DOI: https://doi.org/10.46354/i3m.2019.emss.037

A NEW SOLUTION ENCODING FOR SIMULATION-BASED MULTI-OBJECTIVE WORKFORCE QUALIFICATION OPTIMIZATION

Johannes Karder[a], Viktoria A. Hauder[b], Andreas Beham[c], Klaus Altendorfer[d], Michael Affenzeller[e]

[a],[b],[c],[e]Heuristic and Evolutionary Algorithms Laboratory,
University of Applied Sciences Upper Austria, Softwarepark 11, 4232 Hagenberg, Austria
[d]Department of Operations Management,
University of Applied Sciences Upper Austria, Wehrgrabengasse 1–3, 4400 Steyr, Austria
[a],[c],[e]Institute for Formal Models and Verification
Johannes Kepler University, Altenberger Straße 69, 4040 Linz, Austria
[b]Institute for Production and Logistics Management,
Johannes Kepler University, Altenberger Straße 69, 4040 Linz, Austria

[a]johannes.karder@fh-hagenberg.at, [b]viktoria.hauder@fh-hagenberg.at, [c]andreas.beham@fh-hagenberg.at,
[d]klaus.altendorfer@fh-steyr.at, [e]michael.affenzeller@fh-hagenberg.at

ABSTRACT
Solutions for combinatorial problems can be represented by simple encodings, e.g. vectors of binary or integer values or permutations. For such encodings, various specialized operators have been proposed and implemented. In workforce qualification optimization, qualification matrices can for example be encoded in the form of binary vectors. Though simple, this encoding is rather general and existing operators might not work too well considering the genotype is a binary vector, whereas the phenotype is a qualification matrix. Therefore, a new solution encoding that assigns a number of workers to qualification groups is implemented. By conducting experiments with NSGA-II and the newly developed encoding, we show that having an appropriate mapping between genotype and phenotype, as well as more specialized genetic operators, helps the overall multi-objective search process. Solutions found using the specialized encoding mostly dominate the ones found using a binary vector encoding.

Keywords: workforce qualification, encoding, multi-objective optimization, NSGA-II, simulation

1. INTRODUCTION

One of the main key factors for the success of manufacturing companies is qualified staff. Only if a company's personnel is qualified in such a way that all necessary production steps can be carried out on time, a high customer service level with an associated maximized high customer satisfaction is achievable. At the same time, Europe and thus, the European industry, is facing skills shortage, i.e. there is an increasing amount of vacant jobs for which no qualified human resources are available (OECD 2018). This problem, also referred to as skill gap, can of course be counteracted by raising qualification through internal or external trainings which leads to an increased flexibility in the assignment possibilities of workers (De Bruecker, Van den Bergh, Beliën, and Demeulemeester 2015). However, in many cases there is not even enough workforce available that could be hired and then trained, resulting in a major economic challenge for European companies (EUROCHAMBRES 2019; OECD 2018). As a result, the flexibility should be achieved with the smallest possible amount of necessary qualifications.

Precisely these two contradictory objectives, a maximized service level on the one hand and a minimized sum of necessary qualifications on the other hand represent the two major challenges in the long-term production planning of the flow shop system of our company partner. In order to tackle these problems, we propose a simulation-based multi-objective optimization approach with the optimization framework HeuristicLab (HL), consisting of an innovative encoding for the elitist non-dominated sorting genetic algorithm (NSGA-II) (Deb, Pratap, Agarwal, and Meyarivan 2002) and the simulation framework Sim#.

The paper is structured as follows. In Section 2, we give an overview of the related scientific literature. Next, we present our solution method in Section 3, involving the developed new encoding for the applied multi-objective algorithm and the simulation model. In Section 4, we evaluate our proposed method, showing the potentials of our solution encoding for the NSGA-II as an appropriate algorithm for the presented problem. Finally, we conclude our work and give directions for further research in Section 5.

2. RELATED LITERATURE

Workforce planning is a well-researched topic in the area of operations research. It defines how many workers should be hired or dismissed at which point in time and when they should work for how many periods of time. As a result, workforce planning can consist of staffing and scheduling decisions (De Bruecker, Van den Bergh,

Beliën, and Demeulemeester 2015). Related to this, it is also defined that if a worker is able to perform one specific task, he has one specific skill. There are different determinants which influence skills, such as the qualification, the experience or the age of a worker. Moreover, it is noted that skill determinants have a direct impact on so-called skill consequences, such as the quality of work. However, the link between skill determinants and consequences has rarely been observed up to now (De Bruecker, Van den Bergh, Beliën, and Demeulemeester 2015). Exactly this missing link complies with our object of investigation. We present a simulation-based optimization approach to study the impact of worker qualifications (skill determinant) on the customer service level of a company (skill consequence), assuming that the quality of work is expressed by the customer service level of the partner company. For this purpose we develop an innovative new solution encoding and corresponding variation operators that are able to screen the space of possible workforce qualifications. It is noted that we concentrate on the qualifications of workers, since this is the determining skill factor defined by our company partners.
As we consider the two objectives of service level maximization and qualification minimization, we work on a multi-objective optimization problem. Besides well-known solution methods such as the ε-constraint method (Haimes, 1971) or the recently established Balanced Box Method (Boland, Charkhgard, and Savelsbergh 2015), there is the NSGA-II, proposed by Deb, Pratap, Agarwal, and Meyarivan (2002). This algorithm is the method of choice for many multi-objective optimization problems, including a wide variety of real-world applications (Hu, Bie, Ding, and Lin 2016; Wang, Fu, Huang, Huang, and Wang 2017). Although it is a generic metaheuristic solution approach, depending on the specific problem definition, new solution encodings have to be developed. The combination of a simulation model with a metaheuristic optimization method results in the so-called simulation-based optimization or simulation optimization (Gosavi, 2015). With simulation-based optimization, a simulation model can be used as an objective function for approximating the performance of a real-world system. However, it is also possible to use optimization methods during the execution of a simulation. In general, simulation-based optimization is very often used to get an increased real-world system performance (Affenzeller et al. 2015; Kück, Broda, Freitag, Hildebrandt, and Frazzon, 2017; Lin, Chiu, and Chang 2019).

3. MULTI-OBJECTIVE SIMULATION-BASED WORKFORCE OPTIMIZATION

In the following, we explain our developed method. The optimization part takes responsibility for the assignment of workers to qualifications. The simulation part then simulates the flow shop by integrating the optimization solution candidates and evaluates the corresponding customer service level. The new solution encoding for the implemented NSGA-II is presented in Section 3.1. The simulation model is described in Section 3.2.

3.1. New Solution Encoding

In order to optimize the workforce qualifications, we use the optimization framework HeuristicLab (see https://dev.heuristiclab.com), where we implement a new solution encoding. The *Qualification Encoding* consists of two main properties that define the number of possible qualification configurations: i) the number of qualifications Q and ii) the number of workers W that are present in the system that has to be optimized. A configuration is specified by a variable amount of distinct qualification groups and a number of workers assigned to each group. Qualification groups define which qualifications are present. Each worker w must be assigned to exactly one group and each qualification must be present in at least one qualification group. An example of a qualification configuration is shown in Figure 1. In this example, a total of 6 qualifications need to be distributed among 20 workers. Each qualification group is represented as a Boolean vector in which 1 indicates the presence of a qualification and 0 indicates its absence. In this configuration, 3 workers have been assigned to the first qualification group and therefore possess qualifications 2 and 5.

Configuration

010010	3
100100	4
111000	7
000111	6
	20
Qualification Groups	Workers

Figure 1: Example qualification configuration.

The encoding also provides 11 operators to create, cross, and manipulate such qualification configurations. Metaheuristic algorithms can use these operators in their implementations, e.g. to initialize populations or create offspring. All operators are described in Section 3.1.1. Furthermore, the feasibility of configurations is validated and a repair procedure is devised to correct infeasible configurations, as shown in Section 3.1.2.

3.1.1. Operators

In this section, the implemented operators, i.e. 1 solution creator, 5 crossover operators and 5 mutation operators, are presented.

1) Random Qualification Creator (RQC)

This operator is used to create qualification configurations in a random fashion. A configuration contains at least 1 and a maximum of Q different qualification groups. A qualification group is a set of qualifications that are available for all workers in this group. All qualification groups within one configuration must be unique. Furthermore, all workers must be assigned to a qualification group and one worker must not be present in more than one group. On average, each qualification group created contains one qualification. Initially, one worker is assigned to each group, all remaining workers are then distributed among the groups randomly.

2) Union Average Crossover (UAX)

This crossover creates offspring that contain all groups from its parents by averaging and rounding up or down the number of workers in each group. Rounding up and down is done in an alternating fashion.

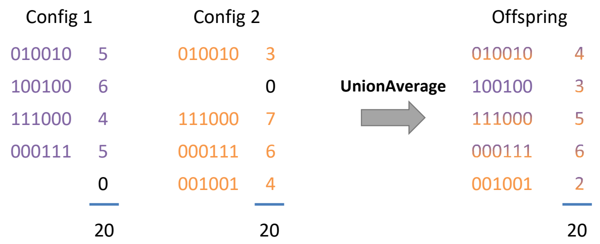

Figure 2: The UAX.

3) Overlap Average Crossover (OAX)

Compared to the UAX, this crossover creates offspring that only contain groups occurring in both parents. For such groups, the number of workers inside these groups is averaged. Workers assigned to groups that are not present in both parents are "lost", i.e. a repair procedure distributes the missing workers among the groups that are present in the offspring. If no common groups are found, a discrete crossover, explained in the following paragraph, is applied.

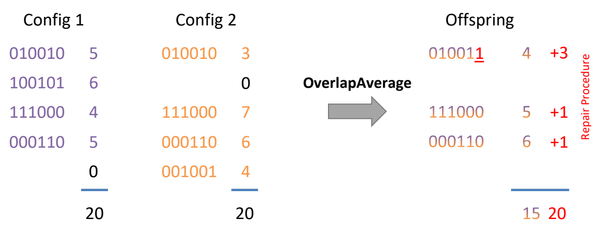

Figure 3: The OAX.

4) Discrete Crossover (DX)

This crossover randomly selects groups from either parent and introduces them in the offspring. A resulting offspring configuration has the same amount of groups as the parent with the least groups. The last group taken from either parent is assigned the number of workers that have not been assigned yet.

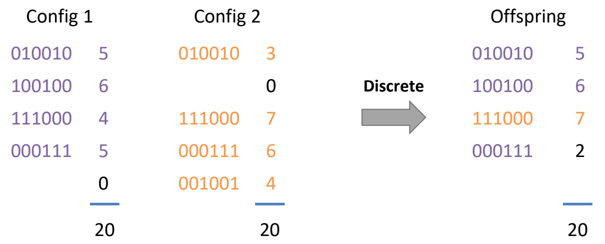

Figure 4: The DX.

5) Set Cover Crossover (SCX)

This crossover aims to find the minimal set of pools present in the parents that covers all qualifications. This so called *Set Cover Problem* is an NP-complete problem itself (Karp 1972). Therefore, a construction heuristic, including a branch & bound algorithm have been implemented to tackle this problem. If the generated offspring is identical to either parent, the DX, as explained before, is applied.

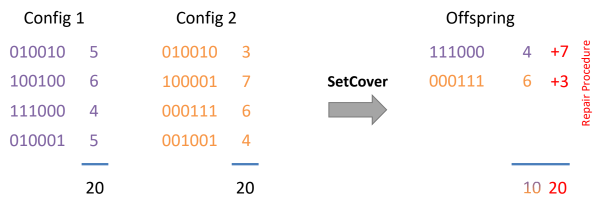

Figure 5: The SCX.

6) Set Pack Crossover (SPX)

The SPX aims to select the greatest number of groups that do not intersect, i.e. that do not have the same qualifications. Finding such sets is equal to the *Set Packing Problem*, which is also NP-complete (Karp 1972). Again, a construction heuristic has been implemented to solve this problem and if the offspring equals either parent, the DX is executed.

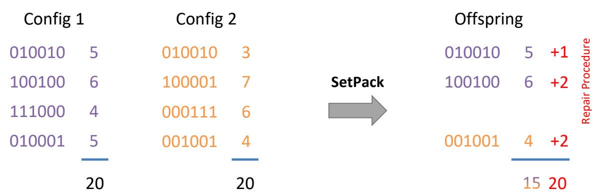

Figure 6: The SPX.

7) Add-Qualification-To-Group Manipulator

This manipulator first selects one group randomly, checks for unset qualifications and aborts if there are none. If unset qualifications are available, one qualification is randomly chosen and set. Finally, the manipulator decides how many workers should be removed from the old group and moved to the new group.

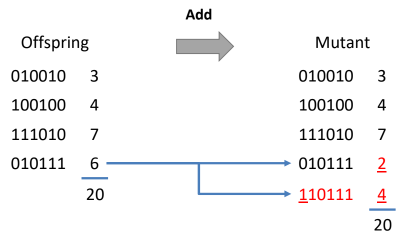

Figure 7: The Add-Qualification-To-Group manipulator.

8) Remove-Qualification-From-Group Manipulator
This manipulator works in principle in the same fashion as the previous one. However, instead of adding a new qualification, it randomly removes one. Again, the number of workers to be transferred from the old to the new group is randomly chosen.

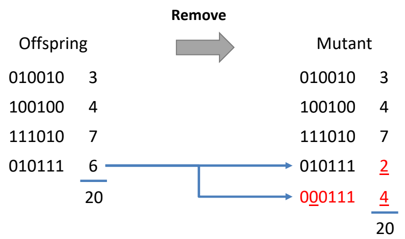

Figure 8: The Remove-Qualification-From-Group manipulator.

9) Swap-Qualification-Within-Group Manipulator
Here, one set and one unset qualification within a randomly chosen group are swapped. The qualifications are chosen randomly and if there are no unset qualifications, the chosen set qualification is removed.

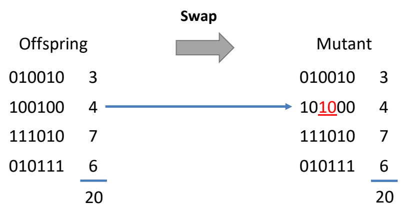

Figure 9: The Swap-Qualification-Within-Group manipulator.

10) Split-Qualification-Group Manipulator
This manipulator splits a randomly chosen qualification group into two disjoint groups. The number of workers are split randomly between the two new groups.

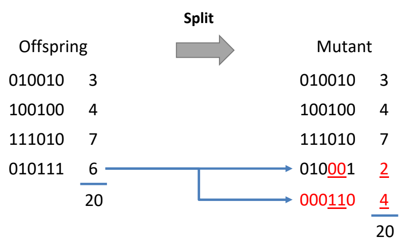

Figure 10: The Split-Qualification-Group manipulator.

11) Merge-Qualification-Group Manipulator
Using this manipulator, qualifications of two randomly chosen groups are merged. All workers from both groups are transferred to the merged group.

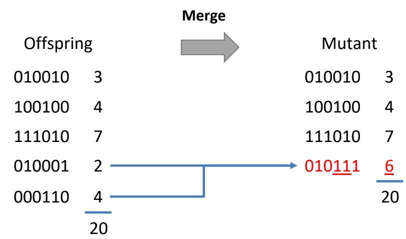

Figure 11: The Merge-Qualification-Group manipulator.

3.1.2. Repair Procedure

The implemented solution encoding also has some constraints that need to be adhered to. First, each qualification must be set in at least one qualification group (see Figure 12).

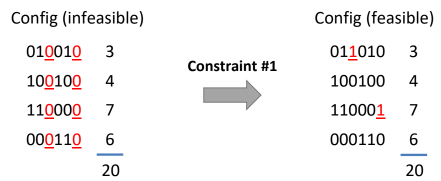

Figure 12: An infeasible configuration, where two qualifications are not present in any qualification group, is repaired.

Second, each worker must have at least one qualification, i.e. each worker must be assigned to exactly one qualification group (see Figure 13).

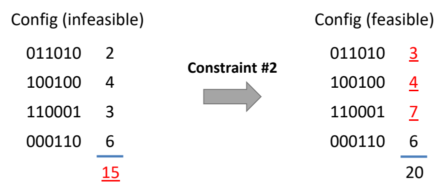

Figure 13: An infeasible configuration, where 5 out of 20 workers are not assigned to any qualification group, is repaired.

Furthermore, a configuration must have unique qualification groups and an empty qualification group (i.e. a group which has either no workers or no qualifications assigned) is not allowed. To ensure these properties, a repair procedure has been implemented and is applied after every solution creation, crossover and mutation operation (except in case of the Merge-Qualification-Group manipulation, which should never yield infeasible configurations).

3.2. Simulated Production System

The simulated production system has been first described by Schober, Altendorfer, Karder, and Beham (2019). We have created an implementation of the described model with the help of the Sim# simulation kernel (see https://github.com/abeham/SimSharp). An overview of this model is depicted in Figure 14.

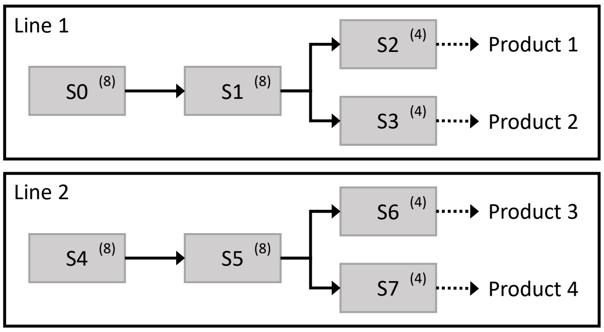

Figure 14: The simulated production system with 2 lines and 4 stations per line, producing 4 different types of products.

Figure 14 shows the production system. All products have to sequentially pass stations from left to right and are split up after the second station between the third and fourth in each line. Each station has its own capacity, i.e. how many workers can process jobs on this particular station at once, depicted in parenthesis. The inter-arrival time of production orders is log-normal distributed and a fixed customer-required lead time is used. The system has been designed for a total of 48 workers. Workers are in a pool, which means that they are idle, and are assigned to stations when required, using a first come first serve (FCFS) dispatching policy. FCFS assigns the first worker that is available and capable of operating the respective machine as determined by the worker's qualifications. Switching between stations costs time. After a machine has been operated by a specific worker, this worker is idle again. For a further detailed examination of the implemented simulation model, the interested reader is referred to the C# source code, which is available on GitHub (see https://github.com/abeham/qualification-model).

4. EXPERIMENTS & RESULTS

This section first discusses the experimental setup in Section 4.1 and then presents and discusses the observed optimization results in Section 4.2.

4.1. Experimental Setup

As already explained, an NSGA-II has been used to optimize the qualification configurations, where the service level is maximized and the total number of required qualifications is minimized. Two experiments (EB1+2) use the binary vector encoding, whereas three other experiments (EQ1–3) use the newly implemented qualification encoding. For statistical significance, all experiments are conducted with 10 repetitions. Table 1 lists the parameters that were used to configure the NSGA-II. Values marked with *EB* are used in all binary vector encoding experiments, *EQ* defines parameter values used in all qualification encoding experiments.

Table 1: The NSGA-II parameters for binary vector and qualification encoding experiments.

Name	Value
PopulationSize	100
Selector	CrowdedTournament
CrossoverProbability	0.5
Crossover	Multi
MutationProbability	0.5
Mutator (Manipulator)	*EB1*: SinglePositionBitflip *EB2*: SomePositionsBitflip (Mut.Prob.: $\frac{1}{6}$) *EQ*: Multi
MaximumGenerations	100
SelectedParents	200

HL supports so called multi-operators, e.g. a multi-crossover or multi-manipulator, which apply 1 of n specified operators randomly. The experiment using the binary vector encoding uses all crossovers available for this encoding, whereas the three experiments that use the qualification encoding utilize different sets of crossover operators:

- EQ1: All proposed crossovers are enabled.
- EQ2: Only the DX is enabled.
- EQ3: Only the DX, UAX and OAX are enabled.

Furthermore, all experiments involving the qualification encoding apply a multi-manipulator, which chooses and executes one of all proposed manipulators randomly every time a mutation operation should be carried out. The binary vector encoding only offers two manipulators, which are evaluated separately (EB1+2). In case of the SomePositionsBitflip manipulator, the operators probability of flipping a bit has been set to $\frac{1}{Q}$, where $Q = 6$, which is the number of qualifications. This means that on average, every 6^{th} bit should be flipped, which corresponds to one qualification per worker.
Each qualification configuration is simulated 20 times. Table 2 shows the used simulation parameters. In the tested scenario, only 46 workers are available, although the system is designed for a total of 48 workers. This makes it harder to find high quality solutions, more specifically, to find solutions that have high service levels. Furthermore, switching stations between lanes costs twice as much as switching stations within a lane.

Table 2: The parameters of the simulation model.

Name	Value
Change Time Ratio	10 %
Cost FGI Inventory	1.0
Cost Tardiness	19.0
Cost WIP Inventory	0.5
Dispatch Strategy	FirstComeFirstServe
Due Date (Fix)	1.0
Due Date (Var)	100.0
Due Date (CV)	0.0
Interarrival Time (CV)	1.0

Line Change Factor	2.0
Observation Time	3600.0
Order Amount Factor	5.0
Personnel Ratio	50 %
Processing Time (CV)	0.25
Qualifications	6
Utilization	95 %
Warmup Time	600.0
Workers	46

A total of 6 qualifications are required to operate all stations of the production system. Stations S0, S1, S2 and S3 are mapped to indices 0, 1 and 2 within a qualification group, S4, S5, S6 andS7 to indices 3, 4 and 5, as shown in Figure 15.

S01 2/3 45 6/7	
010010	10
100100	11
110001	12
001000	13
	46

Figure 15: A valid qualification configuration.

This solution is interpreted as follows:

- Group 1: 10 workers that can operate S1 and S5.
- Group 2: 11 workers that can operate S0 and S4.
- Group 3: 12 workers that can operate S0, S1 and S6+S7.
- Group 4: 13 workers that can operate S2+S3.

4.2. Optimization Results

Figure 16 shows the achieved qualities of all binary vector experiments (EB) and the experiments involving the specialized encoding (EQ).

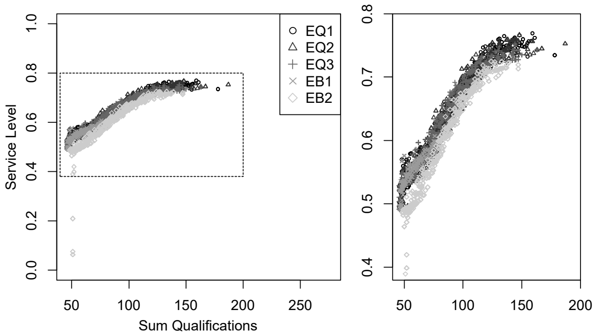

Figure 16: The achieved solution qualities from all conducted experiments.

The sum of qualifications can range from 46 (i.e. one qualification per worker) to 276 (every worker is qualified for every station), the service level ranges from 0.0 to 1.0 (100 %). The dashed box marks the zoomed-in area which is depicted again on the right side of the figure. A visual inspection of this result suggests that that the experiments that use the qualification encoding (EQ1–3) yield better results, compared to the binary vector encoding experiments (EB1+2). When looking at the solutions with the highest service level that where found by EB, one can observe that EQ found solutions with approximately the same service level, but significantly lower sums of qualifications. EB cannot reach the service levels found by EQ. Both sets of experiments found the solution with the minimal number of qualifications, i.e. 46 qualifications, which corresponds to 1 qualification per worker.

The following empirical attainment function plots (López-Ibáñez, Paquete, and Stützle 2010) compare the dominated area of each experiment and thus show how the specialized encoding is able to outperform the binary vector encoding. Figure 17 indicates that there is a significant difference between EB1 and EB2. EB1 finds more solutions with qualities that reside in the lower left corner of the objective space, the upper right corner is dominated by EB2.

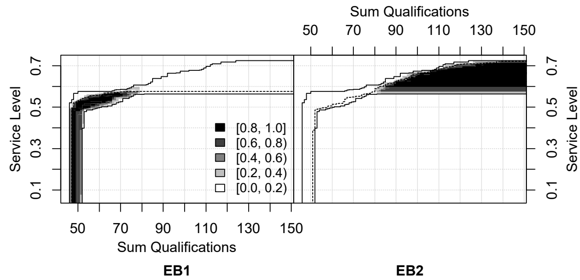

Figure 17: EB1 vs. EB2.

When comparing EB1 to EQ1–3, it is obvious that EQ is dominant in almost all areas of the objective space (see Figure 18; only EQ1 is shown here, comparisons with EQ2+3 look alike).

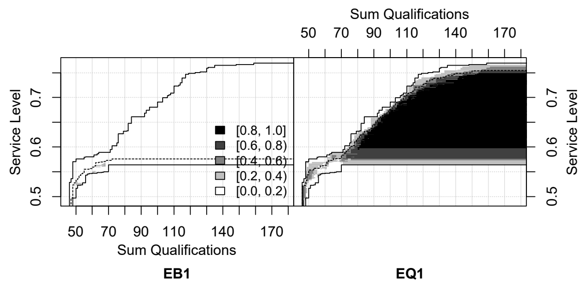

Figure 18: EB1 vs. EQ1.

Even though EB2 reaches higher quality levels than EB1, it is still dominated by EQ in all cases (see Figure 19; again, only EQ1 is shown here, comparisons with EQ2+3 look alike). The highest service levels that are achieved by EB2 are also achieved by EQ, but with significantly less qualifications.

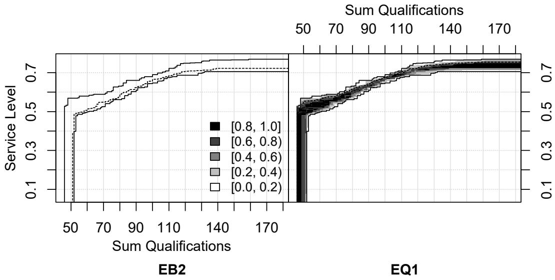

Figure 19: EB2 vs. EQ1.

Finally, there are no significant differences between all EQ experiments, but EQ1 and EQ2 are slightly dominating EQ3 in the upper right area of the objective space (see Figures Figure 20, Figure 21 and Figure 22).

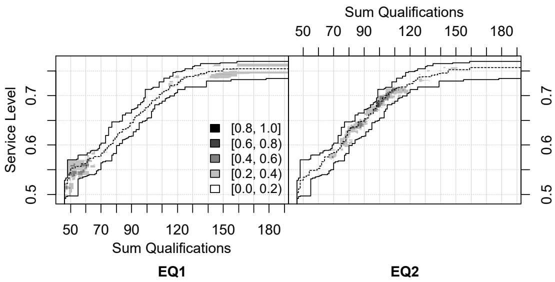

Figure 20: EQ1 vs. EQ2.

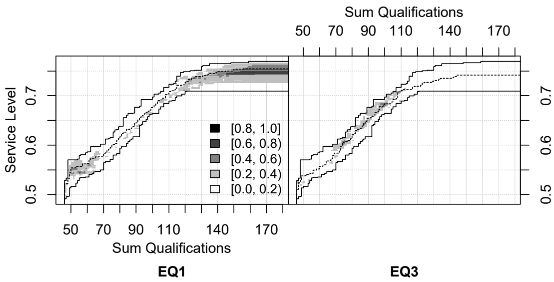

Figure 21: EQ1 vs. EQ3.

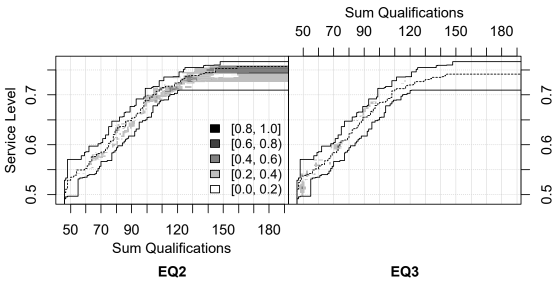

Figure 22: EQ2 vs. EQ3.

For EQ1–3, the chosen crossover and mutation operators have also been analysed with respect to their achieved success ratios. The success ratio $\Psi(c)$ of crossover c in one generation is defined as

$$\Psi(c) = \frac{x_c}{y}$$

, where x_c is the number of offspring that was created by crossover c and y is the overall number of offspring that was created by all crossovers in this generation. The same success ratio can also be calculated for mutation operators. For EQ1, the UAX yielded the most offspring, followed by OAX and DX operators, as can be seen in Figure 23. When inspecting the success ratios of the mutation operators, all success ratios are smaller compared to crossover success ratios and the most successful operator seems to be the Remove-Qualification-From-Group manipulator, as shown in Figure 24 for EQ1.

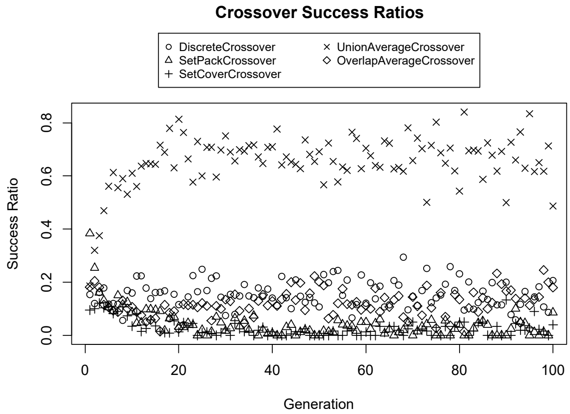

Figure 23: The success ratios of all crossovers used in EQ1 throughout all generations.

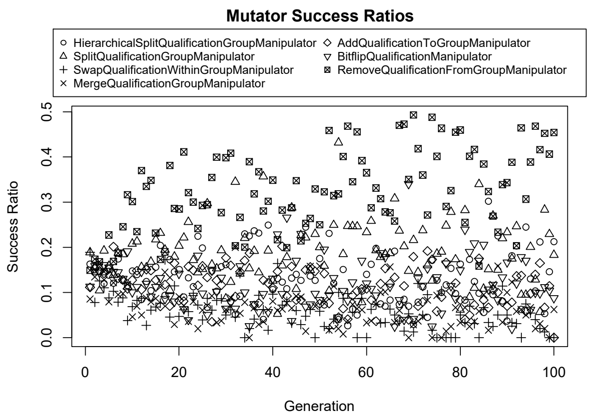

Figure 24: The success ratios of all manipulators used in EQ1 throughout all generations.

5. CONCLUSION & OUTLOOK

First tests with the new encoding show promising results. We conducted multiple experiments with NSGA-II using our new solution encoding and a simpler binary vector encoding. The obtained results show that using the new encoding, solutions with the same amount of qualifications, but higher service levels can be found. Furthermore, the algorithms are able to achieve higher service levels in general. Three different sets of crossovers have been used for testing the new solution encoding. EQ1–3 do not differ much in achieved solution qualities, which indicates that the respective crossover sets are all able to transfer the necessary building blocks to achieve high quality solutions. However, when analyzing crossover success ratios of EQ1, one can observe that the UAX was the most successful crossover in this set.

The proposed encoding can be extended in various ways. An idea is to introduce more solution creators which follow other strategies for constructing solutions. Specialized solution creators could for example use construction heuristics or introduce qualification groups according to predefined patterns, where e.g. a group is qualified for all stations within a line. Such predefined patterns have already been simulated in the aforementioned paper by Schober, Altendorfer, Karder, and Beham (2019), and some manually crafted configurations were even better than solutions optimized by NSGA-II. Another way to extend the proposed encoding is by adding more crossover or manipulation operators.

Finally, the objectives that are optimized can be extended. So far, the only objectives that have been considered were service level and number of qualifications. In the future, we will also take the number of qualification groups into account in order to find good configurations with as few groups as possible.

ACKNOWLEDGMENTS

The work described in this paper was done within the project *Digitale Methoden für verbesserte Personal-qualifizierungsstrategien* (OPTIMAL WORKFORCE, #862008), funded by the Austrian Research Promotion Agency (FFG) and the Government of Upper Austria.

REFERENCES

Affenzeller, M., Beham, A., Vonolfen, S., Pitzer, E., Winkler, S.M., Hutterer, S., Kommenda, M., Kofler, M., Kronberger, G., Wagner, S., 2015. Simulation-Based Optimization with HeuristicLab: Practical Guidelines and Real-World Applications. In: Mujica Mota, M., De La Mota, I.F., Guimarans Serrano, D., eds. Applied Simulation and Optimization: In Logistics, Industrial and Aeronautical Practice. Cham Springer International Publishing, pp.3–38.

Boland, N., Charkhgard, H., Savelsbergh, M., 2015. A criterion space search algorithm for biobjective integer programming: The balanced box method. INFORMS Journal on Computing, 27(4), pp.735–754.

De Bruecker, P., Van den Bergh, J., Beliën, J., Demeulemeester, E., 2015. Workforce planning incorporating skills: State of the art. European Journal of Operational Research, 243(1), pp.1–16.

Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.A.M.T., 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE transactions on evolutionary computation, 6(2), pp.182–197.

EUROCHAMBRES, 2019. EUROCHAMBRES Economic Survey 2019. Available from: https://issuu.com/eurochambres/docs/ees_2019_report_3.0 [Accessed May 2019].

Gosavi, A., 2015. Simulation-based optimization. Berlin: Springer.

Haimes, Y.V., 1971. On a bicriterion formulation of the problems of integrated system identification and system optimization. IEEE transactions on systems, man, and cybernetics, 1(3), pp.296–297.

Hu, Y., Bie, Z., Ding, T., Lin, Y., 2016. An NSGA-II based multi-objective optimization for combined gas and electricity network expansion planning. Applied energy, 167, pp.280–293.

Karp, R.M., 1972. Reducibility Among Combinatorial Problems. In: Miller, R.E., Thatcher, J.W., Bohlinger, J.D., eds. Complexity of Computer Computations. New York Plenum, pp.85–103.

Kück, M., Broda, E., Freitag, M., Hildebrandt, T., Frazzon, E.M., 2017, December. Towards adaptive simulation-based optimization to select individual dispatching rules for production control. In 2017 Winter Simulation Conference (WSC) (pp. 3852–3863). IEEE.

Lin, J.T., Chiu, C.C., Chang, Y.H., 2019. Simulation-based optimization approach for simultaneous scheduling of vehicles and machines with processing time uncertainty in FMS. Flexible Services and Manufacturing Journal, 31(1), pp.104–141.

López-Ibáñez, M., Paquete, L., Stützle, T., 2010. Exploratory Analysis of Stochastic Local Search Algorithms in Biobjective Optimization. In: Bartz-Beielstein, T., Chiarandini, M., Paquete, L., Preuss, M., eds. Experimental Methods for the Analysis of Optimization Algorithms, Berlin Springer, pp.209–222.

OECD, 2018. Skills for jobs. Organisation for Economic Co-operation and Development. Available from: https://www.oecdskillsforjobsdatabase.org/data/Skills%20SfJ_PDF%20for%20WEBSITE%20final.pdf [Accessed March 2019].

Schober, A., Altendorfer, K., Karder, J., Beham, A., 2019. Influence of Workforce Qualification on Service Level in a Flow Shop with two Lines. 9th IFAC Conference on Manufacturing Modelling, Management and Control. Berlin, Germany [accepted].

Wang, H., Fu, Y., Huang, M., Huang, G.Q., Wang, J., 2017. A NSGA-II based memetic algorithm for multiobjective parallel flowshop scheduling problem. Computers & Industrial Engineering, 113, pp.185–194.