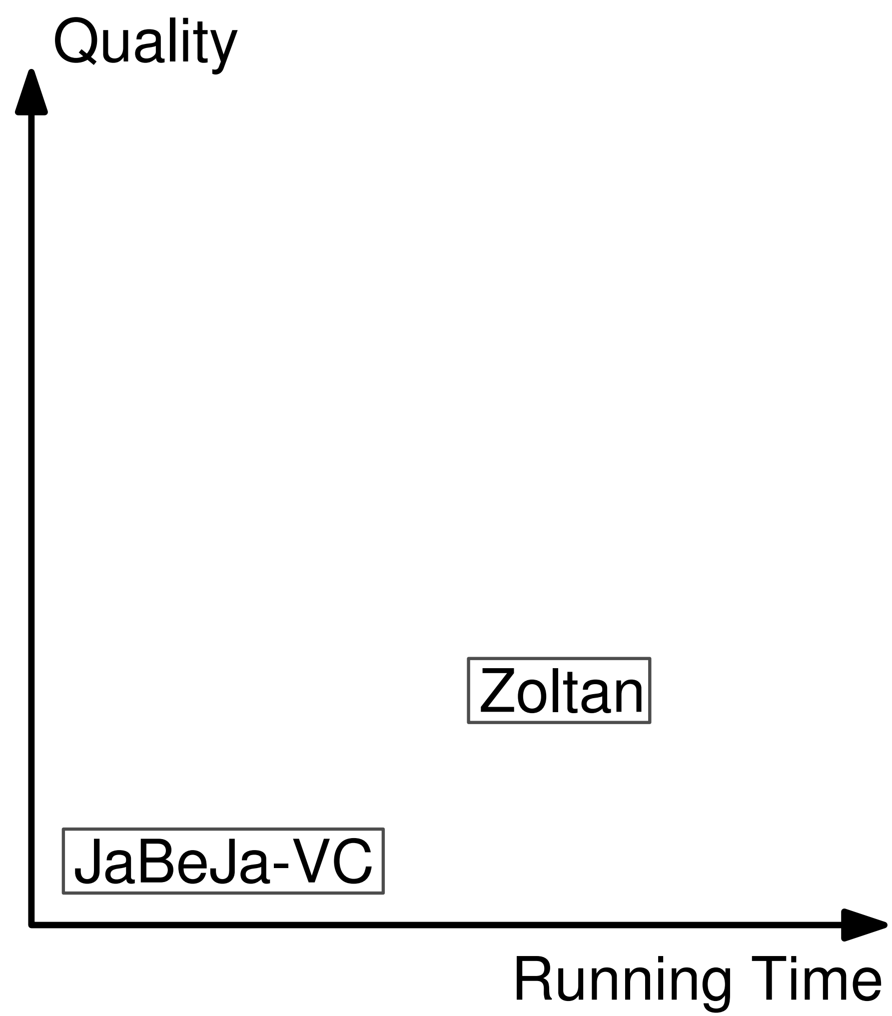
Institute of Theoretical Informatics
Algorithmics Group

Edge Partitioning Algorithms

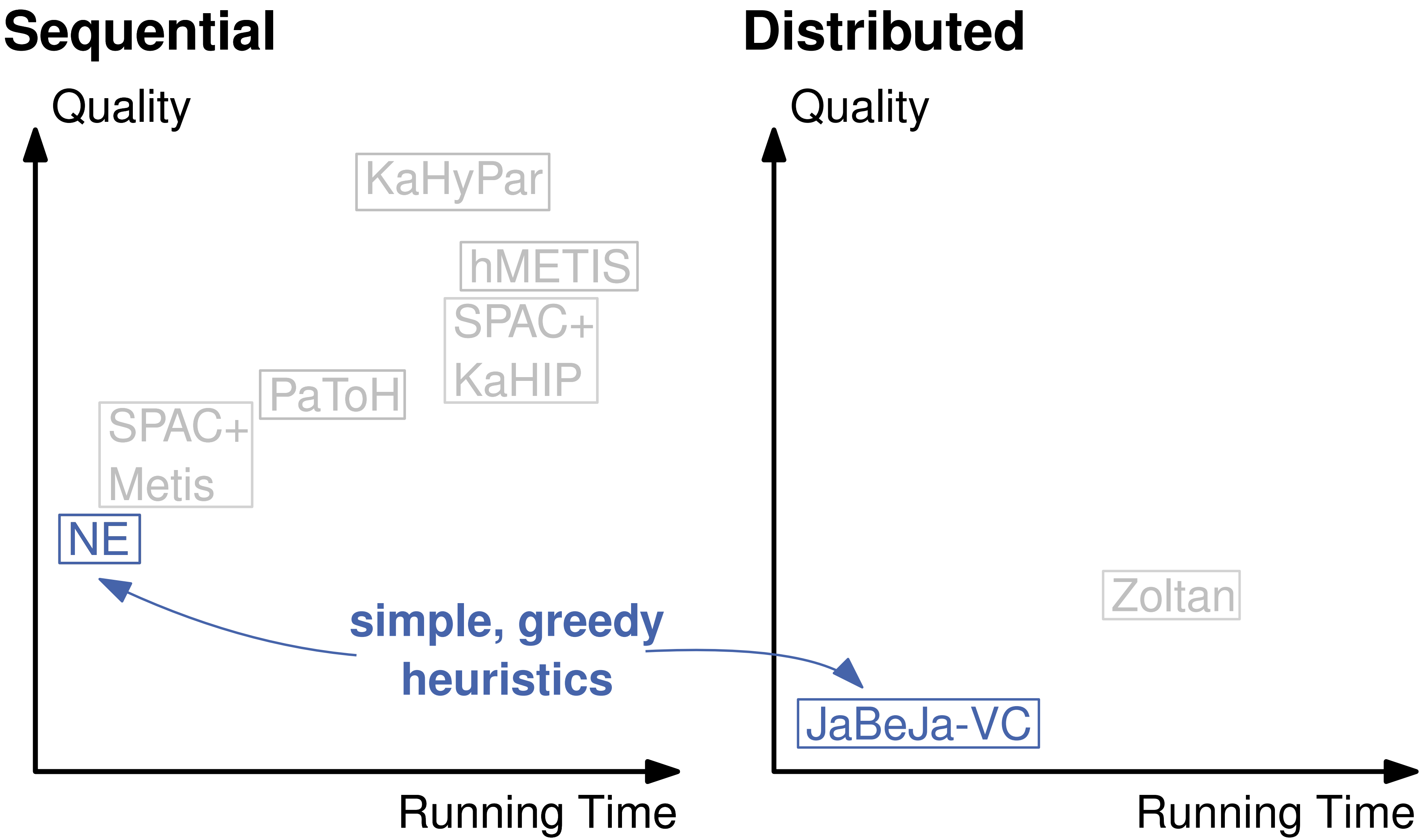
Sequential



Distributed

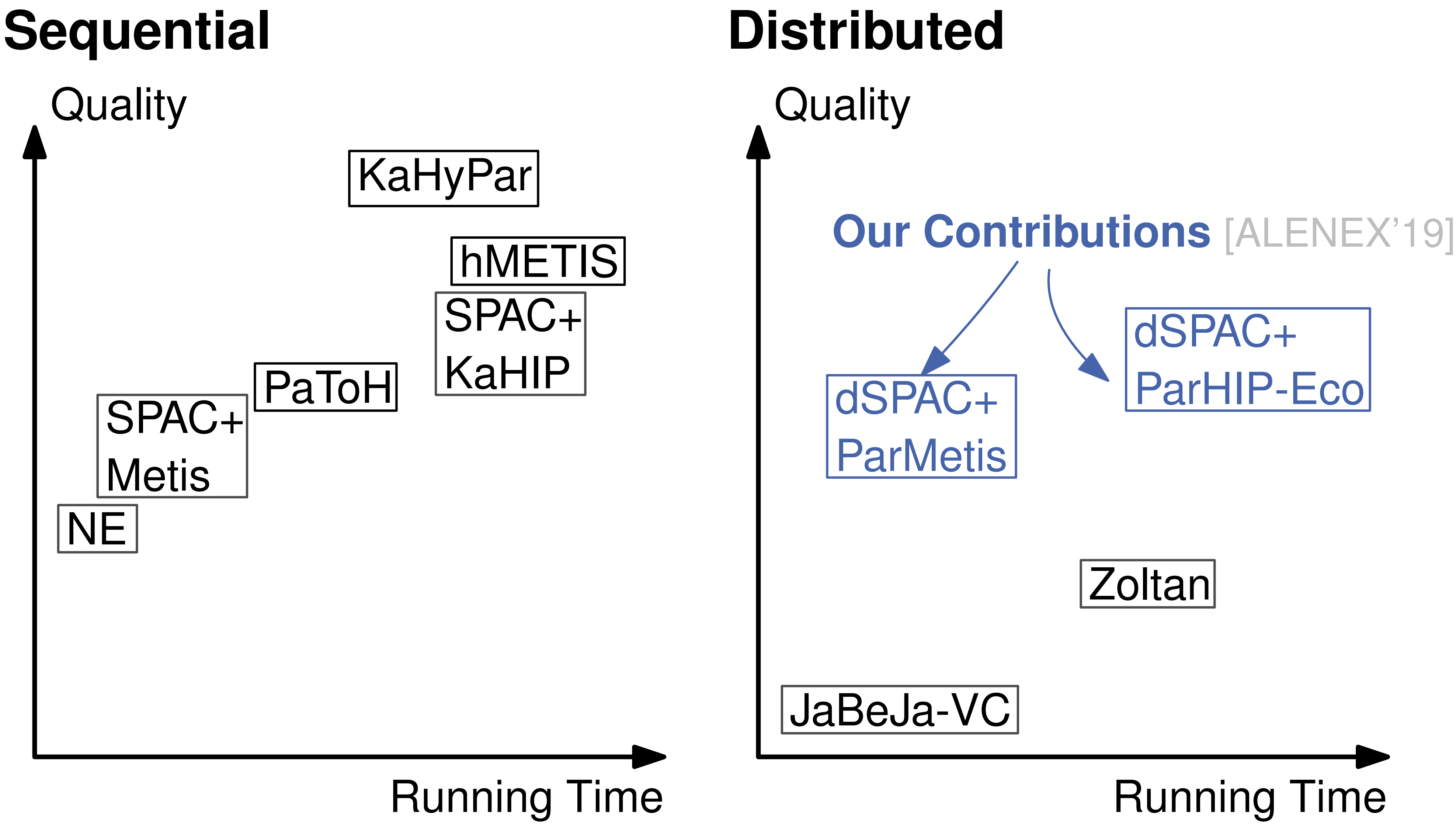


Edge Partitioning Algorithms



Edge Partitioning Algorithms

Karlsruhe Institute of Technology



Institute of Theoretical Informatics
Algorithmics Group

Experiments: Benchmark Setup

- Test suite: 70 graphs
  - Walshaw Graph Archive
  - Sparse Matrix-Vector Multiplication
  - Web & Social Graphs
  - Random Geometric Graphs
- $k \in \{2, 4, 8, 16, 32, 64, 128\}$
- Imbalance: $\epsilon = 3\%$
- Averages of 5 repetitions
- Sequential: 1 core
- Distributed: 32 * 20 cores

Competitors:

HGP's
- KaHyPar-MF
- PaToH
- Zoltan
- Zoltan-AlgD
- hMetis-{R, K}

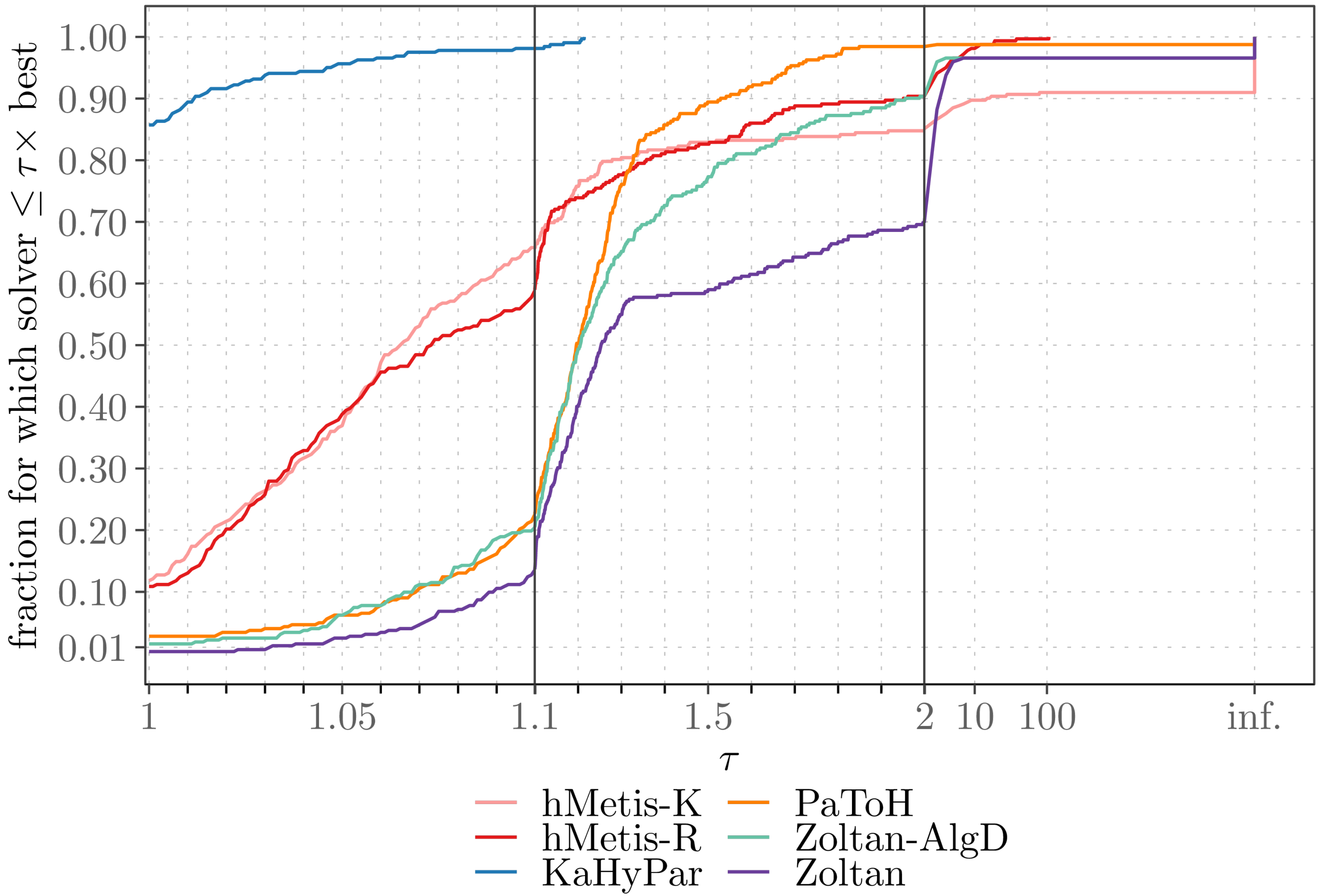
- JaBeJa-VC
- NE

- SPAC + KaHIP
- SPAC + Metis

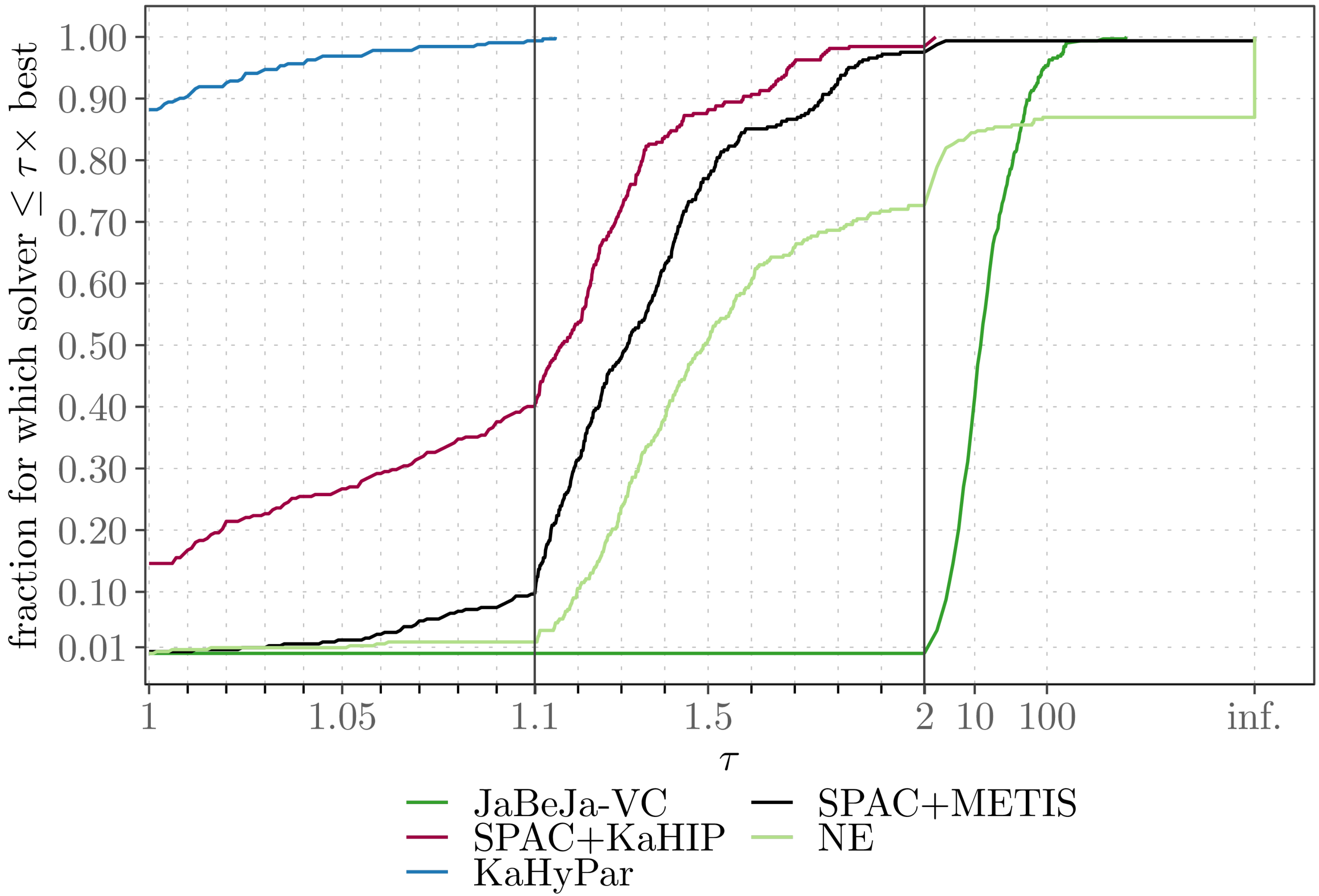
- dSPAC + ParHIP
- dSPAC + ParMetis

Institute of Theoretical Informatics
Algorithmics Group

Experiments: Sequential HGP

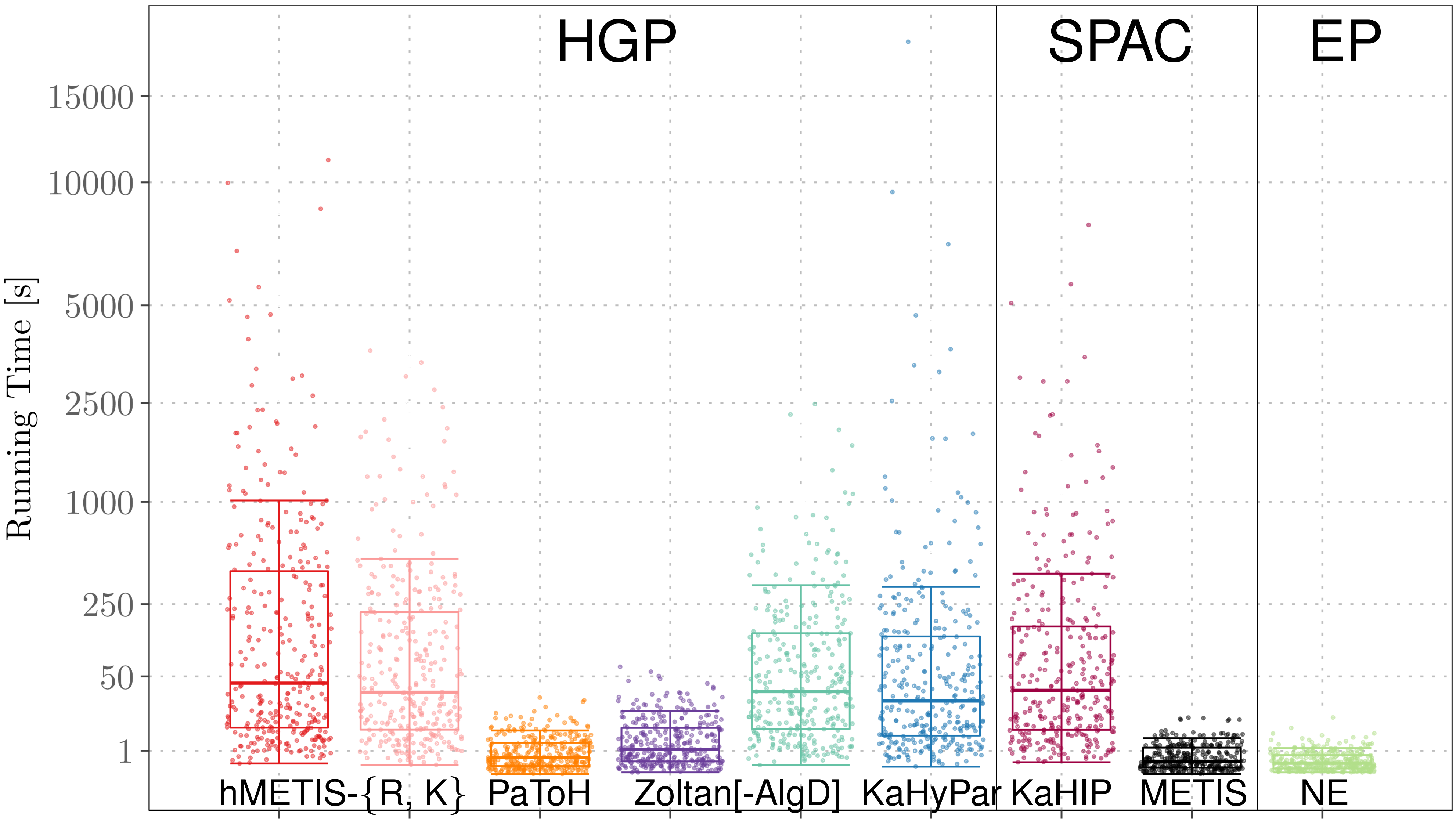


Experiments: Sequential HGP & SPAC+X

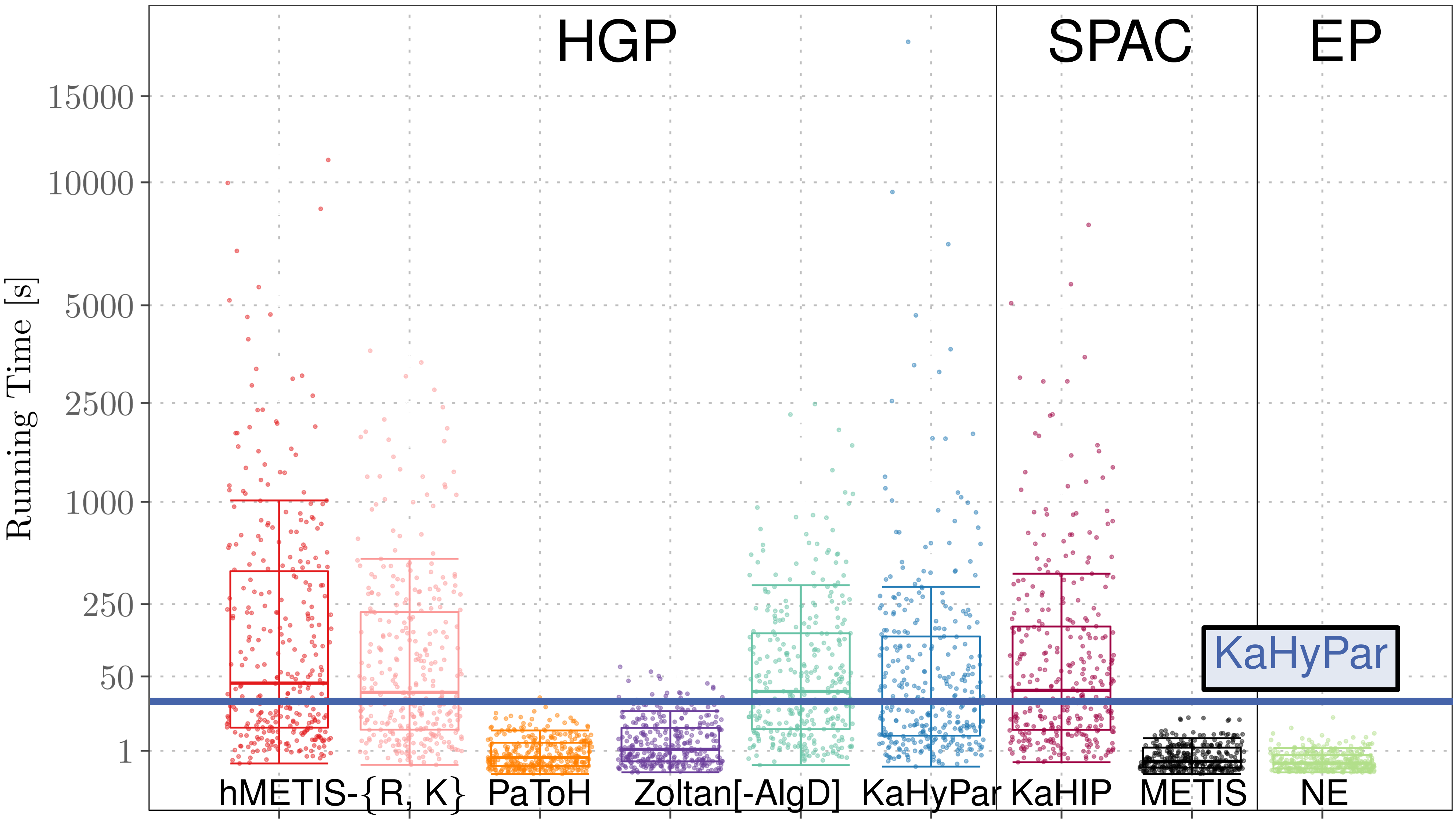


Institute of Theoretical Informatics
Algorithmics Group

Experiments: Sequential Running Time

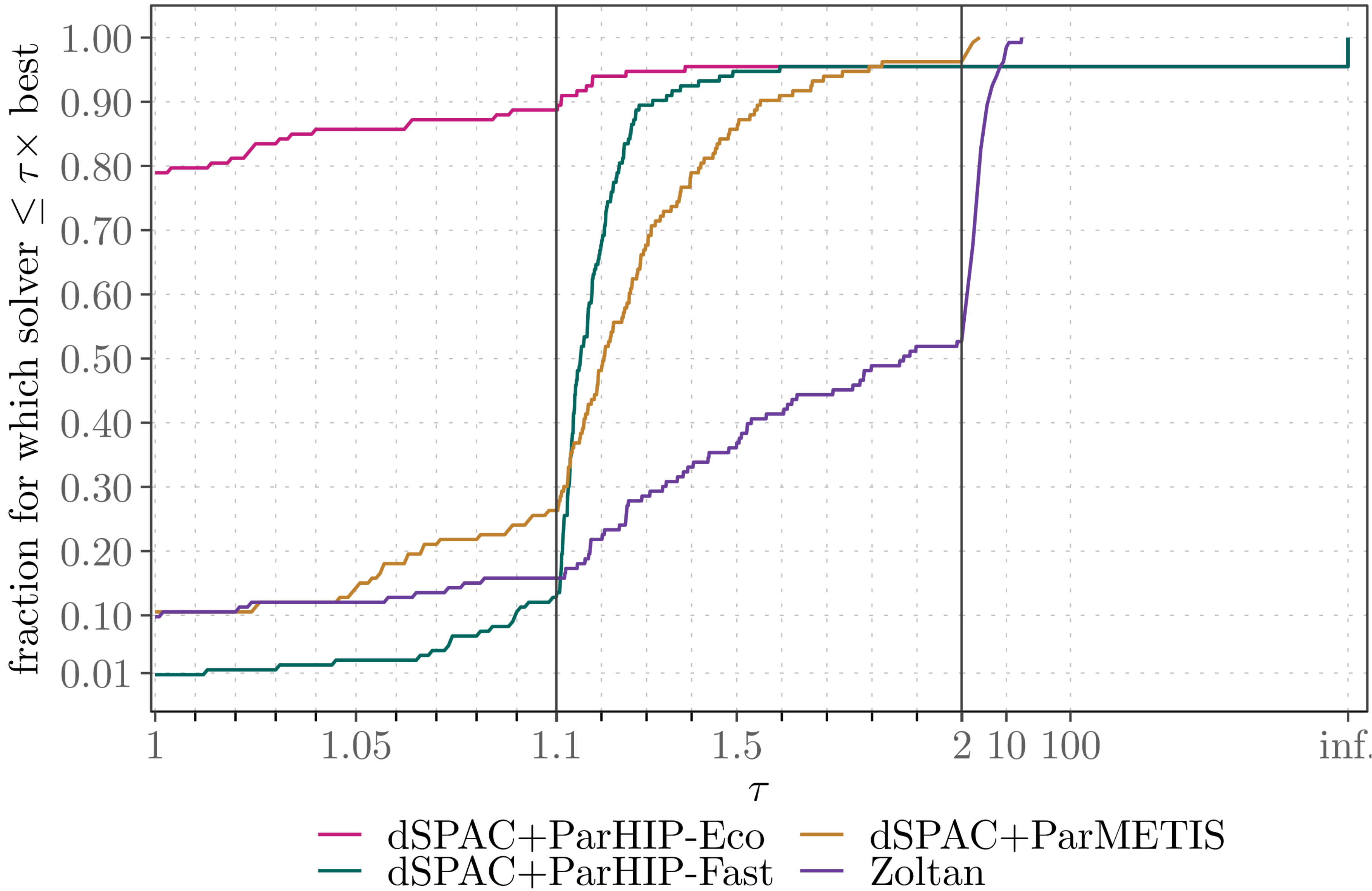


Experiments: Sequential Running Time

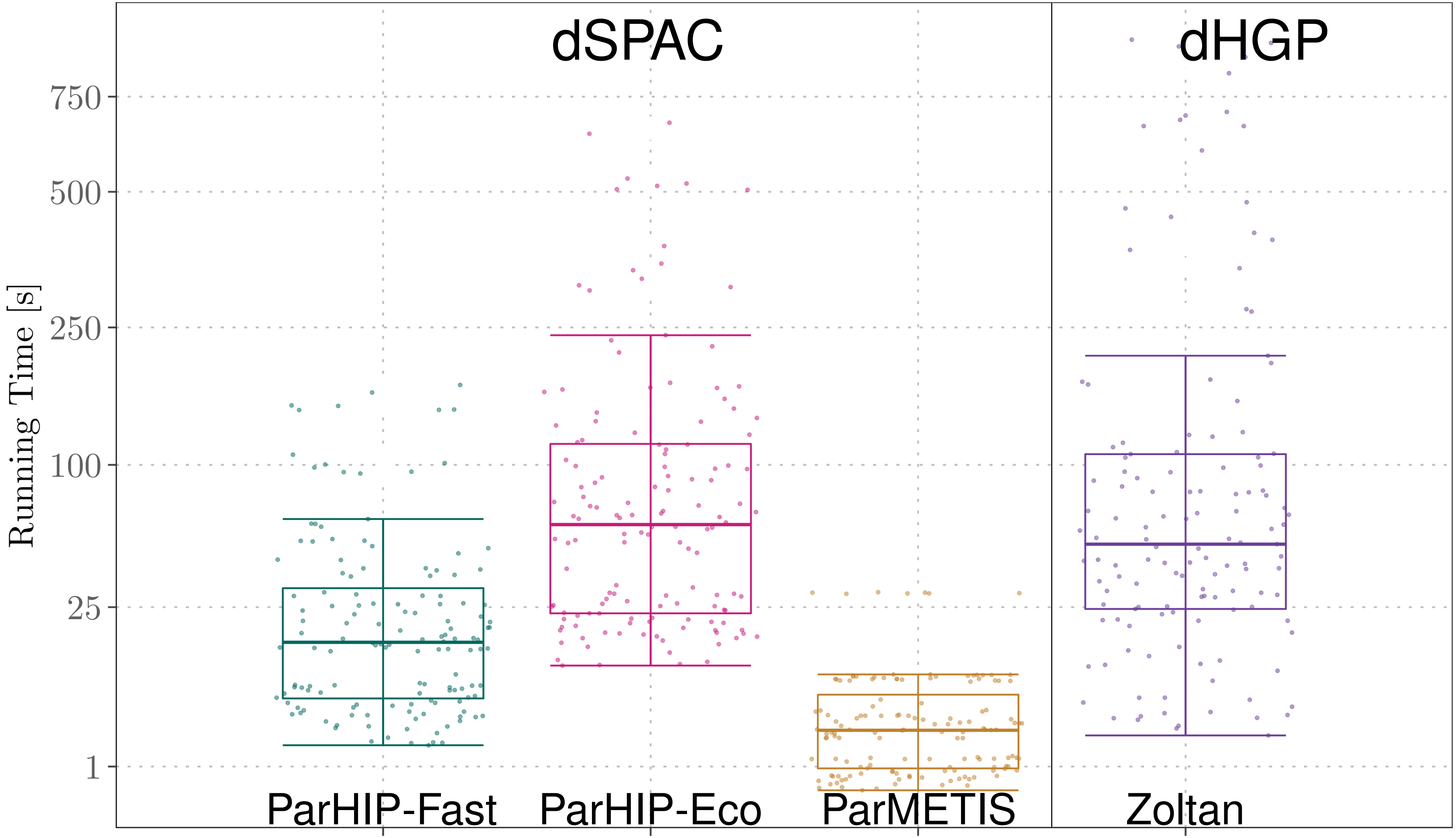


Institute of Theoretical Informatics
Algorithmics Group

Experiments: Distributed HGP & dSPAC+X

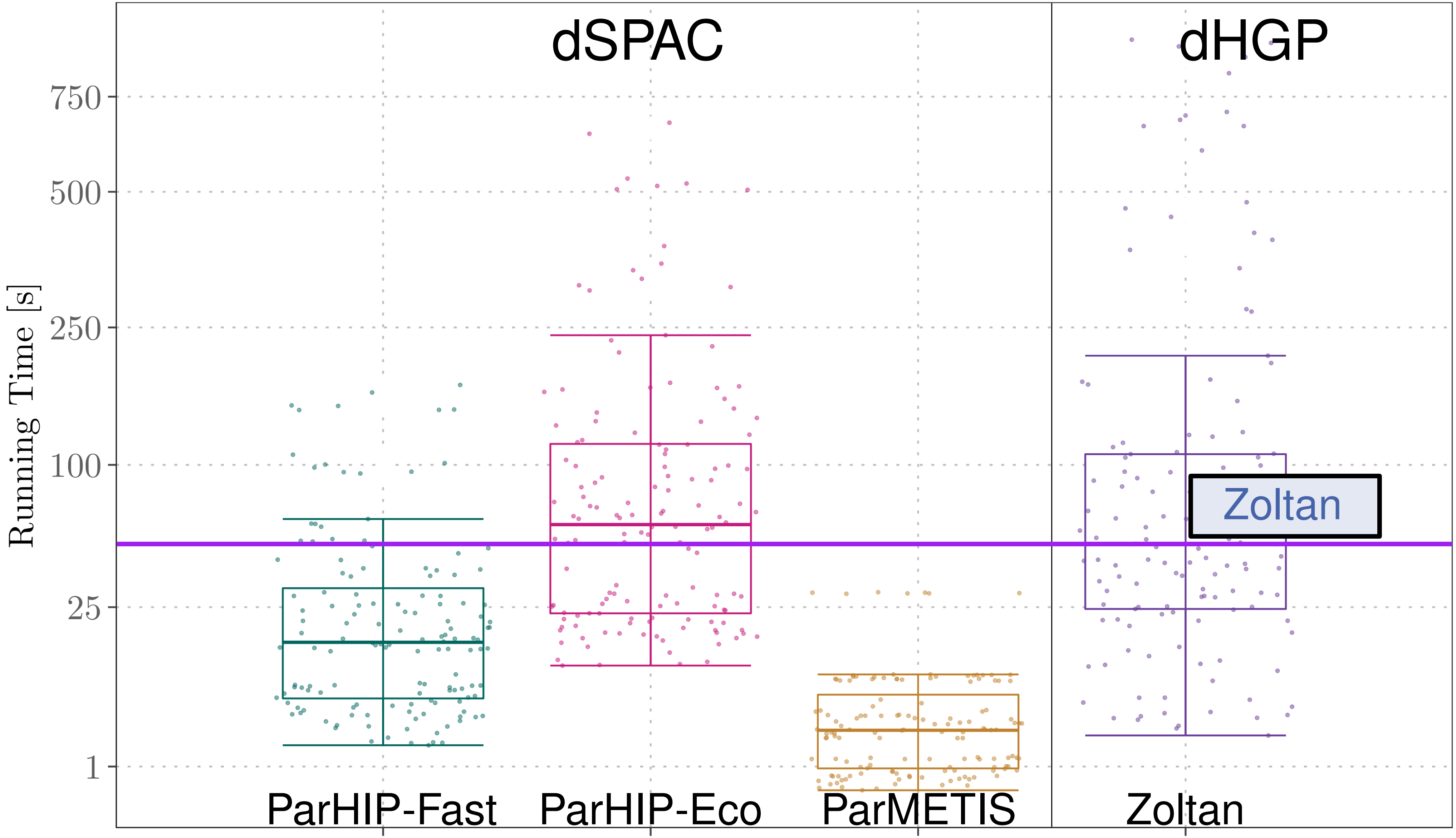


Experiments: Distributed Running Time



Experiments: Distributed Running Time

Karlsruhe Institute of Technology



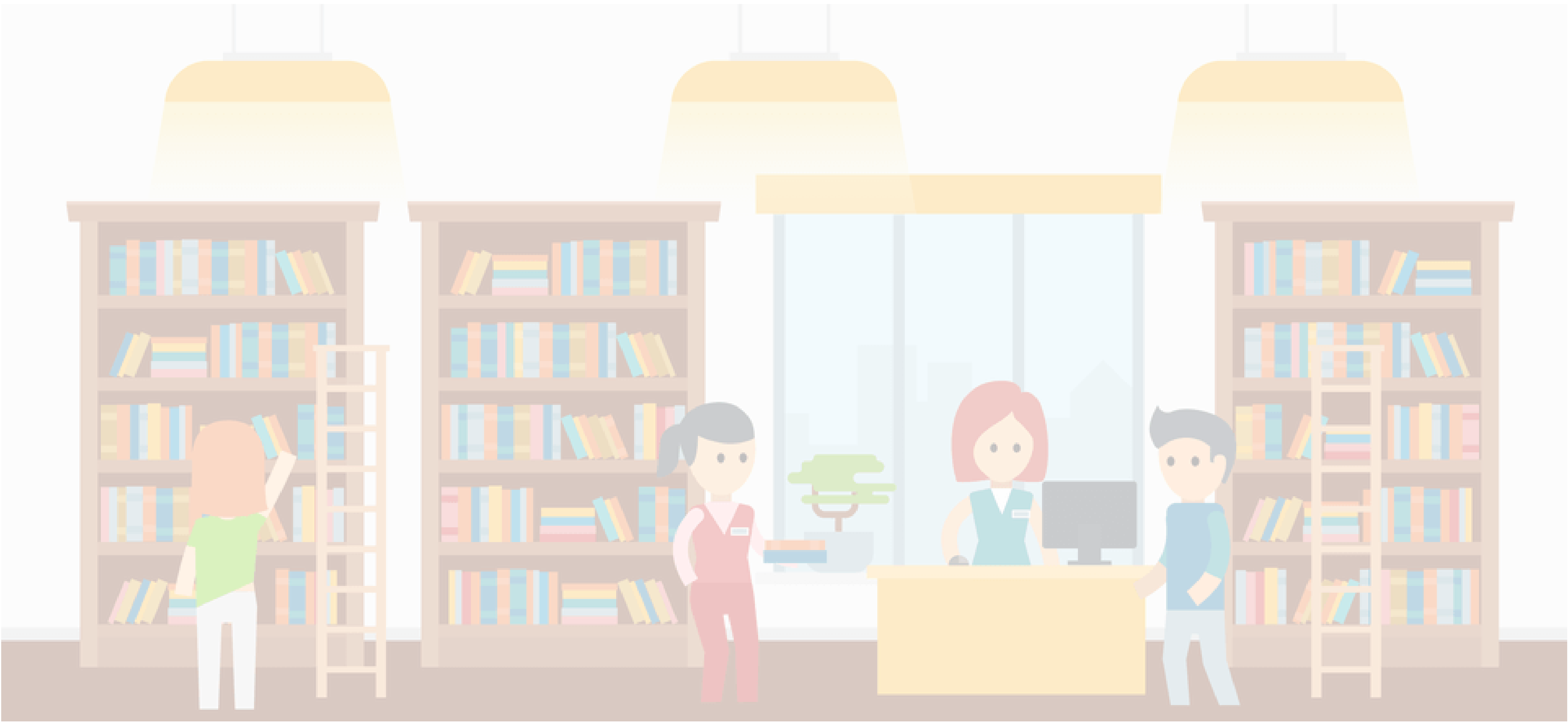
Institute of Theoretical Informatics
Algorithmics Group

Faster Support Vector Machines

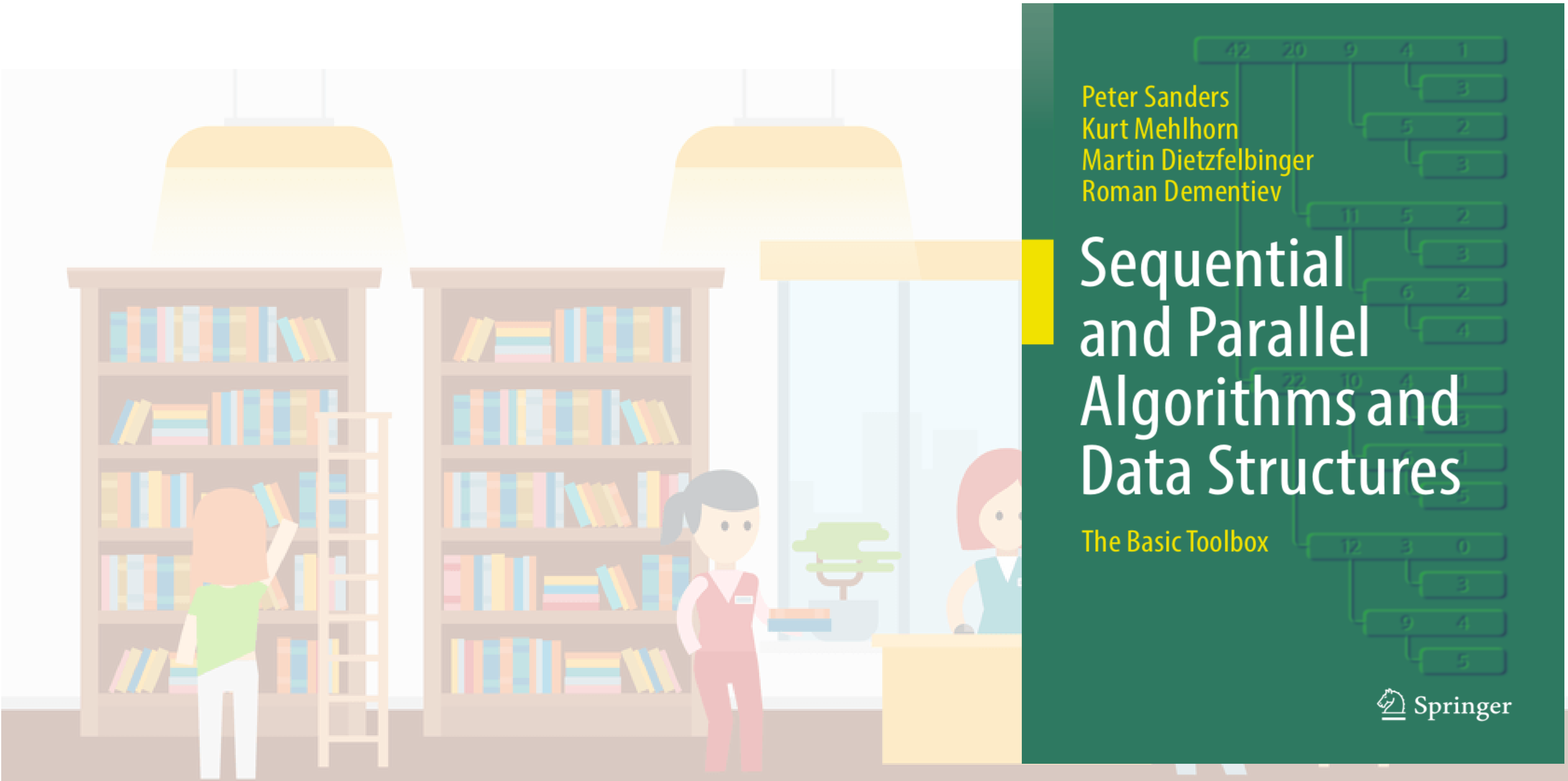
S. Schlag, M. Schmitt, C. Schulz

Binary Classification Problem

Binary Classification Problem



Binary Classification Problem



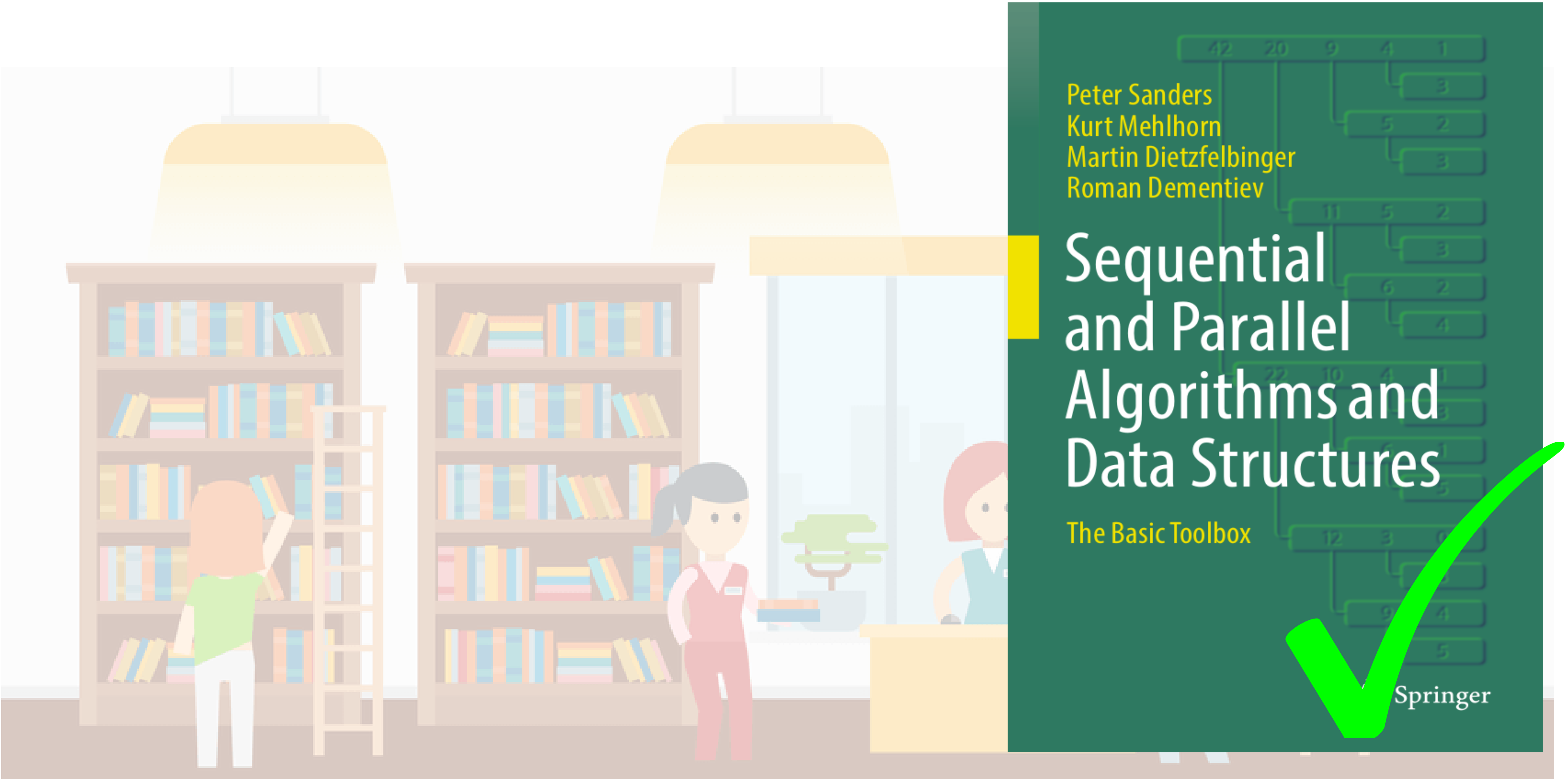
Binary Classification Problem

Institute of Theoretical Informatics
Algorithmics Group

Binary Classification Problem

Institute of Theoretical Informatics
Algorithmics Group

Binary Classification Problem



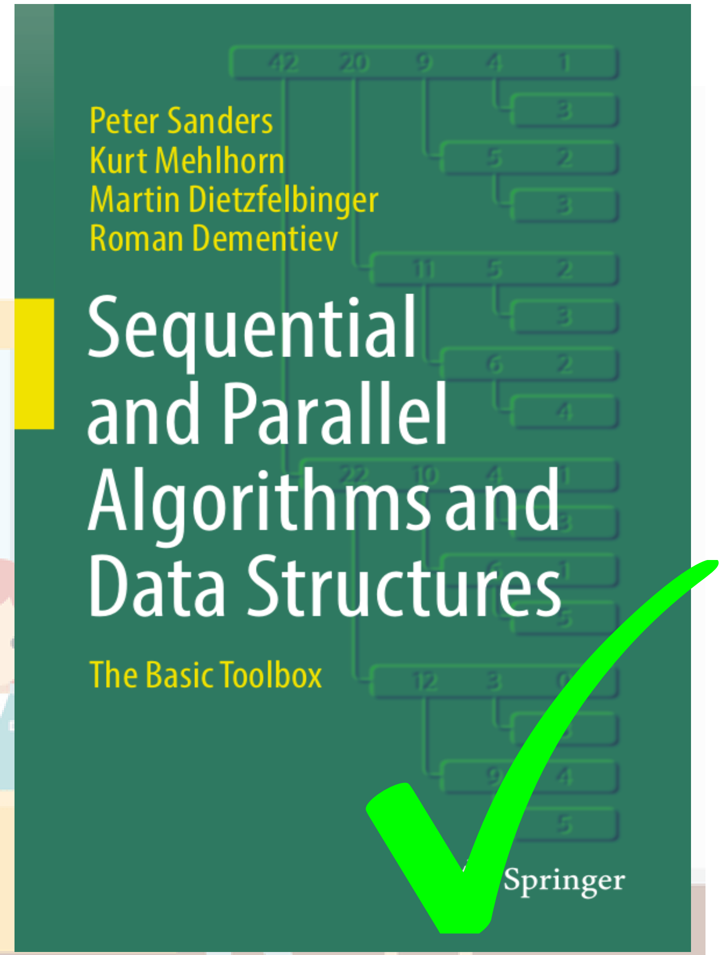
Binary Classification Problem

Very well written

Corollary 12.6. *The approximation ratio of the shortest-queue algorithm is* $2 - 1/m$.

Proof. Let $L_1 = \frac{1}{m}\sum_i t_i$ and $L_2 = \max_i t_i$. The makespan L^* of the optimal solution is at least $\max(L_1, L_2)$. The makespan of the shortest-queue solution is bounded by

$$L_1 + \frac{m-1}{m} L_2 \leq L^* + \frac{m-1}{m} L^* = \left(2 - \frac{1}{m}\right) L^*. \qquad \square$$



Binary Classification Problem

Very well written

Corollary 12.6. *The approximation ratio of the shortest-queue algorithm is $2 - 1/m$.*

Proof. Let $L_1 = \frac{1}{m}\sum_i t_i$ and $L_2 = \max_i t_i$. The makespan L^* of the optimal solution is at least $\max(L_1, L_2)$. The makespan of the shortest-queue solution is bounded by

$$L_1 + \frac{m-1}{m} L_2 \leq L^* + \frac{m-1}{m} L^* = \left(2 - \frac{1}{m}\right) L^*. \qquad \square$$

Very informative illustrations

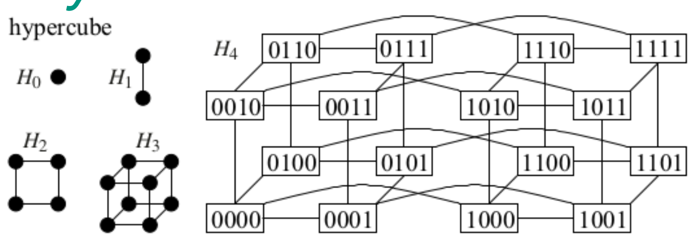
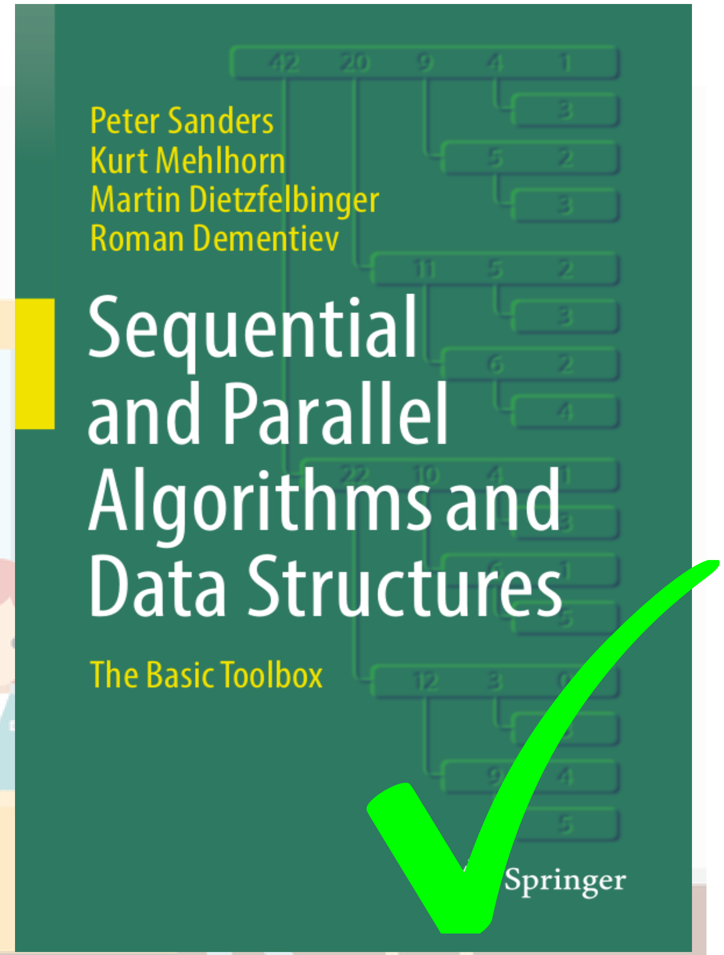


Fig. 13.2. Basic communication topologies for collective communications



Institute of Theoretical Informatics
Algorithmics Group

Binary Classification Problem

Very well written

Corollary 12.6. *The approximation ratio of the shortest-queue algorithm is* $2-1/m$.

Proof. Let $L_1 = \frac{1}{m}\sum_i t_i$ and $L_2 = \max_i t_i$. The makespan L^* of the optimal solution is at least $\max(L_1, L_2)$. The makespan of the shortest-queue solution is bounded by

$$L_1 + \frac{m-1}{m} L_2 \le L^* + \frac{m-1}{m} L^* = \left(2 - \frac{1}{m}\right) L^*. \qquad \square$$

Very informative illustrations

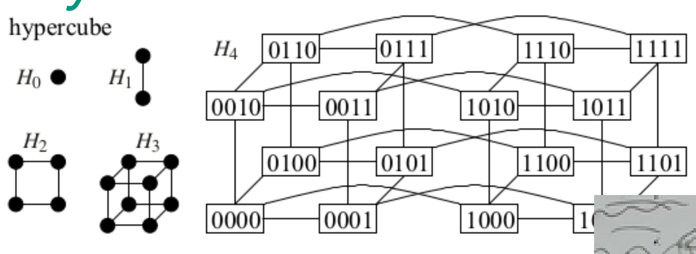
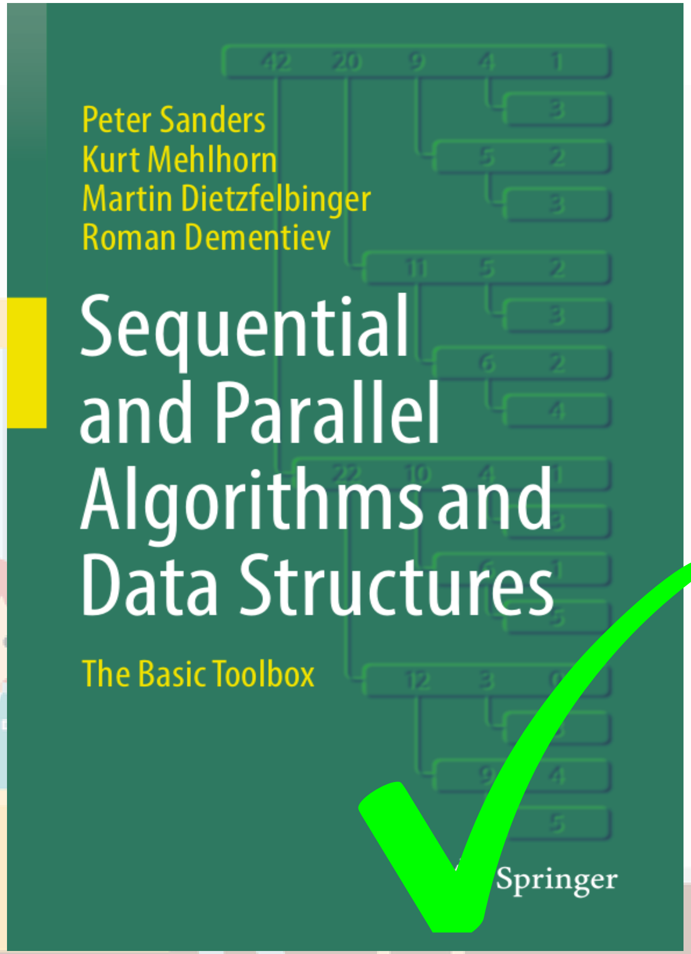


Fig. 13.2. Basic communication topologies for collective comm

Influential Authors



Binary Classification Problem

Very well written

Corollary 12.6. *The approximation ratio of the shortest-queue algorithm is* $2 - 1/m$.

Proof. Let $L_1 = \frac{1}{m}\sum_i t_i$ and $L_2 = \max_i t_i$. The makespan L^* of the optimal solution is at least $\max(L_1, L_2)$. The makespan of the shortest-queue solution is bounded by

$$L_1 + \frac{m-1}{m} L_2 \leq L^* + \frac{m-1}{m} L^* = \left(2 - \frac{1}{m}\right) L^*. \qquad \square$$

Very informative illustrations

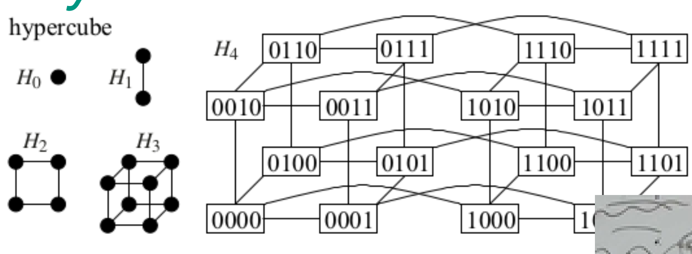
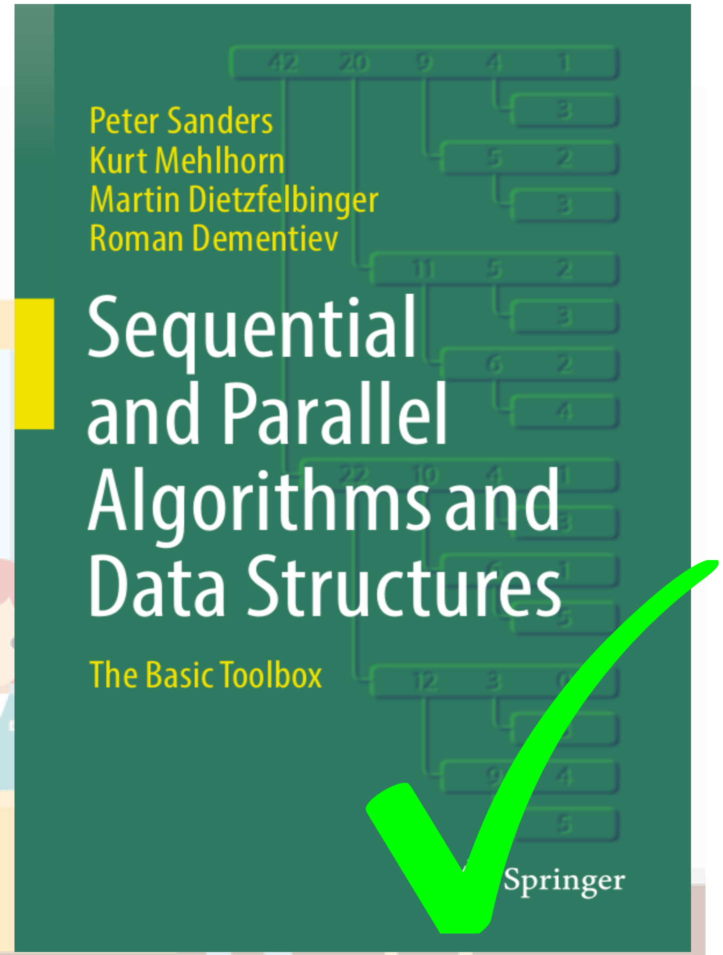


Fig. 13.2. Basic communication topologies for collective comm

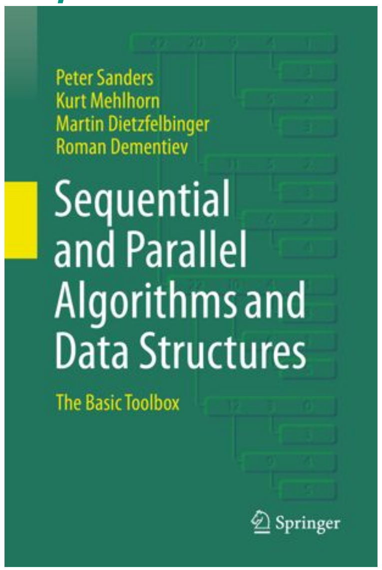
Influential Authors



Institute of Theoretical Informatics
Algorithmics Group

Binary Classification Problem

Example of an excellent book



Train classifier on n **labeled** data points

$$(x_i, y_i)$$

Data point $x \in \mathbb{R}^d$

Features:
text patterns, illustration quality, authors, ...

Label $y \in \{-1, +1\}$

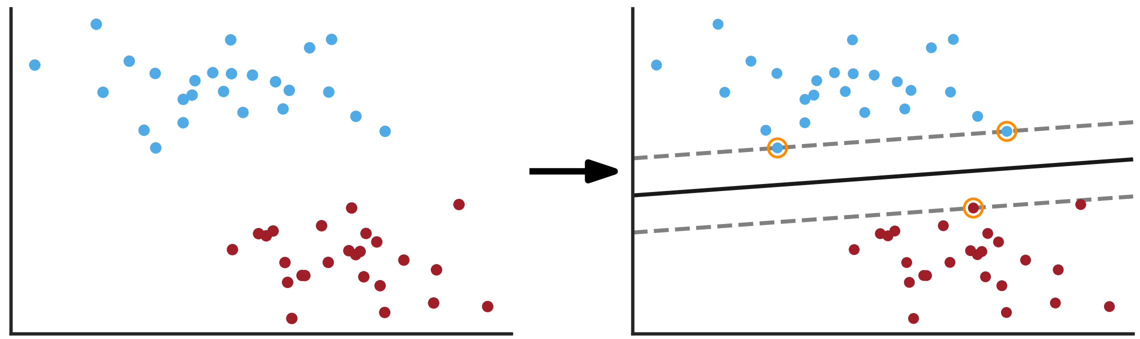
Result:
+1: good book
-1: bad book

Goal: Assign label y_{n+1} to new data points x_{n+1}

Support Vector Machines [CV'97]

Find **hyperplane** with **maximum margin** between classes $\mathbf{C}^{-}$ & $\mathbf{C}^{+}$

Linear SVM



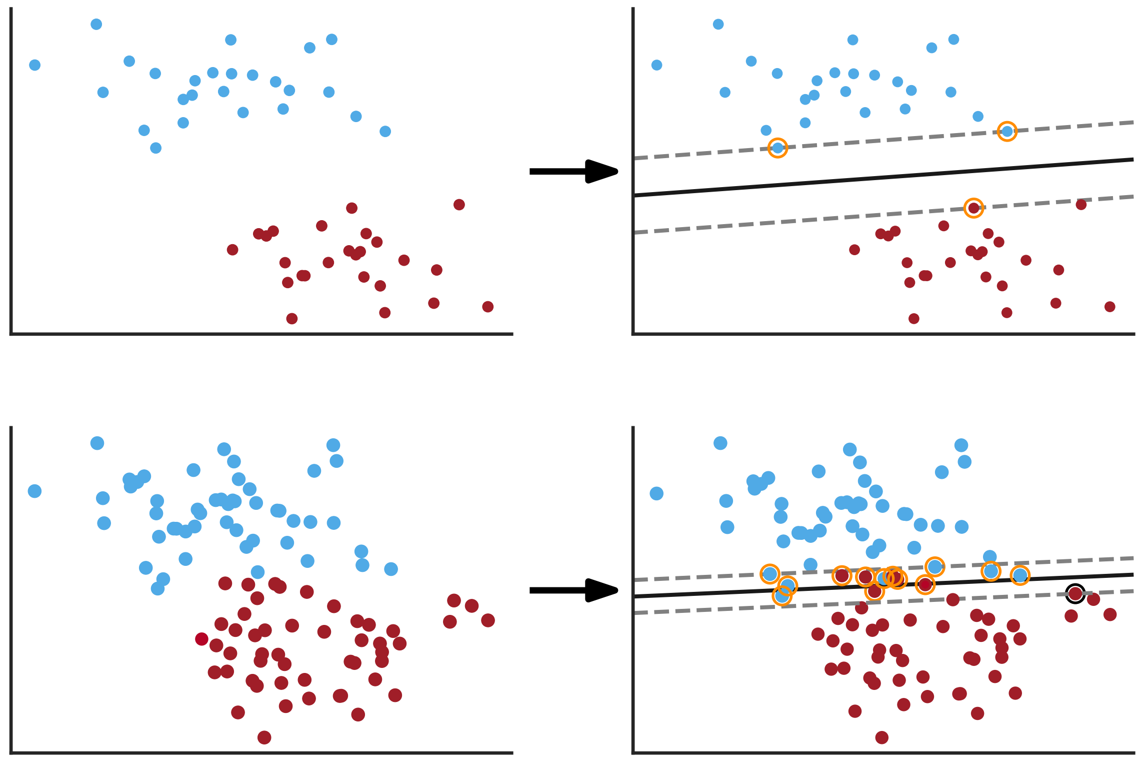
Institute of Theoretical Informatics
Algorithmics Group

Support Vector Machines [CV'97]

Find **hyperplane** with **maximum margin** between classes $\mathbf{C}^-$ & $\mathbf{C}^+$

Linear SVM

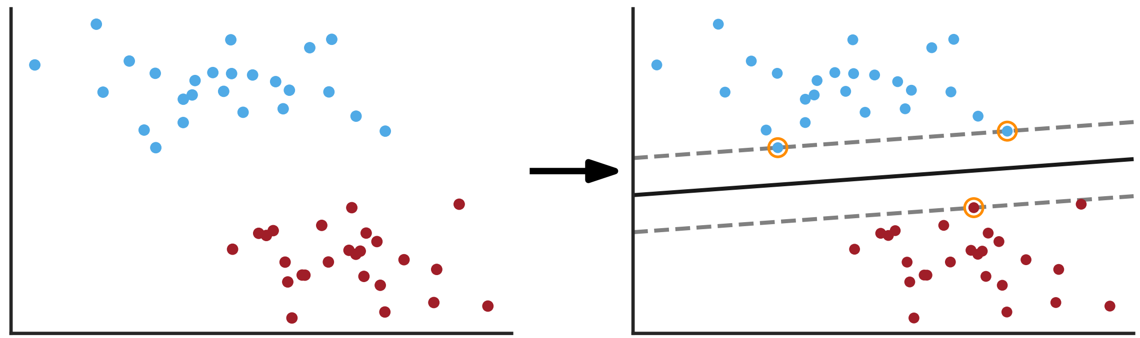
Soft Margin SVM



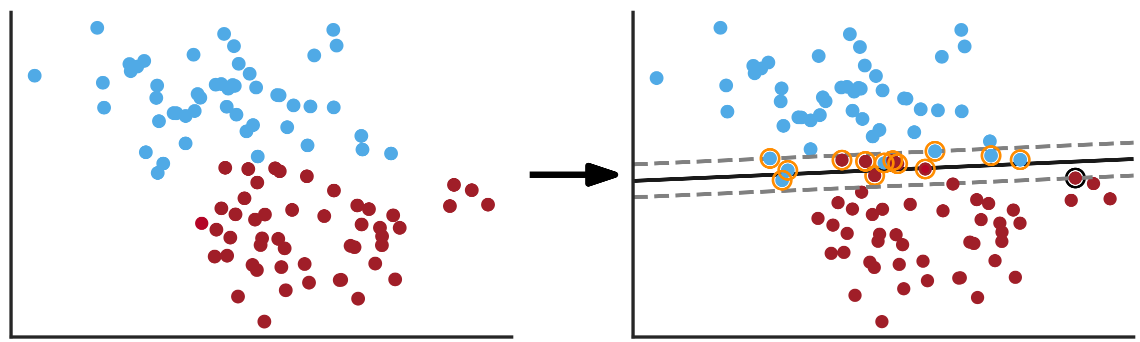
Support Vector Machines [CV'97]

Find **hyperplane** with **maximum margin** between classes $\mathbf{C}^{-}$ & $\mathbf{C}^{+}$

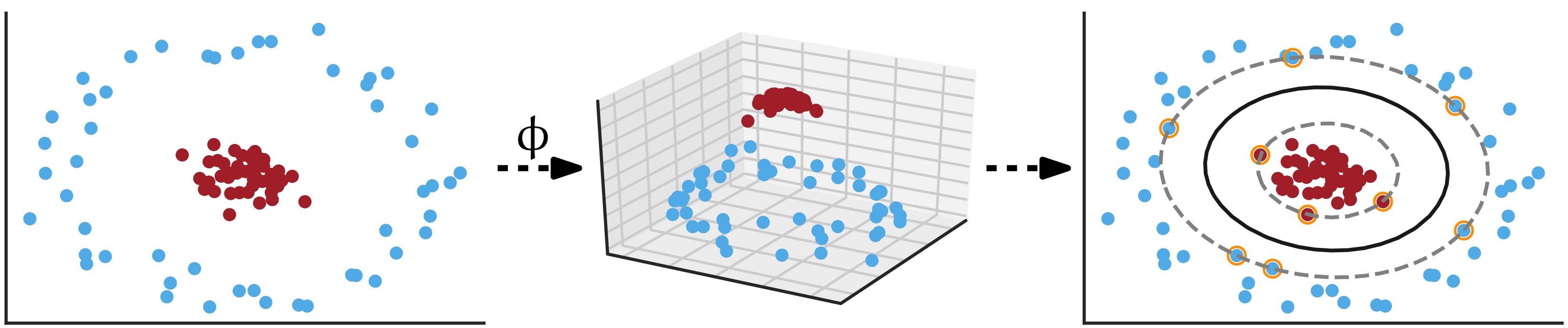
Linear
SVM



Soft Margin
SVM



Nonlinear
SVM

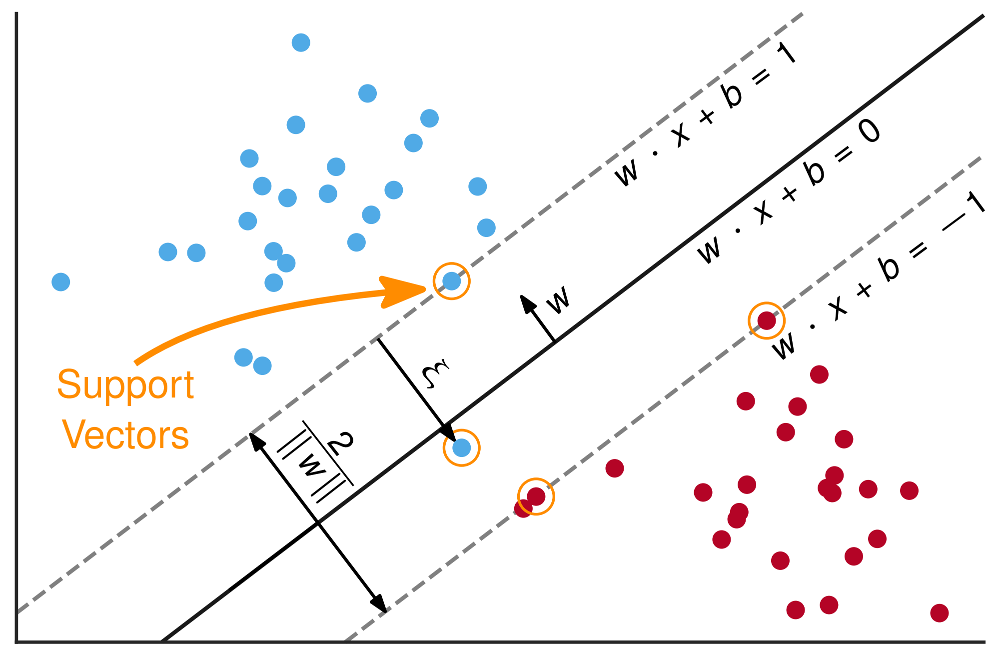


Support Vector Machines [CV'97]

SVM Optimization Problem:

$$\text{minimize} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \xi_i$$

$$\text{subject to} \quad y_i(w \cdot \phi(x_i) + b) \geq 1 - \xi_i, \quad \xi_i \geq 0$$



Support Vector Machines [CV'97]

Karlsruhe Institute of Technology

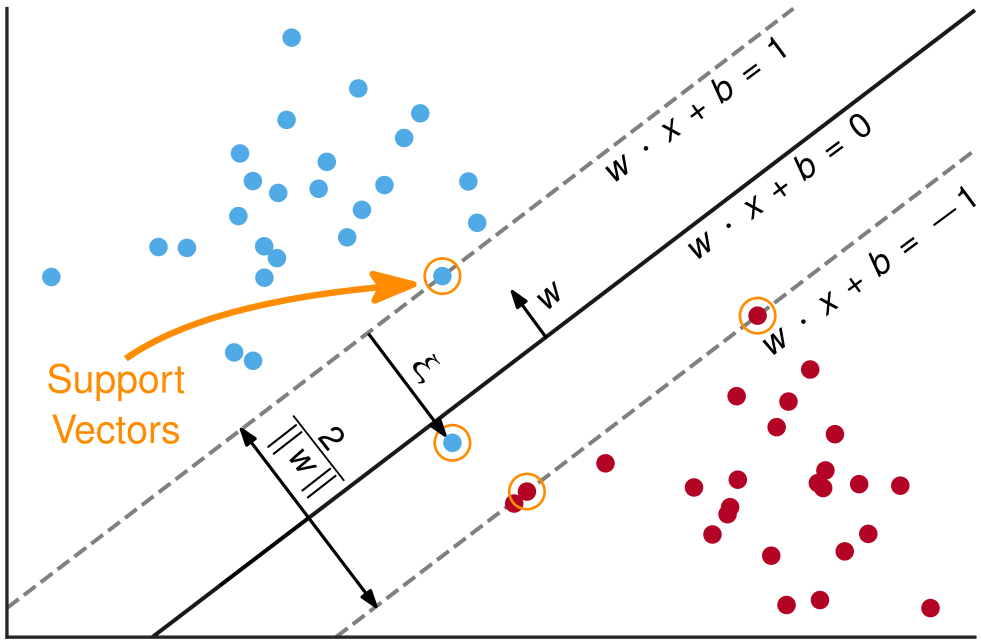
SVM Optimization Problem:

$$\text{minimize} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i$$

Slack penalty

$$\text{subject to} \quad y_i(w \cdot \phi(x_i) + b) \geq 1 - \xi_i, \quad \xi_i \geq 0$$

Slack variables



Support Vector Machines [CV'97]

SVM Optimization Problem:

$$\text{minimize} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i$$

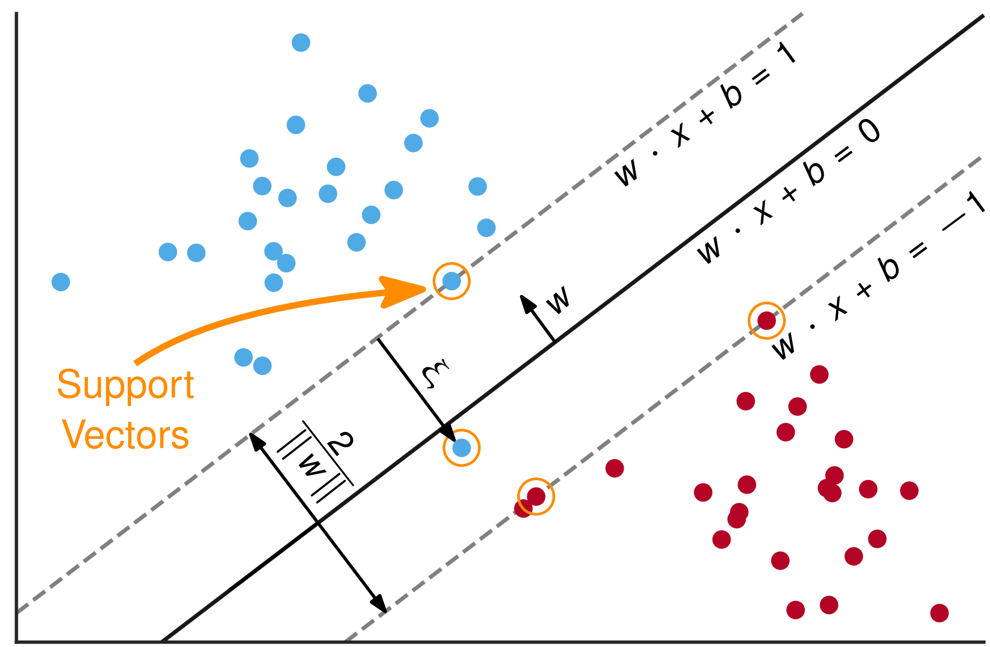
Slack penalty

$$\text{subject to} \quad y_i(w \cdot \phi(x_i) + b) \geq 1 - \xi_i, \quad \xi_i \geq 0$$

Mapping to higher dimensional space

$$\phi : \mathbb{R}^d \rightarrow \mathbb{R}^p (d \leq p)$$

Slack variables



Support Vector Machines [CV'97]

Karlsruhe Institute of Technology

SVM Optimization Problem:

minimize $\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i$

subject to $y_i(w \cdot \phi(x_i) + b) \geq 1 - \xi_i, \quad \xi_i \geq 0$

Slack penalty

Slack variables

Mapping to higher dimensional space

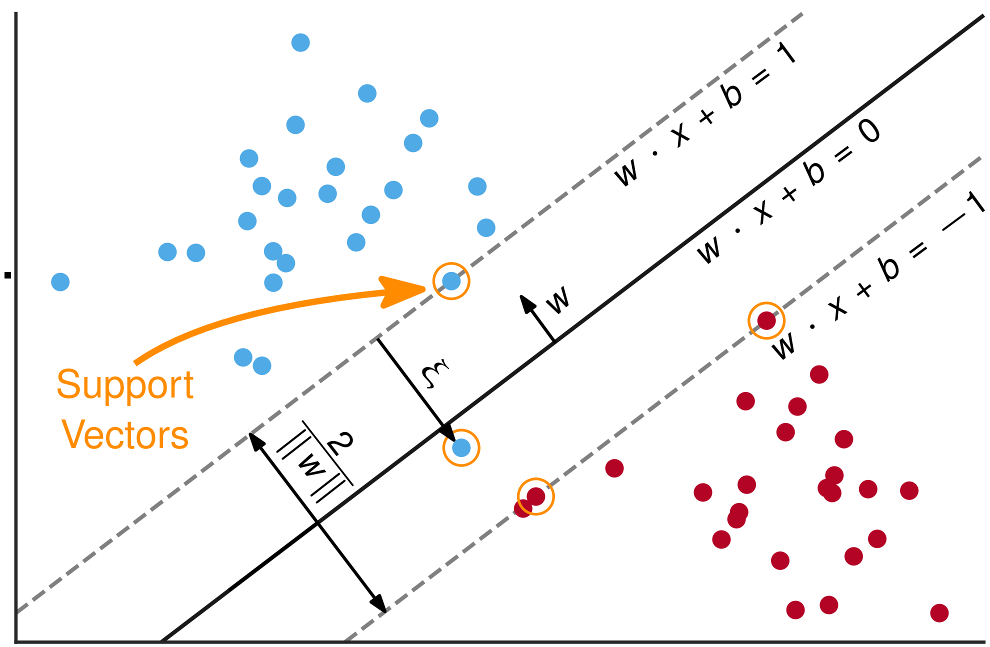
$\phi : \mathbb{R}^d \to \mathbb{R}^p (d \leq p)$

via ...

Kernel Trick:

- Replace inner product by kernel fct.
- $k(x_i, x_j) = \phi(x_i)^T \phi(x_j)$
- Here: $k(x_i, x_j) = \exp(-\gamma\|x_i - x_j\|^2)$

Gaussian kernel (RBF)



Support Vector Machines – Training

Model Selection: instance-specific tuning of several parameters

- Slack penalty C
- Kernel parameters (here: γ)

Complexity:

- Solver running time between $\mathcal{O}\left(n^2\right)$ & $\mathcal{O}\left(n^3\right)$ [GCBDV'04]
- Model selection $\rightsquigarrow$ train many models

Institute of Theoretical Informatics
Algorithmics Group

Support Vector Machines – Training

Model Selection: instance-specific tuning of several parameters

- Slack penalty C
- Kernel pa

Training on **large** data sets becomes **infeasible**!

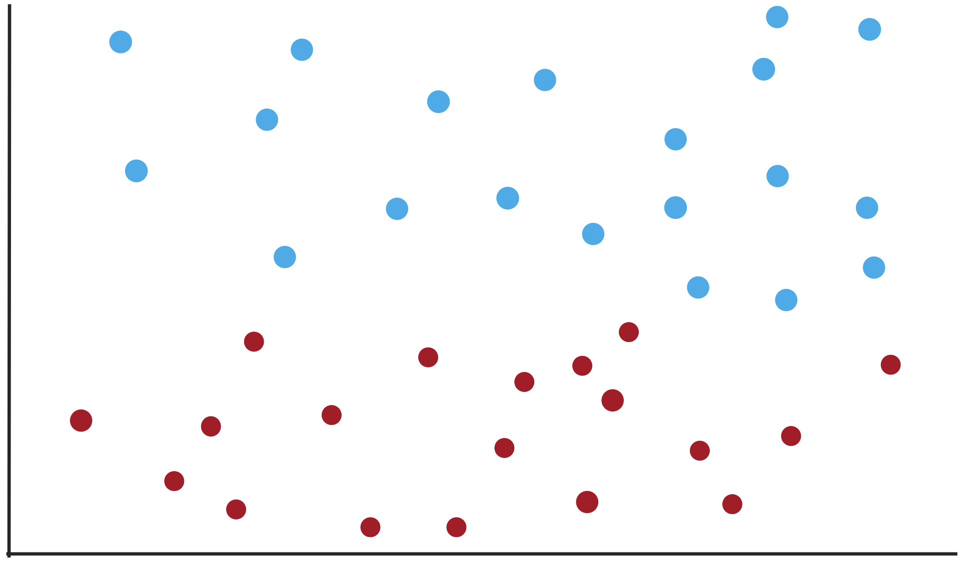
Complexity:

- Solver running time between $\mathcal{O}(n^2)$ & $\mathcal{O}(n^3)$ [GCBDV'04]
- Model selection $\rightsquigarrow$ train many models

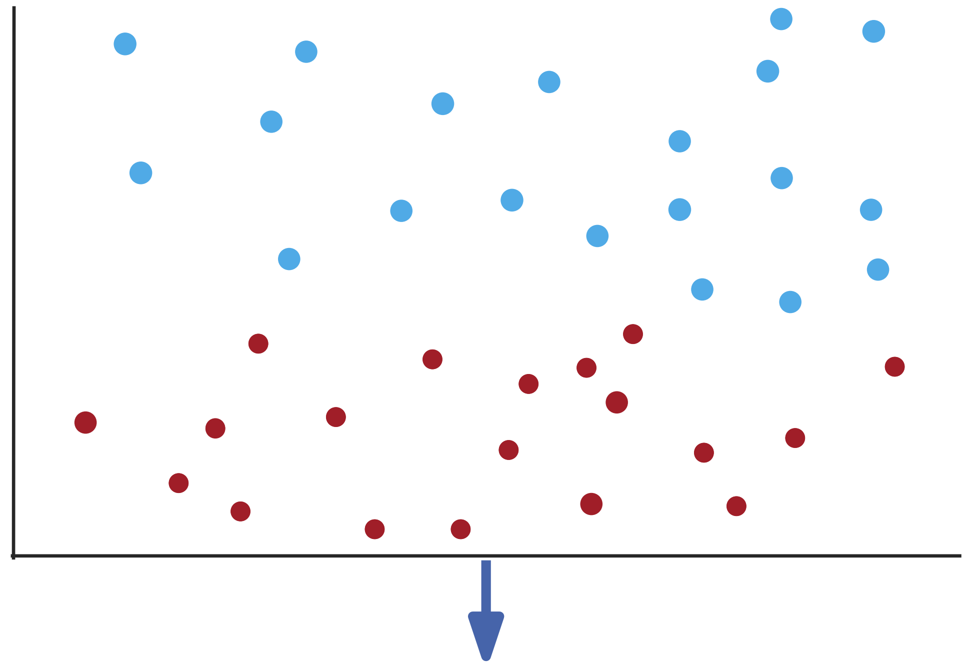
Performance Improvements:

- Sampling
- Parallelization
- Hierarchical techniques
  - Input space
  - Graph representation

From Feature Vectors to Graphs



From Feature Vectors to Graphs

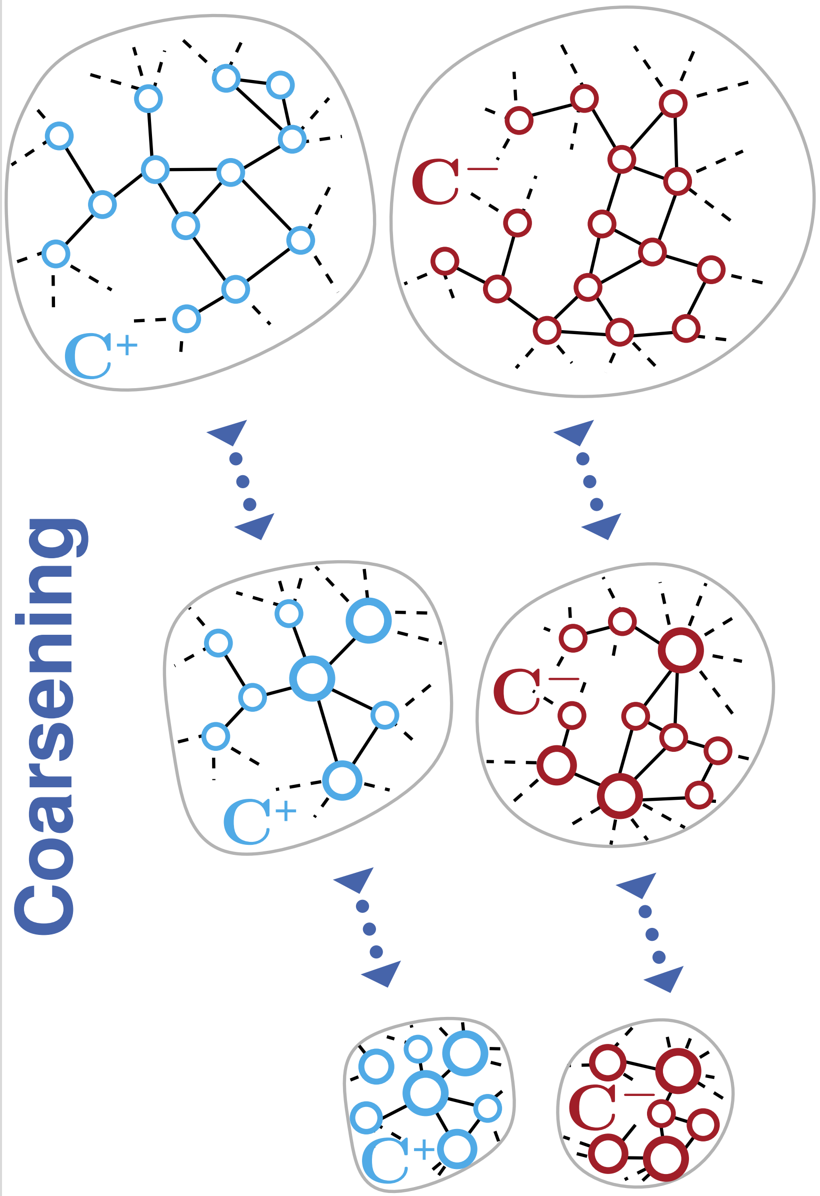


Approx. k-**nearest** neighbors
[RS'15]

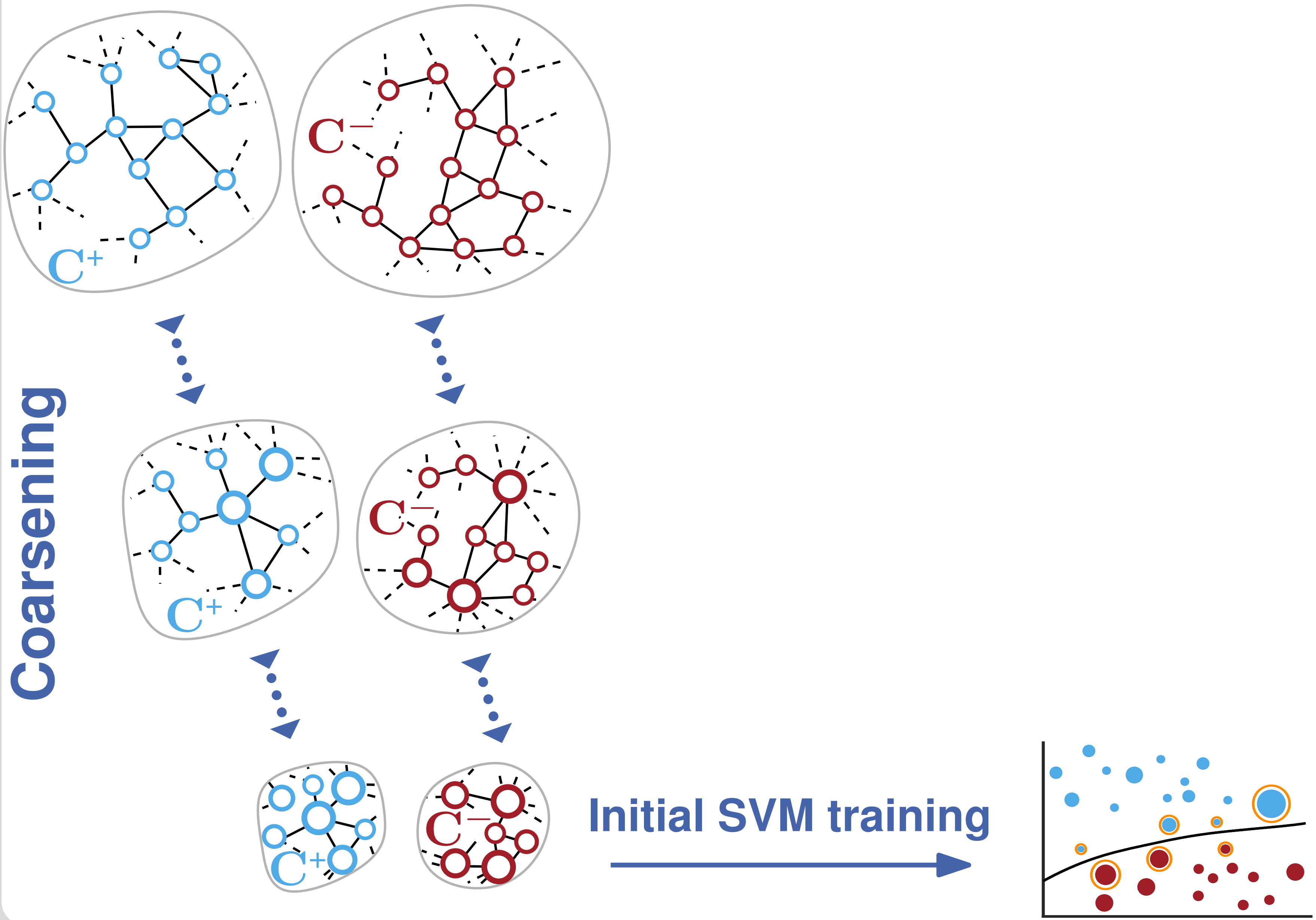
$\omega(e) = \frac{1}{\text{dist}(p,q)}$ ⇒ encode **proximity** information into **edge weights**

Institute of Theoretical Informatics
Algorithmics Group

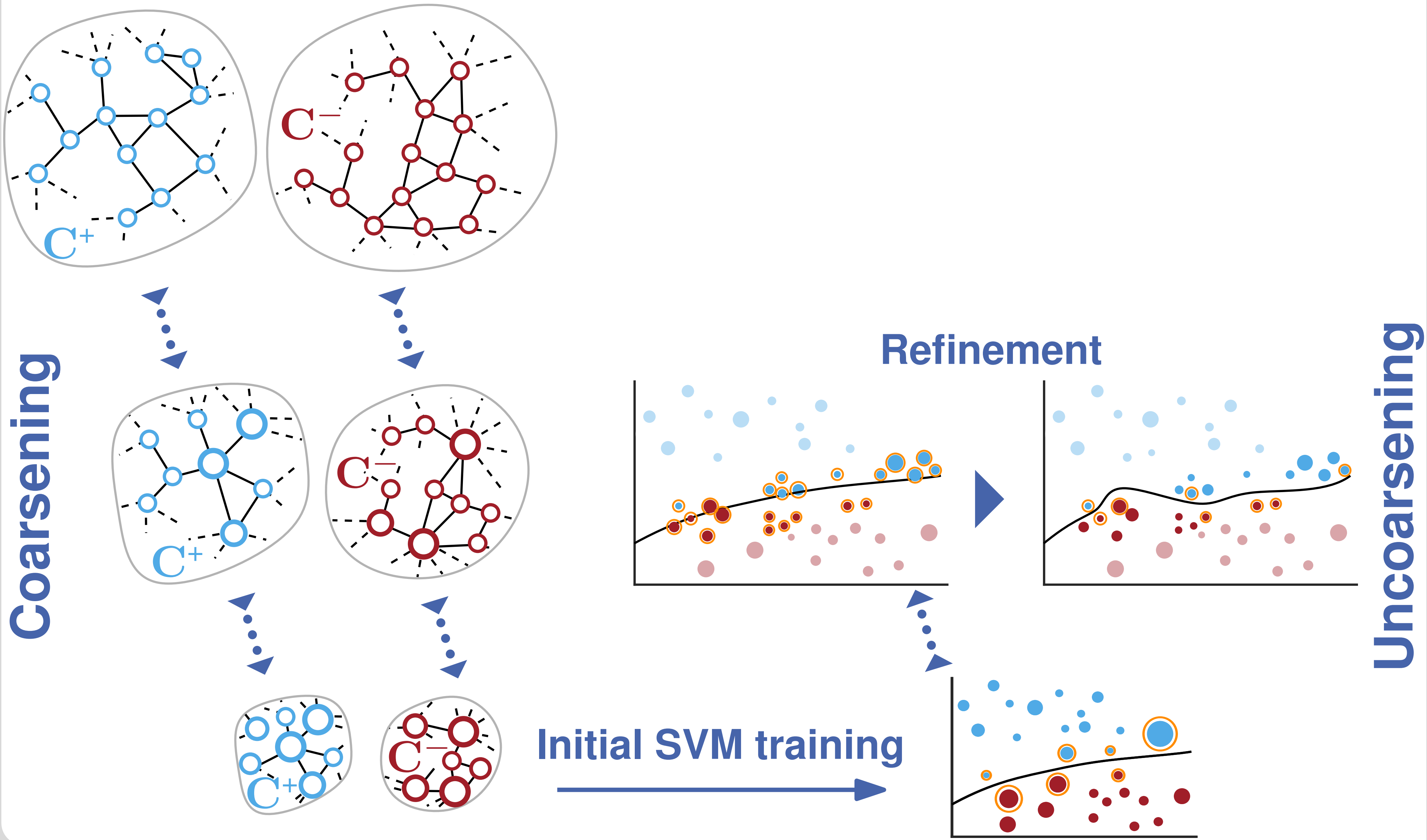
Multilevel Support Vector Machines [RS'15]



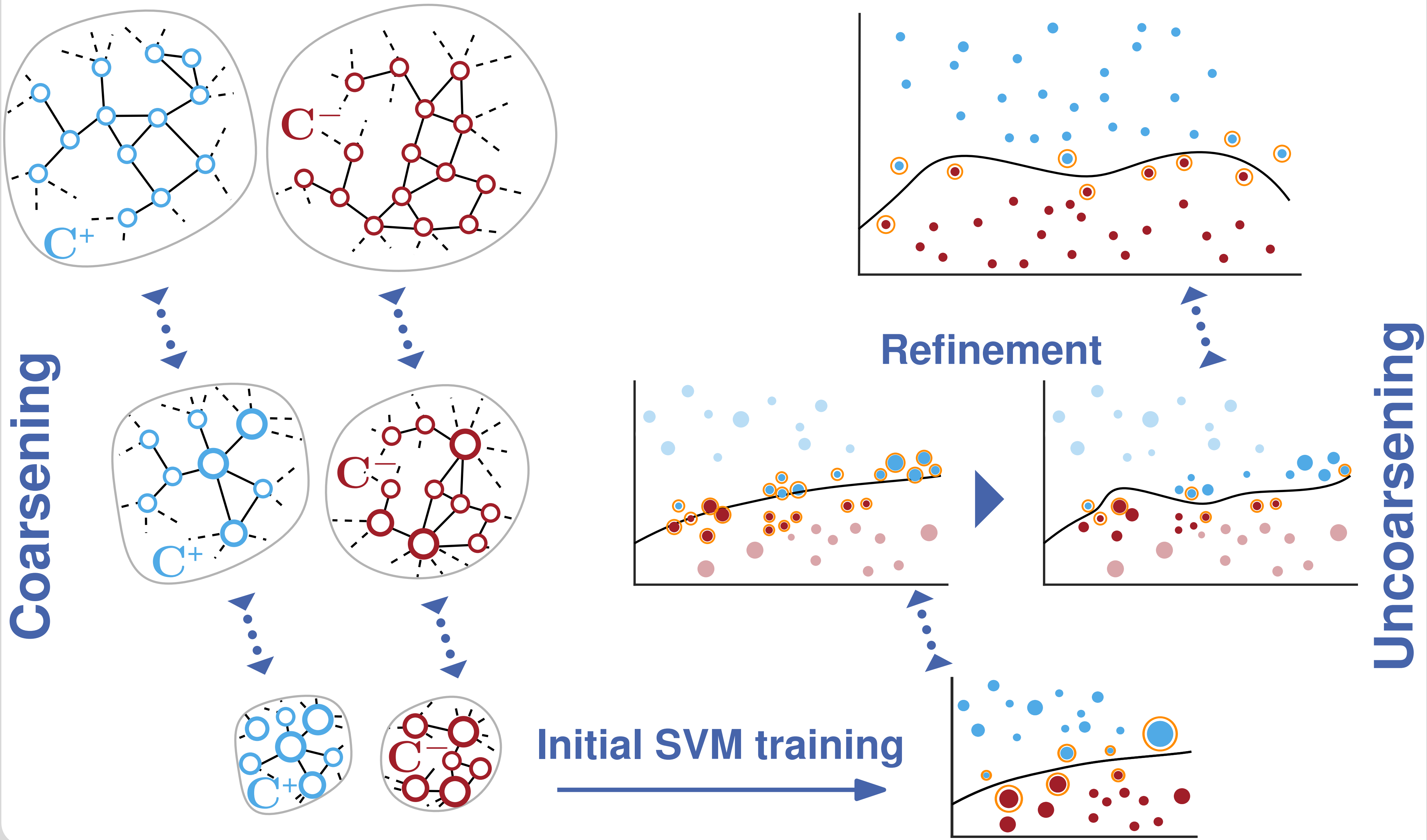
Multilevel Support Vector Machines [RS'15]



Multilevel Support Vector Machines [RS'15]



Multilevel Support Vector Machines [RS'15]



Experimental Setup

Machine: AMD Opteron 6168 with 1.9 GHz, 256 GB of RAM

Implementation:

- approx. k-nearest neighbors: FLANN 1.8.4 [ML'14]
- SVM training: LibSVM 3.22

Configuration:

- $k = 10$ nearest neighbors
- $\ell = 10$ label propagation iterations
- stop coarsening $|C^{+/-}| \approx 500$ nodes

Algorithms:

- KaSVM / $\text{KaSVM}_{\text{fast}}$
- mlsvm-AMG (outperforms DC-SVM & EnsembleSVM) [SJKLRS'17]
- LibSVM

Experimental Setup

Machine: AMD Opteron 6168 with 1.9 GHz, 256 GB of RAM

Implementation:

- approx. k-nearest neighbors: FLANN 1.8.4 [ML'14]
- SVM training: LibSVM 3.22

Configuration:

- k = 10 nearest neighbors
- ℓ = 10 label propagation iterations
- stop coarsening $|C^{+/-}| \approx 500$ nodes

Algorithms:

initially trained model / **no** refinement

- KaSVM / $\text{KaSVM}_{\text{fast}}$
- mlsvm-AMG (outperforms DC-SVM & EnsembleSVM) [SJKLRS'17]
- LibSVM

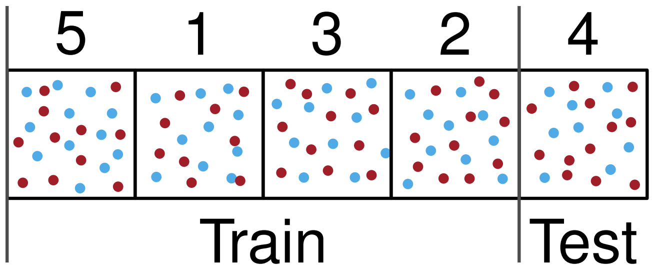
Institute of Theoretical Informatics
Algorithmics Group

Experimental Methodology

Karlsruhe Institute of Technology

k-fold cross validation:

- Shuffle data set → split into $k = 5$ parts
- k training repetitions:
  ⇒ **Training** set: k-1 parts
  ⇒ **Test** set: 1 part
- 5 k-folds per instance



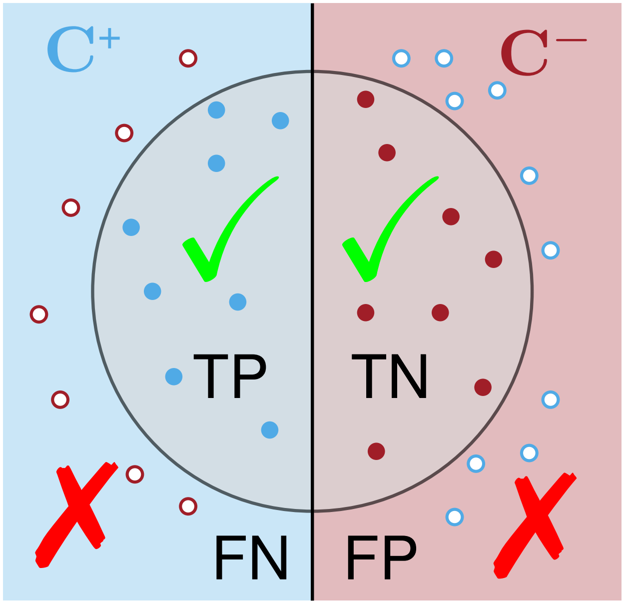
Performance Measures:

$$\text{Accuracy}\ (ACC) = \frac{TP + TN}{FP + TN + TP + FN}$$

$$\text{Sensitivity}\ (SN) = \frac{TP}{TP + FN}$$

$$\text{Specificity}\ (SP) = \frac{TN}{TN + FP}$$

$$\text{Geometric mean} = \sqrt{SP \cdot SN}$$



Institute of Theoretical Informatics
Algorithmics Group

Running Time

Karlsruhe Institute of Technology

	mlsvm-AMG	KaSVM	KaSVM$_{fast}$	LibSVM
		running time [s]		
Advertisement	343	192	70	557
APS failure	1 473	109	13	-
Buzz	110	121	19	-
Census	3 047	657	38	-
Clean (Musk)	14	8	4	320
Cod-rna	80	43	7	15 700
EEG Eye State	123	1 320	0.9	2 700
Forest (Class 3)	10 156	744	99	-
Forest (Class 5)	6 986	1 090	158	-
Forest (Class 7)	5 393	1 990	114	-
Hypothyroid	2	3	0.9	13
Isolet (Class A)	1 627	23	7	856
Letter (Class A)	17	4	2	1 930
Letter (Class B)	55	4	2	1 590
Letter (Class H)	74	9	2	1 970
Letter (Class Z)	31	3	2	1 710
Nursery	7	2	0.7	998
Protein	3 654	41	17	-
Ringnorm	10	13	0.6	161
Skin	81	18	12	38 200
Sleep (Class 1)	1 594	3 080	12	-
Twonorm	7	0.7	0.5	109

Institute of Theoretical Informatics
Algorithmics Group

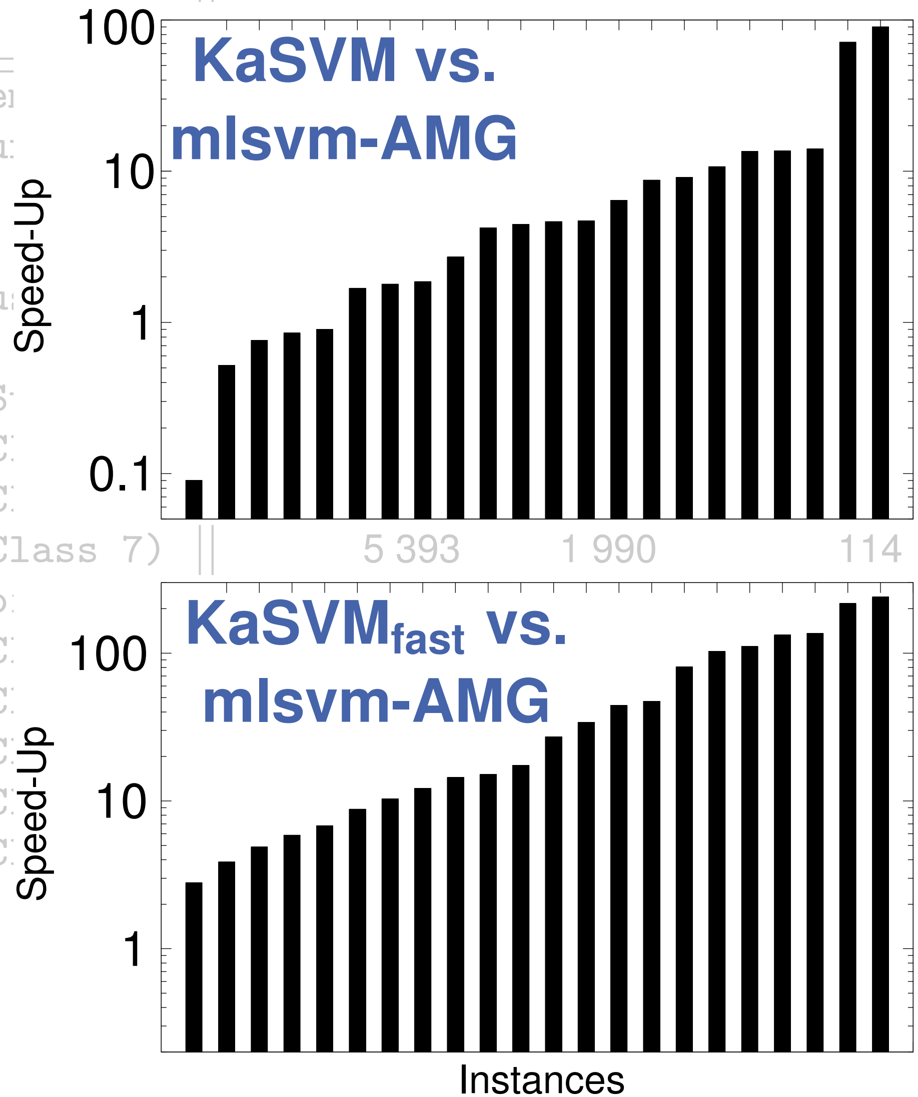
Running Time

	mlsvm-AMG	KaSVM	$KaSVM_{fast}$	LibSVM
		running time [s]		
Advertisement	343	192	70	557
APS failure	1 473	109	13	⊟
Buzz	110	121	19	⊟
Census	3 047	657	38	⊟
Clean (Musk)	14	8	4	320
Cod-rna	80	43	7	15 700
EEG Eye State	123	1 320	0.9	2 700
Forest (Class 3)	10 156	744	99	⊟
Forest (Class 5)	6 986	1 090	158	⊟
Forest (Class 7)	5 393	1 990	114	⊟
Hypothyroid	2	3	0.9	13
Isolet (Class A)	1 627	23	7	856
Letter (Class A)	17	4	2	1 930
Letter (Class B)	55	4	2	1 590
Letter (Class H)	74	9	2	1 970
Letter (Class Z)	31	3	2	1 710
Nursery	7	2	0.7	998
Protein	3 654	41	17	⊟
Ringnorm	10	13	0.6	161
Skin	81	18	12	38 200
Sleep (Class 1)	1 594	3 080	12	⊟
Twonorm	7	0.7	0.5	109

⊟ Running time $> 24h$

Running Time

	mlsvm-AMG	KaSVM	$KaSVM_{fast}$	LibSVM
Advertise[illegible]				557
APS failu[illegible]				-
Buzz				-
Census				-
Clean (Mu[illegible]				320
Cod-rna				15 700
EEG Eye S[illegible]				2 700
Forest (C[illegible]				-
Forest (C[illegible]				-
Forest (Class 7)	5 393	1 990	114	-
Hypothyro[illegible]				13
Isolet (C[illegible]				856
Letter (C[illegible]				1 930
Letter (C[illegible]				1 590
Letter (C[illegible]				1 970
Letter (C[illegible]				1 710
Nursery				998
Protein				-
Ringnorm				161
Skin				38 200
Sleep (Class 1)	1 594	3 080	12	-
Twonorm	7	0.7	0.5	109



Classification Quality

Instance	mlsvm-AMG G-mean	KaSVM G-mean	$KaSVM_{fast}$ G-mean	LibSVM G-mean
Advertisement	**0.91**	**0.91**	0.87	0.941
APS failure	**0.94**	0.92	**0.94**	-
Buzz	**0.95**	0.94	0.93	-
Census	0.81	**0.83**	0.81	-
Clean (Musk)	0.88	**0.96**	0.88	0.996
Cod-rna	**0.94**	**0.94**	0.93	0.855
EEG Eye State	0.75	**0.83**	0.63	0.929
Forest (Class 3)	0.94	**0.95**	0.94	-
Forest (Class 5)	0.79	**0.90**	**0.90**	-
Forest (Class 7)	0.86	**0.93**	0.91	-
Hypothyroid	**0.94**	0.90	**0.94**	0.927
Isolet (Class A)	0.00	**0.99**	0.88	0.990
Letter (Class A)	0.94	**0.96**	0.95	0.995
Letter (Class B)	0.88	**0.92**	**0.92**	0.979
Letter (Class H)	0.76	**0.89**	0.86	0.970
Letter (Class Z)	0.92	**0.95**	**0.95**	0.991
Nursery	**1.00**	**1.00**	**1.00**	1.000
Protein	0.92	**0.93**	0.91	-
Ringnorm	**0.98**	0.97	0.82	0.987
Skin	0.99	**1.00**	**1.00**	1.000
Sleep (Class 1)	0.69	**0.70**	0.41	-
Twonorm	**0.97**	0.96	0.96	0.981

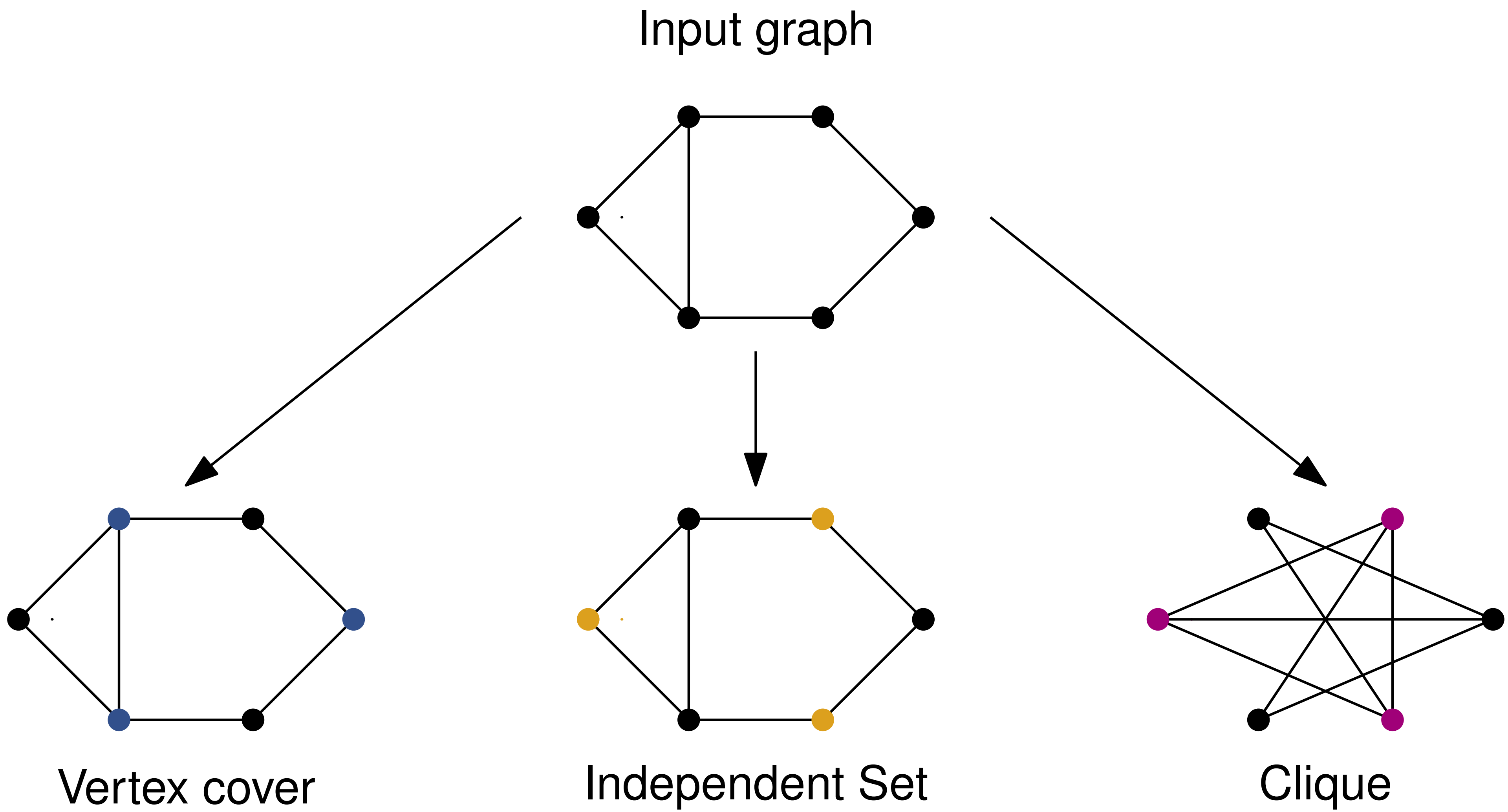
Institute of Theoretical Informatics
Algorithmics Group

PACE Challenge 2019

Vertex Cover Track

D. Hespe, S. Lamm, C. Schulz, D. Strash

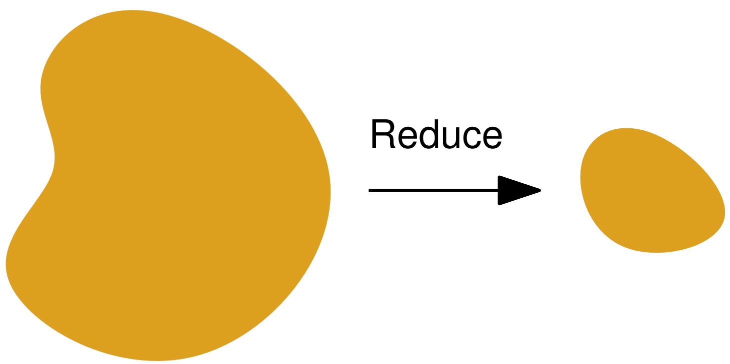
Vertex Cover and Complementary Problems



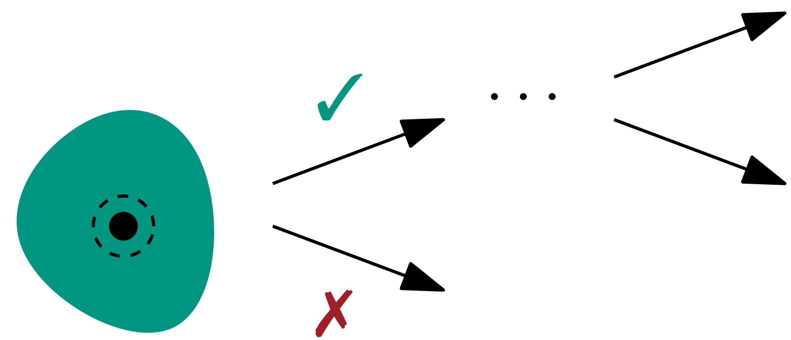
Institute of Theoretical Informatics
Algorithmics Group

Techniques

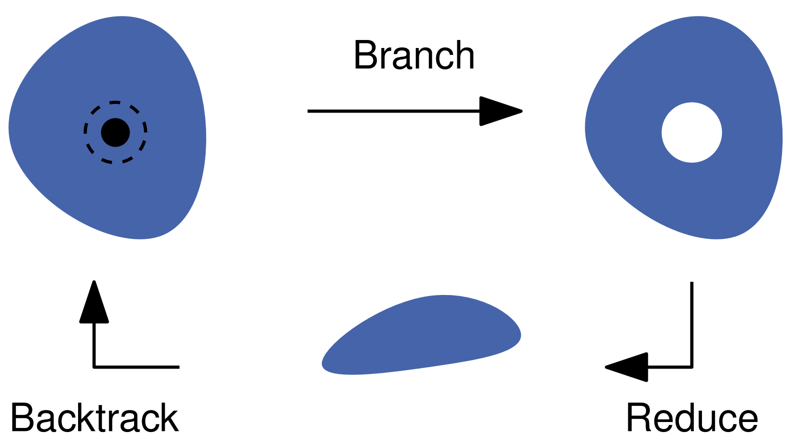
Kernelization



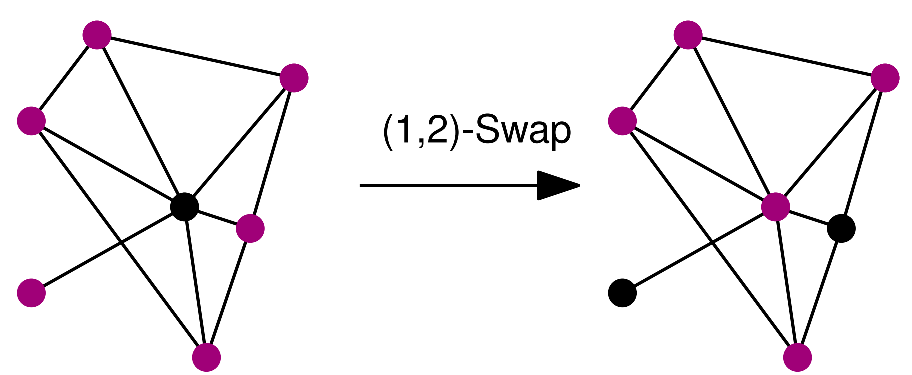
Branch-and-Bound



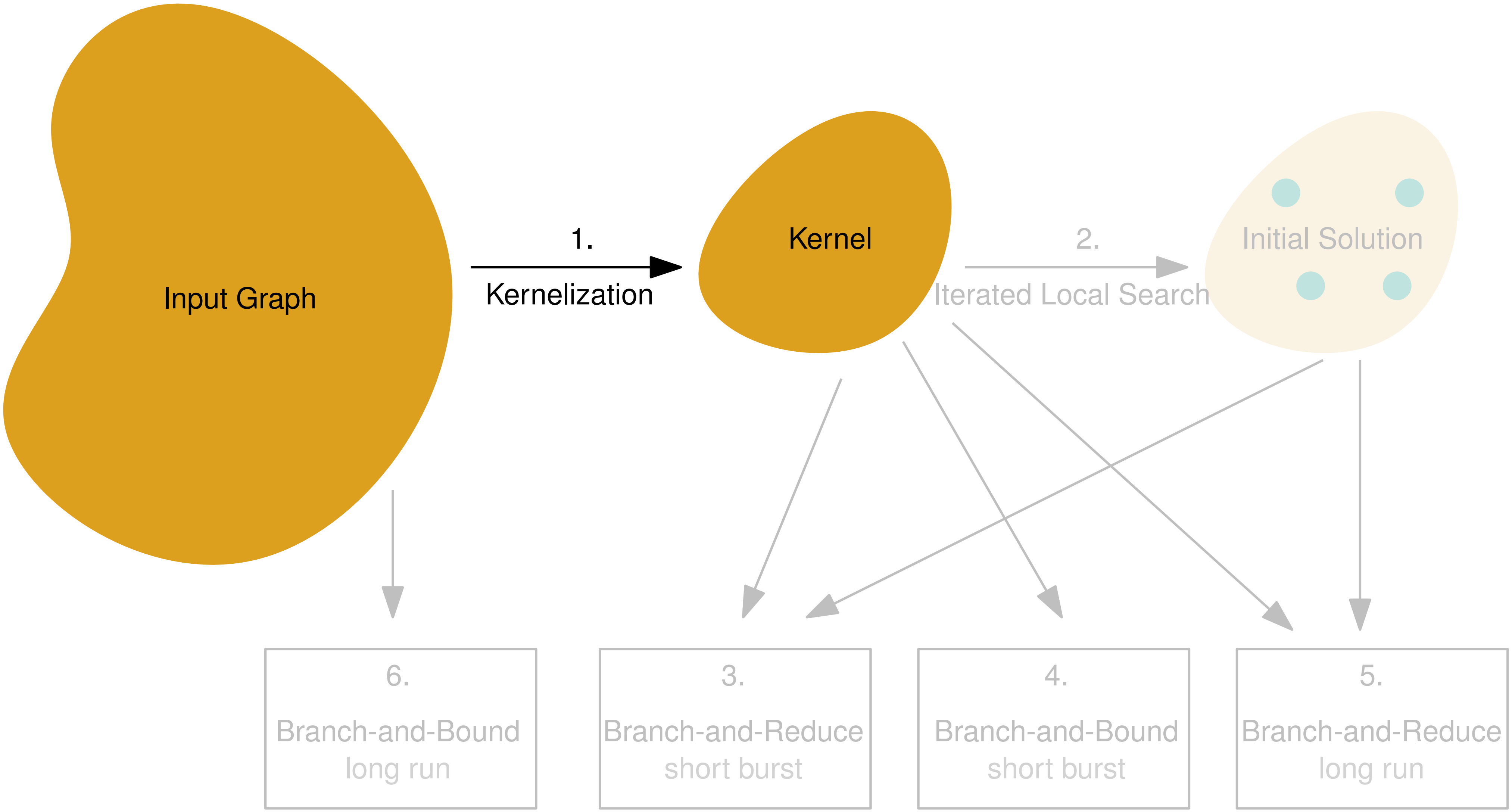
Branch-and-Reduce



Iterated Local Search

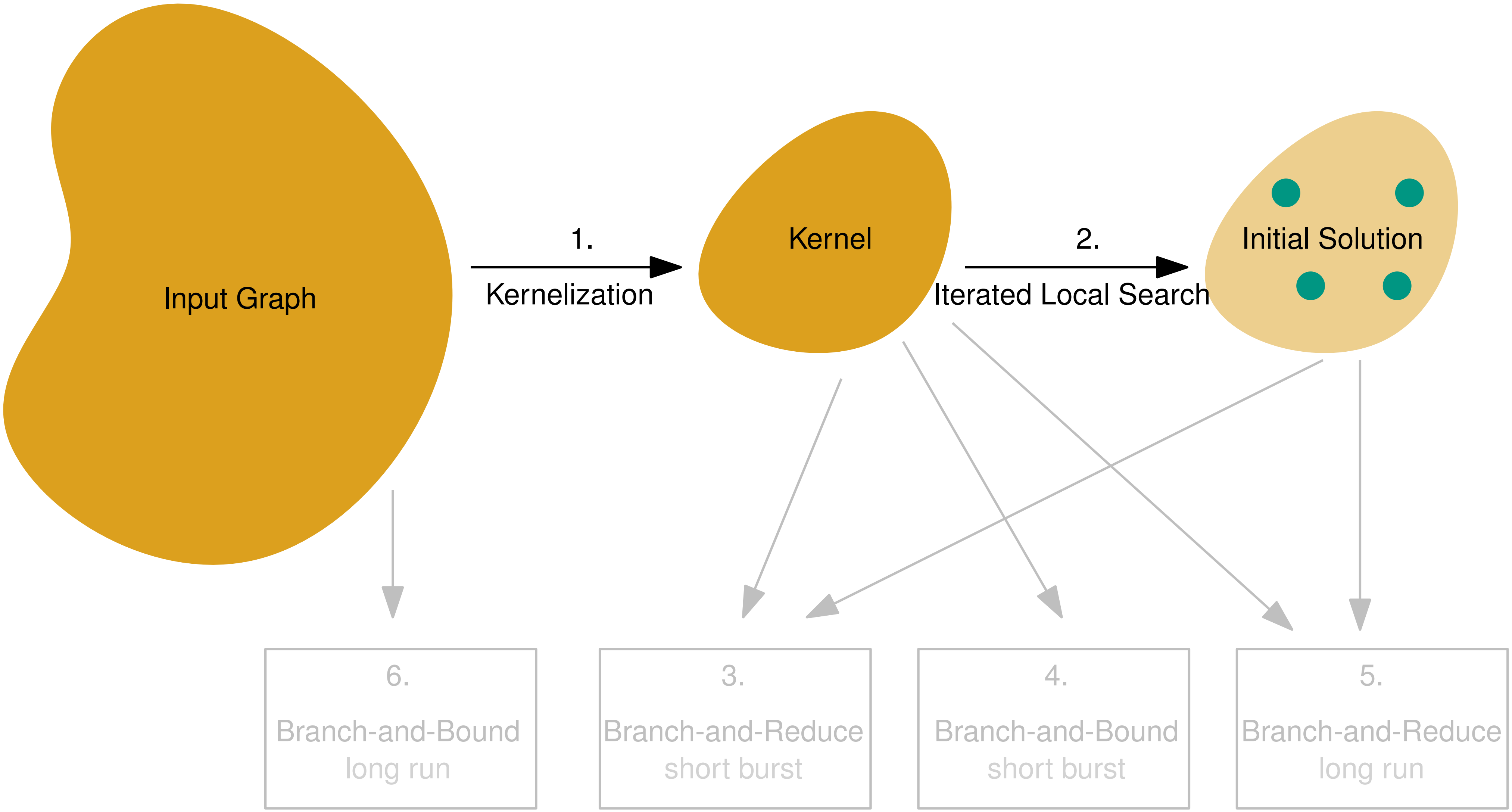


Algorithm Overview



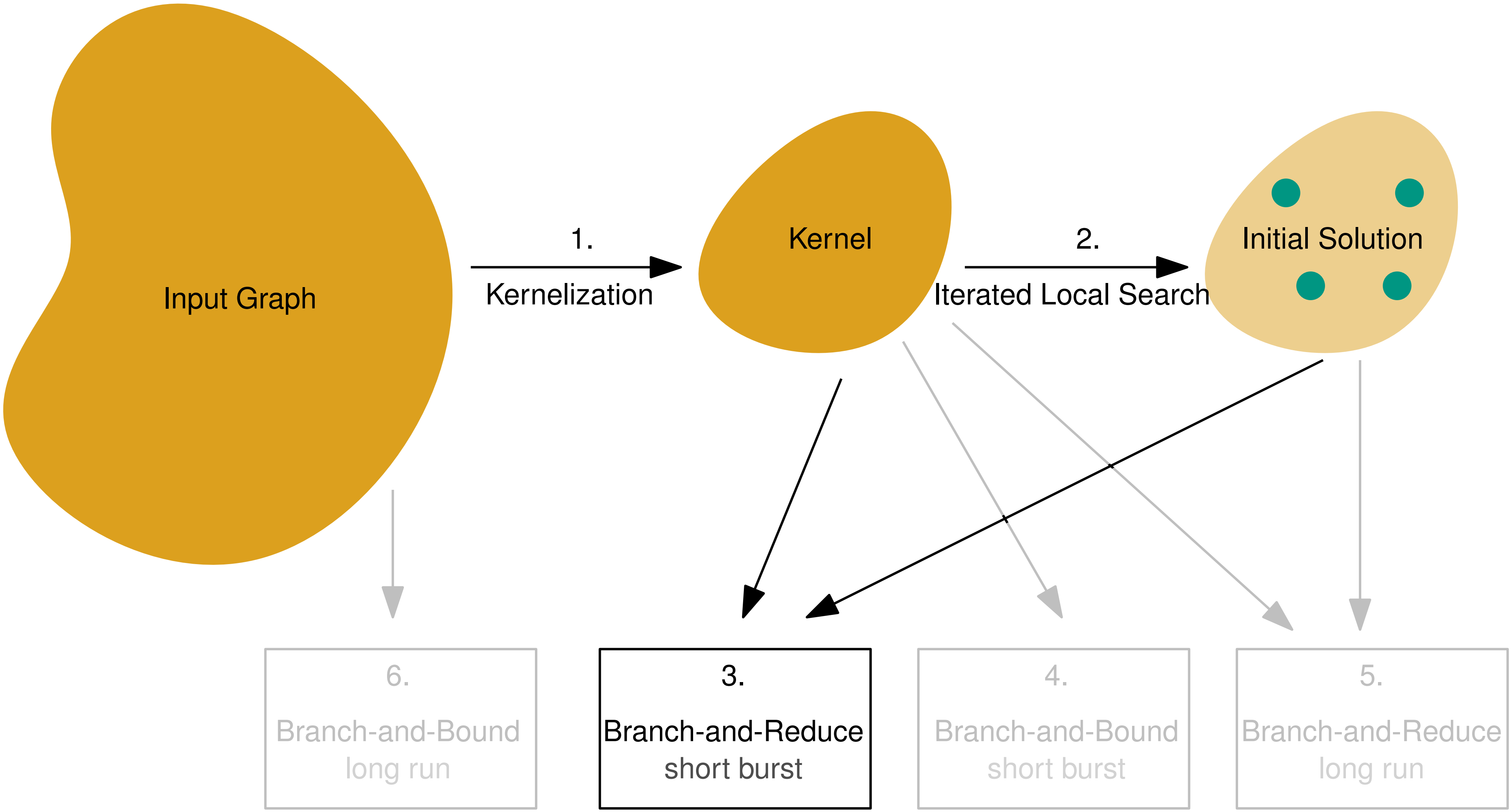
Institute of Theoretical Informatics
Algorithmics Group

Algorithm Overview

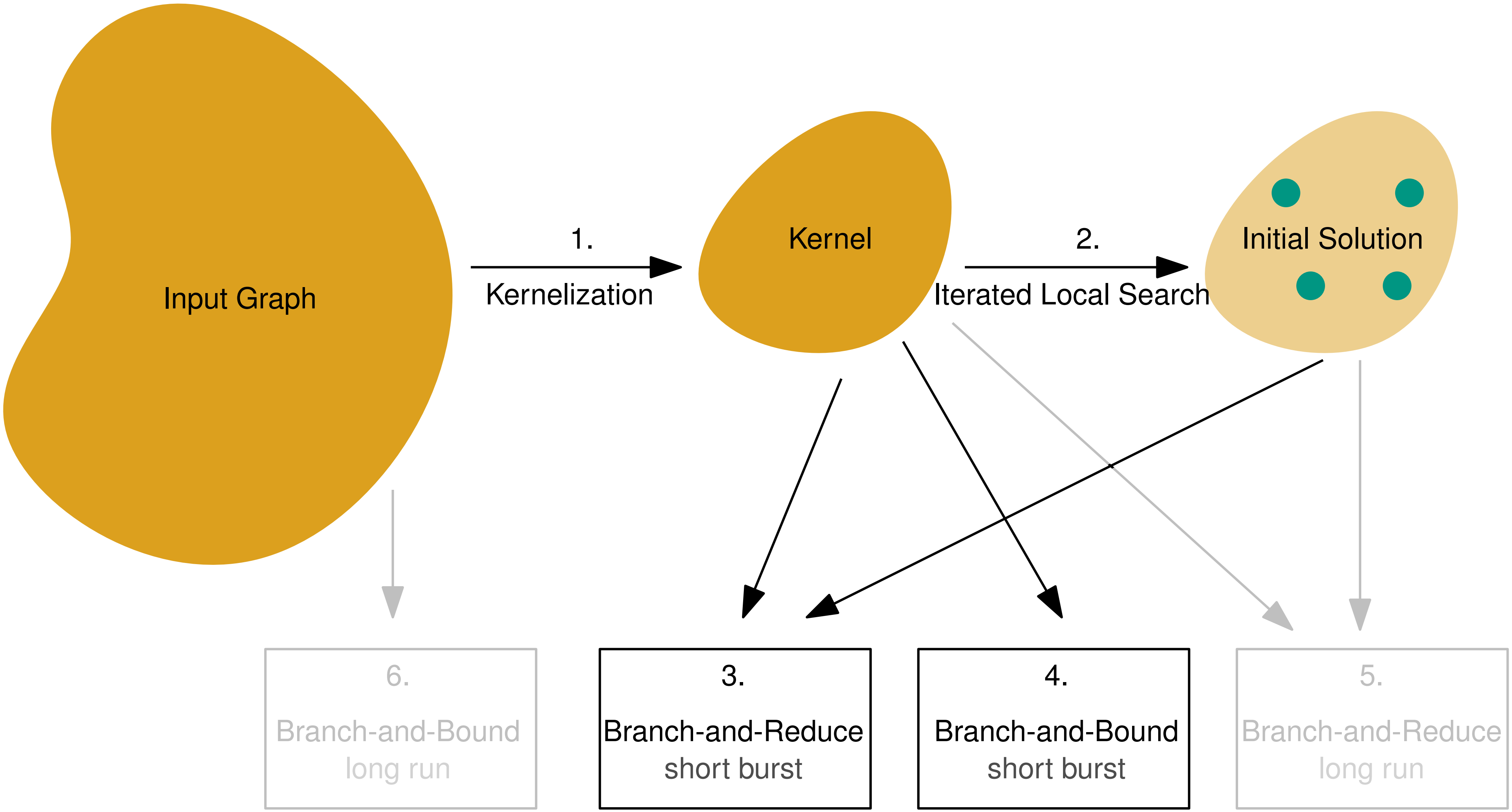


Algorithm Overview

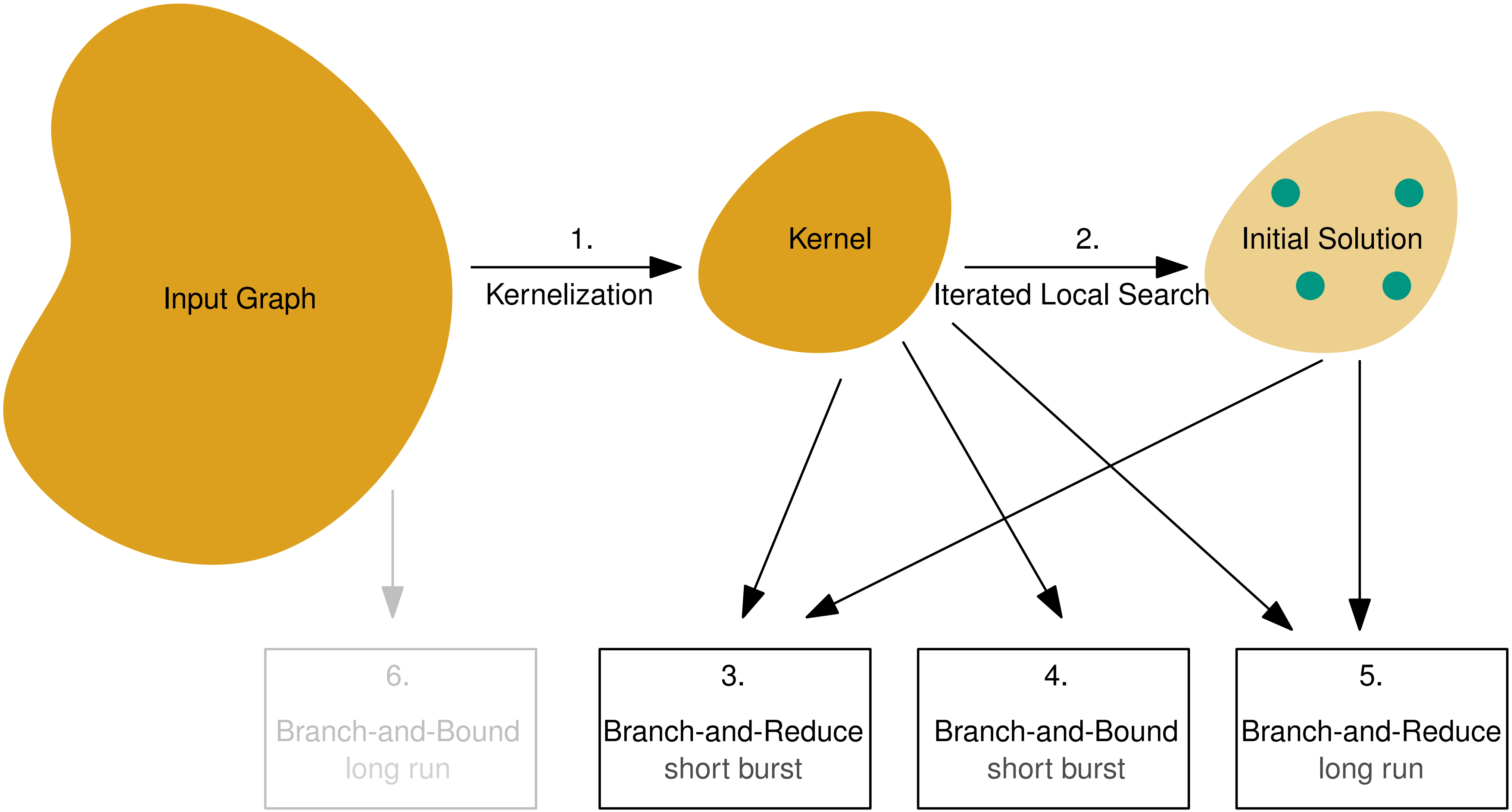
Karlsruhe Institute of Technology



Algorithm Overview

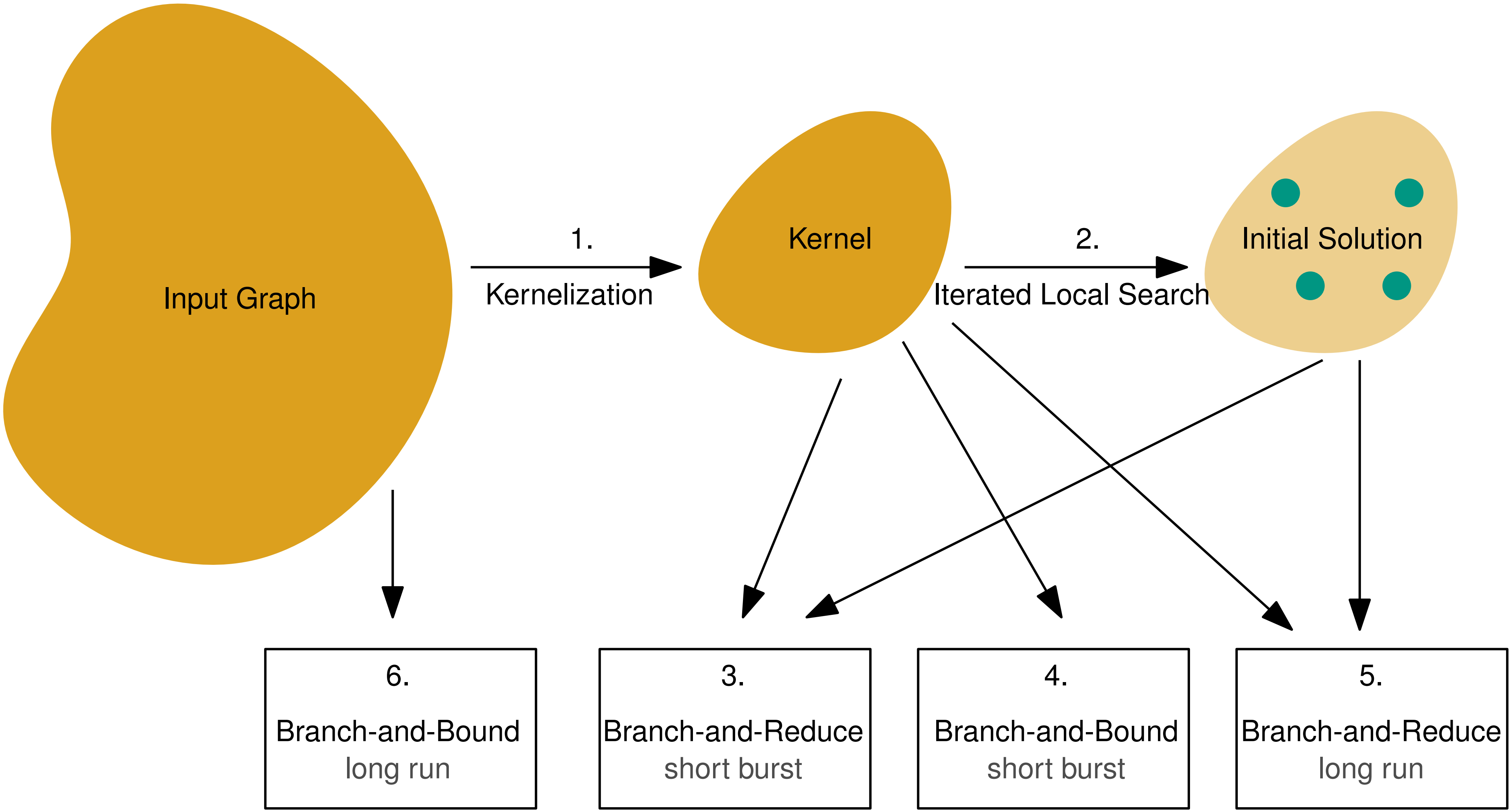


Algorithm Overview

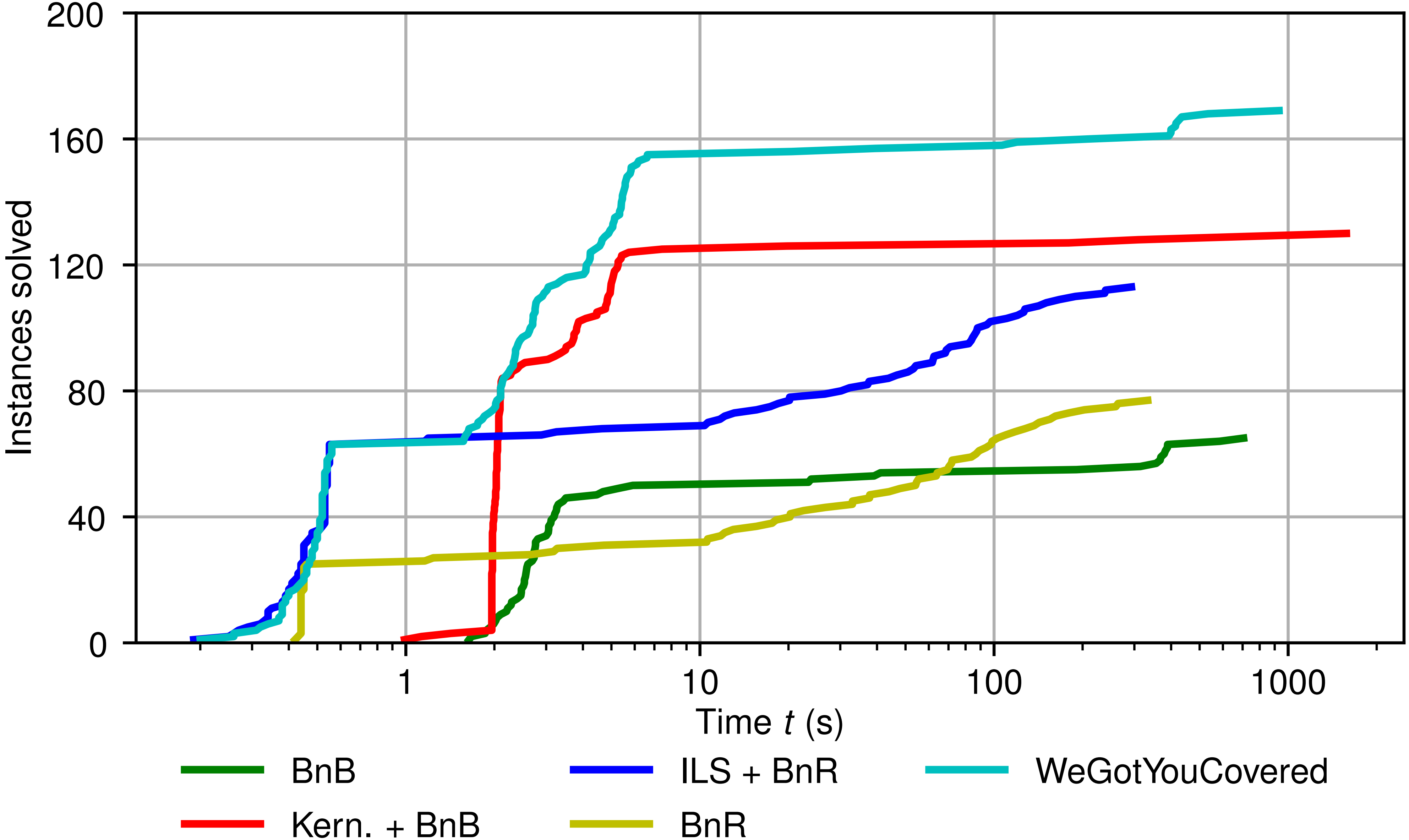


Institute of Theoretical Informatics
Algorithmics Group

Algorithm Overview



Instances Solved Over Time



Institute of Theoretical Informatics
Algorithmics Group

References

- [Gonzalez et. al.'12]
  J. E. Gonzalez, Y. Low, H. Gu, D. Bickson, and C. Guestrin. Power Graph: Distributed Graph-Parallel Computation on Natural Graphs. In *Presented as part of 10th USENIX Symposium on Operating Systems Design and Implementation (OSDI 12)*, pages 17–30. USENIX, 2012.
- [Li et. al.'17]
  L. Li, R. Geda, A. B. Hayes, Y. Chen, P. Chaudhari, E. Z. Zhang, and M. Szegedy. A simple yet efficient balanced edge partition model for parallel computing. *SIGMETRICS Perform. Eval. Rev.*, 45(1):6–6, June 2017.
- [CV'97]
  E. Osuna, R. Freund, and F. Girosi. An improved training algorithm for support vector machines. *In Neural Networks for Signal Processing VII*, pages 276–285, 1997.
- [GCBDV'04]
  H. P. Graf, E. Cosatto, L. Bottou, I. Durdanovic,and V. Vapnik. Parallel support vector machines: The cascade SVM. *In Advances in Neural Information Processing Systems (NIPS)*, pages 521–528, 2004.
- [RS'15]
  T. Razzaghi and I. Safro. Scalable multilevel support vector machines. *In Proc. of the Intl. Conf. on Computational Science (ICCS)*, pages 2683–2687, 2015.
- [SJKLRS'17]
  E. Sadrfaridpour, S. Jeereddy, K. Kennedy, A. Luckow, T. Razzaghi, and I. Safro. Algebraic multigrid support vector machines. *In European Symposium on Artificial Neural Networks (ESANN)*, pages 35–40, 2017.
- [ML'14]
  M. Muja and D. G. Lowe. Scalable nearest neighbor algorithms for high dimensional data. *IEEE Trans. Pattern Anal. Mach. Intell.*, pages 2227–2240, 2014.