Automatically Improving the Anytime Behaviour of Multiobjective Evolutionary Algorithms

Andreea Radulescu[1] Manuel López-Ibáñez[2]
Thomas Stützle[2]

[1]LINA, UMR CNRS 6241, Université de Nantes, Nantes, France
andreea.radulescu@etu.univ-nantes.fr

[2]IRIDIA, CoDE, Université Libre de Bruxelles (ULB), Brussels, Belgium
manuel.lopez-ibanez@ulb.ac.be,stuetzle@ulb.ac.be

March 22nd, 2013

Summary

1. Introduction
2. Anytime optimization
3. Experimental setup
4. Results
5. Conclusion

Introduction
Evolutionary Algorithms

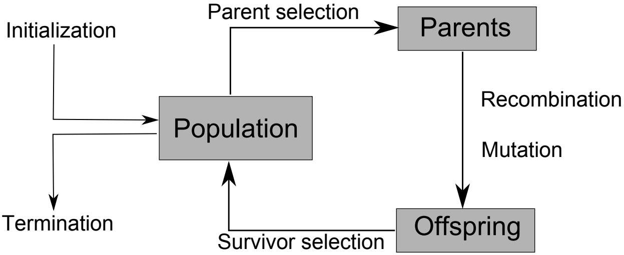


Figure: General framework of Evolutionary Algorithms

Introduction

Manual vs Automatic Tuning

- Finding an appropriate parameter configuration for a specific algorithm is a difficult optimization problem;
- Many evolutionary algorithms are still manually tuned;
- Parameter values are established by conventions, ad hoc choices and experimental comparisons on a limited scale;
- Automatic tuning methods are available that configure evolutionary algorithms without much human effort.

Introduction
Automatic Tuning
Final quality

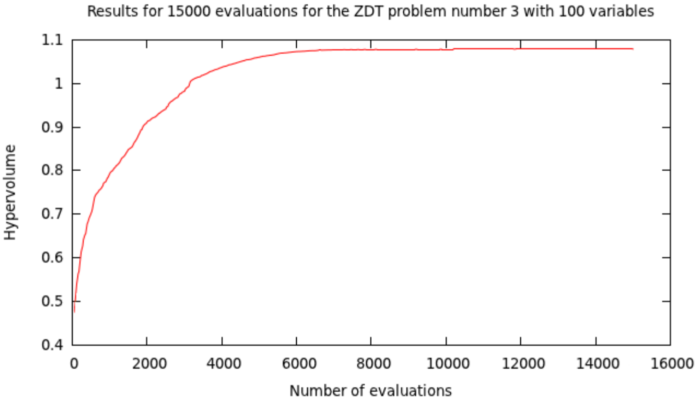


Figure: Quality of Pareto-front approximations through a run

Introduction
Automatic Tuning
Anytime behaviour

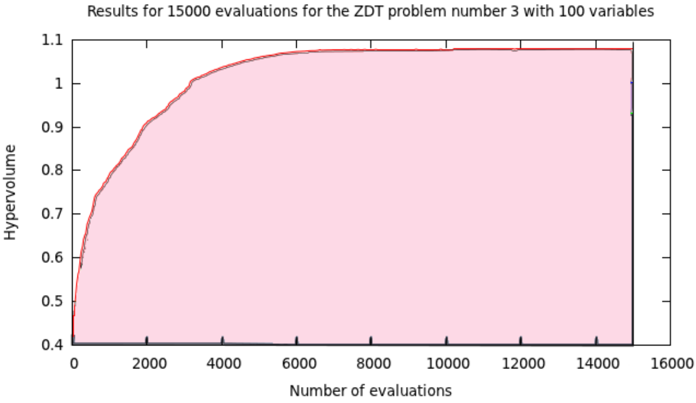


Figure: Anytime behaviour quality

Introduction
Purpose of the research

- Parameter values greatly determine the success of an EA in reaching optimal solutions;
- Improve the anytime behaviour of multi-objective evolutionary algorithms;
- Find automatically the algorithm configurations that produce the best anytime behaviour.

Summary

1. Introduction
2. Anytime optimization
3. Experimental setup
   - Tested MOEAs
   - Benchmark problems
   - Monotonicity of hypervolume in MOEA
   - The irace package
4. Results
   - Tuning for anytime behaviour
   - Tuning for final quality
5. Conclusion

Anytime optimization

- An anytime algorithm returns as high-quality solutions as possible at any moment of its execution;
- The evaluation of the anytime behaviour of an algorithm can be seen as a bi-objective problem;
- The non-dominated front so obtained reflects the quality of the anytime behaviour of the algorithm.

Anytime optimization

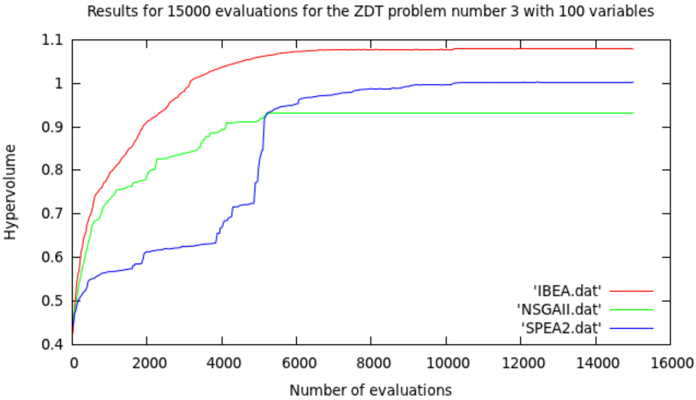


Figure: Anytime behaviour quality

Anytime optimization

Automatically Improving the Anytime Behaviour of MOEAs

A. Radulescu, M. López-Ibáñez, T. Stützle

Introduction

Anytime optimization

Experimental setup

Results

Conclusion

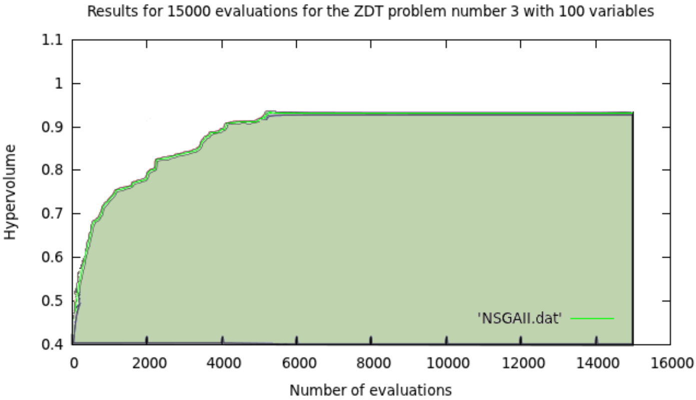


Figure: Anytime behaviour quality

Anytime optimization

Automatically Improving the Anytime Behaviour of MOEAs

A. Radulescu, M. López-Ibáñez, T. Stützle

Introduction

Anytime optimization

Experimental setup

Results

Conclusion

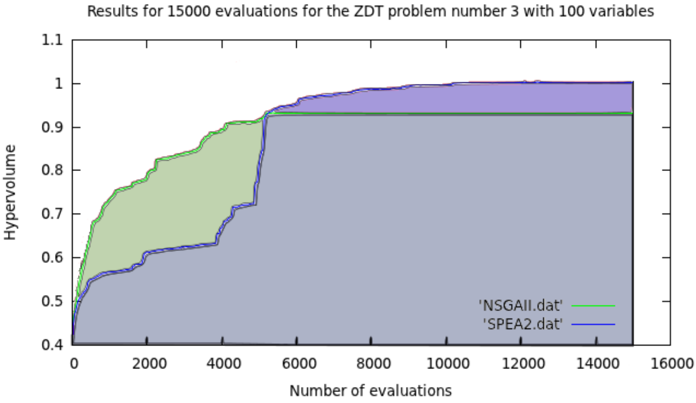


Figure: Anytime behaviour quality

Anytime optimization

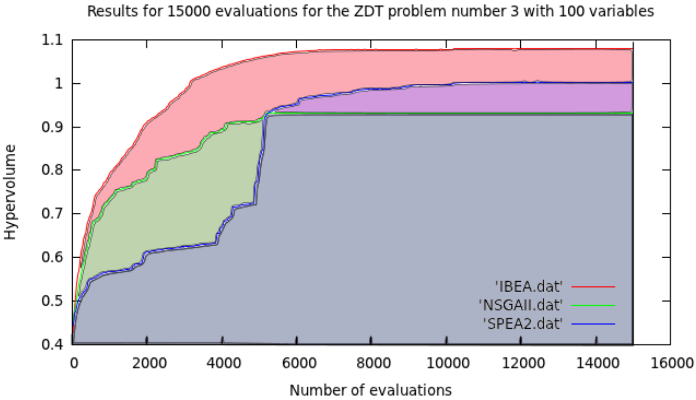


Figure: Anytime behaviour quality

Summary

1. Introduction
2. Anytime optimization
3. Experimental setup
   - Tested MOEAs
   - Benchmark problems
   - Monotonicity of hypervolume in MOEA
   - The irace package
4. Results
   - Tuning for anytime behaviour
   - Tuning for final quality
5. Conclusion

Tested MOEAs

- **IBEA** (Indicator-Based Evolutionary Algorithm)
  - it can be adapted to arbitrary preference information;
  - does not require additional techniques in order to preserve the diversity of the population.
- **NSGAII** (Nondominated Sorting Genetic Algorithm II)
  - Uses a fast nondominated sorting procedure;
  - Implements an elitist-preserving approach.
- **SPEA2** (Strength Pareto Evolutionary Algorithm 2)
  - a fixed-sized archive method;
  - a nearest neighbor density estimation technique.

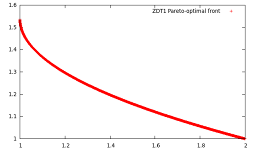
Tested MOEAs
Parameters

Name	Role	Type
crossoverProbability	probability of mating two solutions	real
externalMutationProbability	probability of mutating a solution	real
internalMutationProbability	probability of mutating a variable in a solution	real
populationSize	number of solutions	integer
crossoverDistrIndex	distance between the children and their parents	integer
mutationDistrIndex	distribution of the mutated values	integer
scalingFactor (IBEA)	fitness scaling factor	real
archiveSize (SPEA2)	size of the archive	integer
k(SPEA2)	k-th nearest neighbour	integer

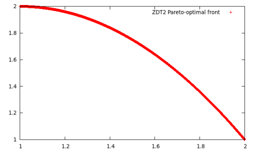
Table: Parameters of IBEA, NSGAII and SPEA2

Benchmark problems
ZDT

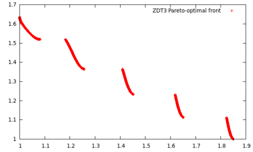
- Scalable to any number of decision variables;
- Controlled difficulty in converging to the Pareto-optimal front;
- Pareto-optimal front easy to construct.



ZDT1 convex



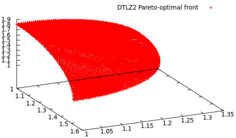
ZDT2 nonconvex



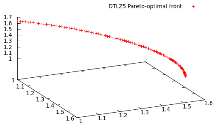
ZDT3 noncontiguous

Benchmark problems
DTLZ

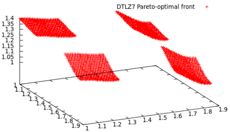
+ scalable to any number of objectives.



DTLZ2

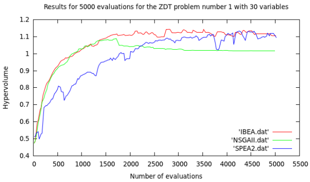


DTLZ5

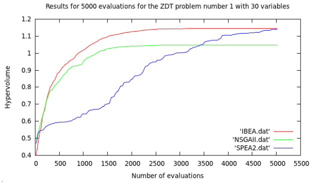


DTLZ7 disconnected set

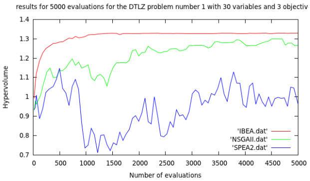
Monotonicity of hypervolume in MOEA



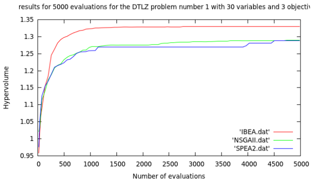
ZDT1 without an external archive



ZDT1 with an external unbounded archive



DTLZ1 without an external archive



DTLZ1 with an external unbounded archive

The irace package

The iterated racing algorithm

- **sampling**: parameter configurations are sampled from a probabilistic distribution (uniformly random at the start);
- **selecting**: parameter configurations are run on a few training problem instances, and statistically worse configurations are discarded until a few remain or computational budget exhausted;
- **updating**: the sampling distribution is modified according to the selected configurations to bias sampling towards best configurations found.

Summary

1 Introduction

2 Anytime optimization

3 Experimental setup
- Tested MOEAs
- Benchmark problems
- Monotonicity of hypervolume in MOEA
- The irace package

4 Results
- Tuning for anytime behaviour
- Tuning for final quality

5 Conclusion

Tuning for anytime behaviour
Anytime behaviour values

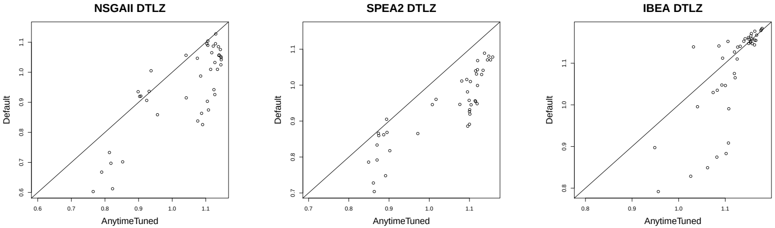


Figure: Anytime behaviour quality for the default configuration versus configurations tuned for anytime behaviour

Tuning for anytime behaviour
Quality variation

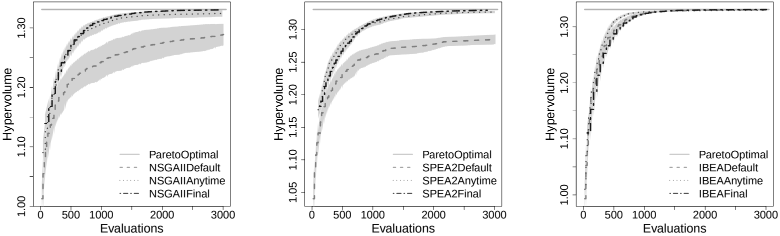


Figure: Variation of the quality of the Pareto front obtained for the three different configurations

Tuning for anytime behaviour
Final values

Automatically Improving the Anytime Behaviour of MOEAs
A. Radulescu, M. López-Ibáñez, T. Stützle
Introduction
Anytime optimization
Experimental setup
Results
Tuning for anytime behaviour
Tuning for final quality
Conclusion

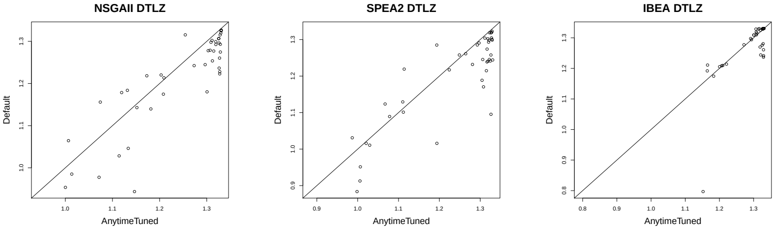


Figure: Final Pareto front quality for the default configuration versus configurations tuned for anytime behaviour

Tuning for final quality

Anytime behaviour values

Automatically Improving the Anytime Behaviour of MOEAs

A. Radulescu, M. López-Ibáñez, T. Stützle

Introduction

Anytime optimization

Experimental setup

Results

Tuning for anytime behaviour

Tuning for final quality

Conclusion

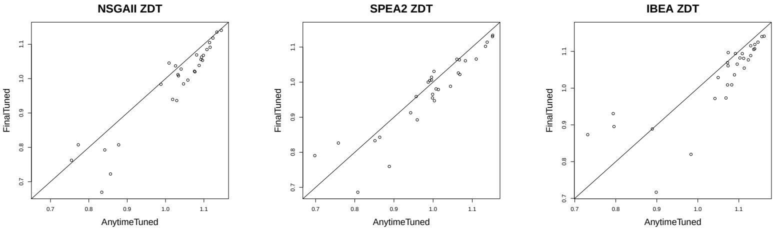


Figure: Difference for **anytime behaviour quality**

Tuning for final quality

Final values

Automatically Improving the Anytime Behaviour of MOEAs

A. Radulescu, M. López-Ibáñez, T. Stützle

Introduction

Anytime optimization

Experimental setup

Results

Tuning for anytime behaviour

Tuning for final quality

Conclusion

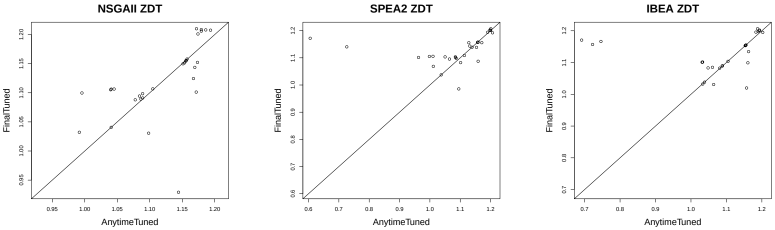
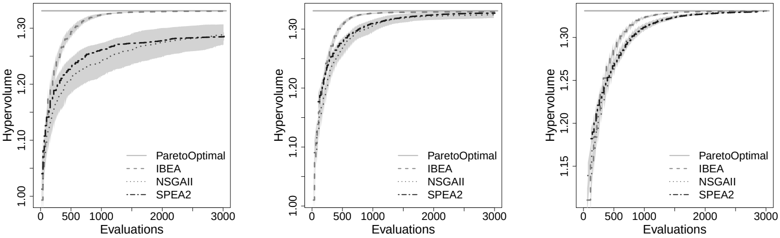


Figure: Difference for the quality of the **final Pareto front**

Tuning for final quality
Quality variation



Default conf. Anytime conf. Final conf.

Figure: Variation of the quality of the Pareto front approximation obtained by the three MOEAs

Conclusion

1. The quality of the anytime behaviour is significantly improved;
2. The tuned configurations improve the search behaviour of MOEAs;
3. The MOEAs are more robust to specific termination criteria;
4. Substantial human effort is saved.

Future work

$\longrightarrow$ The impact of the anytime behaviour tuning can be extended:

- Different categorical parameters;
- Analysis of others MOEAs;
- Tests with different benchmark problems.

Automatically Improving the Anytime Behaviour of Multiobjective Evolutionary Algorithms

Andreea Radulescu[1] Manuel López-Ibáñez[2]
Thomas Stützle[2]

[1]LINA, UMR CNRS 6241, Université de Nantes, Nantes, France
andreea.radulescu@etu.univ-nantes.fr

[2]IRIDIA, CoDE, Université Libre de Bruxelles (ULB), Brussels, Belgium
manuel.lopez-ibanez@ulb.ac.be,stuetzle@ulb.ac.be

March 22nd, 2013