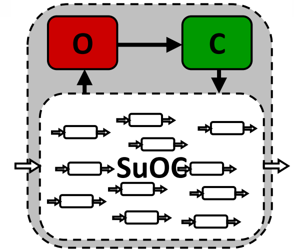
Observation and Control of Collaborative Systems (OCCS)

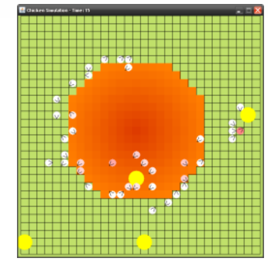
12th colloquium of the DFG SPP Organic Computing
Nuremberg | September 15/16, 2011

J. Branke, E. Cakar, N. Fredivianus, J. Hähner, C. Müller-Schloer, H. Schmeck

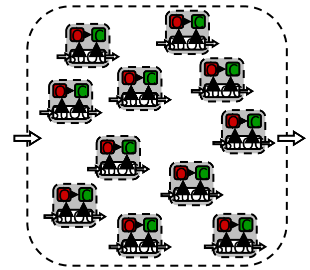
Project overview

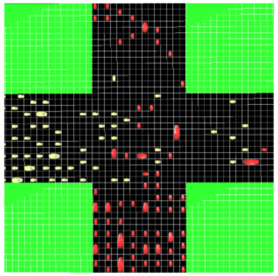
- Phase I
  - Goal: Establishing controlled self-organisation in technical systems
  - Specification of the generic centralised O/C architecture



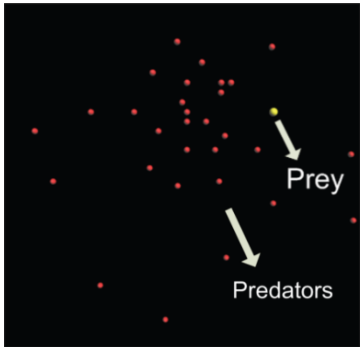


- Phase II
  - Systematic investigation of different distribution possibilities of the O/C architecture
  - Parallel and hierarchical on-line learning with eXtended Classifier Systems (XCSs)



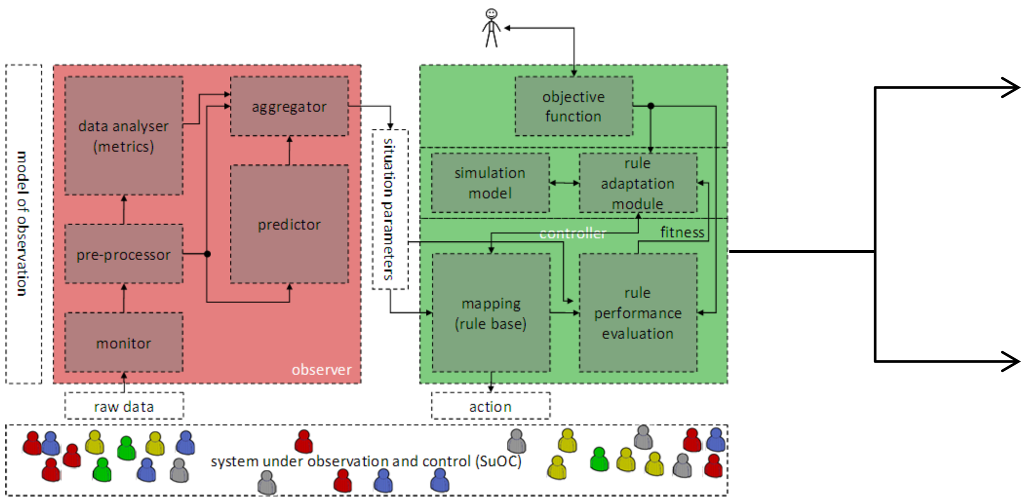


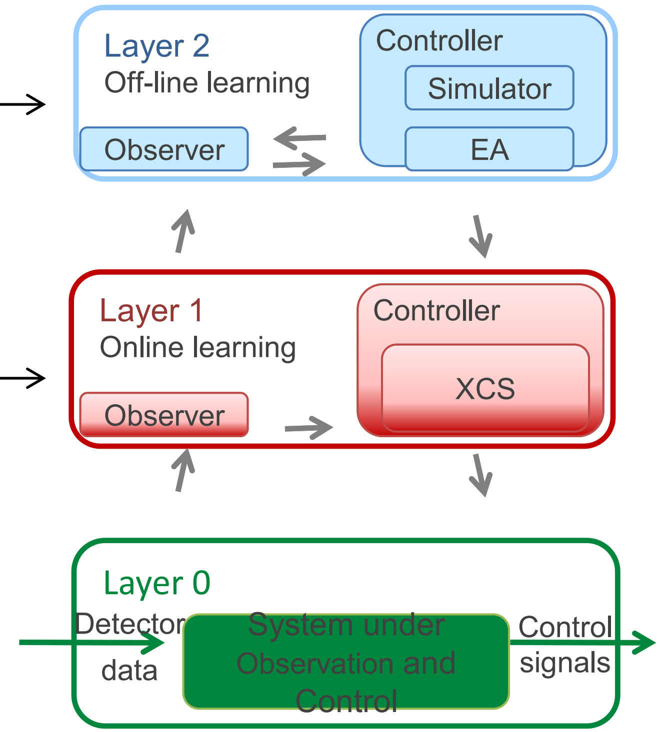
- Phase III
  - Investigation of extended learning mechanisms and experimental evaluation
  - Extension of OCCS methodology to other OC applications



Motivation

- Establishing controlled self-adaptation to create robust and flexible OC systems using the generic O/C architecture





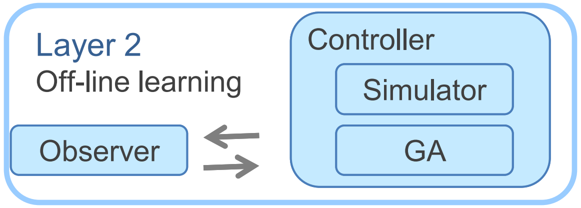
- A highly effective system architecture using two layers with offline and online learning (OTC)
- Objectives:

  Layer 2: Investigation of different optimisation algorithms other than GA

  Layer 1: Investigation of different learning architectures for XCS to speed up the online learning process.

Layer 2 – The optimisation layer

- Offline learning with the population based optimisation algorithm GA

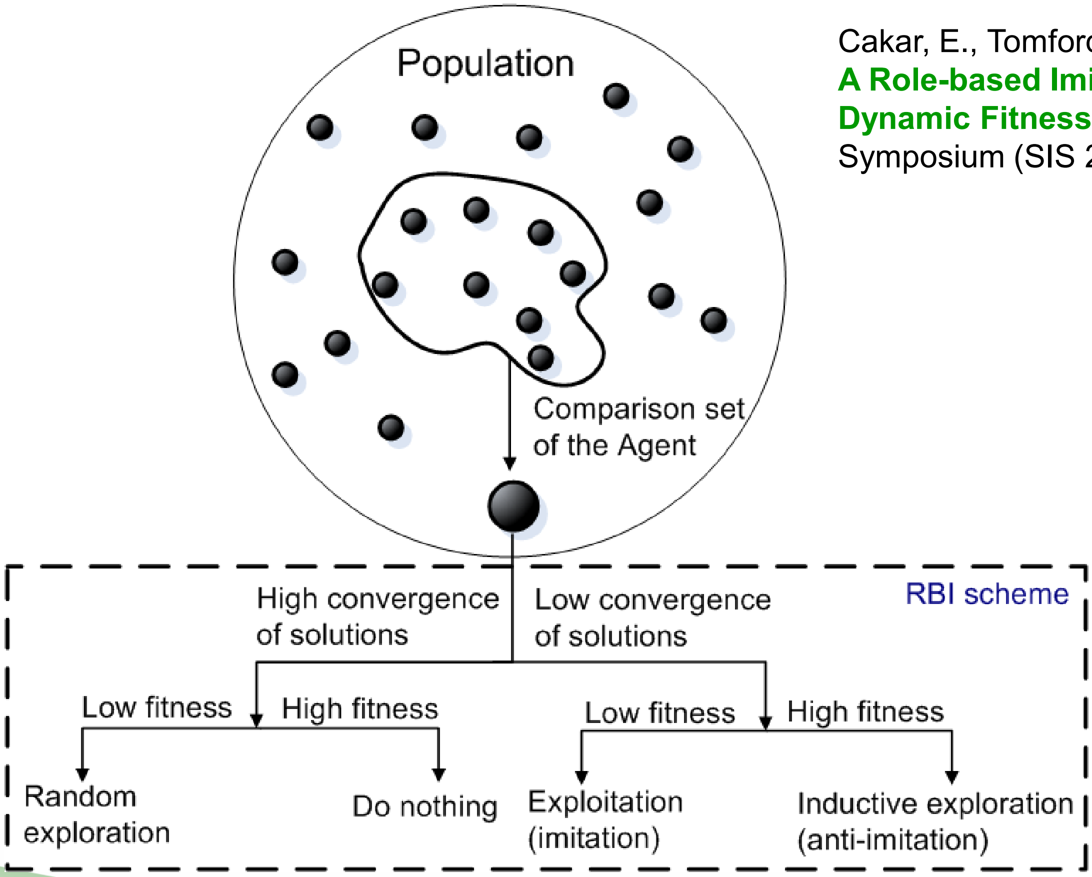


- Question: Is GA the best possible choice?
- There exist many (population-based or trajectory-based) optimisation algorithms that can be used on layer 2:
  - Differential evolution (DE), Particle Swarm Optimisation (PSO), Simulated Annealing (SA) …

- Contribution: A new population-based optimisation algorithm (**R**ole-**B**ased **I**mitation algorithm - **RBI**) that can be used on layer 2 to:
  1. improve the solution quality and
  2. reduce the time to find the optimal solutions.

Layer 2 – RBI

- RBI is a population-based optimisation algorithm.
  - RBI provides a clear distinction of exploring and exploiting individuals according to
    1. the current degree of convergence of a (sub-)population
    2. the relative quality of the agent's solution



Cakar, E., Tomforde S. and Müller-Schloer, C. 2011. **A Role-based Imitation Algorithm for the Optimisation in Dynamic Fitness Landscapes.** In IEEE Swarm Intelligence Symposium (SIS 2011), pages 139 -146, Paris, France, 2011

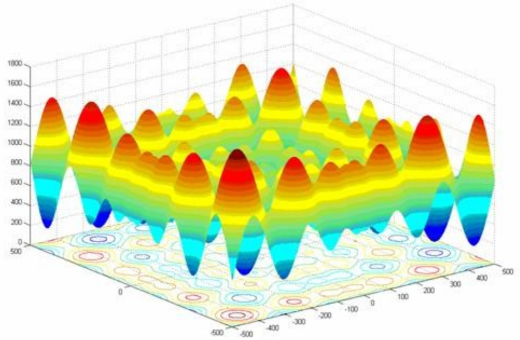
Layer 2 – RBI

- Comparison of RBI to Differential Evolution (DE), Particle Swarm Optimisation (PSO), Genetic Algorithm (GA) and Simulated Annealing (SA)
  1. in **static fitness landscapes** using different benchmark functions from the literature.
  2. in a **dynamic fitness landscape** using a scenario from the predator-prey domain.

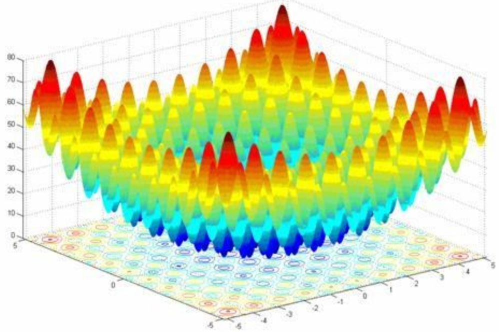
- A static fitness landscape doesn’t change over time while a dynamic fitness landscape may change, e.g. as a function of agent behaviour which is typical for OC scenarios.

Comparison in static fitness landscapes

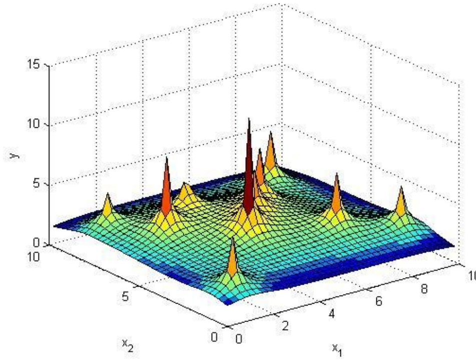
- Comparison of RBI with other algorithms using 21 benchmark functions
  - The benchmark functions are taken from "*A comparative Study of Differential Evolution, Particle Swarm Optimisation and Evolutionary Algorithms on Numerical Benchmark Problems*", Vesterstrom et al., CEC 2004
- F1 - F13 are high-dimensional functions each with 30 dimensions
- F14 – F21 are low-dimensional functions with 2 or 4 dimensions.
- Max number of function evaluations is set to 500,000
- Some of the benchmark functions:



F2- Schwefel function



F9 - Rastrigin function



F14 - Shekel function

Source: http://www-optima.amp.i.kyoto-u.ac.jp

Comparison in static fitness landscapes

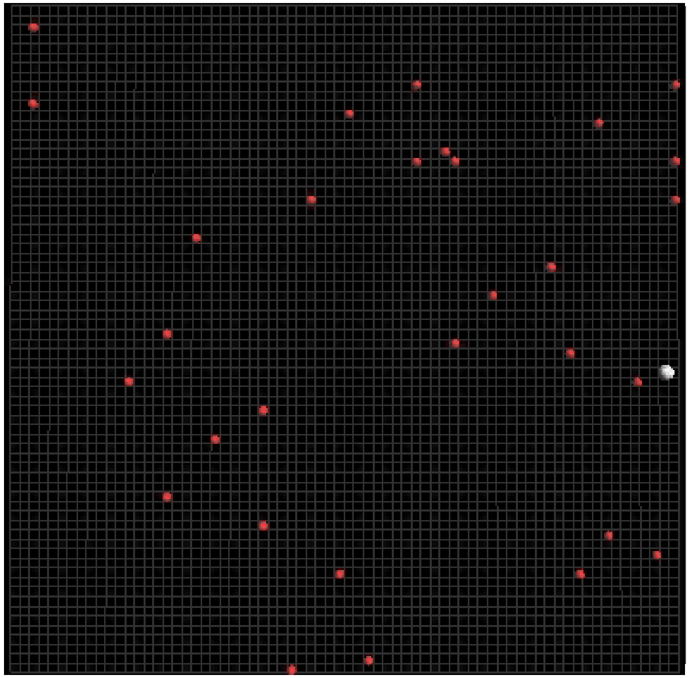
- RBI is better than GA, PSO and SA and on the same level as DE. (Best solutions are shown in grey)

Group		RBI	DE	PSO	GA	SA	Optimum
Multimodal and low-dimensional functions	F14	9.9800384 e-01	9.9800384 e-01	9.9800384 e-01	1.6892421 e+00	9.9800384 e-01	9.9800384 e-01
	F15	3.0748593 e-04	4.6010056 e-04	3.0748599 e-04	3.0814498 e-04	6.1205233 e-04	3.0748593 e-04
	F16	-1.0316285 e+00	-1.0316285 e+00	-1.0316285 e+00	-1.0316285 e+00	-1.0316285 e+00	-1.0316285 e+00
	F17	3.9788735 e-01	3.9788735 e-01	3.9788735 e-01	3.9788735 e-01	3.9788735 e-01	3.9788735 e-01
	F18	3.0 e+00	3.0 e+00	3.0 e+00	3.0 e+00	3.0 e+00	3.0 e+00
	F19	-9.486763 e+00	-10.1532 e+00	-9.735774 e+00	-8.410121 e+00	-9.984785 e+00	-10.1532 e+00
	F20	-10.402941 e+00	-10.402941 e+00	-10.402941 e+00	-10.402941 e+00	-10.049961 e+00	-10.402941 e+00
	F21	-10.536409 e+00	-10.536409 e+00	-10.536409 e+00	-9.772439 e+00	-10.536409 e+00	-10.536409 e+00

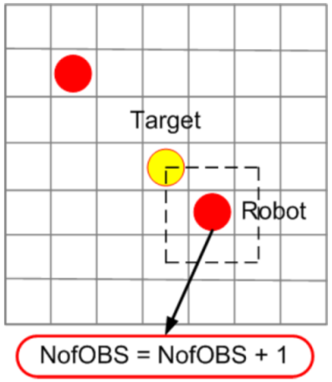
Group		RBI	DE	PSO	GA	SA	Optimum
Unimodal and high-dimensional functions	F1	0.000000 e+00	0.000000 e+00	0.000000 e+00	1.6828756 e-06	2.390598 e-10	0.000000 e+00
	F2	0.000000 e+00	8.142916 e-37	0.000000 e+00	5.466097 e-03	5.140537 e-05	0.000000 e+00
	F3	0.000000 e+00	0.000000 e+00	0.000000 e+00	6.52916 e-04	1.29546415 e-05	0.000000 e+00
	F4	3.223638 e-15	3.7110183 e-08	3.7296683 e-11	1.9280297 e+00	9.731591 e-01	0.000000 e+00
	F5	2.620019 e+01	4.03334 e-22	2.2833145 e+01	3.3183784 e+01	6.4433184 e+00	0.000000 e+00
	F6	0.000000 e+00	0.000000 e+00	0.000000 e+00	0.000000 e+00	0.000000 e+00	0.000000 e+00
	F7	3.419498 e-04	3.215471 e-03	1.6625043 e-03	8.144887 e-04	3.7914343 e-02	0.000000 e+00
Multimodal and high-dimensional functions	F8	-1.2569486 e+04	-1.2569486 e+04	-1.0557845 e+04	-7.4431504 e+03	-1.2558608 e+04	-1.2569486 e+04
	F9	0.000000 e+00	0.000000 e+00	2.527239 e+01	3.0669328 e-04	1.9462983 e-01	0.000000 e+00
	F10	8.970602 e-15	3.996803 e-15	7.312669 e-15	9.4749196 e-04	1.9690224 e-04	4.4408 e-16
	F11	0.000000 e+00	0.000000 e+00	1.0678519 e-03	8.698255 e-03	5.5567506 e-07	0.000000 e+00
	F12	1.5705448 e-32	1.5705448 e-32	3.4549083 e-03	1.2341902 e-08	3.499239 e-10	1.5705448 e-32
	F13	-1.1504403 e+00	-1.1504403 e+00	-1.1504403 e+00	-1.1504400 e+00	-1.1504403 e+00	-1.1504403 e+00

Comparison in dynamic fitness landscapes

- A scenario from the pursuit (predator-prey) domain
- The predators (robots) try to follow and observe the prey (target).



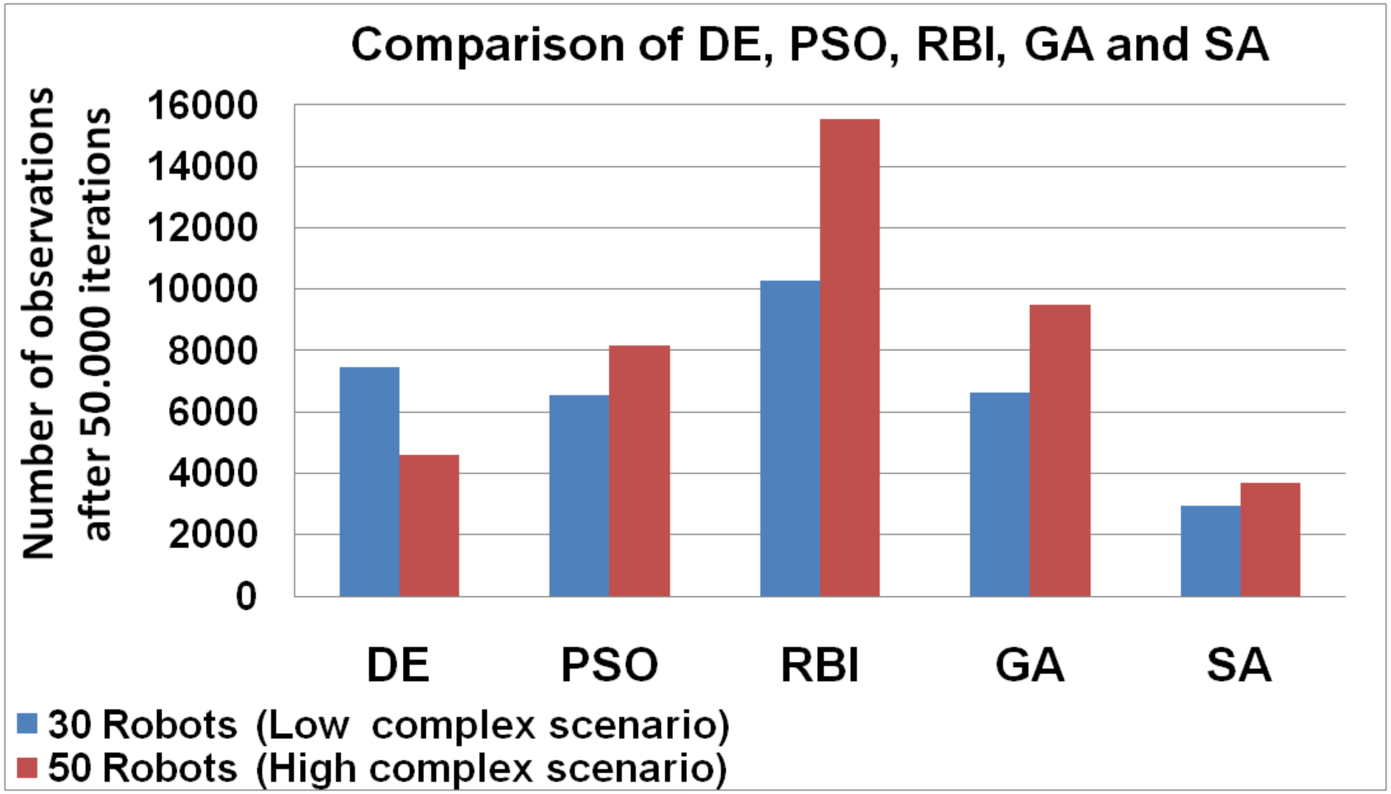
- Grid-based environment
- The target evades the robots and is twice as fast as a robot.
- Each robot counts its number of observations (variable NofOBS) that is incremented each time the target is in the 1-step neighbourhood of the robot.



- Goal of a robot: Maximise the value of its NofOBS
- System performance: The sum of all NofOBS
- Minimum 1 cell distance between two robots: The target cannot be captured.

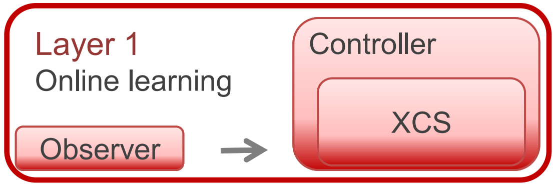
Comparison in dynamic fitness landscapes

- Different scenarios with an increasing level of complexity are investigated.
- Total number of observations is measured after **50,000** iterations.
- Each robot optimises its behaviour every **100** iterations. The number of function evaluations for a single robot is limited to **500** (50,000 / 100).



Layer 1 – The adaptation layer

- Offline learning with an eXtended Classifier System (XCS)



- Questions: How can we improve the learning speed of the XCS, what kind of modifications are to be made?

- Contributions:
  1. Investigation and evaluation of centralised and distributed rule bases for an XCS
  2. Development of a rule combining mechanism (XCS-RC) to create maximally general classifiers that match as many inputs as possible while still being exact in their predictions

XCS – Rule Combining (XCS-RC)

- **XCS – RC** replaces the discovery component of the XCS (*covering* and *genetic operators*) with **rule combining**.
- A pair of classifiers is combined using the inductive reasoning.

Classifiers before combining

Index	Condition (*cl.C*)	Action (*cl.A*)	Prediction (*cl.P*)
1	11010	0	100
2	10110	0	98
3	111##	0	10

Result after combining

Index	*cl.C*	*cl.A*	*cl.P*
1	1##10	0	99
2	111##	0	10

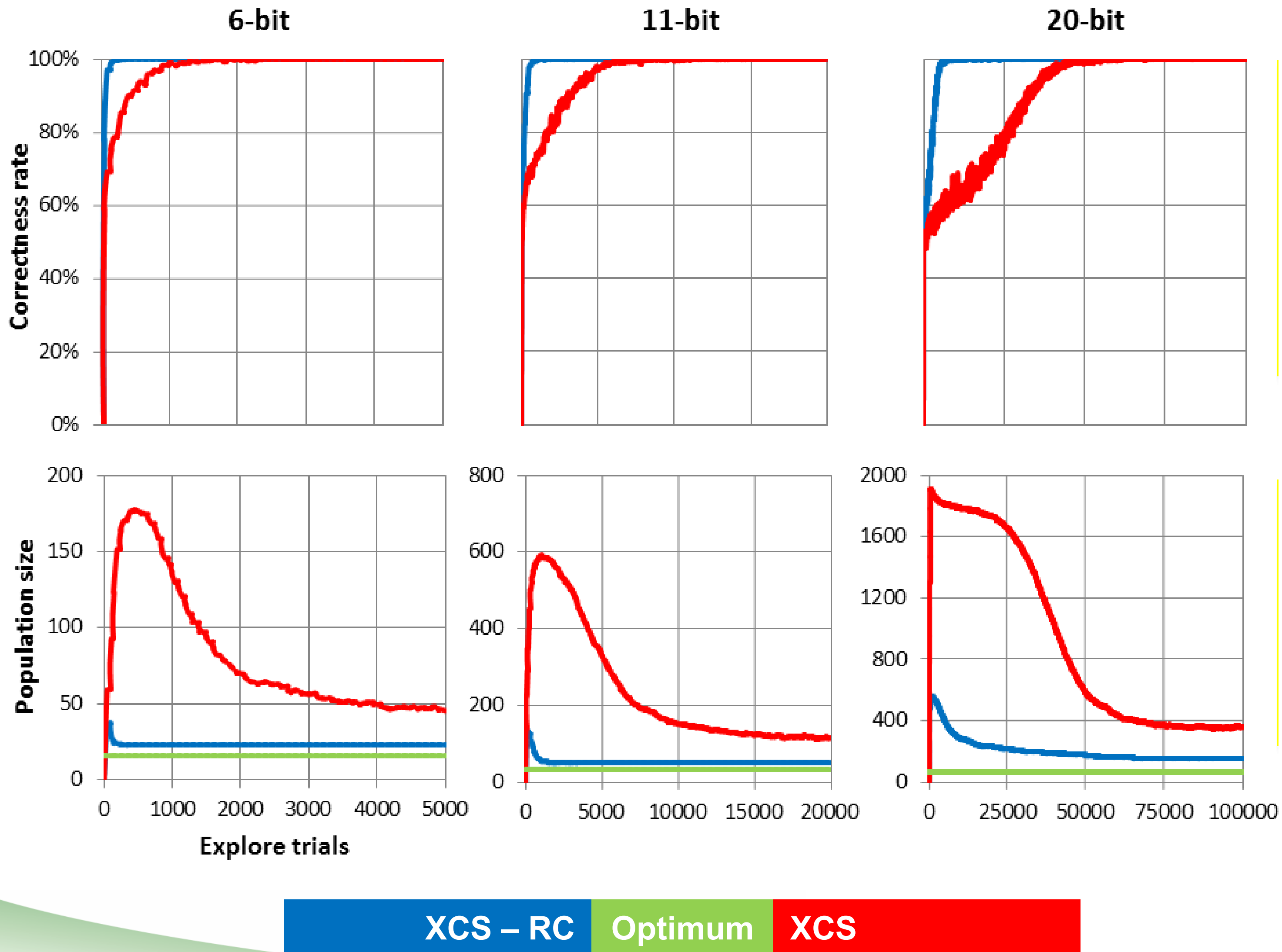
Conflict on „11110“

- Principles: both classifiers have the same **action**, similar **prediction level** and the combining result has **no disproving rule**
- Disproving rule: a classifier that is able to cover the **same condition** as the result of combining but having **significantly different predictions**
- In order to prevent such a conflict, an examination is included in the process

Fredivianus N., Prothmann, H., Schmeck, H. 2010. XCS Revisited: A Novel Discovery Component for the eXtended Classifier System. In Proceedings of 8th International Conference on Simulated Evolution And Learning (SEAL-2010)

Testbench 1

Single-step learning: The multiplexers (average of 20 runs)

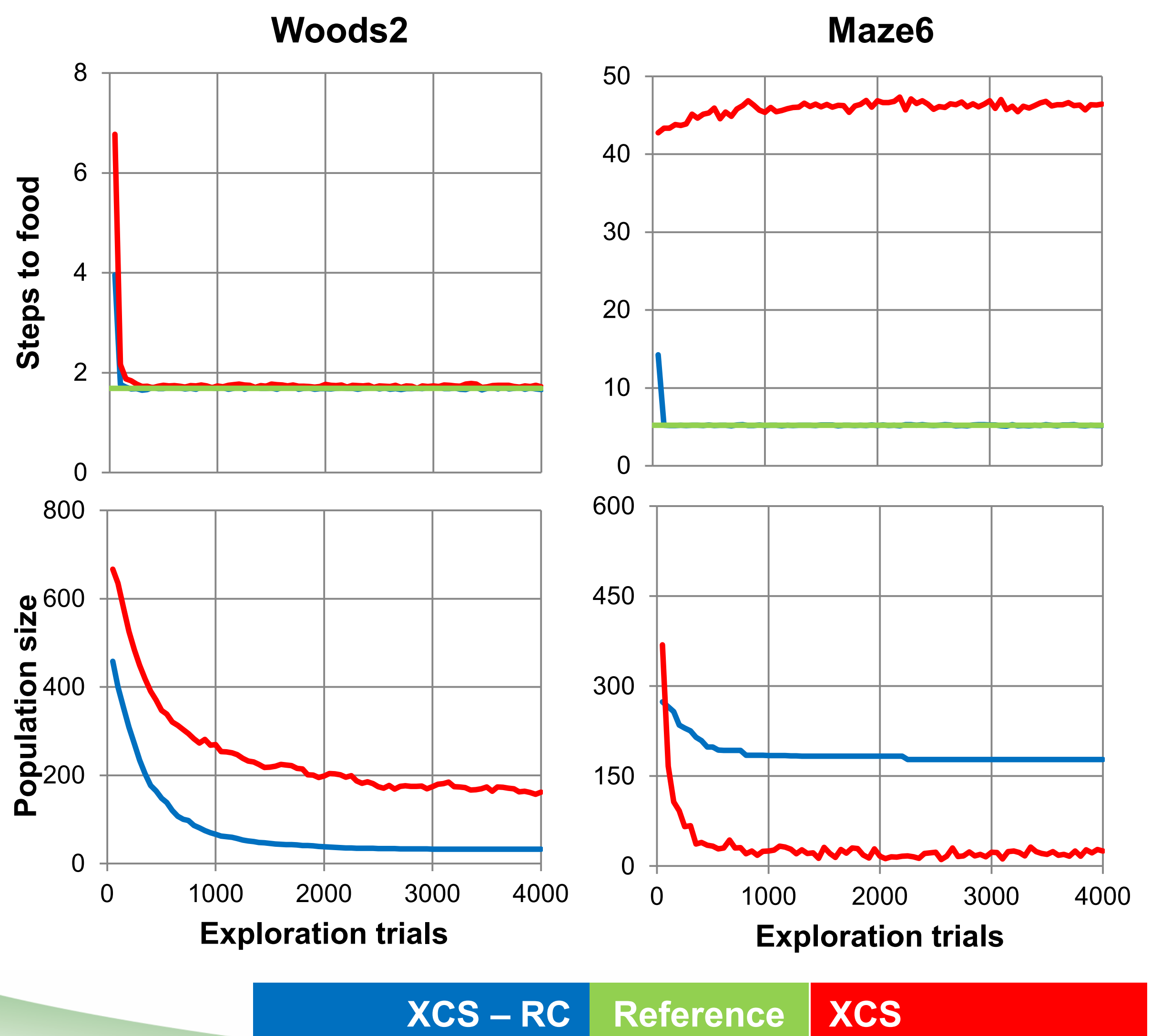


XCS-RC performs quicker in achieving 100% of correctness rate, compared to XCS

XCS-RC minimized the population size more quickly than XCS

Testbench 2

Multi-step learning: The Woods and Maze environments (average of 20 runs)

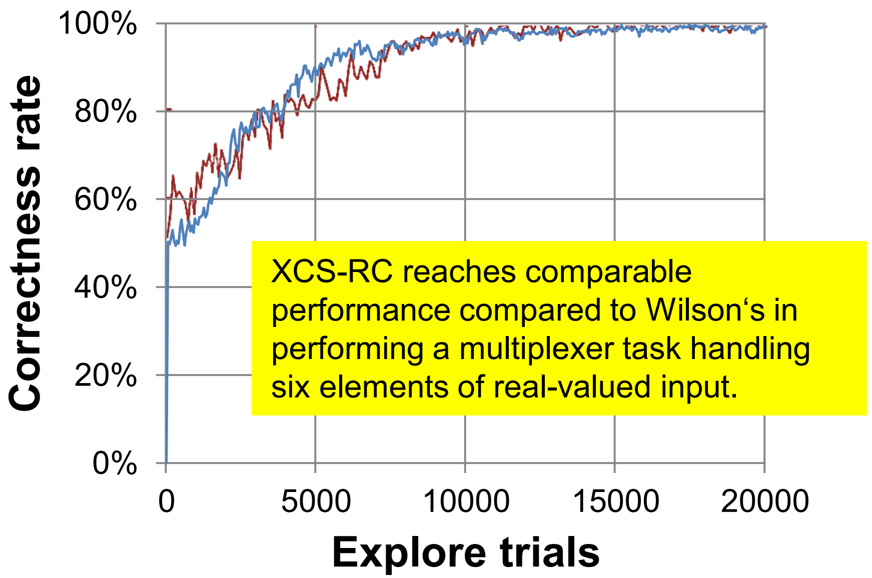


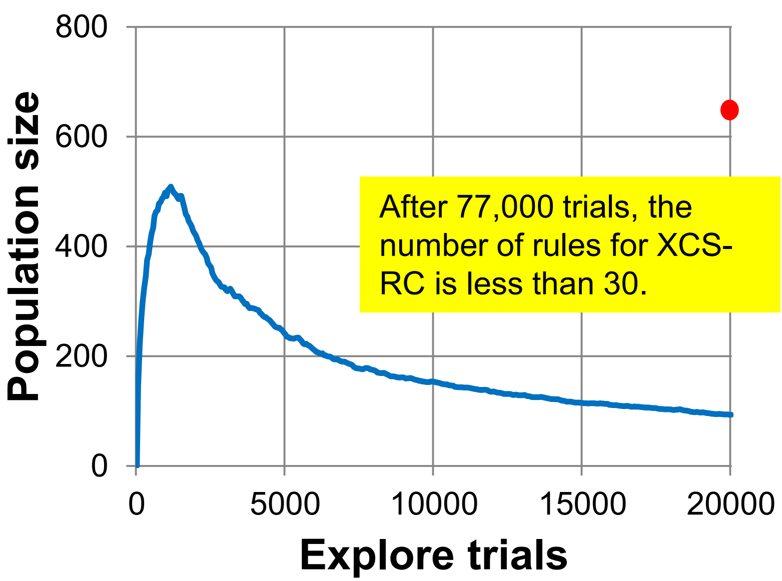
XCS-RC performs well in minimizing steps to food taken by the animat.

Numbers of classifiers in [P] are minimized correctly and significantly by XCS-RC.

XCS-RC and real-valued input

- The principles and mechanism of rule combining are also useful in handling real-valued input, e.g., in the multiplexer task.
- The performance of **XCS-RC** is comparable to the previous investigations (e.g., **Wilson's XCS**) with a high advantage of resource usage.





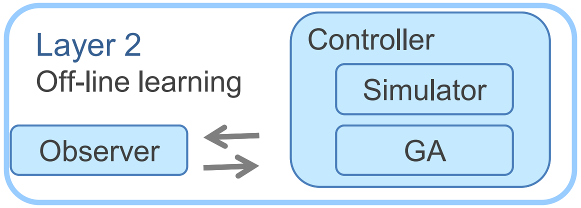
XCS – RC | XCS

- The OTC project implements XCS with real-valued input on its 1st layer.

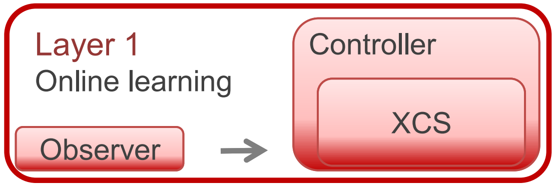
Investigated as a diploma thesis topic by Kais El-Kara under the supervision of Nugroho Fredivianus

Summary

- Summary
  1. Optimisation layer (Layer 2)



     - Development of a new population-based heuristic (Role Based Imitation algorithm - RBI)
     - Better results with RBI in comparison to DE, PSO, GA and SA in static and dynamic fitness landscapes
  2. Adaptation layer (Layer 1)



     - Investigation and evaluation of centralised and distributed rule bases for XCS
     - Higher learning performance with the rule combining mechanism (XCS-RC) in comparison to the standard XCS in single-step and multi-step problems
  3. Application of developed techniques regarding to other OC applications
     - Organic Network Control (ONC) system

       Dynamic Control of Mobile ad-hoc Networks – Network protocol parameter adaptation using Organic Network Control, Tomforde et al., ICINCO 2010
     - Improved results with the OCCS methodology.

Selected publications (1/3)

2011

- Cakar, E., Tomforde S. and Müller-Schloer, C. 2011. **A Role-based Imitation Algorithm for the Optimisation in Dynamic Fitness Landscapes.** In IEEE Swarm Intelligence Symposium (SIS 2011), pages 139 -146, Paris, France, 2011
- Cakar, E., Fredivianus, N., Hähner, J., Branke, J., Müller-Schloer, C., Schmeck, H. 2011. **Aspects of Learning in OC Systems.** In "Organic Computing - A Paradigm Shift for Complex Systems" incollection 3.1, pages 237-251, June 2011.

2010

- Cakar, E. and Müller-Schloer, C. 2010. **Decentralised and Adaptive Collaboration in Multi-Agent Systems.** In Proceedings of the 9th International Symposium on Parallel and Distributed Computing (ISPDC 2010), Istanbul - Turkey
- Fredivianus, N., Richter, U., Schmeck, H. 2010. **Collaborating and Learning Predators on a Pursuit Scenario.** Biologically Inspired Collaborative Computing (BICC 2010), IFIP Advances in Information and Communication Technology, September, 2010
- Fredivianus N., Prothmann, H., Schmeck, H. 2010. **XCS Revisited: A Novel Discovery Component for the eXtended Classifier System.** In Proceedings of *8th International Conference on Simulated Evolution And Learning (SEAL-2010)*
- Lode, C., Richter, U., Schmeck, H. 2010. **Adaption of XCS to Multi-Learner Predator/Prey Scenarios.** In Proceedings of *12th Annual Conference on Genetic and Evolutionary Computation (GECCO 2010), Seiten: 1015-1022, ACM, New York, NY, USA, Juli, 2010*
- Fisch, D., Jänicke, M., Sick, B., and Müller-Schloer, C. 2010. **Quantitative Emergence – A Refined Approach Based on Divergence Measures.** In Proceedings of the 4th International Conference on Self-Adaptive and Self-Organizing Systems (SASO-2010), Budapest – Hungary, Best paper award
- Schmeck, H., Müller-Schloer, C., Cakar, E., Mnif, M., Richter, U. 2010. **Adaptivity and Self-organisation in Organic Computing Systems.** ACM Transactions on Autonomous and Adaptive Systems, Vol. 5, No. 3, Article 10, September 2010
- Müller-Schloer, C. and Schmeck, H. 2010. **Organic Computing: A Grand Challenge for Mastering Complex Systems.** Information Technology (it), Vol. 52, No. 3, pages 135-141, May 2010

Selected publications (2/3)

2009

- Cakar, E. and Müller-Schloer, C. 2010. **Self-Organising Interaction Patterns of Homogeneous and Heterogeneous Multi-Agent Populations.** In Proceedings of the 3th International Conference on Self-Adaptive and Self-Organizing Systems (SASO-2009), San Francisco – California
- Tomforde, S., Cakar, E., Haehner, J. 2009. **Dynamic Control of Network Protocols - a new vision for future self-orgsanised networks** . In Proc. of the 6th Int. Conf. on Informatics in Control, Automation and Robotics – Intelligent Control Systems and Optimization, pages 285-290, 2009.

2008

- Branke, J. and Schmeck, H. 2008. **Evolutionary design of emergent behavior.** In Organic Computing, Würtz, R. P., Eds. Springer, 123–140.
- Cakar, E., Hähner, J., and Müller-Schloer, C. 2008. **Investigation of generic observer/controller architectures in a traffic scenario.** Accepted for publication in INFORMATIK 2008 – Beherrschbare Systeme – dank Informatik.
- Cakar, E., Hähner, J., and Müller-Schloer, C. 2008. **Creating collaboration patterns in multi-agent systems with generic oberserver/controller architectures.** Accepted for publication in Proceedings of the 2nd International ACM Conference on Autonomic Computing and Communication Systems (Autonomics 2008).
- Müller-Schloer, C. and Sick, B. 2008. **Controlled emergence and self-organisation.** In Organic Computing, Würtz, R. P., Eds. Springer, 81–104.
- Ribock, O., Richter, U., and Schmeck, H. 2008. **Using Organic Computing to control bunching effects.** In Proceedings of the 21th International Conference on Architecture of Computing Systems (ARCS 2008), U. Brinkschulte, T. Ungerer, C. Hochberger, and R. G. Spallek, Eds. LNCS, vol. 4934, Springer, 232–244.
- Richter, U. and Mnif, M. 2008. **Learning to control the emergent behaviour of a multi-agent system.** In Proceedings of the 2008 Workshop on Adaptive Learning Agents and Multi-Agent Systems at AAMAS 2008 (ALAMAS+ALAg 2008), F. Klügl, K. Tuyls, and S. Sen, Eds. 33 – 40.
- Richter, U., Prothmann, H., and Schmeck, H. 2008. **Improving XCS performance by distribution.** Accepted for publication in Proceedings of the 7th International Conference on Simulated Evolution And Learning (SEAL 2008).

Selected publications (3/3)

- Schmeck, H. and Müller-Schloer, C. **A characterisation of key properties of environment-mediated multi-agent systems.** In Engineering Environment-Mediated Multi-Agent Systems. Danny Weyns, Sven Brueckner, Yves Demazeau (Eds.), LNCS, 2008.

2007

- Cakar, E., Mnif, M., Müller-Schloer, C., Richter, U., and Schmeck, H. 2007. **Towards a quantitative notion of self-organisation.** In Proceedings of the 2007 IEEE Congress on Evolutionary Computation (CEC 2007), 4222–4229.

  Mnif, M., Richter, U., Branke, J., Schmeck, H., and Müller-Schloer, C. 2007. **Measurement and control of self-organised behaviour in robot swarms.** In Proceedings of the 20th International Conference on Architecture of Computing Systems (ARCS 2007), P. Lukowicz, L. Thiele, and G. Tröster, Eds. LNCS, vol. 4415. Springer, 209–223.

2006

- Branke, J., Mnif, M., Müller-Schloer, C., Prothmann, H., Richter, U., Rochner, F., and Schmeck, H. 2006. **Organic Computing – Addressing complexity by controlled self-organization.** In Post-Conference Proceedings of the 2nd International Symposium on Leveraging Applications of Formal Methods, Verification and Validation (ISoLA 2006), T. Margaria, A. Philippou, and B. Steffen, Eds. Paphos, Cyprus, 185–191.
- Mnif, M. and Müller-Schloer, C. 2006. **Quantitative emergence.** In Proceedings of the 2006 IEEE Mountain Workshop on Adaptive and Learning Systems (IEEE SMCals 2006). 78–84.
- Müller-Schloer, C. and Sick, B. 2006. **Emergence in Organic Computing systems: Discussion of a controversial concept.** In Proceedings of the 3rd International Conference on Autonomic and Trusted Computing (ATC 2006), L. T. Yang, H. Jin, J. Ma, and T. Ungerer, Eds. LNCS, vol. 4158. Springer, 1–16.
- Richter, U., Mnif, M., Branke, J., Müller-Schloer, C., and Schmeck, H. 2006. **Towards a generic observer/controller architecture for Organic Computing.** In INFORMATIK 2006 – Informatik für Menschen!, C. Hochberger and R. Liskowsky, Eds. GI-Edition – Lecture Notes in Informatics (LNI), vol. P-93. Köllen Verlag, 112–119.

2005

- Schmeck, H. 2005b. **Organic Computing – A new vision for distributed embedded systems.** In Proceedings of the 8th IEEE International Symposium on Object-Oriented Real-Time Distributed Computing (ISORC 2005). IEEE Computer Society, 201–203.