White, Gray and Black Box Tuning

Tommy Messelis, Nguyen Dang, Patrick De Causmaecker
KU Leuven
Department of Computer Science
CODEs research group

April 4, 2014
Outline

- Predictors, Algorithm Selection, Portfolio
- Tuning
- Opening the box
Problem:
- Combine two algorithms into one algorithm by selecting the most promising one

Datasets
- PSPLib (Kolisch, R., Sprecher, A., EJOR 1996),

Algorithms
- Algorithm B: Lova et al., Hybrid GA (2009)
Predictors and Algorithm Selection, MRCPSP

- **RCPSP**
  - Jobs, Renewable Resources, Precedence relations, Minimal Makespan
  - NP-hard (Blazewicz et al. (1983)).

- **MRCPSP**
  - Jobs with multiple modes, Resources, Precedence relations, Minimal Makespan
  - Herroelen et al. (1998) m, 1T |cpm, disc, mu|Cmax, Brucker et al. (1999) MPS|prec|Cmax
Predictors and Algorithm Selection, related work


Relative difference Algorithm A / Algorithm B
PSPLib, 5000 evaluations

(b) relative difference
Relative difference Algorithm A / Algorithm B
MMLib, 5000 evaluations
Relative difference Algorithm A / Algorithm B
MMLib, 25000 evaluations

(b) relative difference
Given a new instance of MRCPSP, which is the algorithm will we use?

Approach 1: prediction of the goal function

- Running time (exact, complete)
- **Obtained quality** (heuristic, fixed running time or number of evaluations)
- Prediction based on **features** \(^1\)

---

\(^1\) 686, size, resource constraints, precedence, duration, complexity
Prediction on quality (validation)

<table>
<thead>
<tr>
<th></th>
<th>always-A</th>
<th>always-B</th>
<th>AS1</th>
<th>AS*</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>5000 schedules</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>% correctly classified instances</td>
<td>58.1</td>
<td>49.9</td>
<td>69.0</td>
<td>100</td>
</tr>
<tr>
<td>avg. relative deviation from best (%)</td>
<td>2.30</td>
<td>2.85</td>
<td>1.32</td>
<td>0</td>
</tr>
<tr>
<td>avg. makespan</td>
<td>100.68</td>
<td>99.40</td>
<td>98.51</td>
<td>97.34</td>
</tr>
<tr>
<td><strong>25000 schedules</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>% correctly classified instances</td>
<td>89.6</td>
<td>18.0</td>
<td>80.1</td>
<td>100</td>
</tr>
<tr>
<td>avg. relative deviation from best (%)</td>
<td>0.36</td>
<td>8.01</td>
<td>1.04</td>
<td>0</td>
</tr>
<tr>
<td>avg. makespan</td>
<td>92.25</td>
<td>99.98</td>
<td>92.66</td>
<td>91.90</td>
</tr>
</tbody>
</table>
Approach 2: prediction of the dominance
Class A: algorithm A is best
Class B: algorithm B is best
Classification based on features
Prediction of dominance (validation)

<table>
<thead>
<tr>
<th></th>
<th>AS2a</th>
<th>AS2b</th>
<th>AS2c</th>
<th>AS*</th>
</tr>
</thead>
<tbody>
<tr>
<td>5000 schedules</td>
<td>% correctly classified instances</td>
<td>80.2</td>
<td>77.1</td>
<td>83.2</td>
</tr>
<tr>
<td></td>
<td>avg. relative deviation from best (%)</td>
<td>0.67</td>
<td>0.79</td>
<td>0.51</td>
</tr>
<tr>
<td></td>
<td>avg. makespan</td>
<td>97.93</td>
<td>98.05</td>
<td>97.74</td>
</tr>
<tr>
<td>25000 schedules</td>
<td>% correctly classified instances</td>
<td>92.2</td>
<td>91.4</td>
<td>93.3</td>
</tr>
<tr>
<td></td>
<td>avg. relative deviation from best (%)</td>
<td>0.24</td>
<td>0.27</td>
<td>0.18</td>
</tr>
<tr>
<td></td>
<td>avg. makespan</td>
<td>92.05</td>
<td>92.08</td>
<td>92.02</td>
</tr>
</tbody>
</table>

Weka

- **AS2a REPTree**
- **AS2b DecisionTable**
- **AS2c RandomForest**
Prediction of dominance (reduced feature set)

<table>
<thead>
<tr>
<th></th>
<th>AS2a’</th>
<th>AS2b’</th>
<th>AS2c’</th>
<th>AS2d’</th>
<th>AS*</th>
</tr>
</thead>
<tbody>
<tr>
<td>5000 schedules</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>% correctly classified instances</td>
<td>79.4</td>
<td>77.4</td>
<td>81.7</td>
<td>79.5</td>
<td>100</td>
</tr>
<tr>
<td>avg. relative deviation from best (%)</td>
<td>0.69</td>
<td>0.84</td>
<td>0.57</td>
<td>0.67</td>
<td>0</td>
</tr>
<tr>
<td>avg. makespan</td>
<td>97.92</td>
<td>98.10</td>
<td>97.85</td>
<td>97.87</td>
<td>97.34</td>
</tr>
<tr>
<td>25000 schedules</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>% correctly classified instances</td>
<td>91.7</td>
<td>91.6</td>
<td>93.4</td>
<td>91.2</td>
<td>100</td>
</tr>
<tr>
<td>avg. relative deviation from best (%)</td>
<td>0.26</td>
<td>0.27</td>
<td>0.18</td>
<td>0.30</td>
<td>0</td>
</tr>
<tr>
<td>avg. makespan</td>
<td>92.07</td>
<td>92.06</td>
<td>92.03</td>
<td>92.09</td>
<td>91.90</td>
</tr>
</tbody>
</table>

Weka
- AS2a REPTree
- AS2b DecisionTable
- AS2c RandomForest
Predictors and Algorithm Selection, MRCPSP

When is algorithm selection potentially fruitful?

- $A =$ instances where alg A strictly dominates alg B
- $B =$ instances where alg B strictly dominates alg A

- competitiveness $c = 2\min\{\frac{|A|}{|T|}, \frac{|B|}{|T|}\}$
- equipotency $e = 2\min(\frac{|A|}{|A|+|B|}, \frac{|B|}{|A|+|B|})$.
- reach $r = \frac{|A|+|B|}{|T|}$
- of course $c = e \cdot r$

The fraction where prediction of dominance is correct $\geq 1 - \frac{c}{2}$.
Outline

- Predictors and Algorithm Selection
- Tuning
- Opening the box
Tuning: related work


- i-Race, F-Racce, ParamILS, ...

Problem/Algorithm: Dynamic programming algorithm for single-machine scheduling \(^2\)

two objectives for tuning

- running time
- number of optimal solutions

<table>
<thead>
<tr>
<th></th>
<th>#optimal solutions</th>
<th>#total average runtime (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(c_{\text{default}})</td>
<td>5963</td>
<td>603.2</td>
</tr>
<tr>
<td>(c_{\text{tuned}})</td>
<td>5970</td>
<td>693.1</td>
</tr>
</tbody>
</table>

SIPS (Tanaka and Fujikuma, 2012)

25 parameters

Continuous: 14
Integer: 7
Categorical: 4

Performance measures

The average running time (minimize)
AND
The total number of optimal solutions (maximize)

Single machine scheduling problems

10610 problem instances
(19 sizes: #jobs: 20 - 200)
Use adaptive local search to determine the pareto surface $^3$

$$ f: w = w \times f_1 + (1-w) \times f_2 $$

Problem: each evaluation is a tuning exercise

$^3$e.g. AA-TPLS, Dubois-Lacoste et al. 2011
Problem: each evaluation is a tuning exercise
Proposal:

Parallellization

Step 1
Scalarization$^1_1$  Scalarization$^1_2$  ...  Scalarization$^1_n$

Step k
Scalarization$^k_1$  Scalarization$^k_2$  ...  Scalarization$^k_n$

Results to come!
Predictors and Algorithm Selection
Tuning
Opening the box
  - Further work
Effort

- Datasets
- Running time
- Analysis
objectives reconsidered

- Running time and number of optimal solutions are only known at the end of the run
  - Black box
Opening the box: existing approaches, related work

- Hyperheuristics
  - construction
  - selection
- Algorithm construction
- Programming by optimization
  - http://www.prog-by-opt.net/

---


Opening the box: existing approaches, related work

In particular:

- Visualization of the search trajectory.
Opening the box: potential

- Running time problem
- Observables
- How to obtain observables
  - standard (from metaheuristics)
  - standard (improvement, LLH runtimes, ...)
  - user defined
    - detailed algorithmic
    - implementation bound (data structures, low level algorithms)
Come and see