

**VERTEILTE SYSTEME**

Thomas Weise, Kurt Geihs  
{weise|geihs}@vs.uni-kassel.de  
University of Kassel  
Wilhelmshöher Allee 73  
D-34121 Kassel  
Germany

# DGPF

<http://dgpf.sourceforge.net/>

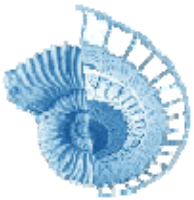
**An Adaptable Framework for Distributed Multi-Objective Search Algorithms Applied to the Genetic Programming of Sensor Networks**

BIOMA 2006

The 2nd International Conference on Bioinspired Optimization Methods and their Applications

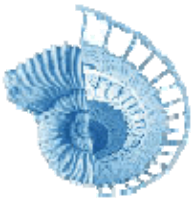
Tuesday, 2006-10-10, Session E, 15:20

Jožef Stefan International Postgraduate School, Jamova 39, 1000 Ljubljana, Slovenia



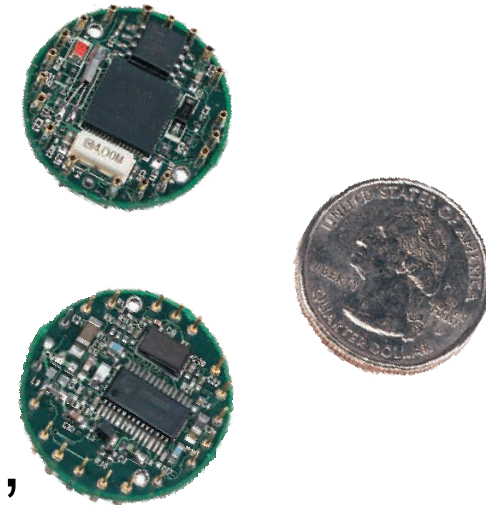
# Contents

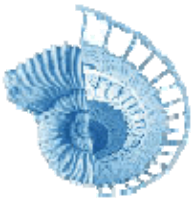
- Sensor Networks
- Representation for Algorithms
- Problems
- Search API
- Heterogeneous Search
- Experiment
- Summary, Future Work, References



# Sensor Networks

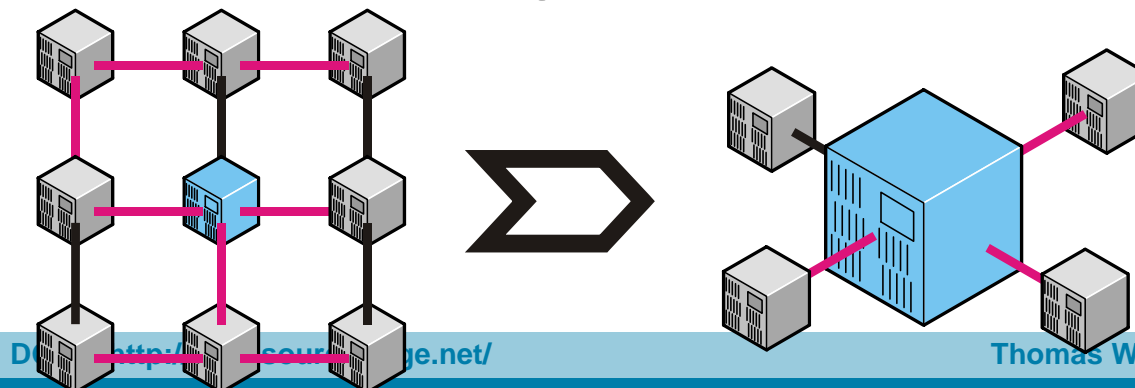
- Sensor nodes are small devices equipped with sensors and short range radio.
- Restricted in memory size, processing speed, and, most important, battery power
- Usually forming large scale distributed systems.
- Typical Applications:
  - Ecology (pollution, seismic activity, ethology)
  - Disaster Prevention (detection of forest fires, prediction of earthquakes, surveillance of contaminated areas)
  - Health Care, Military

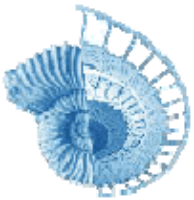




# Sensor Networks – Software Development

- Manual development: cumbersome, many things need to be regarded (routing, energy, ...)
- Using MDA: Model the application and let the MDA-system create the code.
  - Abstract model is simpler to understand
  - MDA tool generates code for modeled algorithm
- Extend this approach with GP: model the results wanted, instead of the algorithms





# Representation for Algorithms

- Algorithms must be simulated for fitness determination

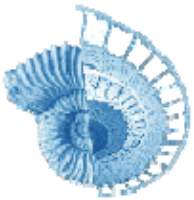
- Basic Instructions

- assembler-like
- Turing complete
- arithmetical operations
- conditional local jumps
- procedure calls
- interrupts for sensor/IO-data

## procedure\_0

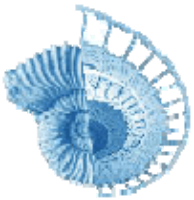
```
0: mem[1] = mem[1] % mem[0]
1: mem[0] = mem[0] % mem[1]
2: if zf then goto 6
3: if zf then call procedure_0
4: zf = (mem[1] >= mem[-1])
5: mem[0] = mem[1]
6: mem[-1] = mem[-1] + mem[1]
7: if zf then goto 3
8: mem[0] = mem[-1]
```

- Simulated on a virtual machine



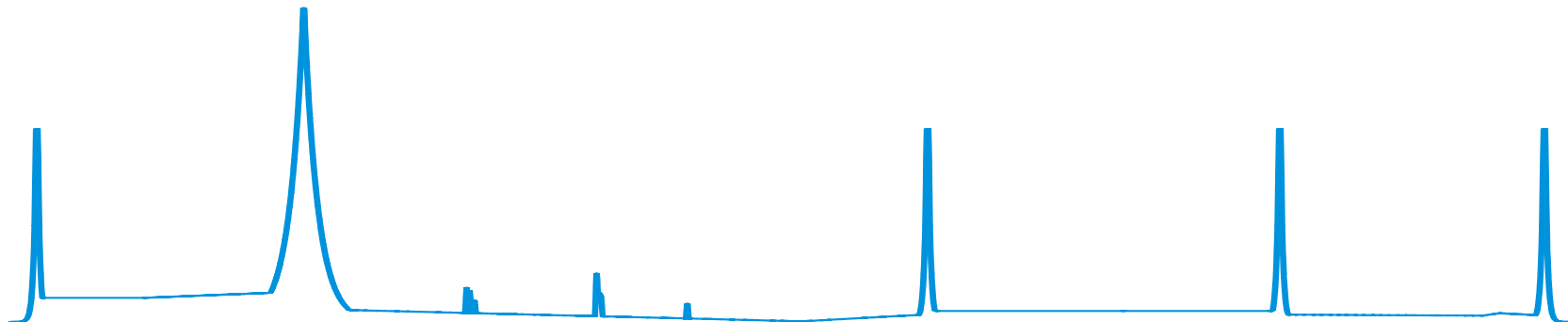
# Representation for Algorithms

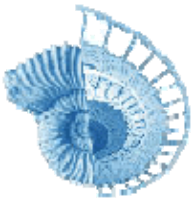
- Many virtual machines running asynchronously in a simulated network
  - messages are undirected radio broadcasts
  - will be received by any node in range
  - interrupt if message arrives
  - one node may only process one message at a time, other simultaneously arriving messages get lost
- Network extension to instruction set:
  - write to output buffer
  - send output buffer as a message



# Problems

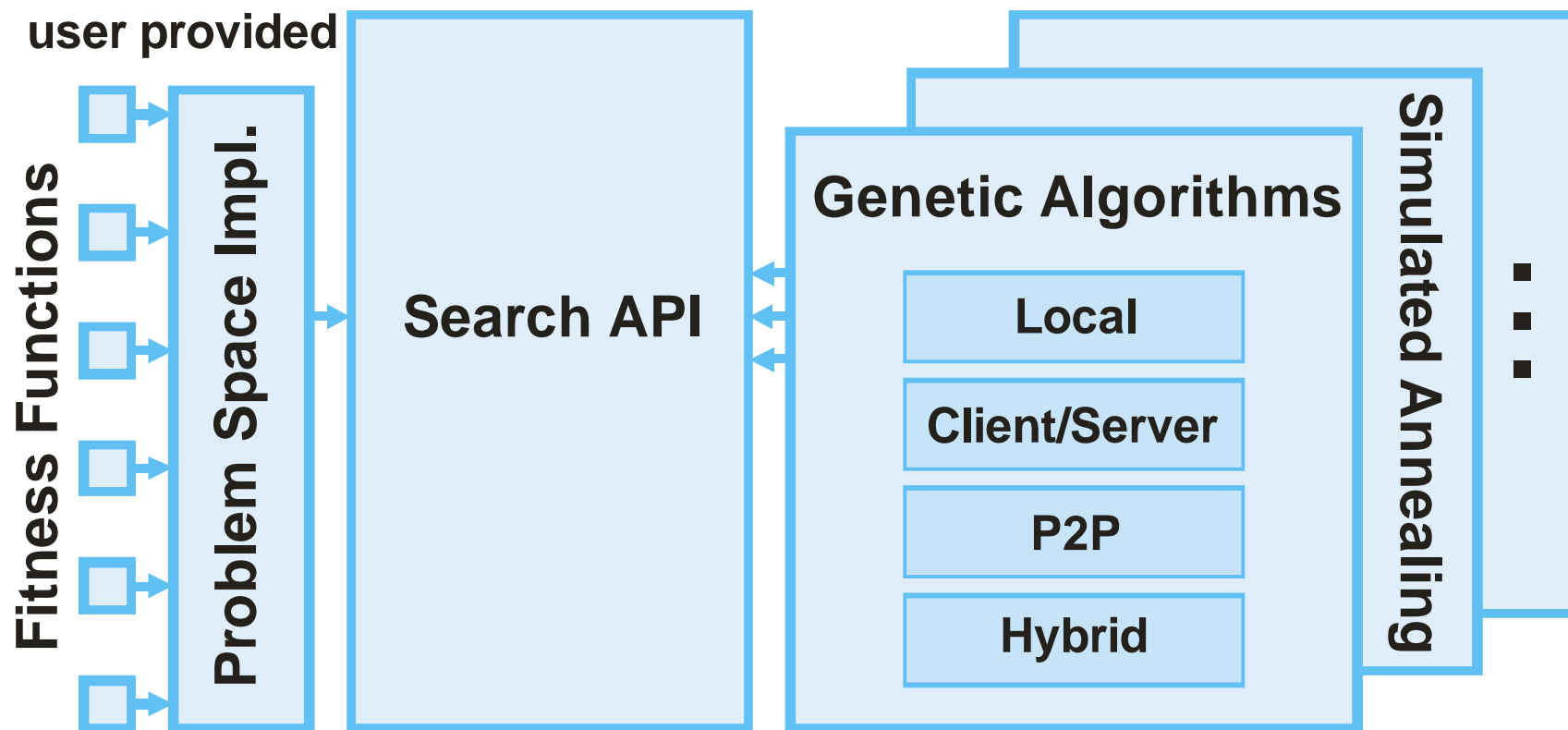
- Fitness landscapes of algorithms are rugged
- Check an algorithm's convergence speed – Fitness Functions need hooks "into the simulation"
- Need for multi-objective optimization:  
    functionality, code size, transmission count...
- Pareto-Optimal Set contains many useless programs  
(again producing useless offspring)



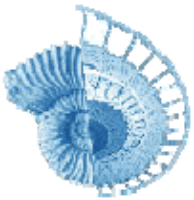


# Search API

- Versatile search API with building blocks for search- and optimization algorithms

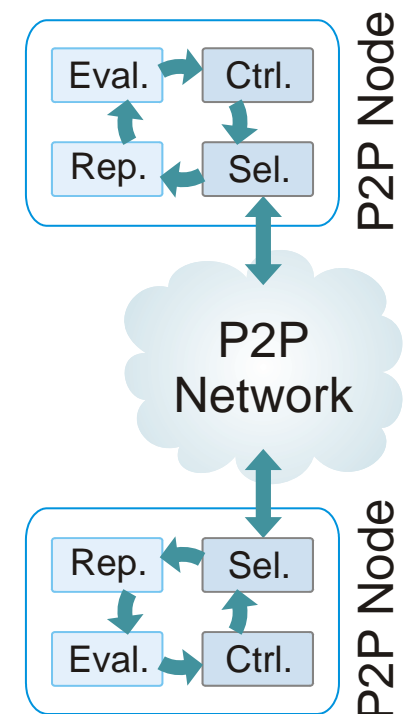
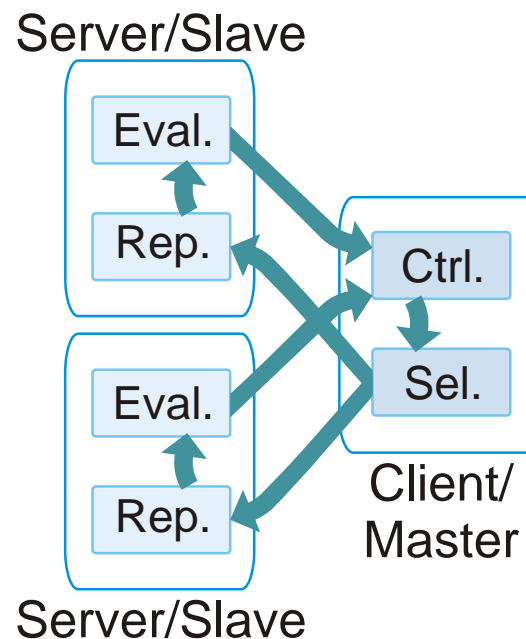
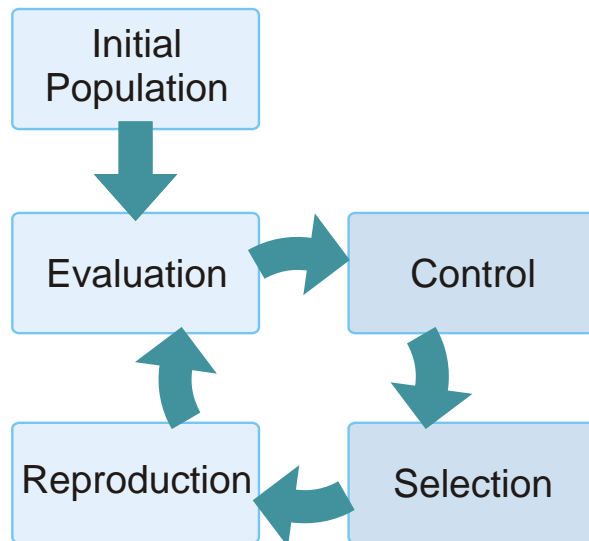


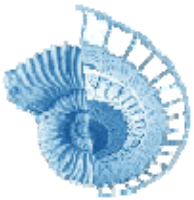




# Search API

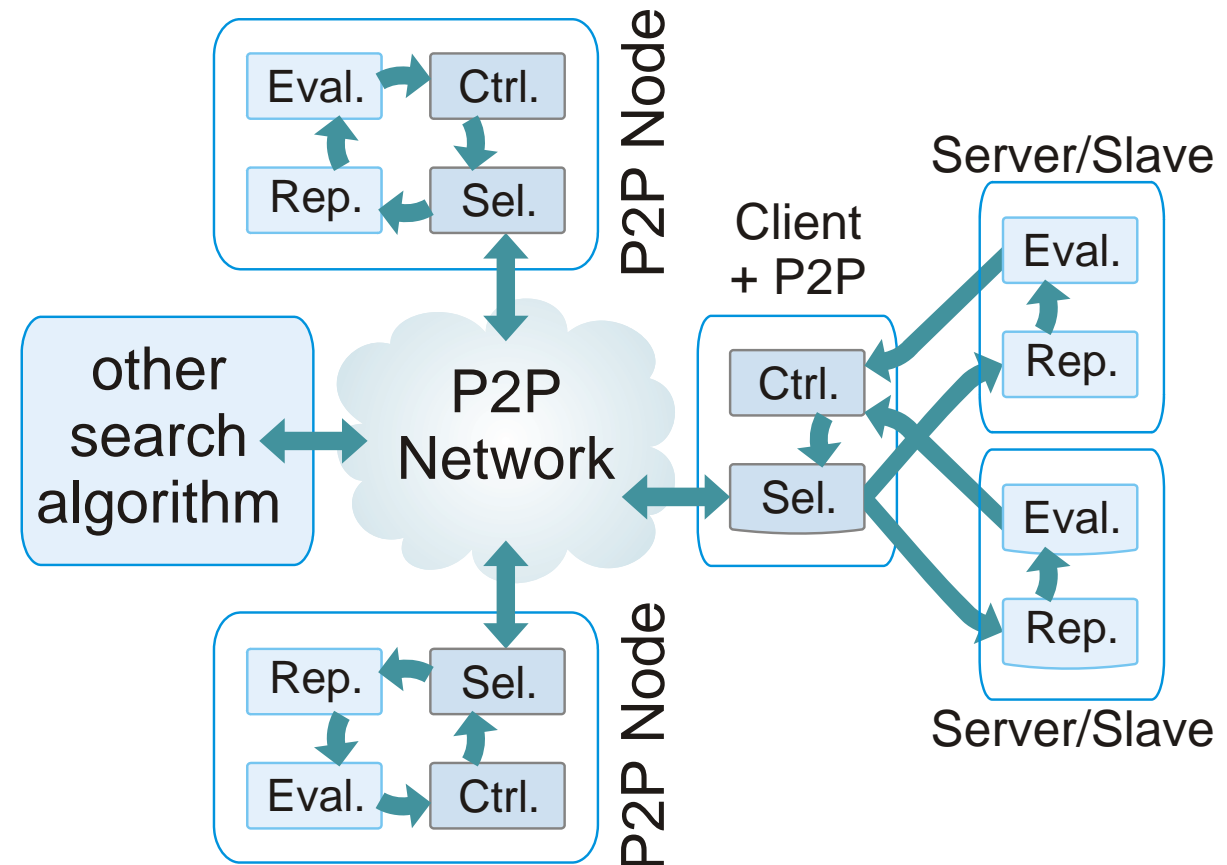
- Semantic mutation and crossover
- Prevalence: comparison of Individuals no longer based on strict domination but is user-defined
- Distribution utilities: C/S, P2P, C/S-P2P

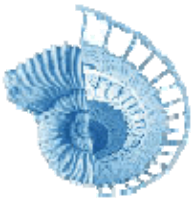




# Heterogeneous Search

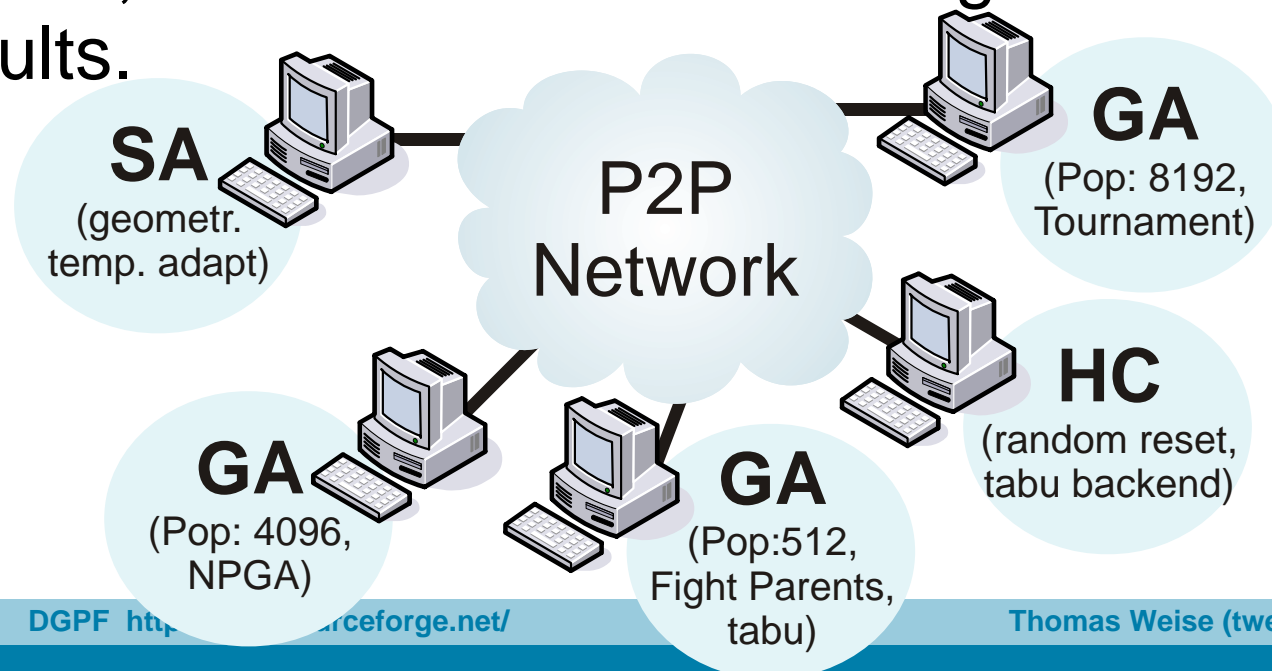
- P2P-Component: used for Island-Hopping in GA but can connect arbitrary search/optimization algorithms
- Genetic Algorithms, Simulated Annealing, and Stochastic Hill Climbing
- Algorithms are auto-adaptive and come with a implicit Tabu-Search backend

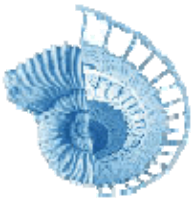




# Heterogeneous Search

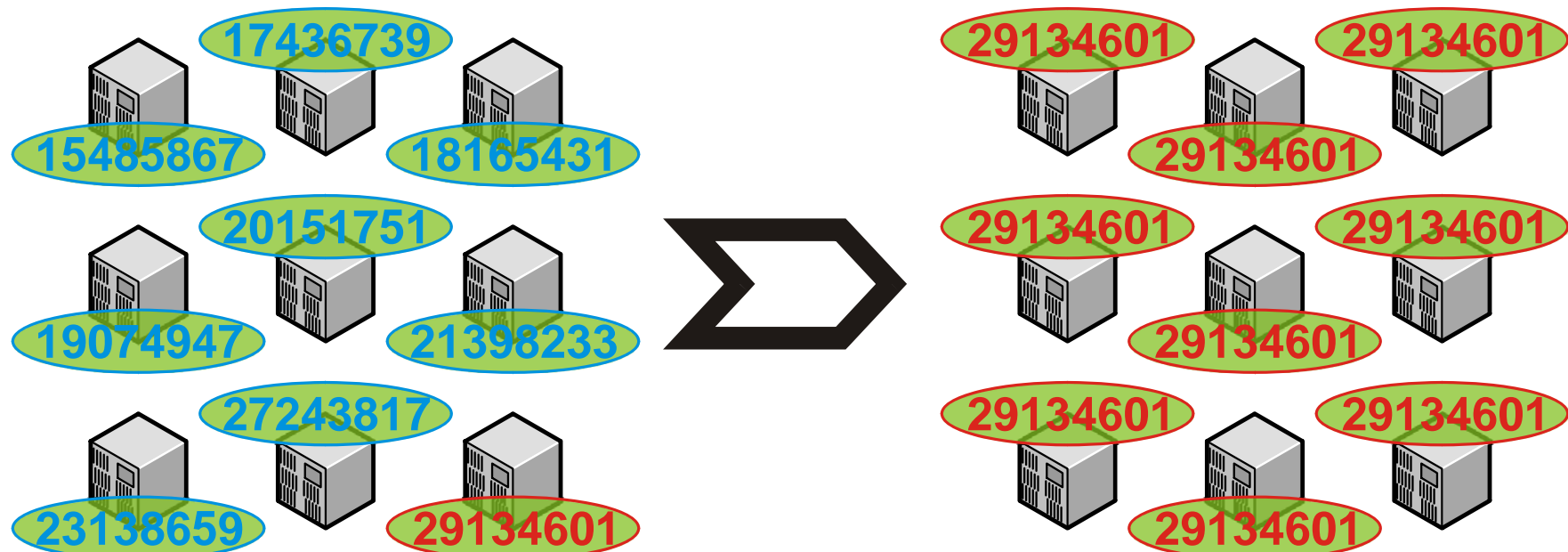
- It is not known beforehand which search/optimization algorithm may perform best for a problem.
- Algorithms may perform differently regarding the stage of the problem space exploration.
- In our case, a mixture of different algorithms yields the best results.

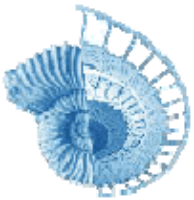




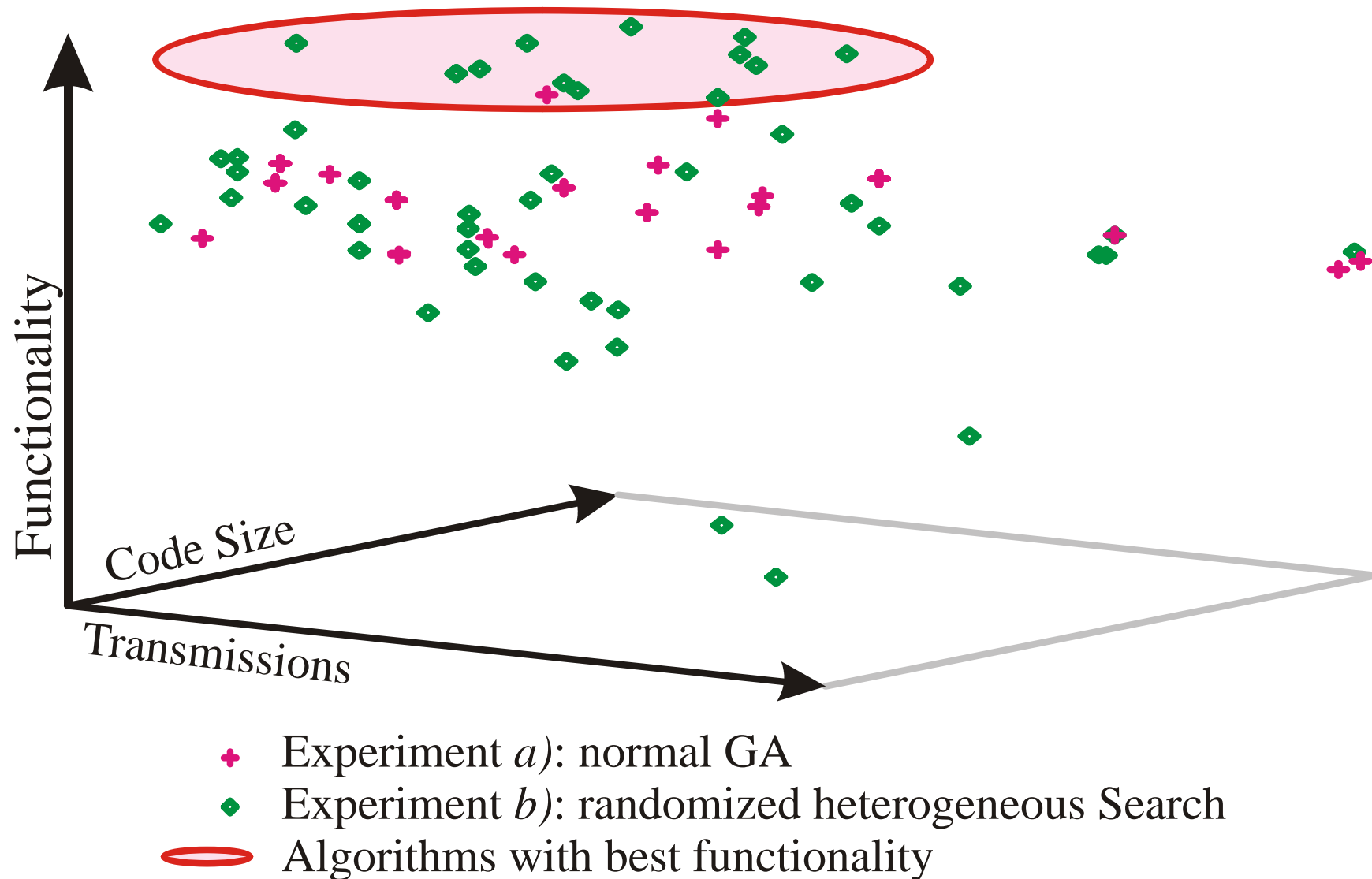
# Experiment

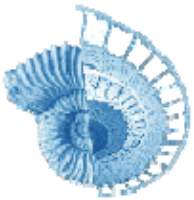
- Evolve an "Election Algorithm"
  - all nodes in the virtual network have an unique id
  - the node with the highest id wins
  - all nodes should receive knowledge of this id
  - multi-objective optimization





# Experiment





# Experiment

*called on startup*

*store 1st variable into  
output buffer*

*send output buffer  
go back to start*

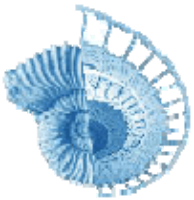
## procedure\_0

```
0: push mem[0]
1: some useless operations used
2: to stall and, as a consequence,
3: reduce transmissions in the
4: simulated/evaluated time span
5: send
6: goto 0
```

*called asynchronously when  
a message comes in  
compare the known and  
the received value  
if no improvement then exit  
exchange values*

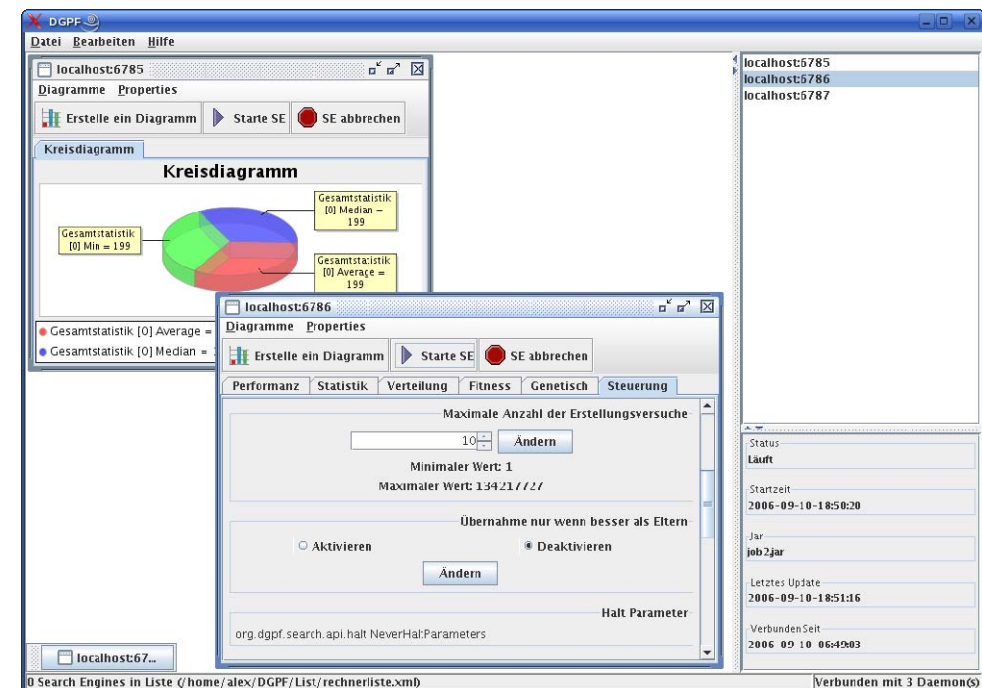
## procedure\_1

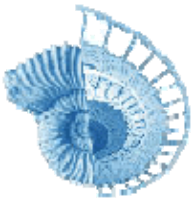
```
0: zf = (mem[-1] < mem[0])
1: if zf then goto 3 // =exit
2: xchg mem[-1], mem[0]
```



# Summary

- Versatile distributed Framework for auto-adaptive, multi-objective search- and optimization algorithms
- Genetic Programming can help to find simple distributed algorithms applicable for sensor networks.
- Heterogeneous searches can provide better performance
- Cluster Management Tool for easy search deployment

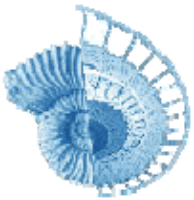




# Future Work

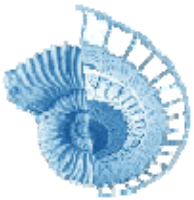
- **Tackle more complex distributed algorithms**
- Transform evolved algorithms into source code
- Extend search API
- Implement extended versions of algorithms like SPEA-II
- Test also PSO and deterministic algorithms like A\*
- Extend auto-adaptation capabilities
- Extend Cluster Management Tool





# References

- Khaled El-Fakihi, Hirozumi Yamaguchiz, Gregor v. Bochmann: A Method and a Genetic Algorithm for Deriving Protocols for Distributed Applications with Minimum Communication Cost, Proceedings of 11th IASTED, 1999
- Lidia Yamamoto, Christian Tschudin, Genetic Evolution of Protocol Implementations and Configurations, IFIP/IEEE International workshop on Self-Managed Systems and Services
- F. Comellas, G. Gimenez: Genetic Programming to Design Communication Algorithms for Parallel Architectures, in Parallel Processing Letters, 1998
- John R. Koza: Non-Linear Genetic Algorithms for Solving Problems. United States Patent 4,935,877. Filed May 20, 1988. Issued June 19, 1990.
- Chee-Yee Chong; Kumar, S.P. Proc, Sensor networks: Evolution, opportunities, and challenges, IEEE, August 2003
- Una-May O'Reilly, Franz Oppacher: Program Search with a Hierarchical Variable Length Representation: Genetic Programming, Simulated Annealing and Hill Climbing, Parallel Problem Solving from Nature III, 1994
- D. Levine: Application of a hybrid genetic algorithm to airline crew scheduling, Computers & Operations Research, 23(6):547-558, 1996
- X. Yao: Optimization by genetic annealing, InProc. of 2nd Austr. Conf. on Neural Networks, 1991



Thanks for your attention.

**Questions?**

Framework available as Open-Source at  
<http://dgpf.sourceforge.net/>  
[weise@vs.uni-kassel.de](mailto:weise@vs.uni-kassel.de)