



BUDAPEST UNIVERSITY OF TECHNOLOGY AND ECONOMICS
FACULTY OF ELECTRICAL ENGINEERING AND INFORMATICS
DEPARTMENT OF TELECOMMUNICATIONS AND MEDIA INFORMATICS

Advanced Approaches in the Application Methodologies of Evolutionary Algorithms

Ph.D. Thesis Booklet

Krisztián Balázs
M.Sc. in Technical Informatics

Supervisor:
Dr. László T. Kóczy
Ph.D., D.Sc.

Budapest, 2013

1 Introduction and research objectives

Soft computing (also known as *computational intelligence*) methods were born in the second half of the 20th century. In efficiency they outperform previous approaches in high difficulty problem fields accepting sub-optimal solutions. The reason is that these techniques solve the problems with relatively low space and time complexity, furthermore they are applicable in cases when the analytic description of the problem is only partially or not known, or when the knowledge about the particular field is uncertain. The price of these advantageous properties must be paid in the inaccuracy and sub-optimality of the solution. Thus, the applicability of soft computing methods is restricted to problem areas, where efficiency and low computational demand are important aspects, on the other hand, some deficiency in accuracy is acceptable.

The three main branches of soft computing are evolutionary algorithms, fuzzy systems and neural networks. Although, they all possess the above properties, there are significant differences between them.

Evolutionary computation techniques are stochastic optimization approaches endeavoring to reach better and better solutions (i.e. to produce “individuals” representing more and more appropriate solutions), analogously to evolutionary processes observed in the nature in the competition between species and individuals.

Fuzzy systems and neural networks have good modeling properties. They are capable of modeling systems, for whose structure at the beginning the accessible knowledge is considerably incomplete (*grey-box problems*) or there is no accessible knowledge at all, however, the responses of the system is known or accessible for certain inputs (*black-box problems*).

A significant difference between these two soft computing approaches is that whereas neural networks hold their knowledge about the problem within the weights in the connections between their elements in a form being hardly extractable and even harder or not at all interpretable, fuzzy systems (constructed with adequate circumspection) gather their knowledge by building rule bases, where every piece of information is in an explicit, interpretable form. This is a huge advantage on the side of fuzzy systems.

Since the soft computing approaches in question (including evolutionary techniques) look back to a past of only a half century, naturally, both their theory and application practice still contain many unsolved questions, even though these methods are studied intensively and applied more and more widely owing to their favorable properties.

The main objective of the thesis is to propose approaches for the application of evolutionary algorithms, with which in particular optimization problem fields higher quality solutions can be found in lower computational complexity than in case of known approaches from the literature. Whenever it is interpretable in the given context, the secondary objective is to make it possible to adjust between different quality measures determining the overall quality of the solutions.

The novelty of these approaches is manifested mainly in:

- unified representations and operators,
- hybrid methods formed by involving both classical and non-classical optimization and modeling techniques,

- adaptively selecting the currently efficiently applicable evolutionary algorithms.

The applied techniques are supported by comprehensive experimental analysis everywhere and by formal analysis, where it is possible.

My research results can be formulated briefly as follows:

1. I successfully hybridized evolutionary algorithms with other state-of-the-art techniques in permutation based problem fields (Thesis statement group 1).
2. I applied evolutionary algorithms and fuzzy modeling within evolutionary methods for improving qualitative properties (Thesis statement group 2).
3. I applied evolutionary methods in various novel ways for different architectures in fuzzy rule based knowledge extraction (Thesis statement group 3).
4. I proposed a meaning preservation approach together with search space narrowing techniques for interpretable fuzzy system construction (Thesis statement group 4).
5. I introduced schedulers for switching between the different optimization algorithms during optimization processes (Thesis statement group 5).

The new methods including general approaches as well as concrete algorithms proposed in the thesis are summarized in Table 1 according to statement groups corresponding to the respective subsections of Section 3. This summary is based on the areas where the newly proposed methods can be applied and the type of improvements they achieve.

Table 1: Summary of the newly proposed methods corresponding to statement groups

	Application field	Concrete problem class	Property optimized	Improvement type
Statement group 1	combinatorial	permutation based optimization	throughput	performance
Statement group 2	combinatorial	scheduling / failure localization	modeling power / reliability	quality
Statement group 3	continuous	fuzzy learning	accuracy	performance
Statement group 4	continuous	fuzzy learning	interpretability	quality
Statement group 5	applicable for both	applicable for all	applicable for all	applicable for both

2 Methodology

In order to reach the previously described aims, during my research I invented and followed four thought patterns. These were guidelines giving general ideas for improving existing algorithms and approaches. Adapting these patterns to concrete cases led to the development of concrete application methods for problem classes and, specifically, for concrete tasks. The thought patterns I used were the following:

- developing efficient representations ([Subsection 3.1](#), [Subsection 3.2](#) and [Subsection 3.3](#)),
- hybridizing techniques and approaches ([Subsection 3.1](#), [Subsection 3.2](#), [Subsection 3.3](#) and [Subsection 3.5](#)),
- introducing reasonable restrictions ([Subsection 3.3](#) and [Subsection 3.4](#)),
- intuitively finding the improvement possibilities ([Subsection 3.4](#)).

The applicability and competitiveness of the newly invented techniques and approaches are verified through multiple stages:

- heuristic arguments (in case of every result),
- formal analysis (where applicable),
- experimental analysis (in case of every result).

3 New results

In this section my new research results will be described briefly.

3.1 Single- and multi-threaded hybrid approaches for permutation based scheduling problems

Thesis statement group 1. *I proposed a new individual representation technique, furthermore I proposed and compared two sets of operators for two different encoding methods and several hybridization approaches for single- and multi-threaded algorithms solving permutation based scheduling problems.*

Related works: [\[J1\]](#), [\[C7\]](#), [\[C8\]](#).

3.1.1 New encoding methods and evolutionary operators for permutation based scheduling problems

I applied two types of individual representation (i.e. two encoding methods) for chromosome based evolutionary techniques applied for permutation based scheduling problems. Permutation based

scheduling problems are optimization tasks, whose search space is the set of permutations corresponding to a particular size.

The first encoding method is based on the permutations themselves, thus the evolutionary operators modify the elements of the permutations directly.

The second encoding method is a newly proposed one, which is an indirect, real value based encoding approach. It is an obvious extension of those representations applied for numerical optimization problems. Although, the operators modify the values of real-valued vectors (arrays)—since the objective function is defined over permutations, the chromosomes represent permutations actually—there is a need to convert the real-valued vectors to permutations somehow. I proposed to do this by ordering the values.

In order to reduce time complexity costs, the chromosomes can be ‘mirrored’ within the individuals in a manner, which makes the modifications caused by the evolutionary operators and the evaluation of the individual simpler performable.

For both the permutation based and the real-value based encoding I proposed three ‘atomic operators’ (mutation, gene transfer and local search), from which evolutionary operators can be constructed for different chromosome based evolutionary algorithms.

Thesis statement 1.1. *I proposed a new real-value based encoding method for permutation based scheduling problems, furthermore I proposed evolutionary operators for permutation based and for the new real-value based individual representation techniques.*

Related work: [J1].

3.1.2 Integrating the new techniques into evolutionary algorithms applied for the PFSP problem

I applied the newly proposed approaches in Genetic Algorithm (GA) [9], Bacterial Evolutionary Algorithm (BEA) [17] and Particle Swarm Optimization (PSO) [11] as well as their memetic variants (which include local search) for solving the Permutation Flow Shop Problem (PFSP) [20]. In this problem there are given jobs and machines. All the jobs should be processed by all the machines one after another. The machines are deployed in a line and a machine can handle one single job at once, that is, the process of the jobs is pipeline-like. There is also given an n-by-m processing time matrix defining the necessary amount of time a job has to stay on a machine, for each job-machine pair. A job can be processed on any machine only if the machine is free (the preceding job has finished on the machine) and the job has already been processed on the preceding machine.

The task is to find a permutation (a sequence) of the jobs, in case of which the total processing time of all the jobs on all the machines (i.e. the so-called makespan) is minimal.

The different evolutionary operators used in my investigations by the evolutionary algorithms GA, BEA and PSO were constructed to the three newly proposed atomic operators.

I carried out simulation runs in order to evaluate and to compare the efficiency of the proposed approaches and the established algorithms. First, the new methods were compared to each other, then the best one was compared to other heuristics: the well-known Iterated Greedy (IG) technique

together with its memetic variant (IGLS) [20] and a genetic algorithm based memetic method (MA) [19], which is e.g. used in combination with IG (MA+MIG method) in multi-processor systems.

For these purposes, a dozen problems were applied from the well-known Taillard's benchmark set [22]. Exactly one problem from each available problem sizes.

For example, for the most difficult problem from the benchmark set the means of the objective function values of the best individuals of the new techniques are presented in Figure 1. The horizontal axis shows the elapsed computation time in seconds and the vertical axis shows the makespan values of the best individuals at the current time.

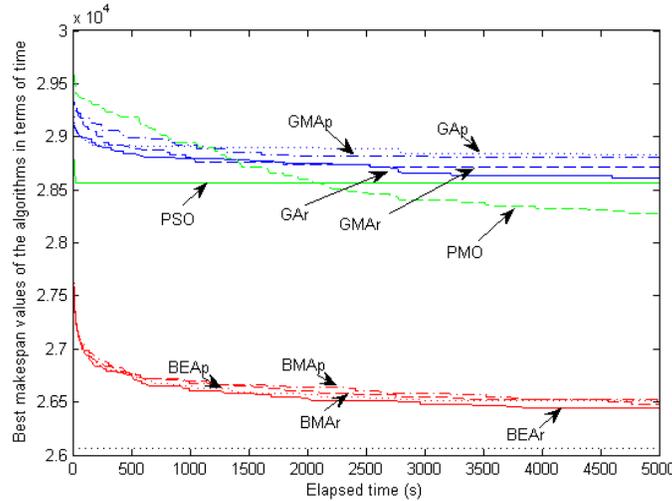


Figure 1: Results for the 500x20 size problem (ta111)

Since BMAr, the real-value based Bacterial Memetic Algorithm (BMA) [4], appeared to be the most efficient algorithm, this technique was involved in further investigations: this method was compared to the Iterated Greedy heuristic and to the genetic algorithm based memetic method.

Although, the best constructed method was more efficient than the genetic algorithm based memetic technique applied in multi-processor systems, it was outperformed by one of the state-of-the-art heuristics, the Iterated Greedy method.

Thesis statement 1.2. *I integrated the two encoding methods and the proposed evolutionary operators into Genetic Algorithm (GA), Bacterial Evolutionary Algorithm (BEA) and Particle swarm Optimization (PSO) as well as into their memetic variants, furthermore I applied the constructed techniques for the Permutation Flow Shop Problem (PFSP). I compared and evaluated the constructed algorithms on a standard benchmark problem set (Taillard's instances), and I provided experimental evidence for the superiority of the real-value based Bacterial Memetic Algorithm (BMAr). Moreover, I compared the best algorithm constructed*

with state-of-the-art techniques, and I provided experimental evidence for the superiority of Iterated Greedy (IG) heuristics.

Related work: [J1].

3.1.3 New hybridization approaches and the Bacterial Iterated Greedy hybrid algorithms for the PFSP problem

Although, even the best established chromosome based evolutionary algorithm, namely the Bacterial Memetic Algorithm could not perform as well as state-of-the art methods for the PFSP task, like the well-known Iterated Greedy heuristic, the idea arose that some combinations of this evolutionary and the IG techniques may result fruitful algorithms. A reason behind this idea is that in case of multi-threaded IG methods the combination with evolutionary algorithms was able to improve the performance as studies about parallel IG technique involving the genetic algorithm based memetic technique (MA+MIG) showed (e.g. [19]).

It is also worth trying to replace the genetic algorithm based memetic method in this hybrid heuristic with the Bacterial Memetic Algorithm, which appears to be more effective for the PFSP task and which shows better properties in other fields of optimization, too.

Generalizing these ideas I proposed approaches to hybridize arbitrary population based evolutionary algorithms (PBEA) with any other non-population based iterated heuristic (NPBIH) as follows.

If only one thread is considered, i.e. there is no parallelization in the algorithms, in order to obtain a hybrid heuristic the only way to combine the PBEA with the NPBIH is to embed one of them into the other one, because executing the two methods one after another does not really mean a hybrid method.

If the base method is the PBEA and the embedded heuristic is the NPBIH, then the latter can be considered as an additional evolutionary operator (just like a local search operator added to the pure evolutionary method in order to obtain a memetic technique), which is executed a predefined number of iterations long in each generation of the PBEA for each individual in the population of candidate solutions.

At the beginning of the optimization process if the initial population was generated by some distribution, the initial individuals would be low-grade. If the initial candidate solutions were created by a deterministic initial heuristic (DIH) (e.g. the so-call NEH method [18] in case of IG) the individuals would be much better, however, they all would be the same, thus the diversity of the population would be zero, which is definitely unfavorable. Hence the initial population is generated by a stochastic variant of the DIH (e.g. the Stochastic NEH heuristic proposed by myself) in order to obtain fair population diversity together with promising initial candidate solutions.

If the base method is an NPBIH and the embedded algorithm is a PBEA, then the situation is different from the previous one, because the NPBIHs consider only one candidate solution at once, whereas the PBEAs maintain a whole population of the permutations. Therefore, at the embedding point a number of individuals must be derived from one candidate solution. This means that the embedding point must be such a point in the base algorithm, where the base heuristic can easily run on side-roads, i.e. it can fork to slightly different ways, forming the individuals of the population of the PBEA.

Beyond the basic embedding types, sub-types can also be created by the omission of algorithm parts (e.g. operators). Omission can often be viable, because although the resulting technique is simpler and thus it makes less improvement iteration-by-iteration, the algorithm becomes faster and thus more iterations can be executed within the same amount of time. Therefore, the maimed method might have higher efficiency than the original one.

Unlike discussed in the single thread case, if multiple threads are considered, i.e. there are more than one algorithms running parallel, even if the original techniques are executed on every thread, a hybrid method can be obtained by running different methods on different threads. An example to such a hybrid multi-threaded heuristic is the above mentioned MA+MIG method, where the original algorithms are running on all the threads, mostly IG techniques parallel, but there is one exceptional thread, where the genetic algorithm based memetic method is executed.

I proposed similar hybrid techniques by exchanging the heuristics on the threads. Considering the previously discussed chromosome based evolutionary methods and single-threaded techniques two hybrid approaches are proposed for multi-threaded optimization. They are originated from the MA+MIG method by making the following changes, respectively:

1. The genetic algorithm based memetic heuristic and the iterated greedy threads are exchanged with a PBEA and NPBIH threads, respectively.
2. The genetic algorithm based memetic heuristic and the iterated greedy threads are exchanged with a PBEA and hybrid PBEA-NPBIH threads, respectively.

Following the above steps I hybridized bacterial techniques with the iterated greedy methods.

Table 2 shows the results of the comparison of IGLS, BMAR and the best established single-threaded bacterial iterated greedy technique (SBIG), where 12 benchmark problems were considered, one instance from each size (BKMV—Best Known Makespan Value¹, Rel. diff—Relative difference).

As it can be observed, on the more difficult half of the benchmark problems SBIG always gave the best makespan values. After this, it is not strange that considering all the benchmark tasks whereas BMAR has lower efficiency than IGLS, their combination, the new hybrid method clearly outperformed both other techniques fulfilling my original expectations.

Thesis statement 1.3. *I proposed new hybridizing approaches for combining population based evolutionary algorithms with any other non-population based iterated heuristic.*

I applied the proposed combination approach for hybridizing Bacterial Evolutionary Algorithm (BEA) with single- and multi-threaded Iterated Greedy (IG) heuristics. I compared and evaluated the established hybrid Bacterial Iterated Greedy algorithms on the same standard benchmark problem set (Taillard's instances). I compared the best established single- and multi-threaded hybrid algorithms with single- and multi-threaded IG heuristics, and I provided experimental evidence for the superiority of the newly proposed techniques.

Related works: [C7], [C8].

¹<http://mistic.heig-vd.ch/taillard/problemes.dir/ordonnancement.dir/ordonnancement.html>

Table 2: Results of IGLS, BMAr and SBIG

Problem				BMAr		IGLS		SBIG	
ID	Size	BKMV	Time limit	Result	Rel. diff.	Result	Rel. diff.	Result	Rel. diff.
ta001	20x5	1278	10	1278	0.00%	1278	0.00%	1278	0.00%
ta011	20x10	1582	10	1586.4	0.28%	1583.2	0.08%	1582.6	0.04%
ta021	20x20	2297	10	2303.8	0.30%	2301.6	0.20%	2304.2	0.31%
ta031	50x5	2724	50	2724	0.00%	2724	0.00%	2724	0.00%
ta041	50x10	2991	50	3045.6	1.83%	3035.2	1.48%	3035.4	1.48%
ta051	50x20	3847	50	3945.6	2.56%	3925	2.03%	3917.8	1.84%
ta061	100x5	5493	200	5493	0.00%	5493	0.00%	5493	0.00%
ta071	100x10	5770	200	5788.2	0.32%	5786.8	0.29%	5781.4	0.20%
ta081	100x20	6202	200	6392.4	3.07%	6350	2.39%	6319.2	1.89%
ta091	200x10	10862	1000	10882.2	0.19%	10888.6	0.24%	10874.6	0.12%
ta101	200x20	11181	1000	11432.4	2.25%	11392.4	1.89%	11373.2	1.72%
ta111	500x20	26059	5000	26476.4	1.60%	26395.6	1.89%	26298.2	0.92%

3.2 Improving qualitative properties during combinatorial optimization by evolutionary algorithms and fuzzy modeling

Thesis statement group 2. *I proposed new evolutionary approaches partly together with fuzzy modeling techniques for improving qualitative properties of complex systems during combinatorial optimization.*

Related works: [C6], [C17].

3.2.1 Novel encoding methods for resource scheduling problems

I applied two representations for resource scheduling problems.

The first problem formulation is a new *naive representation* of the task. In this case tuples are used to represent the schedules. For this purpose I introduced two new notions: *demand unit* and *demand*. In order to have an as simple frame algorithm as possible, optimization deals with *demands* instead of resources. This way the partial results can be merged easier, because shifts are formed from the demands belonging to different parts of the problem, and then these shifts are assigned to the resources.

A demand unit expresses a certain activity (e.g. driving or having a break in case of Crew Scheduling Problems), which must be fulfilled by a resource (e.g. driver). A list of consecutive demand units is a demand. Each demand can be considered as a partial work schedule. Since exactly one demand belongs to each slot in the schedule, the applied tuples are elements of $D^{|R|}$, where D is the set of demands and $|R|$ is the number of slots. The objective (cost) function is defined over the tuples, and the task is to find the best tuple, i.e. the tuple fulfilling the given constraints (e.g. business criteria) besides having the lowest cost.

A more sophisticated problem formulation is based on the well-known Set Covering Problem (SCP). In case of this NP-hard problem there are given a set U (*universe*), a family of sets over the universe C (*covering set family*) and a cost function $c : C \mapsto \mathbb{R}^+$, which defines the cost of each

covering set. The task is to cover all the elements of the universe with some of the covering sets, so that the sum of the cost of the applied covering sets (*total cost*) be minimal.

I proposed the corresponding modifications of evolutionary operators crossover, mutation, bacterial mutation and gene transfer for the naive representation.

3.2.2 Handling business requirements during evolutionary optimization using fuzzy modeling

Business criteria can be formulated as constraints defining the feasible part of the search space. The solutions in the infeasible part of the search space are not acceptable due to business considerations. However, often business criteria cannot be defined exactly or they have some flexibility. In order to deal with this uncertainty and to adopt the flexibility, the feasible set can be defined as a fuzzy set. That is, the business criteria can be formulated as fuzzy constraints.

Some of the criteria are stricter, whereas the others are more lenient, thus the fuzzinesses of the constraints are different, and, in general, the more important the fulfillment of a criterion, the steeper the sides of the membership function in the corresponding fuzzy constraint.

Since the role of the fitness values in evolutionary algorithms is to determine the goodness of the individuals from the point of view of the solution given to the scheduling task, the fitness function directs the evolution. Therefore, the most obvious choice is to make the fitness function be able to deal with as many objectives as possible. The total cost and the fuzzy constraints derived from the business criteria can be included in the fitness function. The total cost can serve as the main part of the function, whereas the number and degree of constraint violations can add a penalty term. Giving more fuzziness to the constraints refines the penalty term and makes the fitness function more smooth. Thus, the evolutionary process becomes more sensible and might produce better solutions.

Thesis statement 2.1. *I proposed two new encoding methods for resource scheduling problems, and a new approach for handling business requirements by fuzzy modeling during evolutionary optimization.*

Related work: [C6].

3.2.3 Integrating the new techniques into evolutionary algorithms applied for the CSP problem

The Crew scheduling problem (CSP) [6] is a well-known optimization problem, which is a common abstraction of many business processes, where crew or other notions of resources are to be assigned to certain tasks so that boundary conditions and informal business criteria are also met. Crew scheduling problems are primarily found in ground and air transportation companies, logistics and related businesses.

For the CSP problem I applied both proposed representation types considering *rounds* instead of *slots* and *drivers* instead of *resources* everywhere.

Bacterial Memetic Algorithm can use the naive formulation of the problem. In this case the tuples describing the schedules are represented by the chromosomes (individuals). Each gene of the chromosome corresponds to a single round. The bacterial operators are responsible for changing the

values of the genes, i.e. assigning new demands to the rounds in the schedules, in order to obtain as good solution as possible.

In case of Ant Colony Optimization [7] the SCP based problem formulation can be applied. If the whole covering of the universe, different covering sets and the total cost of a given covering are represented by the food resource, the corresponding partial paths and the length of a given path, respectively, then the method is able to produce a covering set system having a quasi-minimal cost.

Thesis statement 2.2. *I integrated the designed encoding methods and fuzzy modeling techniques into Bacterial Evolutionary Algorithm (BEA) and Ant Colony Optimization (ACO); furthermore I applied the constructed techniques for the Crew Scheduling Problem (CSP).*

Related work: [C6].

3.2.4 Improving system reliability in optical networks by failure localization using evolutionary optimization

Over the last decades optical networks took an important role in communication systems mainly due to their very high data transfer capabilities. As a result, the reliability of optical networks became crucial in nowadays information society. This is well illustrated by surveys (e.g. [10]) describing some examples of failure costs, which showed that one single downtime hour can easily result in the loss of millions of dollars due to the lack of communication.

Rather recent failure localization approaches (e.g. [1], [23] and [3]) apply so-called *monitoring paths (m-paths)*, *monitoring cycles (m-cycles)*, or generally, *monitoring trails (m-trails)*. They perform failure localization by forming the set of m-trails (including m-paths and m-cycles) so that they solve the *Unambiguous Failure Localization (UFL)* problem [3]. The UFL task is to cover the graph of the optical network by connected edge sets (m-trails) in a way that the failure of a link causes the failure of such a combination of the m-trails, which unambiguously identifies the failed link. This can be achieved if every edge in the graph of the optical network is uniquely covered by m-trails, i.e. there are no identical sets of m-trails corresponding to different edges. For cost efficiency purposes, the UFL problem should be solved using as few m-trails as possible. This requirement transforms the failure localization problem to an optimization task.

Most of the existing methods solving the UFL task (including the ones described in [1], [23] and [3]) have a significant deficiency as they require the freedom to use arbitrarily established m-trails in the graph of the optical network. However, considering only a predefined set of paths in the graph as permitted m-trails can be a more realistic, cost-effective and better way to achieve maximal throughput of a network, thus more reasonable assumption than allowing the application of arbitrary m-trails.

I proposed an approach formulating this latter, restricted UFL problem as a special SCP task and solving it using the Ant Colony Optimization (ACO) method.

I carried out simulation runs in order to demonstrate the applicability of the newly proposed evolutionary approach for solving the link failure localization problem in optical networks, furthermore to compare it with some recent methods, namely Random Code Swapping (RCS) [23] and Link Code Construction (LCC) [3], whose applied m-trail sets are not predefined.

In the simulation runs eight benchmark problems were used based on randomly generated connected graph topologies.

From the simulation results it can be observed that with the consideration of the significant handicap of the newly proposed method compared to RCS and LCC, namely, ACO was the only algorithm considering the set of permitted m-trails, the proposed novel approach has promising inherent possibilities.

Thesis statement 2.3. *I proposed an Ant Colony Optimization (ACO) based failure localization approach for improving system reliability in optical networks, which is able to deal with a predefined set of possible monitoring trails.*

I formulated the restricted version of the Unambiguous Failure Localization (UFL) problem as a modified Set Covering Problem (SCP), namely Unambiguous Set Covering Problem (USCP), then I adapted the ACO technique to this modified SCP task.

Related work: [C17].

3.3 New methods in fuzzy rule based knowledge extraction

Thesis statement group 3. *I proposed and compared a family of new methods for fuzzy rule based knowledge extraction approaches applying different fuzzy architectures.*

Related works: [C16], [C1], [C3], [C2], [C5], [C4], [C18], [C19], [S1], [J6], [C10], [J3], [J4], [C11], [J5].

3.3.1 Application of evolutionary algorithms in dense and sparse flat fuzzy systems

I applied several types of evolutionary algorithms in dense and sparse flat fuzzy rule based knowledge extraction systems.

For flat rule bases trapezoidal membership functions were used. The optimized parameters were the characteristic points of the membership functions (i.e. the vertices of the trapezoids) of the rules in the rule base.

I carried out simulation runs in order to compare the efficiency of the inference techniques combined with different numerical optimization methods for solving machine learning problems and to discover the difference between the properties of systems applying dense and sparse rule bases. I applied all reasonable combinations of Mamdani-inference [15], Stabilized Kóczy-Hirota (SKH)-interpolation [24] methods and several chromosome based evolutionary optimization algorithms (including fix as well as variable length optimization techniques, i.e. the methods working on fix and the ones coping with variable rule base sizes) for three machine learning problems (the one dimensional pH [4], the two dimensional ICT [4] and the six dimensional Nawa-Furuhashi problem [17]).

In case of variable length methods I proposed new evolutionary operators for handling fuzzy rules. During the execution of each operator the chromosome of the particular individual expands, shrinks or keeps its size randomly. If the chromosome expands, a whole new random rule is inserted between

two random consecutive existing rules (or before the first or after the last rule). If the chromosome shrinks, a whole randomly selected rule is removed from the existing ones.

Since bacterial based algorithms performed better than genetic and particle swarm based ones, for variable length simulations the modifications of bacterial based techniques (BEAv, BSDv, BMAv) were applied.

Based on the simulation results I observed the following tendencies for the fixed length methods in case of inference techniques applied on both dense and sparse rule bases:

- Bacterial techniques seemed to be better than the corresponding genetic and particle swarm methods.
- PSO always performed better than GA and sometimes relatively close to BEA.
- Among particle swarm methods PSO gave better results than the algorithms using gradient steps.
- PMA (PSO using LM steps) seemed to be the worst technique using Levenberg-Marquardt gradient steps.
- GMA (GA using LM steps) had the highest convergence speed among genetic algorithms.
- Usually, after a sufficient time, BMA (BEA using LM steps) was not worse than any other algorithm. The more difficult the problem is, the stronger the advantage of the technique appears.

It might be said that BMA advances ‘slowly but surely’ to the optimum. ‘Slowly’, because in most of the cases at the beginning of the optimization process it did not have the highest convergence speed. ‘Surely’, because during the simulations it did not lose so much from its efficiency than the other techniques.

For example, [Figure 2](#) shows the fitness development of the algorithms in terms of time in case of the Nawa-Furuhashi problem together with Mamdani-inference.

In case of variable length methods as the dimensionality of the learning problem grows, the advantage of the interpolative inference technique that comes from the ability to use sparse rule bases appears better and better. Since the size of the rule base was penalized by a factor in the fitness function, it even occurred that for simple problems the variable length methods produced higher fitness values than the fixed length techniques.

Thesis statement 3.1. *I proposed evolutionary operators for variable length bacterial algorithms for modifying the number of the fuzzy rules during fuzzy rule based knowledge extraction processes.*

I applied Genetic Algorithm (GA), Bacterial Evolutionary Algorithm (BEA), Particle Swarm Optimization (PSO) and their memetic variants (GSD, BSD, PSD, GMA, BMA and PMA), where Steepest Descent (SD) and Levenberg-Marquardt (LM) techniques were used as local search methods, as well as the newly constructed variable length bacterial algorithms (BEAv, BSDv and BMAv) for establishing both dense and sparse flat fuzzy rule based knowledge

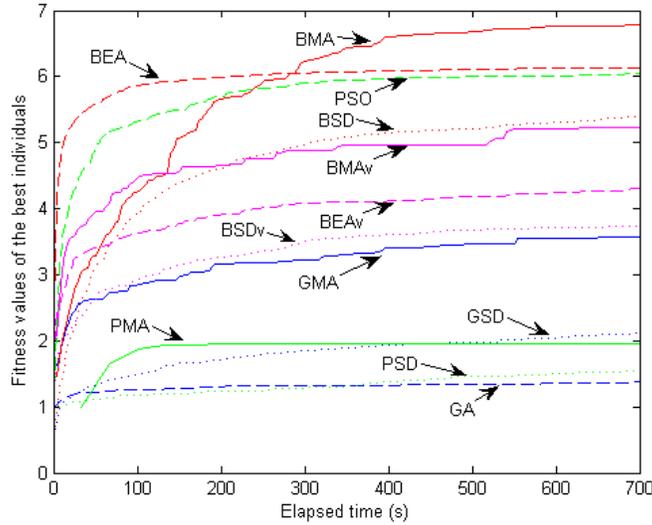


Figure 2: Results for the six-dimensional learning problem applying Mamdani-inference based learning

extraction systems using the characteristic points of the membership functions within the fuzzy rules as genes for the evolutionary techniques. I compared and evaluated the established systems on a variety of benchmark problems, and I provided experimental evidence for the superiority of bacterial techniques (BEA, BMA, BEAv and BMAv). I compared the variable length methods to the fixed length techniques and I provided experimental evidences for cases, when the application of the newly proposed approaches led to better results by smaller rule base complexity.

Related works: [C16], [C1], [C3], [C2], [C18], [C19], [S1], [J6], [J4].

3.3.2 New evolutionary approaches for hierarchical and hierarchical-interpolative fuzzy architectures

I proposed a new schema for representing and modifying the hierarchical fuzzy rule bases in individuals, i.e. the encoding of the rule bases, and the manners how the steps of the optimization algorithms modify the candidate solutions in order to achieve as optimal knowledge base as possible.

In the optimization processes the trapezoidal fuzzy sets of the rules are always represented by their characteristic points (i.e. the positions of their vertices).

In evolutionary programming techniques the candidate solutions can be represented by *expression trees* [12], [13]. The expression trees comprise two type of nodes: *non-terminal (inner)* and *terminal (leaf)* nodes.

The non-terminal nodes hold the antecedent parts of one or more rules. In case of each non-terminal node the number of considered input dimensions (number of *decision variables*) is a random value, whose maximum can be parameterized. A decision variable can only be such a dimension, which has not been decision variable in the ancestors of the current node. (This way the size of the expression tree becomes limited.) Henceforth, these decision variables (i.e. the ones appearing in no ancestors of the node) will be called as *free decision variables*. The consequent part of a rule is a child node of the particular node, which can be either a non-terminal or a terminal node. Thus, the number of rules a non-terminal node holds is determined by the number of the children of the particular node. The number of children of the nodes is a random value, whose maximum, i.e. the maximum number of rules belonging to a node, can be parameterized.

Each terminal node (leaf) holds one single fuzzy set, as a conclusion (output) set.

There can be non-terminal nodes as well as terminal nodes on the same level, hence meta-rules and rules can appear together.

I introduced memetic programming techniques as evolutionary programming algorithms (e.g. [12], [5]) containing the following additional local search operator.

Before local search steps, all numerical parameters from all the nodes are collected in a single numerical vector. Then, gradient methods are applied on this vector. During the objective function evaluations within the local search steps the elements of the vector are written into the expression tree to the corresponding places temporarily. This way the individual can be evaluated. At the end of the local search phases, if the gradient steps caused improvements, the new vector is written into the candidate solution again for the evolutionary operators, otherwise the original values are kept in the nodes. Since any of the numerical parameters might be changed by the local search, the whole individual (obviously, except for the structure) is optimized when this operation is performed.

Thesis statement 3.2. *I proposed a pair of new dense hierarchical and hierarchical-interpolative rule base representation techniques suitable for evolutionary programming optimization methods; furthermore I introduced memetic programming methods combining evolutionary programming algorithms with local search techniques.*

The new encoding methods are based on expression trees. The non-terminal nodes represent rule bases with only antecedent parts, whereas the terminal nodes hold the consequent sets of the fuzzy rules.

As local search steps I proposed a technique collecting the numerical parameters from the nodes of the expression tree to a numerical vector and performing gradient steps based on this vector. I applied this method as an operator within evolutionary programming techniques, which approach formed memetic programming algorithms.

Related works: [C5], [C4], [C10], [J3], [J4], [C11], [J5].

3.3.3 Application of the new approaches in hierarchical and hierarchical-interpolative fuzzy systems

Based on the previous point I applied Genetic Programming (GProg) [12], Bacterial Programming (BProg) [5] and their memetic variants for constructing hierarchical and hierarchical-interpolative

fuzzy rule bases with the following considerations.

Since a decision variable cannot be such a dimension, which has already been a decision variable in any ancestor of the current node, in case of crossover in GProg the following conflict of decision variables may easily arise. Consider two individuals being combined by the crossover operator. Assume that a particular variable x_i appears below the crossover point in one of the individuals and that the same decision variable x_i appears above the crossover point in the other one. In this case the operator results in an expression tree, where the same decision variable x_i appears on different levels corrupting the encoding schema (discussed above). In order to resolve this conflict, besides keeping the structure of the sub-tree, the transferred nodes change their decision variables so that the new decision variables be such dimensions, which have not been decision variables in the ancestors. This introduces a necessary restriction, namely, the number of decision variables in the transferred sub-tree must not be more than the number of free decision variables in the receiver tree at the crossover point. Thus, when selecting a crossover point this restriction must be regarded.

The same problem arises for gene transfer operator in BProg, where the same solution of this technical difficulty is applied.

In case of mutation in GProg and bacterial mutation in BProg there are no technical difficulties; only new sub-trees or nodes must be generated with the consideration of the encoding rules described in the previous point.

As inference algorithms I applied simple extensions of the Mamdani-inference and the Stabilized KH-interpolation methods. These extensions are necessary, because there are child nodes in the consequent parts of the rules. Since the conclusions of sub-trees are fuzzy sets (specially, the contents of terminal nodes can also be considered as conclusions), they can be substituted into the consequent parts of the rules of the parent nodes. This can be performed recursively.

I carried out simulation runs on five different machine learning benchmark problems (Friedman, Daily electricity, Nawa-Furuhashi, Stock prices and Treasury data sets) [2], [17], in order to demonstrate the applicability of the established fuzzy rule based learning systems, furthermore to compare them with each other from the point of view of speed of learning and accuracy of the resulting knowledge base.

In the experiments bacterial programming techniques outperformed the corresponding genetic programming ones and BProg appeared to be the most appropriate technique to combine with hierarchical systems (see e.g. the results for the Nawa-Furuhashi problem in [Figure 3](#)).

Thesis statement 3.3. *I applied Genetic Programming (GProg), Bacterial Programming (BProg) and their respective memetic variants (GProg+SD, BProg+SD, GProg+LM and BProg+LM), where Steepest Descent (SD) and Levenberg-Marquardt (LM) techniques were used as local search methods, together with the newly proposed encoding method for establishing hierarchical and hierarchical-interpolative fuzzy rule based knowledge extraction systems. I compared and evaluated the established systems on a variety of benchmark problems, and I provided experimental evidence for the superiority of bacterial techniques.*

Related works: [C5], [C4], [C10], [J3], [J4], [C11], [J5].

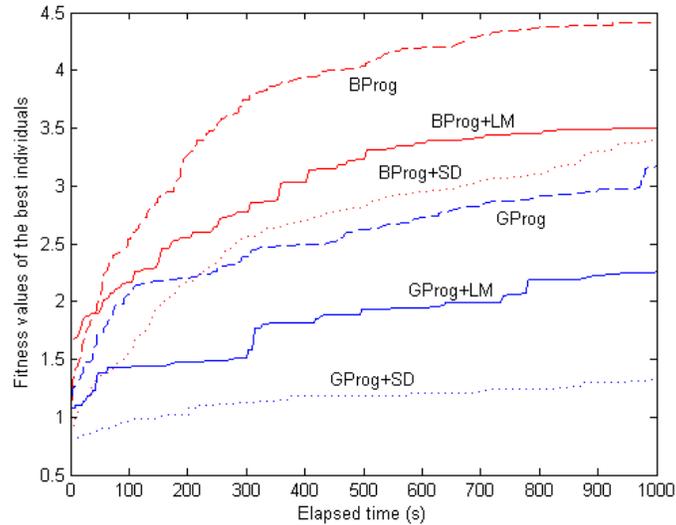


Figure 3: Fitness values for the Nawa-Furuhashi benchmark problem

3.4 New meaning preservation approaches for constructing fuzzy systems with adjustable interpretability-accuracy trade-off

Thesis statement group 4. *I proposed a new meaning preservation approach and a new parameterizable search space narrowing technique for evolutionary algorithms optimizing fuzzy knowledge bases, which make it possible to adjust between accuracy and interpretability.*

Related works: [C12], [C15], [J2].

3.4.1 A new meaning preservation approach for interpretable fuzzy rule bases

Due to the trade-off between interpretability and accuracy, conventional approaches have the following huge disadvantage. Depending on the resulting rule base of the learning process, totally different sets can be labeled with the very same linguistic terms, i.e. the vocabulary applied is not persistent throughout the wide range of problems. It might even occur, when two partitions of the input space in two contexts differ essentially, that the same sets are labeled with different linguistic terms.

However, even if the sets belonging to the linguistic terms were defined exactly and thus if the terms denoted the same sets in each resulting rule base, i.e. if the vocabulary was persistent, the interpretation of the result could be significantly different for one human and for another, because due to the ambiguity of natural languages the meaning of a natural language term may differ for different people.

This is the reason why I proposed a new, personalized approach for constructing interpretable fuzzy systems. The main idea of this approach is to use the linguistic terms in the same sense as

the user uses them, i.e. to have a *common vocabulary* with the user. For example, if before the interpretation the rule was “if the weather is 30°C then turn on the air conditioning”, the interpretation should say “if the weather is very hot then turn on the air conditioning” to the one living in the frigid zone, whereas it should say “if the weather is quite warm then turn on the air conditioning” to the one living in the intertropical zone.

As interpretability means that the knowledge is formulated in a manner that makes the information directly understandable for the user, the easiest way to meet the requirements of interpretability is to hold the information in a representation being familiar to the user. Trivially, such representations can be the natural languages. But there are some problems with them due to their imprecision. If people hear or read something being formulated in a natural language, they associate a meaning to the heard or read text. However, one can associate something, whereas the other one can associate something else, because there are no exact definitions of phrases in natural languages.

In order to use the terms in the same sense as the user uses them, the user must be *interviewed*. A simple interview could be to ask the user to define adjectives as fuzzy sets. However, supposing someone not being familiar with fuzzy sets at all (which is a rather realistic assumption), the interview can be worked out by using fuzzy membership elicitation techniques (see e.g. [8]). Then, a simple interview could be e.g. a survey. For example, to get to know what the user means under ‘hot’, a sequence of questions could be asked: “How much do you feel 15, 20, 25, 30, 35, 40 degrees ‘hot’ (but not ‘very hot’) on a 0 – 5 range (0 – not at all, 5 – totally)?” After that, the adequate fuzzy sets can be constructed easily.

Linguistic terms may not involve only adjectives (e.g. ‘hot’), but modifiers, so-called linguistic hedges, too (e.g. ‘a bit’, or ‘very’). These modifiers can be considered to be transformations of the sets of the adjectives being under modification. Therefore, if the user is interviewed about ‘cold’, ‘hot’ and about how the user modifies the meaning of an adjective if it is combined with the linguistic hedge ‘very’, the meanings of ‘very cold’ and ‘very hot’ need not be interviewed, because they can be computed by applying the transformation of ‘very’ on the fuzzy sets of ‘cold’ and ‘hot’. This may lead to a complexity reduction. (Obviously, the transformations should be carefully defined based on a well-designed interview, because e.g. in case of ‘very’ shifting the certain sets with a positive value may be suitable for ‘hot’, but it is surely not a proper action for ‘cold’.)

The whole procedure may work in reverse, too. Instead of interviewing, the user could be *trained*, i.e. the user could be told about the meaning of certain terms (adjectives and modifiers).

Based on the user defined linguistic terms fuzzy rules and rule bases can be constructed easily. However, not every rule base constructed from these terms will fulfill the interpretability conditions, and thus not all of them will be interpretable (e.g. due to lack of consistency). These ones will be called invalid, whereas the ones fulfilling the interpretability conditions can be referred to as valid *interpretable solutions*.

Thesis statement 4.1. *I proposed a new meaning preservation approach for constructing interpretable fuzzy rule bases. The main idea of this approach is to use the linguistic terms in the same sense as the user uses them, i.e. to have a “common vocabulary” with the user.*

In order to use the terms in the same sense as the user does, the user can be “interviewed”. A

simple interview could be to ask the user to define adjectives as fuzzy sets or it can be worked out by using fuzzy membership elicitation techniques.

Linguistic terms may not involve only adjectives (e.g. ‘hot’), but modifiers, so-called linguistic hedges, too (e.g. ‘a bit’, or ‘very’). These modifiers can be considered to be transformations of the sets of the adjectives being under modification.

The whole procedure may work in reverse, too. Instead of interviewing, the user could be “trained”, i.e. the user could be told about the meaning of certain terms (adjectives and modifiers). It could be done by using the reverse of the procedure of the interview.

Related works: [C12], [J2].

3.4.2 A new search space narrowing technique for evolutionary algorithms applied for constructing interpretable fuzzy rule bases

The interpreted information can be characterized by a finite, but in practice, a limited amount of features, because a human can deal with only a limited number of information units. Furthermore, a human cannot distinguish between units of information being too close to each other in meaning, i.e. the granularity of distinguishable information is not infinitely small, and hence the space of possible solutions is bounded. Thus, the set of interpretable solutions will be considered finite and will be denoted by X_0 , hereafter.

If a fuzzy system is constructed from samples by applying supervised machine learning techniques and interpretability is the main objective of this process, the task of learning is to determine an $x_0^* \in X_0$, such that $\forall x \in X_0 : A(x_0^*) \geq A(x)$, where $A(\cdot)$ is the measure of (relative) accuracy, which is a strictly monotonic decreasing function of the error, which can be calculated e.g. based on the differences between the outputs of the system and the desired outputs. This x_0^* can be achieved by global searching numerical optimization algorithms after a sufficient time. The result of the learning process is the most accurate knowledge base among interpretable solutions. Clearly, the stress is on interpretability in this case.

Let X_∞ denote the largest considerable set of parameter vectors of the particular fuzzy rule base. If a fuzzy system is constructed from samples by applying supervised machine learning techniques and accuracy is the main objective of this process, the task of learning is to determine an $x_\infty^* \in X_\infty$, such that $\forall x \in X_\infty : A(x_\infty^*) \geq A(x)$, where $A(\cdot)$ is the same (relative) accuracy function as it was above. This x_∞^* can be approximated with arbitrary accuracy by global searching numerical optimization algorithms (recall that global search methods stochastically converge to the global optimum). The result of the learning process is the most accurate knowledge base regardless of interpretability.

It is obvious that if a sequence of search spaces being nested into each other $X_0 \subset X_{r_1} \subset X_{r_2} \subset \dots \subset X_{r_n} \subset X_\infty$ (where the sequence of r_i indices is a strictly monotonic increasing sequence) is defined, then it has a positive probability that an optimal solution in a broader space has higher accuracy than all the elements of a narrower space.

However, if $r_i > 0$ and $x_{r_i}^* \notin X_0$ (where $x_{r_i}^*$ is the optimal solution within X_{r_i}), an interpretation can also be given, if there is an interpreter function $\mathfrak{I} : X_\infty \mapsto X_0$, such that $\mathfrak{I}(x_{r_i}^*)$ is somehow the

‘closest’ element from X_0 to $x_{r_i}^*$, i.e. $\forall x \in X_0 : d(x_{r_i}^*, \mathcal{I}(x_{r_i}^*)) \leq d(x_{r_i}^*, x)$, where $d : X_\infty \times X_\infty \mapsto \mathbb{R}^+ \cup \{0\}$ is a translation invariant metric. That is, the interpretation of a solution $x_{r_i}^* \notin X_0$ is the closest interpretable solution $x_0 \in X_0$ to $x_{r_i}^*$ according to an arbitrarily fixed translation invariant distance function.

It is clear that $\mathcal{I}(x_\infty^*)$ can never be more accurate than x_0^* by definition as well as x_0^* can never be more accurate than x_∞^* .

This shows (matching intuitive expectations) that interpretability and accuracy are conflicting requirements: if an interpretable knowledge base is constructed, it is less accurate, and if a more accurate one is constructed, expectedly, after interpretation it becomes less accurate than if interpretation had been the main, and accuracy had only been a secondary objective.

These conflicting approaches can be combined with different weights to intermediate approaches if both the accuracy of the non-interpreted knowledge base and the accuracy of the interpreted one are important. Such combinations can be achieved by narrowing the search space of possible knowledge bases and producing a sequence of nested search spaces $X_0 \subset X_{r_1} \subset X_{r_2} \subset \dots \subset X_{r_n} \subset X_\infty$. For example, if the search spaces are unions of solid hyper-spheres (i.e. hyper-balls) constructed around all the elements of X_0 , where the hyper-balls have the same radius values, furthermore a broader and a narrower search space differ from each other only in the radius value.

It would be greatly favorable, if, as a benefit, there was a tendency showing that the interpreted solution was expectedly more accurate in case of a narrower search space, because this way by choosing a narrower search space from the sequence, although, the accuracy of the non-interpreted knowledge base would be lower, the accuracy of the interpreted knowledge base would be higher. Therefore, roughly speaking, one could balance between interpretability and accuracy by selecting the proper search space.

Thesis statement 4.2. *I proposed a new search space narrowing technique for evolutionary algorithms, with which interpretable fuzzy systems can be constructed so that it is possible to adjust between accuracy and interpretability.*

Let $X_r := \{x \in X_\infty \mid \exists x_0 \in X_0 : d(x, x_0) < r\} \subseteq \mathbb{R}^n$ be the “narrowed search space” w.r.t. a given radius $r \in \mathbb{R}_0^+ \cup \{\infty\}$ and a given translation invariant metric d , where X_0 denotes the set of interpretable rule bases and X_∞ denotes the set of possible rule bases.

In this case with the application of the sequence of nested search spaces $X_0 \subset X_{r_1} \subset X_{r_2} \subset \dots \subset X_{r_n} \subset X_\infty$ it is possible to adjust between accuracy and interpretability.

Related works: [C12], [C15], [J2].

3.4.3 Formal and experimental analysis of the narrowing technique

Based on a stochastic mathematical model I successfully verified the mentioned favorable property formally.

Theorem 1. *Assume that the search spaces are defined as above. Then by choosing a narrower search space from the sequence, although, the expected accuracy of the non-interpreted knowledge base is lower, the expected accuracy of the interpreted knowledge base is higher.*

The following theorem highlights the reason why the closest interpretable solution is proposed to be chosen during interpretation.

Theorem 2. *Assume that the search spaces are defined as above. Then in case of arbitrary x_r^* the closest interpretable solution gives the highest expected accuracy.*

In order to demonstrate the applicability of the proposed approaches and to confirm experimentally that the discussed favorable property holds, I carried out simulation runs.

For this purpose the so-called pH problem was applied. The results of the learning processes applying Mamdani-inference and Stabilized KH-interpolation techniques are collected in [Table 3](#) and in [Table 4](#), respectively.

In the tables the values marked with * are theoretically wrong ones, because the optimal interpretable solution cannot be worse than any other interpretable solutions (cf. e.g. [Table 3](#), Maximum-metric, radius 0), furthermore the optimum point of a search space cannot be less accurate than the optimum of one of its subsets (cf. e.g. [Table 4](#), Manhattan-metric, radius 0.1). These results originate from the insufficient quasi-optimum achieved by the optimization process. Hence they are considered as invalid results, and thus, they are omitted from the analyzation of the observations.

It can be clearly seen that except the results marked with †, i.e. in 31 out of 33 valid cases, $\text{MSE}(x_{r_i}^*) \geq \text{MSE}(x_{r_j}^*)$ and $\text{MSE}(\mathcal{J}(x_{r_i}^*)) \leq \text{MSE}(\mathcal{J}(x_{r_j}^*))$ hold, if $r_i \leq r_j$.

Since accuracy is a monotonic decreasing function of the error, if $r_i \leq r_j$, then in 31 out of 33 valid cases $A(x_{r_i}^*) \leq A(x_{r_j}^*)$ and $A(\mathcal{J}(x_{r_i}^*)) \geq A(\mathcal{J}(x_{r_j}^*))$ also hold.

Based on the experimental results it can be stated that the simulation results are convincing and they confirm the theoretically expected properties of the proposed approaches for constructing interpretable fuzzy systems.

Table 3: Results for the pH problem in case of Mamdani-inference technique

radius(r)	Manhattan-metric		Euclidean-metric		Maximum-metric	
	$\text{MSE}(x_r^*)$	$\text{MSE}(\mathcal{J}(x_r^*))$	$\text{MSE}(x_r^*)$	$\text{MSE}(\mathcal{J}(x_r^*))$	$\text{MSE}(x_r^*)$	$\text{MSE}(\mathcal{J}(x_r^*))$
0	$4.71 \cdot 10^{-3}$	$4.71 \cdot 10^{-3}$	$4.71 \cdot 10^{-3}$	$4.71 \cdot 10^{-3}$	* $4.98 \cdot 10^{-3}$	* $4.98 \cdot 10^{-3}$
0.01	$4.06 \cdot 10^{-3}$	$5.31 \cdot 10^{-3}$	$3.46 \cdot 10^{-3}$	$5.31 \cdot 10^{-3}$	$2.74 \cdot 10^{-3}$	$5.31 \cdot 10^{-3}$
0.02	$3.64 \cdot 10^{-3}$	† $2.26 \cdot 10^{-2}$	$2.73 \cdot 10^{-3}$	$5.31 \cdot 10^{-3}$	$1.64 \cdot 10^{-3}$	$5.31 \cdot 10^{-3}$
0.05	$3.26 \cdot 10^{-3}$	$5.31 \cdot 10^{-3}$	$1.82 \cdot 10^{-3}$	$5.31 \cdot 10^{-3}$	$2.81 \cdot 10^{-4}$	† $4.80 \cdot 10^{-2}$
0.1	$1.96 \cdot 10^{-3}$	$1.13 \cdot 10^{-2}$	$5.60 \cdot 10^{-4}$	$1.58 \cdot 10^{-2}$	$6.90 \cdot 10^{-5}$	$2.16 \cdot 10^{-2}$
0.2	$7.40 \cdot 10^{-4}$	$1.58 \cdot 10^{-2}$	$2.17 \cdot 10^{-4}$	$1.82 \cdot 10^{-2}$	* $1.48 \cdot 10^{-4}$	* $2.73 \cdot 10^{-2}$

Thesis statement 4.3. *I formally proved that the newly proposed search space narrowing technique possesses the desired adjusting properties if certain assumptions hold true, that is:*

If $0 \leq r_1 < r_2 \leq R$ (where R is the “critical radius”) then

$$\mathbb{E}A(\mathcal{J}(x_{r_2}^*)) < \mathbb{E}A(\mathcal{J}(x_{r_1}^*)) \leq \mathbb{E}A(x_{r_1}^*) < \mathbb{E}A(x_{r_2}^*).$$

Table 4: Results for the pH problem in case of Stabilized KH-interpolation method

radius(r)	Manhattan-metric		Euclidean-metric		Maximum-metric	
	$\text{MSE}(x_r^*)$	$\text{MSE}(\mathcal{J}(x_r^*))$	$\text{MSE}(x_r^*)$	$\text{MSE}(\mathcal{J}(x_r^*))$	$\text{MSE}(x_r^*)$	$\text{MSE}(\mathcal{J}(x_r^*))$
0	$1.42 \cdot 10^{-2}$	$1.42 \cdot 10^{-2}$	$1.42 \cdot 10^{-2}$	$1.42 \cdot 10^{-2}$	$1.42 \cdot 10^{-2}$	$1.42 \cdot 10^{-2}$
0.01	$1.40 \cdot 10^{-2}$	$1.42 \cdot 10^{-2}$	$1.39 \cdot 10^{-2}$	$1.42 \cdot 10^{-2}$	$1.32 \cdot 10^{-2}$	$1.46 \cdot 10^{-2}$
0.02	$1.40 \cdot 10^{-2}$	$1.47 \cdot 10^{-2}$	$1.36 \cdot 10^{-2}$	$1.46 \cdot 10^{-2}$	$1.20 \cdot 10^{-2}$	$1.46 \cdot 10^{-2}$
0.05	$1.12 \cdot 10^{-2}$	$1.55 \cdot 10^{-2}$	$1.22 \cdot 10^{-2}$	$1.46 \cdot 10^{-2}$	$8.46 \cdot 10^{-3}$	$1.55 \cdot 10^{-2}$
0.1	* $1.18 \cdot 10^{-2}$	* $1.47 \cdot 10^{-2}$	$9.24 \cdot 10^{-3}$	$1.55 \cdot 10^{-2}$	$7.19 \cdot 10^{-3}$	$1.55 \cdot 10^{-2}$
0.2	$1.02 \cdot 10^{-2}$	$1.69 \cdot 10^{-2}$	$7.90 \cdot 10^{-3}$	$1.55 \cdot 10^{-2}$	$6.23 \cdot 10^{-3}$	$3.12 \cdot 10^{-2}$

The equality in the middle holds exactly when $r_1 = 0$.

I confirmed the formally proved adjusting properties on a variety of benchmark problems experimentally.

Related works: [C15], [J2].

3.5 Adaptive scheduling of optimization algorithms

Thesis statement group 5. I proposed a new adaptive scheduling schema for adaptively switching between optimization algorithms during an optimization process.

Related works: [C9], [C14], [C13].

3.5.1 The problem of adaptive scheduling in optimization processes

There are a huge number of numerical optimization algorithms known from the literature. A large part of them is formed by the iterative ‘oracle-based’ (also known as ‘black box’) techniques that evaluate the objective function in each iteration to compute new states. They can be used in the field of complex optimization problems, because these techniques claim only a few assumptions about a given problem, thus they are rather general algorithms. A part of them are invented intuitively (e.g. Steepest Descent [21] and Levenberg-Marquardt [14], [16] methods), another part of them are inspired by natural processes (e.g. Genetic Algorithm [9] and Bacterial Evolutionary Algorithm [17]). However, there is a huge cost of this generality: there are no exact results which technique is how efficient in general, or even in a particular problem field at all.

Therefore, there are only heuristics for deciding which ones to use and how to parameterize them. These are based on intuition and simulation results (e.g. [17]).

There are often big differences in the difficulty of the algorithms, which result in different type of characteristics. A simpler technique can be faster, but less efficient and a more difficult one can be much slower, but much more efficient from iteration to iteration.

Often, in the early part of the optimization it is easy to reach better and better states in the problem space, but after a long iteration period it is quite difficult to find a better state. Thus, simpler algorithms can be useful at the beginning (as a more global search) due to their higher iteration speed,

whereas more difficult algorithms can be a better choice at the end (as a more local search) due to their higher efficiency.

Therefore, it would be often useful to apply simpler algorithms at the beginning and more difficult ones at the end of a numerical optimization process. More generally, it would be desired to find out which optimization algorithm to use on a particular part of a problem. (This problem is more general than “Which algorithm should be applied on an optimization problem?”). Rephrasing the desire: an effective schedule of the optimization algorithms should be found adaptively to the particular numerical optimization problem. (Obviously, not only optimization algorithms can be scheduled, but whole optimization architectures, or simply parameterizations, too.)

This problem defines a decision tree, where the levels, the edges and the vertices represent iteration levels, executions of optimization algorithms and states, respectively. It can be explored totally or partially. The former has a huge computational demand (exponential), the latter has lower (either linear), depending on the size of the part.

Thesis statement 5.1. *I introduced and formally defined the problem of adaptive scheduling of optimization algorithms.*

It is often useful to apply simpler algorithms at the beginning and more difficult ones at the end of a numerical optimization process. Generalizing this idea, it would be desired to find out which optimization algorithm to use for a particular part of a problem. That is, an effective schedule of the optimization algorithms should be found adaptively to the particular numerical optimization problem.

Related work: [C9].

3.5.2 Scheduler algorithms

I proposed the following three scheduling methods for solving the above described problem.

Greedy Scheduler. *Greedy scheduler (GS)* executes all the algorithms simultaneously, and after each iteration (or time slot) the currently best candidate solution is selected according to its objective function value. In case of population based techniques (like BEA or BMA) the best population is selected based on some quality measure, e.g. the fitness value of the best individual. In the subsequent iteration (or time slot) every algorithm is initialized to the selected candidate solution or population and executed simultaneously. Then, another comparison is performed, and so forth.

The seemingly hopeless drawback of this simple scheduler is its huge overhead originating from the parallel execution of all algorithms during the whole optimization process.

Fast Greedy Scheduler. Sometimes the optimization problem has favorable properties and particular optimization algorithms can perform best during long periods. In this case the number of switches between the algorithms in an optimal schedule may be low.

In order to exploit this advantage of such problems, I proposed another version of the above discussed Greedy Scheduler. If the scheduling problem has the mentioned favorable property, this scheduling method is faster than the previous one.

The *Fast Greedy Scheduler (FGS)* does not compare the optimization algorithms in each step, i.e. it does not execute them simultaneously all the time, but only after a predefined ‘*blind running*’ time, while it applies only the last locally best algorithm.

This way the mentioned drawback of the previous scheduler can be eased.

Monotonic Fast Greedy Scheduler. If it can be assumed that the characteristics, i.e. the convergence speeds in terms of fitness level, are monotonic decreasing functions, then the following straightforward improvement can be applied to Fast Greedy Scheduler, resulting in the approach *Monotonic Fast Greedy Scheduler (MFGS)*.

After each blind running phase the convergence speed of the active algorithm is compared to the last calculated convergence speeds of the inactive ones, which are the values calculated at the end of the last performed comparing phase. If the convergence speed of the active method is still greater than or equal to all the other values, then at the current fitness level the active algorithm must still be the most efficient one due to the monotonicity of the characteristics assumed. In this case the subsequent comparing phase is postponed and the blind running continues. During the remaining part of this lengthened phase the convergence speed is continuously compared to the last calculated convergence speeds of the inactive methods. When the efficiency of the active algorithm decreases below the highest last measured efficiency level of the inactive methods the postponed comparing phase takes place, and then the algorithms are executed simultaneously for a short period.

With this improvement MFGS clearly reduces the comparing overhead of the schedulers further.

Thesis statement 5.2. *I proposed three new scheduler algorithms for adaptively switching between optimization algorithms during an optimization process. Furthermore, I discussed the efficiency of the newly proposed schedulers in different circumstances and the relation between the techniques.*

Related works: [C9], [C14], [C13].

3.5.3 Experimental analysis of optimization processes applying the proposed schedulers

I carried out simulation runs on benchmark problems in order to demonstrate the usability of the recently discussed theory. They show the efficiency of MFGS during fuzzy rule based learning applied for the Friedman function, Stock prices and Treasury data set [2].

The learning architectures applied Mamdani- and SKH-inference methods, furthermore Bacterial Evolutionary Algorithm (BEA) and Bacterial Memetic Algorithm (BMA) as optimization algorithms to find proper parameters for the fuzzy rules.

Observing the results given by the different fuzzy systems, perhaps the most obvious fact is that MFGS was outperformed in the five as well as in the nine dimensional cases (Friedman function and Stock prices data set, respectively), however, for the fifteen dimensional problem (Treasury data set) MFGS was the most efficient (in case of both Mamdani- and SKH-inference methods). The explanation of these results is the following.

For the simpler problem it is not worth using the scheduling technique, because its overhead deteriorates the advantage gained by the possibility of adaptively switching to the currently best performing optimization algorithm.

However, in case of the more difficult problem the advantage of adaptively switching is higher than the disadvantage of the overhead originating from the comparing requirements, and thus the fuzzy systems involving the scheduling approach outperform the other ones (cf. Figure 4, where the results for the Treasury data set is presented in case of the SKH-inference technique).

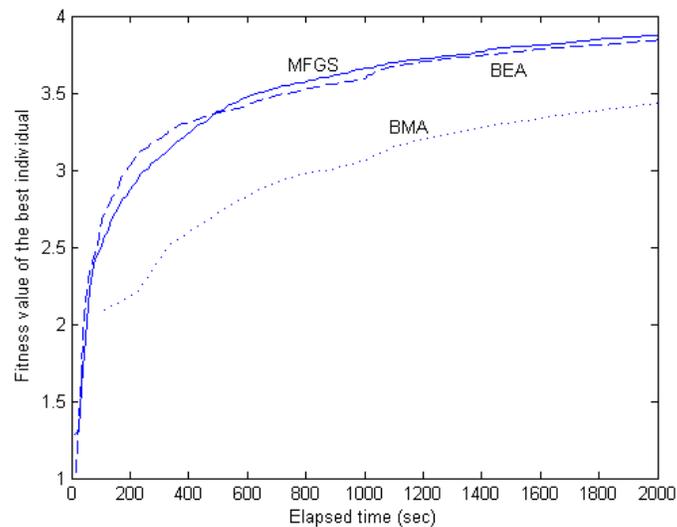


Figure 4: Fitness values for the Treasury data set in case of SKH-interpolation.

Looking at the results, one might have a presumption that BMA was overwhelmed in contrast with the conclusion of Subsubsection 3.3.3. However, after a careful study it can be noticed that at the end of the optimization process the fitness curve of BMA is the steepest foreshadowing an intersection with the curve of BEA. Indeed, in Subsubsection 3.3.3 BMA showed a “slowly but surely” behavior, and if BMA had enough time for running, it outperformed any other techniques.

Thesis statement 5.3. *I compared and evaluated optimization processes applying and not applying the newly proposed schedulers, and I provided experimental evidence for the improved efficiency of optimization processes applying the newly proposed schedulers in case of more difficult problems.*

Related works: [C14], [C13].

4 Application of the results

The new results are beneficial in theoretical operations research, discrete and continuous optimization and machine learning basic research as well as in industrial technical and information technological fields, like production optimization, routing, resource management, or knowledge extraction from data.

With the application of the results related to **Thesis statement group 1** for production lines the throughput of the production processes can be increased by the advanced optimization properties of the new methods.

The advantages of the approaches proposed related to **Thesis statement group 2** are the capability of better modeling uncertain business constraints and dealing with other practical restrictions, with which other existing methods cannot cope.

The results related to **Thesis statement group 3** and **Thesis statement group 4** can be applied for extending the quantitative (accuracy) and qualitative (interpretability) properties of fuzzy rule based machine learning components in artificial intelligent systems.

Finally, due to their rather general nature, the approaches proposed related to **Thesis statement group 5** are able to increase the performance and quality in all the mentioned fields.

I have already applied my results in the frame of several scientific basic researches as well as industrial research-and-development projects, where I successfully solved process optimization for production lines, resource assignment and machine learning tasks.

To mention some already published application examples: production efficiency improvement of production lines by adequately sorting the jobs being processed (**Thesis statement group 1**) [J1], scheduling of drivers onto mass transportation vehicles (**Thesis statement group 2**) [C6], failure localization in optical networks (**Thesis statement group 2**) [C17], extraction of interpretable rules regarding the consumption of vehicles from data sets containing consumption, mass, power and other features of the vehicles (**Thesis statement group 3** and **Thesis statement group 4**) [J2].

Related works

Journal publications

- [J1] K. Balázs, Z. Horváth, and L. T. Kóczy. Different chromosome based evolutionary approaches for the permutation flow shop problem. *Acta Polytechnica Hungarica*, 2(2):115–138, 2012. Estimated impact factor: **0.385**.
- [J2] K. Balázs and L. T. Kóczy. New meaning preserving approaches for constructing fuzzy systems with adjustable interpretability-accuracy trade-off. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*. Submitted for publication.
- [J3] K. Balázs and L. T. Kóczy. Fuzzy szabályalapú modellek és rendszerek felépítése evolúciós technikák segítségével. *Híradástechnika*, 66:44–50, 2011. In Hungarian.

- [J4] K. Balázs and L. T. Kóczy. Constructing dense, sparse and hierarchical fuzzy systems by applying evolutionary optimization techniques. *Applied and Computational Mathematics*, 11(1):81–101, 2012. Estimated impact factor: **0.551**.
- [J5] K. Balázs and L. T. Kóczy. Hierarchical-interpolative fuzzy system construction by genetic and bacterial memetic programming approaches. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 20(supp02):105–131, 2012. Estimated impact factor: **1.781**.
- [J6] Zs. Dányádi, K. Balázs, and L. T. Kóczy. A comparative study of various evolutionary algorithms and their combinations for optimizing fuzzy rule-based inference systems. *Scientific Bulletin of “Politechnica” University of Timisoara, Romania*, 55(69):247–254, 2010.

Article in book series

- [S1] K. Balázs, L. T. Kóczy, and J. Botzheim. Comparative investigation of various evolutionary and memetic algorithms. In I. J. Rudas, J. Fodor, and J. Kacprzyk, editors, *Computational Intelligence in Engineering*, volume 313 of *Studies in Computational Intelligence*, pages 129–140. Springer, 2010.

Conference papers

- [C1] K. Balázs, J. Botzheim, and L. T. Kóczy. Comparative analysis of various evolutionary and memetic algorithms. In *Proceedings of the 10th International Symposium of Hungarian Researchers on Computational Intelligence and Informatics, CINTI 2009*, pages 193–205, Budapest, Hungary, 2009.
- [C2] K. Balázs, J. Botzheim, and L. T. Kóczy. Comparative analysis of interpolative and non-interpolative fuzzy rule based machine learning systems applying various numerical optimization methods. In *World Congress on Computational Intelligence, WCCI 2010*, pages 875–982, Barcelona, Spain, 2010.
- [C3] K. Balázs, J. Botzheim, and L. T. Kóczy. Comparison of various evolutionary and memetic algorithms. In *Proceedings of the International Symposium on Integrated Uncertainty Management and Applications, IUM 2010*, pages 431–442, Ishikawa, Japan, 2010.
- [C4] K. Balázs, J. Botzheim, and L. T. Kóczy. Hierarchical fuzzy system construction applying genetic and bacterial programming algorithms with expression tree building restrictions. In *World Automation Congress, WAC 2010*, pages 1–6, Kobe, Japan, 2010.
- [C5] K. Balázs, J. Botzheim, and L. T. Kóczy. Hierarchical fuzzy system modeling by genetic and bacterial programming approaches. In *World Congress on Computational Intelligence, WCCI 2010*, pages 1866–1871, Barcelona, Spain, 2010.

- [C6] K. Balázs, M. Farkas, and L. T. Kóczy. Proposal on formalizing uncertain business criteria as fuzzy constraints in multi-objective csp optimization. In *International Fuzzy Systems Association World Congress and Asia Fuzzy Systems Society International Conference, IFSA 2011*, pages 1–5, Surabaya, Indonesia, 2011.
- [C7] K. Balázs, Z. Horváth, and L. T. Kóczy. Hybrid bacterial iterated greedy heuristics for the permutation flow shop problem. In *World Congress on Computational Intelligence, WCCI 2012*, pages 1–8, Brisbane, Australia, 2012.
- [C8] K. Balázs, Z. Horváth, and L. T. Kóczy. Multi-threaded bacterial iterated greedy heuristics for the permutation flow shop problem. In *13th IEEE International Symposium on Computational Intelligence and Informatics, CINTI 2012*, pages 63–66, Budapest, Hungary, 2012.
- [C9] K. Balázs and L. T. Kóczy. A remark on adaptive scheduling of optimization algorithms. In *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, IPMU 2010*, pages 719–728, Dortmund, Germany, 2010.
- [C10] K. Balázs and L. T. Kóczy. Hierarchical-interpolative fuzzy system construction by genetic and bacterial programming algorithms. In *IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2011*, pages 2116–2122, Taipei, Taiwan, 2011.
- [C11] K. Balázs and L. T. Kóczy. Genetic and bacterial memetic programming approaches in hierarchical-interpolative fuzzy system construction. In *World Congress on Computational Intelligence, WCCI 2012*, pages 1–8, Brisbane, Australia, 2012.
- [C12] K. Balázs and L. T. Kóczy. New parameterizable search space narrowing technique for adjusting between accuracy and interpretability in fuzzy systems. In *13th IEEE International Symposium on Computational Intelligence and Informatics, CINTI 2012*, pages 323–328, Budapest, Hungary, 2012.
- [C13] K. Balázs and L. T. Kóczy. Adaptive scheduling of optimization algorithms in the construction of interpolative fuzzy systems. In *IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2013*, Hyderabad, India, 2013. To appear.
- [C14] K. Balázs and L. T. Kóczy. Constructing dense fuzzy systems by adaptive scheduling of optimization algorithms. In *International Fuzzy Systems Association World Congress, IFSA 2013*, Edmonton, Canada, 2013. To appear.
- [C15] K. Balázs and L. T. Kóczy. A stochastic model for analyzing the interpretability-accuracy trade-off in interpretable fuzzy systems using nested hyperball structures. In *8th conference of the European Society for Fuzzy Logic and Technology, EUSFLAT 2013*, Milano, Italy, 2013. Accepted for publication.
- [C16] K. Balázs, L. T. Kóczy, and J. Botzheim. Comparison of fuzzy rule-based learning and inference systems. In *Proceedings of the 9th International Symposium of Hungarian Researchers*

on *Computational Intelligence and Informatics, CINTI 2008*, pages 61–75, Budapest, Hungary, 2008.

- [C17] K. Balázs, P. B. Soproni, and L. T. Kóczy. Improving system reliability in optical networks by failure localization using evolutionary optimization. In *IEEE International Systems Conference, SysCon 2013*, Orlando, USA, 2013.
- [C18] Zs. Dányádi, K. Balázs, and L. T. Kóczy. A comparative study of various evolutionary algorithms used for fuzzy rule-based inference and learning systems. In *IEEE International Joint Conferences on Computational Cybernetics and Technical Informatics, ICC3-CONTI 2010*, pages 49–54, Timisoara, Romania, 2010.
- [C19] Zs. Dányádi, K. Balázs, and L. T. Kóczy. Using multiple populations of memetic algorithms for fuzzy rule-base optimization. In *Proceedings of the 11th IEEE International Symposium on Computational Intelligence and Informatics, CINTI 2010*, pages 113–118, Budapest, Hungary, 2010.

References

- [1] S. Ahuja, S. Ramasubramanian, and M. Krunz. Single-link failure detection in all-optical networks using monitoring cycles and paths. *IEEE/ACM Transactions on Networking (TON)*, 17(4):1080–1093, 2009.
- [2] J. Alcalá-Fdez, A. Fernandez, J. Luengo, J. Derrac, S. García, L. Sánchez, and F. Herrera. Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. *Journal of Multiple-Valued Logic and Soft Computing*, 17(2–3):255–287, 2011.
- [3] P. Babarczi, J. Tapolcai, and P.-H. Ho. Srlg failure localization with monitoring trails in all-optical mesh networks. In *International Conference on Design of Reliable Communication Networks, DRCN 2011*, pages 188–195, Krakow, Poland, 2011.
- [4] J. Botzheim, C. Cabrita, L. T. Kóczy, and A. E. Ruano. Fuzzy rule extraction by bacterial memetic algorithms. In *Proceedings of the 11th World Congress of International Fuzzy Systems Association, IFSA 2005*, pages 1563–1568, Beijing, China, July 2005.
- [5] J. Botzheim, C. Cabrita, L. T. Kóczy, and A. E. Ruano. Genetic and bacterial programming for B-spline neural networks design. *Journal of Advanced Computational Intelligence and Intelligent Informatics*, 11(2):220–231, February 2007.
- [6] B. Crawford, C. Castro, and E. Monfroy. A hybrid ant algorithm for the airline crew pairing problem. In *MICAI 2006: Advances in Artificial Intelligence*, volume 4293 of *Lecture Notes in Computer Science*, pages 381–391. Springer, 2006.

- [7] M. Dorigo, V. Maniezzo, and A. Colorni. The ant system: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics—Part B*, 26(1):29–41, 1996.
- [8] D. Dubois and H. M. Prade. *Fundamentals of fuzzy sets*. Springer, New York, 2000.
- [9] J. H. Holland. *Adaption in Natural and Artificial Systems*. The MIT Press, Cambridge, Massachusetts, 1992.
- [10] R. W. Kember. *Fibre Channel: A Comprehensive Introduction*. Northwest Learning Associates, 1998.
- [11] J. Kennedy and R. C. Eberhart. Particle swarm optimization. In *Proceedings of the IEEE International Conference on Neural Networks*, pages 1942–1948, Perth, Australia, 1995.
- [12] J. R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA, USA, 1992.
- [13] J. R. Koza. *Genetic Programming II: Automatic Discovery of Reusable Programs*. MIT Press, Cambridge MA, USA, May 1994.
- [14] K. Levenberg. A method for the solution of certain non-linear problems in least squares. *Quart. Appl. Math.*, 2(2):164–168, 1944.
- [15] E. H. Mamdani and S. Assilian. An experiment in linguistic synthesis with a fuzzy logic controller. *Int. J. Man-Mach. Stud.*, 7:1–13, 1975.
- [16] D. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *J. Soc. Indust. Appl. Math.*, 11(2):431–441, Jun. 1963.
- [17] N. E. Nawa and T. Furuhashi. Fuzzy system parameters discovery by bacterial evolutionary algorithm. *IEEE Transactions on Fuzzy Systems*, 7(5):608–616, Oct. 1999.
- [18] M. Nawaz, E. E. Ensore Jr., and I. Ham. A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem. *OMEGA, The International Journal of Management Science*, 11(1):91–95, 1983.
- [19] M. G. Ravetti, C. Riveros, A. Mendes, M. G. C. Resende, and P. M. Pardalos. Parallel hybrid heuristics for the permutation flow shop problem. Research technical report, AT&T Labs, 2010.
- [20] R. Ruiz and T. Stützle. A simple and effective iterated greedy algorithm for the permutation flowshop scheduling problem. *European Journal of Operational Research*, 177:2033–2049, 2007.
- [21] J. A. Snyman. *Practical Mathematical Optimization: An Introduction to Basic Optimization Theory and Classical and New Gradient-Based Algorithms*. Springer, New York, 2005.

- [22] E. Taillard. Benchmarks for basic scheduling problems. *European Journal of Operational Research*, 64(2):278–285, 1993.
- [23] J. Tapolcai, B. Wu, and P.-H. Ho. On monitoring and failure localization in mesh all-optical networks. In *IEEE INFOCOM*, pages 1008–1016, Rio de Janeiro, Brasil, 2009.
- [24] D. Tikk, I. Joó, L. T. Kóczy, P. Várlaki, B. Moser, and T. D. Gedeon. Stability of interpolative fuzzy kh-controllers. *Fuzzy Sets and Systems*, 125(1):105–119, 2002.