

ParamILS: Iterated Local Search in Parameter Configuration Space

Jiří Kubalík
Department of Cybernetics, CTU Prague

Substantial part of this material is based on the paper
Frank Hutter, Holger H. Hoos, Kevin Leyton-Brown, and Thomas Stützle: ParamILS: An Automatic Algorithm Configuration Framework,
Journal of Artificial Intelligence Research (JAIR),
volume 36, pp. 267-306, October 2009.
See <http://www.cs.ubc.ca/~hutter/papers/Hutter09PhD.pdf>



<http://cw.felk.cvut.cz/doku.php/courses/a0m33eoa/start>

FocusedILS

The question is how to choose the optimal number of training instances, N ?

- Using too small N leads to good training performance, but poor generalization to previously unseen test benchmarks.
- On the other hand, we cannot evaluate every parameter configuration on an enormous training set - if we did, search progress would be unreasonably slow.

FocusedILS is a variant of ParamILS that **adaptively varies the number of training samples** considered from one parameter configuration to another in order **to focus samples on promising configurations**.

- $N(\theta)$ denotes the number of runs available to estimate the cost statistic $c(\theta)$.

FocusedILS

The question is how to choose the optimal number of training instances, N ?

- Using too small N leads to good training performance, but poor generalization to previously unseen test benchmarks.
- On the other hand, we cannot evaluate every parameter configuration on an enormous training set - if we did, search progress would be unreasonably slow.

FocusedILS is a variant of ParamILS that **adaptively varies the number of training samples** considered from one parameter configuration to another in order **to focus samples on promising configurations**.

- $N(\theta)$ denotes the number of runs available to estimate the cost statistic $c(\theta)$.

The question is how to compare two parameter configurations θ_1 and θ_2 for which $N(\theta_1) \leq N(\theta_2)$?

- *What if we computed the empirical statistics based on the available number of runs for each configuration?*

Can lead to systematic bias if, for example, the first instances are easier than the average ones.

FocusedILS: Procedure $better_{Foc}(\theta_1, \theta_2)$

Domination: Configuration θ_1 dominates θ_2 when at least as many runs have been conducted on θ_1 as on θ_2 , and the performance of $\mathcal{A}(\theta)$ on the first $N(\theta_2)$ runs is at least as good as that of $\mathcal{A}(\theta_2)$ on all of its runs.

θ_1 dominates θ_2 if and only if $N(\theta_1) \geq N(\theta_2)$ and $\hat{c}_{N(\theta_2)}(\theta_1) \leq \hat{c}_{N(\theta_2)}(\theta_2)$.

FocusedILS – procedure $better_{Foc}(\theta_1, \theta_2)$ implements a comparison strategy based on the domination

1. first it acquires one additional run for the configuration i having smaller $N(\theta_i)$, or one run for both configurations if $N(\theta_1) = N(\theta_2)$;
2. then, it continues performing runs in this way until one configuration dominates the other.
It returns true if θ_1 dominates θ_2 , and false otherwise.

It keeps track of the total number, B , of configurations evaluated since the last improving step.

- Whenever $better_{Foc}(\theta_1, \theta_2)$ returns true, B extra (bonus) runs are performed for θ_1 and B is reset to 0.
- This way it is ensured that many runs are performed with good configurations \implies the error made in every comparison of two configurations θ_1 and θ_2 decreases on expectation.

Recommended Material

Frank Hutter, Holger H. Hoos, Kevin Leyton-Brown, and Thomas Stützle: ParamILS: An Automatic Algorithm Configuration Framework. In *Journal of Artificial Intelligence Research (JAIR)*, volume 36, pp. 267-306, October 2009.

Other papers and SW available at <http://www.cs.ubc.ca/labs/beta/Projects/ParamILS/>

