Proceedings of the 2014 Winter Simulation Conference
A. Tolk, S. Y. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller, eds.

GENETIC ALGORITHMS FOR CALIBRATING AIRLINE REVENUE MANAGEMENT SIMULATIONS

Sebastian Vock

Department of Information Systems
Freie Universität Berlin
Garystrasse 21
14195 Berlin, GERMANY

Steffen Enz

Business Information Systems & Operations Research
Technische Universität Kaiserslautern
Gottlieb-Daimler-Strasse, Building 47
67663 Kaiserslautern, GERMANY

Catherine Cleophas

Research Group Advanced Analytics
RWTH Aachen University
Kackertstrasse 7
52072 Aachen, GERMANY

ABSTRACT

Revenue management (RM) theory and practice frequently rely on simulation modeling. Simulations are employed to evaluate new methods and algorithms, to support decisions under uncertainty and complexity, and to train RM analysts. To be useful in practice, simulations have to be validated. To enable this, they are calibrated: model parameters are adjusted to create empirically valid results. This paper presents two novel approaches, in which genetic algorithms (GA) contribute to calibrating RM simulations. The GA emulate analyst influences and iteratively adjust demand parameters. In the first case, GA directly model analysts, setting influences and learning from the resulting performance. In the second case, a GA adjusts demand input parameters, aiming for the best fit between emergent simulation results and empirical revenue management indicators. We present promising numerical results for both approaches. In discussing these results, we also take a broader view on calibrating agent-based simulations.

1 INTRODUCTION

Revenue management (RM) aims to channel demand to maximize overall revenue. The concept assumes a perishable product, fixed capacity, low marginal cost, and distinct demand segments. Automated RM systems forecast expected demand from historical sales data and use this forecast to calculate the optimal availability of product-price-combinations. State of the art RM models account for customer choice and aim to optimally exploit customers' willingness to pay. An overview of application areas is provided e.g. in Cleophas et al. (2011), mathematical methods of RM are surveyed e.g. in Talluri and Ryzin (2006). By combining data analysis, forecasting, and optimization, revenue management represents a picture-book example of business analytics (Davenport and Harris 2013).

Simulation modeling allows RM researchers and practitioners to analyze the potential consequences of new strategies: Immediate real-world implementation would require considerable effort and pose a major financial risk. Simulation experiments enable trial-runs under deliberately varied market conditions. Simulations can model stochastic demand and long-term effects of feedback loops that are part of the RM process. Finally, RM performance is difficult to analyze empirically. It is over-determined by economic conditions, competitor actions, and the company's own efforts. In a simulation that provides ceteris paribus conditions, any change in RM performance is caused by a corresponding change in RM strategy.

Agent-based simulations are especially well-suited to explicitely model the interactions of suppliers and customers. These interactions are the focus of revenue management. Furthermore, in practice, RM is not fully automated but is complemented by human analysts (Zeni 2003). Analysts supplement the demand forecast as well as the parameters and the results of the optimization. They can also be modeled as agents.

To be applicable for decision support, simulations have to be validated – researchers have to show that simulation results fit empirical observations. Calibrating model parameters to achieve this fit is a pre-condition for successful validation.

However, calibrating agent-based simulations for decision support is difficult. Agents' decision rules have to be explicitly formulated and parameterized (Gilbert 2008). Direct information such the decision rules for customers and RM analysts is rarely available. This calls for indirect calibration: Setting input parameters, processing experiments, validating the results by comparing them to empirical data, and adjusting the parameters until validity is achieved. Indirectly calibrating RM simulations can be both time-consuming and frustrating as these steps have to be repeated iteratively. Researchers aiming to manually calibrate the simulation have to wait until an experiment was processed, consider the output, and decide on how to adjust the input. As the relationship between input and output is not straight-forward in an agent-based simulation, it is not easy to predict which adjustments will increase validity.

The required effort and the repetitive nature render the task of manual, indirect calibration unwieldy and unpopular. To avoid the necessity of extensive indirect calibration, significantly simplified models could reduce the number of parameters. However, such models are not capable of modeling the emergent results observable when simulating individual agents interacting. At the same time, employing complex simulation models while neglecting the effort of calibration endangers the validity and thereby trustworthiness of RM simulations. When they are not rigorously calibrated and validated, simulations are neither a reliable research method nor a reliable tool for practical decision support; they remain toys (North and Macal 2007).

To facilitate and enhance calibration through automation, this contribution proposes the use of genetic algorithms (GA). We suggest representing a particular set of input parameters by the genetic code of a GA and phrasing the genetic performance function as one of validity or RM performance. Thereby, we automatize the gradual evolution of better input parametrizations. Setting up the algorithms and monitoring this evolution still require manual attention from researchers. Nevertheless, the automation of large parts of the calibration process renders it more efficient and thereby less likely to be avoided or neglected.

The next section briefly summarizes relevant research on simulation modeling in revenue management and on genetic algorithms in simulation modeling. An approach to employing genetic algorithms to model analyst influences is detailed in Section 3; an approach to employing genetic algorithms to adjust demand parameters is detailed in Section 4. For both approaches, we present numerical results based on the simulation system REMATE. We discuss the results in the larger context of calibrating agent-based simulations for different business analytics problems in Section 5.

2 STATE OF THE ART

This state of the art summarizes research on simulation modeling for revenue management and on genetic algorithms in simulation modeling. In both regards, we pay particular attention to agent-based modeling.

2.1 Simulation Modeling for Revenue Management

Simulation modeling plays a crucial role in RM theory and practice. Simulations can be used to evaluate new forecast methods and optimization algorithms. For example, Cleophas, Frank, and Kliewer (2009) uses simulations to evaluate forecast performance. They can support decisions under uncertainty by modeling market aspects that are not included in the automated system, such as competitive pricing (Isler and Imhof

2008). Through repeated what-if analyses as well as through participative game play, they can be used to teach concepts of RM (Cleophas 2012b).

Several simulation systems designed to support RM research and practice exist. Creating a simulation system to evaluate the success of an RM method is established research practice (Frank, Friedemann, and Schröder 2008). PODS (Passenger Origin Destination Simulator) has been developed in a cooperation of Boeing and MIT since the 1990s (Hopperstad 1995). PODS supported studies on topics ranging from the evaluation of demand estimation methods (Zickus 1998) to effects of low-fare competition for network carriers (Belobaba and Wilson 1997). PODS presentations frequently emphasize the challenge presented by calibrating new and realistic network scenarios.

The simulation system used in this paper, REMATE (Revenue Management Training for Experts), has been developed at Deutsche Lufthansa in cooperation with several German universities since 2009. REMATE models airlines and customers as agents interacting in dynamic markets (Cleophas 2012a). The simulation employs a stochastic, discrete-event-based paradigm and implements state-of-the-art algorithms for RM forecasting and optimization to model airline RM. Customers are modeled as maximizing their utility according to rational choice and request tickets following a Poisson distribution. The stochastic variation of parameters such as demand volume and willingness to pay is calibrated on empirical data. With regard to REMATE, too, calibration has been identified as a complex task. Individual instances of RM simulation modeling are described in many research contributions:

Agent-based simulation models allows to explicitely model the dynamic interactions of customers and suppliers and to gauge implications of emergent phenomena (Gilbert 2008): Both companies implementing RM and customers seeking the best alternative at the lowest price can be modeled as agents (Cleophas 2012c). By allowing agents to learn and communicate, complex aspects of real markets that cannot be efficiently included in analytical models of mathematical optimization can be represented in the simulation.

As described in Frank, Friedemann, and Schröder (2008), several challenges arise when employing simulations to support revenue management research. First of all, software verification has to ensure that market model and RM algorithms are correctly implemented. There exists extensive research on the verification of simulation systems; Sargent (2013) lists relevant aspects of this topic.

Demand calibration poses a particular challenge for RM simulations, as information about the true characteristics of demand is difficult to gauge even for the real-world systems. Such information is usually estimated based on sales, which present a censored view: Sales data only list customers who are willing to buy at the price offered at the time of their request (Weatherford and Pölt 2002). Customers who choose to abstain or to accept a competing offer are not reported by sales data. Furthermore, the decision rules underlying sales are not transparent.

Representing RM analysts in simulations is another unsolved problem. The need for such analysts is frequently acknowledged: (Zeni 2003) emphasize the role of analysts in supplementing the demand forecast. (Isler and Imhof 2008) stress the role of RM analysts in dealing with competitive markets. Commercial RM systems include interfaces that enable analysts to influence the forecast or the optimization results. To our knowledge, PODS does not explicitly model RM analysts. While REMATE allows manual analyst influences, it does not represent analysts as intelligent agents.

This contribution presents an approach employing GAs to model analyst influences and to calibrate demand for RM simulations. As the remainder of this section shows, GAs are a fitting tool for this task and one that is already established in the domain of simulation modeling.

2.2 Genetic Algorithms in Simulation Modeling

Genetic algorithms (GA) are a meta-heuristic inspired by nature (Simon 2013): A population of instances, each implementing a different solution as an individual ”genetic code”, is initialized and evaluated on a particular problem instance. The best performers are cross-bred to generate new instances. Mutations introduce random variations to avoid the algorithm converging on locally optimal solutions. The algorithm stops after a given number of iterations or when a steady state has been reached, in which solution performance no longer changes.

GA are well-suited to solve complex combinatorial optimization problems. Application areas range from lot sizing (Goren, Tunali, and Jans 2010) over resource allocation (Hegazy 1999) to pricing (Macías and Guitart 2011). In solving optimization problems, GAs are frequently cited as a type of "simulation optimization" (Paul and Chanev 1998): To identify the best solution using a GA, a simulation of the problem is required that evaluates the performance of alternatives.

Predictive analytics and model fitting can also be phrased as optimization problems: These aim to minimize the distance between a model's output and training or validation data. Accordingly, predictive analytics and model fitting, too, are application areas for GAs. For example, (Sarimveis and Bafas 2003) formulates a prediction problem as a non-linear optimization model and applies a GA to develop solutions online. Models that can be fit using GA range from abstract mathematical curves (Lybanon and Messa Jr 1999), to agent-based models (Midgley, Marks, and Kunchamwar 2007). In the latter example, the authors consider "destructive validation", elaborating on an idea phrased in (Miller 1998). However, if GA can be applied to search for invalid parametrizations, it appears intuitive that they can also be applied to search for valid parametrizations. This contribution presents an approach to employ GA in such a way that it supports airline revenue management simulations.

As described in Gilbert (2008), GAs can be employed to model intelligent agents, too: The evaluation of solutions and crossbreeding of successful candidates serves as a model of learning. Thereby, a GA can model an individual or a population attempting to solve a problem, such as utility maximization (Chen and Huang 2007). Employing genetic algorithms to model RM analysts represents a novel variant of this concept. In this context, GA are applied to improve revenue performance beyond the level of automated systems given imperfect forecast performance.

3 CALIBRATING ANALYST INFLUENCES

RM parameters define the availability of product-and-price-combinations. They can be influenced through relative or absolute changes by analysts. Analyst influences are observable and stored in databases. However, there exists no possibility to evaluate their impact. No situation with exactly the same demand yet without these influences can be observed. Therefore, we cannot empirically calculate and evaluate the effect of analyst influences. For this purpose, simulation modeling is required. This section describes a novel approach to calibrate influences set by RM analysts.

3.1 Approach

Our RM model is defined through three dimensions: supply, demand and RM methods. The parameters of the model define a time-dependent availability over the sales horizon. The sales horizon considered starts 360 days before departure and ends at the day of departure. To divide the booking horizon in discrete parts, we define a set D of time slices d. In the following we only use these discrete time slices to define influences. To model supply, we have to define a set F of flights f and a set C of compartments c used on each flight. F uniquely defines a set J of origin-destination (OD) combinations j. Combining an OD with a specific departure time defines a set O of origin-destination-itineraries (ODI) o. Let B be a set of booking classes b used on each ODI and $p_b^o \in P$ the corresponding price for each ODI-booking class combination. The demand side is defined by a set $Y = \{Y_j, j \in J\}$, where Y_j defines a set of customers y_j for OD j.

In the RM model, two availability parameters can be influenced by analysts. The first availability parameter $e_{f,c,d}$ is defined per flight, compartment and time before departure. E denotes the set of all such parameters within a simulation scenario. $e_{f,c,d}$ models the opportunity costs of a unit of capacity for the airline on this (f,c,d)-combination. We call influences targeting this parameter supply-side-influences (SI). Let $T_{o,b}$ be the set of parameters per ODI and booking class for the second availability parameter and T the set of all parameters used in a scenario. $t_{o,b,d}$ defines the specific parameter for an itinerary, booking class and time before departure combination. This parameter represents the marginal revenue for a single customer. We call influences targeting this parameter demand-side-influences (DI). Let $s_k = \{i_{k,l} \in \mathbb{R} | l \in I_k\}$ be a strategy consisting of influences and S the index-set for k. We use the values $i_{k,l}$ to manipulate the parameters calculated by the RM system. In terms of GAs, we subsequently refer to these strategies as "individuals" and to single influences as "genes".

Initialization: For the approach described in this section, we initialize the values of all influences by drawing uniformly distributed random values from a defined interval. There exist alternative ways to initialize individuals, for example setting the initial values manually.

Supply-side-influences: To define the range of randomly drawn initial influences per compartment and flight, we calculate the average net price $\overline{p}_{c,f}$ for all booking classes $b \in B_{c,f}$ within a compartment-flight combination (c, f). We denote with $\hat{b}_c$ the number of booking classes within a specific compartment c. Since prices are defined per ODI, we have to slightly transform all corresponding prices to calculate

$$\overline{p}_{c,f} = \frac{1}{\hat{b}_c} \cdot \sum_{b \in B_{c,f}} p_b^f.$$

In a next step, we define the range from which the random initial values $p_{c,f}$ are drawn, depending on a parameter $r_1 \in [0,1]$. This range is used to define the initial value $e_{f,c,d}$ as a realization of $X(r_1) \sim U\left(\left[-r_1 \cdot \overline{p}_{c,f}, r_1 \cdot \overline{p}_{c,f}\right]\right)$.

Demand-side-influences: To define initial values per itinerary and class, we calculate the mean $\bar{t}_{o,b}$ of all parameter values $t_{o,b,d}$ within an ODI-class combination (o,b)

$$\bar{t}_{o,b} = \frac{\sum_{d \in D} t_{o,b,d}}{|T_{o,b}|}.$$

In a next step, we define the range from which the initial random values $t_{o,b}$ are drawn. This range depends on a parameter $r_2 \in [0,1]$ and is used to define a random variable $X(r_2) \sim U([-r_2 \cdot \bar{t}_{o,b}, r_2 \cdot \bar{t}_{o,b}])$. Accordingly the initial value $t_{b,o}$ is defined as a realization of $X(r_2)$.

We based the formulation of our GA on the structure described in (Yu and Gen 2010). A generation consists of a set A of n_A parent individuals, from which a mating pool M with n_M individuals and subsequently a population P consisting of n_P individuals is derived. The individuals within the mating pool are a subset of the parents that form the basis for creating the new population. Based on individual fitness values, we select a subset of n_A individuals from the population as the parents for the following generation.

Calculating the fitness value: A frequent objective pursued by RM analysts, as by automated RM systems, is revenue maximization. Therefore, we define the fitness value v_k as the revenue for individual s_k. According to this definition, our aim is to maximize the fitness value.

Constructing the mating pool: Before breeding a new population, we construct a mating pool $M \subseteq A$ by imitating natural selection according to (Yu and Gen 2010). We draw individuals from the set of parents based on relative fitness: Individuals with a higher fitness value a more likely selected for the mating pool. Denote by

$$\rho_k = \frac{v_k}{\sum_{i \in A} v_i} \in [0,1]$$

the probability for individual $s_k \in A$ to be selected. According to those probabilities we draw n_M individuals where we allow multiple drawings of the same individual. So it is possible that the same individual is selected several times for the mating pool. For our approach, we use a slightly different calculation of the relative fitness: Before calculating the new probability $\hat{\rho}_k$ we transform the fitness-value of the n_A parents with a dynamically chosen parameter $r \in \mathbb{R}$ according to $\hat{v}_k = v_k - r$. The transformed probability $\hat{\rho}_k$ of individual s_k to be selected as mate is then calculated as

$$\hat{\rho}_k = \frac{\hat{v}_k}{\sum_{i \in A} \hat{v}_i} \in [0,1].$$

This transformation appears useful due to the tight and large fitness values of the n_A parents. The resulting probabilities create a quasi-uniform distribution. The above mentioned transformation results in better distributed probabilities and guarantees a greater amount of variability within the selection process.

Crossovers and mutations: Creating a new population is based upon crossover recombination techniques for real numbers following the framework outlined in Yu and Gen (2010) and Simon (2013). After recombining two individuals from the mating pool, mutation guarantees a level of variation. Mutations are realized independently for each influence included in the newly created individual by adding a random factor m. To this end, we draw m from a normal distribution with mean 0 and standard deviation $\sigma \in \mathbb{R}$.

The simplest way to create new individuals is to select some of the best individuals from the parent generation and mutate every gene with m. We call this method *Parent-Mutation*. For all influences $i_{k,l}$ with $l \in I_k$ of the new individual s_k^* we calculate the new value $i_{k,l}^*$ as $i_{k,l}^* = i_{k,l} + m$. Additionally, we use multiple crossover methods to provide sufficient variation and to address the specific characteristics of an RM simulation. In the following, we do not use any preferences for the location of the offspring in our crossover methods. Our basic crossover method is the general Hoelder-mean

$$M_p(s_1, \ldots, s_n) = \left(\frac{1}{n} \sum_{i=1}^{n} s_i^p \right)^{1/p},$$

which will be used with different parameters p to define multiple crossovers. First we implement the *arithmetic crossover*. For all influences of new individual s_k^* we calculate the new value $i_{k,l}^*$ as the arithmetic mean of all individuals, using the general Hoelder-mean with $p = 1$. The *harmonic crossover* is the second technique. For all influences of new individual s_k^* we calculate the new value $i_{k,l}^*$ as the harmonic mean of all individuals, using the Hoelder-mean with $p = -1$. The *blend crossover* is adapted from (Yu and Gen 2010). For all influences $i_{k,l}^*$ with $l \in I_k$ of new individual s_k^*, the blend crossover (BLX) is defined for two individuals s_1, s_2 (Yu and Gen 2010). We define a uniform distributed random variable

$$X_{BLX} \sim U\left([i_{1,l} - \alpha\,|i_{1,l} - i_{2,l}|\,, i_{2,l} + \alpha\,|i_{1,l} - i_{2,l}|]\right)\ ,\ i_{1,l} < i_{2,l}.$$

The new influence $i_{k,l}^*$ is calculated as realization of X. The blend crossover is similar to the arithmetic crossover. All offspring are represented as a linear combination of the two parents. To increase variation, we expand the range to allow values outside the line connecting the two parents. The additional amount of range depending on the distance between s_1 and s_2 is defined by a parameter $\alpha \in [0, 1]$.

Additionally, we define *situation-specific crossover* methods. These crossovers depend on the special scenario characteristics. For example, for a scenario including substitutable itineraries, the crossover method can exchange parts of the strategy between the itineraries. For scenarios including only a single flight and multiple influence types we defined a SI-DI crossover. Here the influences $i_{1,l}$ and $i_{2,l}$ of two individuals are used to calculate a new individual in the following way

$$i_{k,l}^* = \begin{cases} i_{1,l} & \text{if } i_{k,l} \text{ is a demand-side influence,} \\ i_{2,l} & \text{if } i_{k,l} \text{ is a supply-side influence.} \end{cases}$$

3.2 Results

To demonstrate the potential of our approach, this section provides numerical results based on a simulation study. The scenario considered includes a single itinerary, defined by one direct flight. On this itinerary, 12 booking classes are offered in two compartments at one constant price per class. The implemented RM algorithms were adapted from Fiig et al. (2009). Demand parameters were calibrated using empirical flight data.

Each experiment includes 100 simulation runs: The first 50 runs are used to initialize the RM algorithms, from run 51 on, influences are applied. When calculating the fitness value, we average revenue resulting from runs 51-100, reflecting the part of the experiment affected by analyst influences. The GA parameters generated for the simulation study are defined as follows: We use $|D| = 13$ time slices, a generation as well as the mating pool consists of $n_A = n_M = 10$ parent-individuals. According to the number of crossover and mutation methods used, a population contains $n_P = 27$ individuals. In preliminary studies, this number resulted in an appropriate balance between run-time and positive convergence effects. We set $r_1 = 0.3$, $r_2 = 0.3$, $\alpha = 0.5$, and r dynamically equal to the fitness value of the 15^{th} individual of the current population. For drawing the mutation factor m, we propose $\sigma = 0.05$ in our simulation setup.

To create the GA population, we apply all crossover techniques presented in the previous section. Within the first 40 generations, parent mutations perform exceedingly well when evaluating the average rank of all individuals created across generations. BLX and SI-DI crossover show a slightly poorer performance; harmonic crossover comes in last. When evaluating the minimum rank achieved, SI-DI crossover achieves slightly superior results. The parent mutation precisely exhibits the behavior expected from experiences gathered with manual influence setting: Small changes for all influences can lead to large result deterioration. Rank variance for the set of individuals created by the parent mutation method is high; compared to all other crossover techniques, ranks are not stable across generations.

To benchmark the automated setting of analyst influences using GA, we compare the results to reference values achieved by using a ”psychic” forecast. For each simulation run, this forecast is created based on knowledge of the generated demand: The actual amount of customers willing to buy particular booking classes is predicted. We define a non-deterministic expected achievable value (EAV) that can be calculated based on an averaged psychic forecast over all runs. Obviously, it is possible to beat the EAV because it does not represent the deterministic optimum for this scenario. Additionally, we calculate a base case (BC) by executing the simulation given a realistic forecast and no analyst influences.

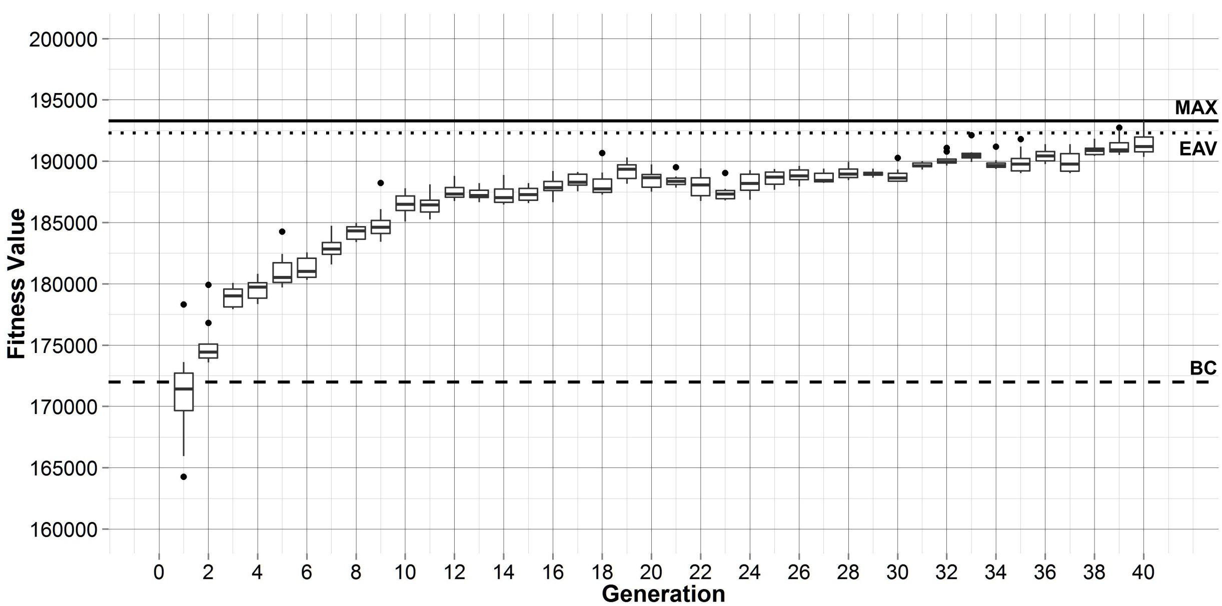

Figure 1: Fitness values for the ten best individuals over all generations for calibrating user influences.

Figure 1 shows box plots representing the fitness values of the 10 best-performing individuals of each generation. While the y-axis displays fitness in terms of achieved revenue as a monetary value, the x-axis displays the generation considered. Each box plot consists of a box with a band inside, lines extending vertically, called whiskers, and in some cases single dots above or underneath the whiskers. The bottom of the box represents the first quartile of the underlying data, the top the third quartile. The horizontal line inside the box represents the median, while the end of the whiskers include all values that are within

$1.5 \cdot IQR$ of the upper respectively lower quartile, following the definition of Tukey (Tukey 1977). Any data not included in the range of the whiskers is plotted as an outlier, symbolized by single dots. The horizontal black line over the whole figure marks the fitness value of the best individual over all generations (MAX). The EAV is displayed by the dotted horizontal line, while the BC is displayed as dashed line.

Figure 1 illustrates a convergence of fitness indicators over the generations. Within the first 10 generations, revenue can be increased by about 8%, whereas over the last 10 generations, the incremental increase is about 3%. In generation 39, the GA could even find a solution which exceeds the EAV. Additionally, the variance within a generation decreases over the generations. Considering the individuals' ranks and the corresponding crossover-methods reveals that individuals created by the situation-specific crossover SI-DI and the parent-mutation are most likely to be among the top performers.

3.3 Discussion

From our experience, manually calibrating a complex simulation is difficult and requires considerable effort and time. In particular, big alterations in validation indicators frequently result from small parameter changes. With this in mind, we were surprised by the high performance and fast convergence of the algorithm presented here. The performance within the first 15 generations shows that the GA provides a suitable way to calibrate user influences. Especially the fast convergence behavior compared to manual approaches is astonishing. To avoid an over-fitting the model and parameters were designed using state-of-the-art approaches before the data used for the numerical approaches was known.

Our expectation of large alterations from small changes could be confirmed by analyzing the different crossover methods individually. However, the overall convergence of top performing individual solutions represented by Figure 1, contradicts this expectation. This can be explained by the combination of a variety of crossover methods. Especially the SI-DI crossover exceeds our expectations. We conclude that focusing on the structure of the problem while designing a GA for calibration is extremely helpful. For further analysis, the algorithm should be tested on larger scenarios. Here, we expect an even higher performance boost in contrast to manual calibration.

4 CALIBRATING DEMAND

When using simulations to support RM decisions, simulation results have to be empirically validated. Realistically replicating RM methods and supply appear to be merely a problem of data collection and verification. Estimating demand parameters from real data to parametrize scenarios, however, constitutes a greater challenge. This section introduces an approach that uses a GA to automatically adapt demand input parameters. The aim is to generate simulation results that fit empirical booking-data as well as possible.

4.1 Approach

In the simulation model used to generate numerical results, customer types are used as templates to stochastically draw individual customers. These customer types are defined by a set of mean parameters and distributions. Calibrating demand means adjusting these parameters. The objective is to generate customers whose interaction with the RM system results in bookings fitting the empirical indicators. For this purpose, we modify the approach presented in Section 3.

We define an individual $s_k, k \in S$ as a uniquely defined set T of n_T customer types and a parameter $d_k(c) \in [0,1]$ describing the relative share of type c. Demand for an individual is defined by $d_k = \sum_{c=1}^{n_T} d_k(c)$. The number of customer types, the supply, and RM algorithms are kept constant during calibration.

Let L be the set of parameters within an individual. A single customer type is defined by n different parameters. From this follows that the number of parameters defining an individual can be calculated as $|L| = n_t \cdot n + {n_t}^2$. Based on the simulation model, for all parameters $i_{k,l}$ within a customer type, it holds that $i_{k,l} \in \mathbb{R}^+ \setminus \infty$. The values have to be chosen in intervals such as $i_{k,l} \in \left[0, i_l^{max}\right]$ for all parameters.

Initialization: As mentioned in Section 3, there exist several alternative ways to parametrize the first generation of GA individuals. For demand calibration, we use parameters that were manually generated during earlier calibration attempts. We mutate the parameters to create a basic variation within the first generation. The mutation factor for each parameter $i_{k,l}$ is drawn from a normally distributed random variable

$$X \sim N\left(i_{k,l}, max\left\{0.1, \frac{i_{k,l}}{10}\right\}\right).$$

Mutations and crossover: The crossover and mutation techniques used in this approach follow those described in Section 3. Since $i_{k,l} \geq 0, \forall l \in L$, we can introduce the geometric mean as an additional crossover method. Besides, the situation-specific crossover has to be modified to fit the new requirements. We use a slightly modified parent-mutation method, too, as it is defined for initialization. The *geometric crossover* will be used for all influences $i_{k,l}$ with $l \in I_k$ of new individual s_k^*. We calculate the new value $i_{k,l}^*$ using the Hoelder-mean with $p \to 0$.

Situation-specific crossover: The idea behind this crossover technique is to exploit the specific structure of the currently examined demand situation. Let s_1, s_2 denote two individuals. We then define the following crossover to calculate the parameters $i_{k,l}^*$ of our new individual s_k^*:

$$i_{k,l}^* = \begin{cases} i_{1,l} & \text{if } i_{k,l} \text{ corresponds to customer type } t \in T_1, \\ i_{2,l} & \text{if } i_{k,l} \text{ corresponds to customer type } t \in T_2, \end{cases}$$

where we define the set $T_1 := \left\{t \in T \mid t \leq \left\lfloor \frac{n_t}{2} \right\rfloor\right\}$ and $T_2 := \left\{t \in T \mid t \geq \left\lceil \frac{n_t}{2} \right\rceil\right\}$.

Calculating the fitness value: The objective of demand calibration is to fit customer types so that generated customers cause bookings that fit an empirically given ideal. Manual calibration compares the number of final bookings per booking class b, flight f and individual s_k with a given target value. This requires the absolute number of bookings for each booking class-flight combination $\dot{x}_{b,f}$ and the number of simulation bookings $x_{b,f,k}$ for each individual s_k. The fitness value for individual s_k is then calculated as

$$v_k = \sum_{f \in F} \sum_{b \in B} \left|\dot{x}_{b,f,k} - x_{b,f}\right|.$$

According to this definition the objective is to minimize the fitness value.

4.2 Results

The results presented here are based on the scenario introduced in Section 3 using 50 simulation runs. The demand is defined by $n_t = 6$ customer types. The GA modifies parameters defining the willingness-to-pay and the customer-choice function; this affects $n = 27$ parameters per customer type leading to an overall number of 162 parameters per scenario. The RM system relies on the psychic forecast that was used to generate the EAV presented in Section 3. We set $n_a = n_m = 10$. According to the additionally introduced crossover methods the population size increases to $n_p = 32$ individuals. For this approach we do not transform the fitness value during the creation of the mating pool, as the magnitude of the values is smaller.

Since for all parameters $i_{k,l}, l \in L$, defining customer types holds that $i_{k,l} \geq 0$, we can use all crossover methods for this approach, including the geometric crossover. For the BLX crossover we set $\alpha = 0.5$. Parent mutations and arithmetic crossover perform exceedingly well when evaluating the average rank of all individuals created across all generations. Geometric, harmonic, and customer crossover show a slightly poorer performance; BLX crossover comes in last. Evaluating the minimum rank achieved by the different crossover methods, BLX and parent mutations achieves slightly poorer results than all other methods. Looking at the variance within the achieved ranks the parent mutation shows exactly the behavior that would be expected from experiences gathered with manual influence setting. A good performance is shown by customer crossover, the result deterioration is half as large as from the other crossover techniques.

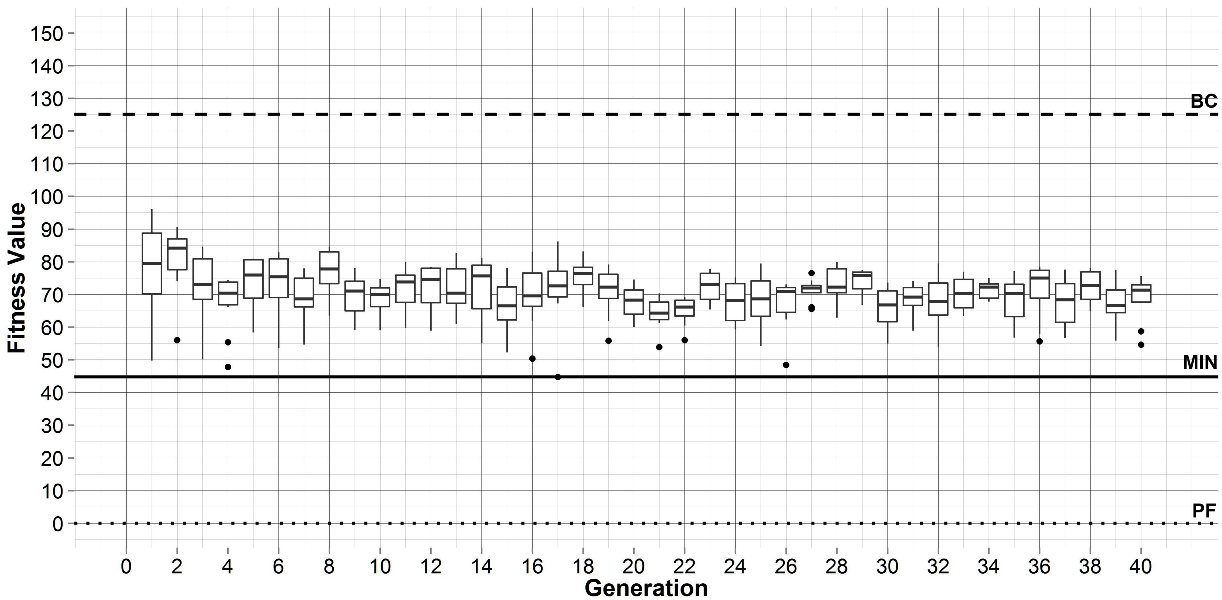

Figure 2: Fitness values for the ten best individuals over all generations for calibrating demand.

We calculate the fitness value using averaged bookings per booking class, flight and individual $x_{b,f,k}$. As target values $\dot{x}_{b,f}$, we use empirical booking data. Figure 2 shows GA performance in terms of box plots. Every box plot represents the ten best-performing individuals of a generation and their fitness values. The x-axis of the graph indicates the generation, while the y-axis shows the fitness value. The box plots follow the description given in Section 3. Two reference cases are displayed as horizontal lines within the graph: The base case (BC), indicated by a dashed line, shows the fitness resulting from customer types used for initialization. The perfect fit (PF), where the fitness value is 0, presents a lower bound. Note that this lower bound is unrealistic as it is hardly feasible to achieve a perfect fit. The continuous horizontal line marks the fitness value of the best individual across all generations (MIN).

4.3 Discussion

Compared to the results of Section 3, we observe both slower convergence and higher variance of individual fitness within the generations. This can be explained by the substantially more complex objective function and the high number of genes. To achieve improved convergence, modeling the mutation probability as a variable factor appears promising.

The GA clearly outperformed manual calibration in terms of efficiency. We were not able to obtain similar fitness values by manual calibration within such a short time: Calibrating the simulation using the GA took about one week including the parametrization of the GA. Manual calibration of similarly sized scenarios has required up to 10 weeks of dedicated work.

We were able to improve fitness values further by additional manual calibration of the fittest individual obtained by the GA. This fact may be addressed through additional crossover methods, taking the structure of the problem into account. For example, in the current implementation, we do not use information describing for which booking classes the fit is particularly low. We could take this information into account by introducing a crossover that mates an individual with high fitness in higher booking classes with an individual with high fitness in lower booking classes. Here, an intelligent way to define which genes should be incorporated into the new individual has to be found. Finally, a more sophisticated way to calculate the fitness value could further improve GA performance. Using the sum of the squared deviations or another definition incorporating situation specific details could lead to better convergence.

5 CONCLUSION

This contribution first outlined the potential benefits of employing simulations for decision support in revenue management. We highlighted the challenges of calibrating such simulations for empirical validation, particular with regard to analyst influences and demand. To meet this challenge, we proposed the use of GAs and introduced two novel approaches. In numerical experiments, we achieved promising results: The system including GA-set analyst influences systematically outperformed a system lacking such influences. Demand calibration using GA achieved an improved model fit and proved to be significantly more time-efficient than manual calibration.

By introducing formal approaches and presenting numerical results, we were able to show that GAs clearly are a versatile tool for calibrating agent-based simulations. This was demonstrated for the revenue management domain, but can be expected to apply to other domains, as well. Indirectly calibrating heterogeneous groups of agents is difficult and time-consuming; GAs offer a way of effectively and efficiently automatizing large parts of this process. As mentioned in the respective sections, both approaches presented here still hold potential for technical improvement and extension. Such a possible extension of our approach could be the generalization to input data modeling methods.

Research on agent-based simulations in general requires further ideas for automated calibration. Such methods would make simulation modeling as a rigorous method for the purpose of decision support easier to implement. The possibility of rigorously validating simulations without excessive calibration effort would make this tool even more applicable and popular across research domains.

REFERENCES

Belobaba, P. P., and J. L. Wilson. 1997. "Impacts of yield management in competitive airline markets". *Journal of Air Transport Management* 3 (1): 3–9.

Chen, S.-H., and Y.-C. Huang. 2007. "Relative risk aversion and wealth dynamics". *Information Sciences* 177 (5): 1222–1229.

Cleophas, C. 2012a. "Assessing multi-agent simulations–inspiration through application". In *Highlights on Practical Applications of Agents and Multi-Agent Systems*, edited by J. Bajo Perez, J. Corchado Rodríguez, M. Moreno, V. Julián, P. Mathieu, J. Canada-Bago, A. Ortega, and A. Fernández Caballero, 163–170. Springer.

Cleophas, C. 2012b. "Designing serious games for revenue management training and strategy development". In *Proceedings of the 2012 Winter Simulation Conference*, edited by C. Laroque, J. Himmelspach, R. Pasupathy, O. Rose, and A. M. Uhrmacher, 140. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Cleophas, C. 2012c. "Multi-agent modelling for revenue management". *Journal of Revenue & Pricing Management* 11 (2): 240–242.

Cleophas, C., M. Frank, and N. Kliewer. 2009. "Simulation-based key performance indicators for evaluating the quality of airline demand forecasting". *Journal of Revenue & Pricing Management* 8 (4): 330–342.

Cleophas, C., I. Yeoman, U. McMahon-Beattie, and E. Veral. 2011. *The applications of revenue management and pricing*, 9–16. Palgrave Macmillan.

Davenport, T. H., and J. G. Harris. 2013. *Competing on analytics: the new science of winning*. Harvard Business Press.

Fiig, T., K. Isler, C. Hopperstad, and P. Belobaba. 2009. "Optimization of mixed fare structures: Theory and applications". *Journal of Revenue & Pricing Management* 9 (1): 152–170.

Frank, M., M. Friedemann, and A. Schröder. 2008. "Principles for simulations in revenue management". *Journal of Revenue & Pricing Management* 7 (1): 7–16.

Gilbert, N. 2008. *Agent-based models*. Number 153. Sage.

Goren, H. G., S. Tunali, and R. Jans. 2010. "A review of applications of genetic algorithms in lot sizing". *Journal of Intelligent Manufacturing* 21 (4): 575–590.

Hegazy, T. 1999. “Optimization of resource allocation and leveling using genetic algorithms”. *Journal of Construction Engineering and Management* 125 (3): 167–175.

Hopperstad, C. 1995. “Potential Application of the Boeing Passenger Origin-Destination Simulator (PODS) to Yield Management System Planning”. Technical report, Revision C. Boeing Commercial Airplane Group.

Isler, K., and H. Imhof. 2008. “A game theoretic model for airline revenue management and competitive pricing”. *Journal of Revenue & Pricing Management* 7 (4): 384–396.

Lybanon, M., and K. C. Messa Jr. 1999. “Genetic algorithm model fitting”. In *Practical Handbook of Genetic Algorithms*, edited by L. D. Chambers, 269–346.

Macías, M., and J. Guitart. 2011. “A genetic model for pricing in cloud computing markets”. In *Proceedings of the 2011 ACM Symposium on Applied Computing*, 113–118. ACM.

Midgley, D., R. Marks, and D. Kunchamwar. 2007. “Building and assurance of agent-based models: An example and challenge to the field”. *Journal of Business Research* 60 (8): 884–893.

Miller, J. H. 1998. “Active nonlinear tests (ANTs) of complex simulation models”. *Management Science* 44 (6): 820–830.

North, M. J., and C. M. Macal. 2007. *Managing business complexity: discovering strategic solutions with agent-based modeling and simulation*. Oxford University Press.

Paul, R. J., and T. S. Chanev. 1998. “Simulation optimisation using a genetic algorithm”. *Simulation Practice and Theory* 6 (6): 601–611.

Sargent, R. G. 2013. “Verification and validation of simulation models”. *Journal of Simulation* 7 (1): 12–24.

Sarimveis, H., and G. Bafas. 2003. “Fuzzy model predictive control of non-linear processes using genetic algorithms”. *Fuzzy Sets and Systems* 139 (1): 59–80.

Simon, D. 2013. *Evolutionary Optimization Algorithms*. John Wiley & Sons.

Talluri, K. T., and G. J. Ryzin. 2006. *The theory and practice of revenue management*, Volume 68. Springer.

Tukey, J. W. 1977. *Exploratory data analysis*. Addison-Wesley.

Weatherford, L. R., and S. Pölt. 2002. “Better unconstraining of airline demand data in revenue management systems for improved forecast accuracy and greater revenues”. *Journal of Revenue & Pricing Management* 1 (3): 234–254.

Yu, X., and M. Gen. 2010. *Introduction to Evolutionary Algorithms*. Springer.

Zeni, R. H. 2003. “The value of analyst interaction with revenue management systems”. *Journal of Revenue and Pricing Management* 2 (1): 37–46.

Zickus, J. S. 1998. “Forecasting for airline network revenue management: revenue and competitive impacts”. Technical report, Massachusetts Institute of Technology.

AUTHOR BIOGRAPHIES

SEBASTIAN VOCK graduated from the Technische Universität Darmstadt with a M.Sc. in Mathematics; he is now a PhD student of the Freie Universität Berlin. His doctoral work is sponsored by a Lufthansa fellowship and focuses on flexibility and long-term orientation in Airline RM. His email address is sebastian.vock@fu-berlin.de.

STEFFEN ENZ is working on his PhD thesis at the Technische Universität Kaiserslautern after having graduated from the University of Frankfurt with a Degree in Physics. His doctoral work is sponsored by a Lufthansa fellowship and focuses on meta-heuristics for dynamic pricing in Airline RM. His email address is enz@wiwi.uni-kl.de.

CATHERINE CLEOPHAS is Associate Professor for Advanced Analytics in the Research Area Operations Research and Management at RWTH Aachen University. Her research fields include revenue management, modeling and forecasting the interaction of suppliers and demand, and agent-based, stochastic simulation systems. Her email address is catherine.cleophas@rwth-aachen.de.