2014 IEEE Congress on Evolutionary Computation (CEC)
July 6-11, 2014, Beijing, China

Swarm/Evolutionary Intelligence for Agent-Based Social Simulation

Andreas Janecek and Tobias Jordan and Fernando Buarque de Lima-Neto

***Abstract*— Several micro economic models allow to evaluate consumer's behavior using a utility function that is able to measure the success of an individual's decision. Such a decision may consist of a tuple of goods an individual would like to buy and hours of work necessary to pay for them. The utility of such a decision depends not only on purchase and consumption of goods, but also on fringe benefits such as leisure, which additionally increases the utility to the individual. Utility can be used then as a collective measure for the overall evaluation of societies. In this paper, we present and compare three different agent based social simulations in which the decision finding process of consumers is performed by three algorithms from swarm intelligence and evolutionary computation. Although all algorithms appear to be suitable for the underlying problem as they are based on historical information and also contain a stochastic part which allows for modeling the uncertainty and bounded rationality, they differ greatly in terms of incorporating historical information used for finding new alternative decisions. Newly created decisions that violate underlying budget constraints may either be mapped back to the feasible region, or may be allowed to leave the valid search space. However, in order to avoid biases that would disrupt the inner rationale of each meta heuristic, such invalid decisions are not remembered in the future. Experiments indicate that the choice of such bounding strategy varies according to the choice of the optimization algorithm. Moreover, it seems that each of the techniques could excel in identifying different types of individual behavior such as risk affine, cautious and balanced.**

I. Introduction

SIMULATING the human decision finding process can be of great use, not only for neural scientists or psychologists but also for economists. Assumptions like *rational individuals* that maximize their utility rationally while being provided with perfect information, help economists to understand and simulate reality. Though, those assumptions are very restrictive. Human behavior seems not to be fully rational, but bounded rational. In order to simulate such bounded rational human behavior, agent based economic models are becoming increasingly important [1]. Those micro economic models also assume that individuals try to rationally maximize their utility and have access to all necessary information. Since humans have to deal in reality with imperfect information (e.g uncertainty) and limited processing capacity, it is desirable, to develop methods that better resemble human decision making, in order to improve economic prediction models.

Andreas Janecek is with the University of Vienna, Austria, Research Group Theory and Applications of Algorithms. Tobias Jordan and Fernando Buarque de Lima-Neto are with University of Pernambuco, Recife/PE, Brazil, Polytechnic School of Engineering / Computing Engineering Program (email: {andreas.janecek@univie.ac.at, tobias@poli.br, fbln@ecomp.poli.br

In this paper, we present a new social simulation in which the decision of agents is based on Swarm Intelligence (SI) and evolutionary computation. While SI methods are based on the collective behavior of inherently decentralized and self-organized autonomous entities, evolutionary methods are based on Darwinian principles such as crossover, selection and mutation [2], [3]. Three different optimization algorithm are used to simulate the decisions of individuals within an agent based economic simulation model: the two well known methods Particle Swarm Optimization (PSO) [2], [4] and Differential Evolution (DE) [5], and the recently developed Fish School Search (FSS) [6] algorithm. PSO and FSS can be classified as SI methods, while DE is usually classified as an evolutionary method [1]. Although all of these algorithms are population-based meta-heuristics, they differ greatly in terms of historical information used for finding new decision. While PSO inherently exploits the information of the *best* position/decision found so far, DE computes new positions/decisions as *combination* of *other* current decisions. In both, PSO and DE, all particles perform solely independent movements, *i.e.*, there is no collective movement of the whole swarm into a specific direction. Contrary to that, FSS is partly based on individual movements but also includes *collective* movements that are based on (the quality of) current positions/decisions of all individuals.

The underlying optimization problem of this work stems from utility theory [7]. We assume that households are facing a multi-variant optimization problem with the goal to maximize their utility by choosing the respective best combination (tuple) of differently priced and differently preferred goods (summarized as *consumption*), and the required number of working hours needed in order to afford these tuple of goods. Each additionally bought good *and* each additional hour of leisure (*i.e.*, time spent not working) increases the utility.

Although the optimization algorithms used in this work do not guaranty for finding optimal solutions for the given optimization problem (all of them can be classified as *meta-heuristics*), we believe that some properties of meta-heuristics are well suited for this non trivial task. All three algorithms are in some sense able to resemble human decision making since they are based on previous and current information, and, even more importantly, also contain a probabilistic part which allows for modeling the uncertainty usually involved in the human decision making process. Similar to human choice behavior that has been proven to be heuristic [8], the swarm allows for parallel reflection and selection of different alternatives with a mixture of proba-

[1]For simplicity, we use the term "particle" to refer to population members in PSO, FSS *and* DE.

bilistic and past oriented measures. In every iteration, the meta-heuristic algorithms attempt to continuously improve the currently best solution. As a result, meta-heuristics allow for approximating the simulation process desirably close to human choices. In contrast to meta-heuristics, conventional (non-iterative) methods which are able to solve the given problem exactly (e.g., Lagrangian method) are not that well suited for simulating such a behavior. The search behavior of the three examined meta-heuristics is different. Customizing parameters of the algorithm does not lead to such a significant difference in search behavior, but seems to be well suited for more detailed approximation of search processes. Therefore, this work focuses on the examination of different heuristic search approaches.

Related work. Following the categorization of John Holland [3], *Agent Based Modeling (ABM)* refers to the computational study of social agents as evolving systems of autonomous interacting agents. ABM is a tool for studying social systems from a complex adaptive system perspective. From this perspective, a researcher is interested in how macro phenomena are emerging from micro level behavior among a heterogeneous set of interacting agents [3]. ABM has been applied in various fields, e.g., for observing racial segregation [9], political opinion building [10], consumer behavior [11], and various other fields. The research area of *Agent Based Social Simulation (ABSS)* can be found in the intersection of the three disciplines agent based computing, social sciences and computational simulation [12]. Applications of ABSS in social sciences are of Serrano Filho,Lima-Neto *et al.* in demography [13], Alvaro and Lima-Neto [14] in econometry and Jordan, Cordeiro, Lima-Neto et. al. in simulation of taxation systems [15]. One advantage of social sciences conducted with ABM is that it allows for debugging and understanding macro phenomena better, hence, allowing for simulating on an experimental base without being faced to ethical or numerical problems. A detailed summary of sociology in ABSS can be found in [16]. Emerging from this ABM approach, a particular field of research has been established: *Agent Based Macroeconomics*, *i.e.*, studying macroeconomic contexts with ABM. This type of macroeconomic simulation avoids problems with other simulation methods and gives new possibilities of research [17]. A review of this research field can be found in [18].

The applicability of SI for optimizing business processes in economics has been discovered over a decade ago, e.g., [19]. Since then, PSO has been used to improve various kinds of business models. Two recent studies focus on improving cluster analysis within a decision making model [20], and on optimizing product-mix models [21]. Another recent study discusses the applicability of computational intelligence and ABM for financial forecasting [22]. In terms of social simulation, PSO has been applied for simulating human behavior in emergency situations [23], and for insurgency warfare simulation [24]. In [25], an agent-based model for analyzing human behaviors using PSO is presented. However, this study does not focus on simulating consumer's behavior. Prior to the publication in [25], we have presented first ideas on a new agent based social simulation which uses PSO to simulate the decision finding process of consumers [26]. Although several heuristic methods for optimizing and simulating the consumer's decision making process have been proposed, e.g., [8], we are not aware of any other publication than [26] that has investigated the application of SI methods to this task. In [26], the analysis of two different bounding strategies that map particles violating the underlying budget constraints to a feasible region has revealed that the "slower" bounding strategy (which needs significantly more iterations to find the optimal solution) appears to be more appropriate for simulating the uncertainty and curiosity involved in the human decision making process. In this paper we extend our study in [26] by (i) studying the performance of two additional optimization approaches based on swarm behavior, (ii) evaluating specific bounding strategies for each of these algorithms, and (iii) analyzing the behavior of the swarm for different numbers of iterations.

Synopsis. We briefly review the applied optimization algorithms in Section II and give a formal description of the underlying decision problem in Section III. Modifications of the meta-heuristic algorithms needed in order to customize them to the decision problem are summarized in Section IV. Experimental evaluation is presented in Section V, and Section VI concludes the paper and discusses ideas for future research in this context.

II. Optimization Algorithms

The three different optimization algorithms used in this paper are briefly reviewed in this section. The applicability of these algorithms to the underlying optimization task is further discussed in Section IV.

Particle Swarm Optimization (PSO, [2], [4], cf. Algorithm 1) is a stochastic global optimization technique inspired by the social behavior of swarms, where every individual or *particle* traces a trajectory in the allowed search space. Each particle i stores its current location $\vec{l}_i$ and velocity $\vec{v}_i$, the best location it has visited so far $\vec{l}_i^b$ ("personal best"), and the best location $\vec{g}^b$ visited so far by the whole swarm ("global best"). In each iteration, the particles move through the search space based on their current weighted velocity $\vec{v}_i$ incremented or decremented by a weighted sum consisting of the differences of $\vec{l}_i^b$ and $\vec{l}_i$, and $\vec{g}_b$ and $\vec{l}_i$, respectively. At the end of each iteration, the new locations are evaluated and $\vec{l}_i^b$, and $\vec{g}_b$ are updated.

Differential Evolution (DE, [27], [5], cf. Algorithm 2) is a simple but effective stochastic function minimizer. In DE, a particle i is moved around in the search-space by using simple mathematical formulation. At each iteration, a new candidate solution for particle i is generated: a population member $p1$ is chosen randomly, and for each dimension, the weighted difference between two randomly selected population members $p2$, $p3$ is added to $p1$ with a pre-defined

Algorithm 1 – General structure of PSO

Initialize $\vec{l_i}$ and $\vec{v}_i$, the inertial weight w, and two acceleration coefficients $c1$ and $c2$
repeat
  Update velocity: $\vec{v}_i = w \cdot \vec{v}_i + c1 \cdot rand \cdot (\vec{l_i^b} - \vec{l_i}) + c2 \cdot rand \cdot (\vec{g}^b - \vec{l_i})$
  Update position: $\vec{l_i} = \vec{l_i} + \vec{v}_i$
  Evaluation of fitness of new position $f(\vec{l_i})$
  Update $(\vec{l_i^b})$ if $f(\vec{l_i}) < f(\vec{l_i^b})$, and $(\vec{g}_b)$ if $f(\vec{l_i}) < f(\vec{g}^b)$
until termination (time, iterations, convergence, ...)

probability [2]. The candidate vector is accepted for the next generation if and only if it yields a better fitness than i.

Algorithm 2 – General structure of DE

Initialize $\vec{l_i}$, differential weight F, crossover prob. CR
repeat
  For each particle i do:
  Random *selection* of particles $p1$, $p2$, $p3$
  Create crossover probability vector $\vec{R}$ based on CR
  Create candidate: $\vec{c} = \vec{l}_{p1} + \vec{R}^T \cdot F \cdot (\vec{l}_{p2} - \vec{l}_{p3})$
  Update: if $f(\vec{c}) < f(\vec{l_i})$, then $\vec{l_i} = \vec{c}$
until termination (time, iterations, convergence, ...)

The Fish School Search (FSS) algorithm [6], [28] is a recently developed SI algorithm by Bastos-Filho and Lima-Neto based on the social behavior of schools of fish. By living in swarms, the fish improve survivability of the whole group due to collaboration. Moreover, the fish perform *collective* tasks in order to achieve synergy (e.g. finding locations with more food). The location of each fish represents a possible solution and the individual success of a fish is measured by its weight — consequently, promising areas can be inferred from regions where bigger ensembles of fish are located. FSS is based on *feeding* and *swimming* operators: *Feeding* represents updating the weight of the fish based on the successfulness of the current movement. Three different *swimming* operators (individual movement, collective instinctive movement, and collective volitive movement) move the fish according to the feeding operator. For more details about this algorithm the reader is referred to [6].

III. Individual Decisions

In Microeconomics, the accepted theory is that individuals, hence households, behave rationally. This implies that purchases and labor should be in close affinity. Accordingly, we assume that each household tries to maximize its utility by either increasing consumption or leisure (*i.e.*, time spent not working). Consumption and leisure are two interacting goods — under fixed environmental conditions increasing consumption usually decreases leisure, and vice versa [26]. We define some declaration and notation in the following:

[2] cf. the crossover probability vector $\vec{R}$ in Algorithm 2: For each dimension d, $\vec{R}_d$ is set to 1 if $rand() <= CR$, and set to 0 if $rand() > CR$.

Algorithm 3 – General structure of FSS

Initialize $\vec{l_i}$, stepsize parameters, weights (w_i = 1)
repeat
  Swimming 1: **Individual** movement for each fish
  Feeding: update weights for all fish based on new locations
  Swimming 2: **Collective** instinctive movement, *i.e.*, movement towards overall direction
  Swimming 3: **Collective** volitive movement, *i.e.*, dilation from or contraction towards baricenter
until termination (time, iterations, convergence, ...)

- *Goods.* We are considering a fixed number of n distinct goods (for simplicity, we do not distinguish between basic and luxury goods). Each good does not resemble a single item but rather a collection or group of similar items, e.g., food, transport, housing, communication, education, and so on. The amount of each of the n goods consumed by an individual is abbreviated as x_i, and all goods are stored in an n-dimensional item vector $\vec{x} = (x_1, x_2, ..., x_n)$.
- *Prices.* In order to purchase each of the n goods, an individual has to pay a different price for each good (*i.e.*, for each group of items). The prices p_i for each of the n goods are stored in an n-dimensional price vector $\vec{p} = (p_1, p_2, ..., p_n)$.
- *Work.* Work is measured as the number of working hours per decision unit (day, month, year, ...) and abbreviated as w. We assume that work is limited by the maximum number of working hours per decision unit w_{max}, which is defined by external authorities (e.g. by a legal framework, medical reasons, ...). We point out that it is also possible to set w_{max} to the number of hours during some relevant period of time. However, in this case one would assume that sleeping and all other necessary daily duties count as leisure.
- *Leisure time.* In most cases, working is a "bad" for individuals, however, it is possible to measure *negative* work as a "good". *Leisure time* refers to the time an individual has available to her/him during some relevant period, *i.e.*, the time this individual does not have to work $w_{max} - w$.
- *Preferences.* Each individual has different preferences for each of the n goods, $\vec{\lambda} = (\lambda_1, \lambda_2, ..., \lambda_n)$. Additionally, each individual has some preference for leisure time (denoted as λ_{lt}). We note that the sum of all preferences (for leisure time and all goods) is equal to 1, *i.e.*, $\lambda_{lt} + \sum_{i=1}^{n} \lambda_i = 1$.

Utility Function. As underlying value allocation function for this work serves the one designed by Cobb and Douglas [29]. This function provides the model with "well-behaved" preferences, which are regarded as standard preferences for the valuation of alternatives in micro economic theory [29]. In the most-simple form an individual's utility function $u(c, l)$ has only two arguments, consumption c and leisure l, and has at least the following properties: Utility is always strictly increasing in consumption (*i.e.*, $\partial u / \partial c > 0$), and leisure (*i.e.*, $\partial u / \partial l > 0$). In the above utility function, consumption is the product of all goods to the power of the individual's preferences, *i.e.*, $c = \prod_{i=1}^{n} x_i{}^{\lambda_i}$. The second argument, leisure, is computed as leisure time to the power of the preference for leisure time, *i.e.*, $l = (w_{max} - w)^{\lambda_{lt}}$. Using a basic Cobb-Douglas utility function in the form $u(x_1, x_2) = x_1^c \times x_2^d$, utility can be computed as

$$u(c,l) = c \times l \equiv \left(\prod_{i=1}^{n} x_i{}^{\lambda_i} \right) \times (w_{max} - w)^{\lambda_{lt}} . \quad (1)$$

Constraints. The Cobb-Douglas function imposes some implicit minimum consumption constraints for each good, since the utility will decrease if either one or several of the goods, and/or leisure are smaller than 1. However, this property of the utility function should not harm our algorithm, since we consider any x_i as an agglomeration of different types of goods. A zero consumption would mean to unrealistically disclaim the consumption of one x_i in total. In addition to this implicit minimum consumption constraint, the above utility function is subject to two explicit constraints which cannot be violated:

- *Constraint 1: Maximum work time.* Work w is limited by the maximum amount of work hours per relevant period w_{max}, such that $w \leq w_{max}$.

- *Constraint 2: Limited budget.* The expenses e cannot be higher than the total salary s, such that $e \leq s$, where e is calculated as the sum of the products of the amount of each good times the price for this good, $e = \sum_{i=1}^{n} x_i \, p_i$, and s is calculated as work w times salary per hour s_h [3], $s = w \times s_h$.

IV. Agent-based meta-heuristics

In this study, we assume that each of the optimization algorithms PSO, DE and FSS works as meta-heuristic decision system *within* the mind of a single agent or individual. Mathematically, the position/location vector $\vec{l}$ of each heuristic concatenates the item-vector $\vec{x}$ and work w, the values to be optimized by the optimization algorithms. In each iteration, the swarm computes new possible decisions for the agent, and each particle resembles one considered solution. Similar to [30], we simplify the comparison between a human choice and the respective optimization algorithm by dividing the selection process in two steps, (i) finding, and (ii) evaluating candidate solutions (*i.e.*, tuple/decision).

(i) The human decision *finding* process is influenced by internal and external factors: Internal factors are, e.g., curiosity and experience of an individual. Historical positions of the algorithms can be used as means to simulate experience of an individual. Formerly successful solutions are hereby favored and new solutions might be in close proximity to remembered ones. Curiosity, as well as the huge number of external factors can be simulated by the stochastic part involved in all three algorithms, as for PSO, DE and FSS each new location is partly based on some random movement. Algorithm-specific properties and comparison to the human decision *finding* process are discussed in Section IV-B

(ii) When an individual has found a new solution or tuple, the utility of this solution is *evaluated*. Since individuals aim at maximizing their utility, the solution that is expected to provide the maximum utility is chosen. In case of the Cobb-Douglas utility function (Eq. 1), the solutions are evaluated according to the preferences the individual has for the respective good and the quantities the solution contains. For our approach, this implies that the fitness of a particles' (fish') position $\vec{l}$ is evaluated by computing the utility of this position (Eq. 1). The best position found so far is then regarded as the current solution (*i.e.*, decision) of the agent.

A. Coping with unfeasible decisions

Unfeasible decisions either involve negative consumption of a good or violate the constraints mentioned in Section III. There are three situations when a consumer decision is marked as invalid:

1) A valid decision does not allow for consuming negative amounts of any good nor a negative amount of leisure time. Hence, all negative values of $\vec{x}$ and leisure time are set to 0.

2) Constraint 1 in Section III cannot be violated. If work w increases the maximum amount of working hours w_{max}, w is reset to w_{max}.

3) Constraint 2 in Section III cannot be violated, *i.e.*, the expenses e cannot exceed the salary s.

In case of the first and/or second two situations (negative consumption, exceeded w_{max}), the invalid decision are handled automatically by all optimization algorithms. Technically, this means that a position of a particle that is out-of-bounds, *i.e.*, outside the allowed search space, is mapped back to the feasible range. More precisely, in situations (1) and (2) it is not mapped to a random position within the search space but rather bounded to the border of the search space (the lower bound in situation (1) and the upper bound in situation (2)).

Situation (3), *i.e.*, $e > s$, is handled differently by each optimization algorithm, *i.e.*, each optimization algorithm uses a specific strategy which handles such invalid solutions. We note that bounding is not the only solution to deal with invalid solutions — sometimes it is possible to allow movements to invalid solutions, however, without remembering them. In other cases it is preferable to simply discard invalid solutions. More details are discussed in the next section.

B. Algorithm-specific analysis and customization

In **PSO**, the personal best and global best position can be used as means to simulate an individuals' experience. At the beginning, PSO involves a global search strategy where new solutions can be spread across the whole search space. While this depends to some extent on the size of the parameters and the initialization of the particle's velocities, it is a typical behavior of PSO. This indicates a high level of curiosity since new locations may be far away from current locations.

[3] We are not investigating the influence of any kind of taxes (including income tax) on consumption or leisure. We assume that s_h is the disposable salary (income) per hour of an individual, *i.e.* her/his after tax income.

PSO bounding. As a result, there is a rather high probability that newly found positions are out-of-bounds. PSO deals with such situations using the "let-them-fly" bounding strategy, which actually does not map a particle to a new position, but "ignores" its current position for updating. A solution outside the allowed search space is marked as invalid and is not used for updating $\vec{l}_i^b$ nor for updating $\vec{g}_b$, respectively (cf. Line 6 in Algorithm 1).

In **DE**, each newly found location is a mathematical combination of three other locations/decisions. These three historical decisions may simulate the experience of individuals. Depending on the distribution of decisions, the new positions may be rather close to the current positions if the swarm is relatively compact, but may also be far away from the current location if the swarm is wide-spread across the search space.

DE bounding. Since each candidate solution in DE is a combination of other solutions, it is not possible to allow particles to leave the search space, as there is a high probability that future candidate solutions which are based on this solution will be invalid as well. Thus, we limit the update of DE (Line 7 in Algorithm 2) such that only valid candidate solution are considered. Compared to PSO, this limits the level of curiosity, since all new locations are restricted to the feasible search area. In the remainder of this paper we refer to this bounding strategy as "no-invalid-move".

In **FSS**, the quality of the current location *and* the weight of each fish can be used to simulate an individual's experience. In this regard the weight provides the agent with experience that a particular solution has already generated good results in the past and is therefore worth to be considered in more detail. The individual movement relies solely on the quality of current locations, while the collective movements are influenced by the fishs' weights. These collective movements are one of the main difference of FSS compared to PSO and DE. Collective movements help to keep the fish close to each other. Transcribed to the agents choice, this implicates that considered solutions are rather small variations of current decisions of the agent. Contrary to the collective movements, the individual movement represents the curiosity part and the search for new, eventually better solutions.

FSS bounding. Due to the operators of FSS, the bounding strategies of FSS are different compared to those of PSO and DE. After the individual movement, only valid solutions are considered (*i.e.*, we apply no-invalid-move strategy from DE). This implies that invalid individual movements do not contribute to the collective movements. Contrary to the individual movements, the collective instinctive movement operator and the collective volitive movement operator use the so called "bound-to-border" strategy, in which invalid solutions are bound to the border of the search space. Technically, particles are mapped to the indifference map corresponding to the current salary. This is done by computing the ratio r between salary and expenses $r = s/e$, and multiplying all elements of $\vec{x}$ element-wise with this ratio, $\vec{x} = \vec{x} \cdot r$. Since $r < 1$, the elements of $\vec{x}$ are diminished, and, as a result, the expenses are diminished until e equals s.

C. *Optimization algorithms parameters vs. human decision*

Table I gives an overview of possible interpretations of the connection between some selected features of the discussed algorithms and their corresponding interpretation in the process of human decision making.

TABLE I

POSSIBLE INTERPRETATIONS BETWEEN OPTIMIZATION HEURISTICS AND HUMAN PERSPECTIVE IN DECISION FINDING

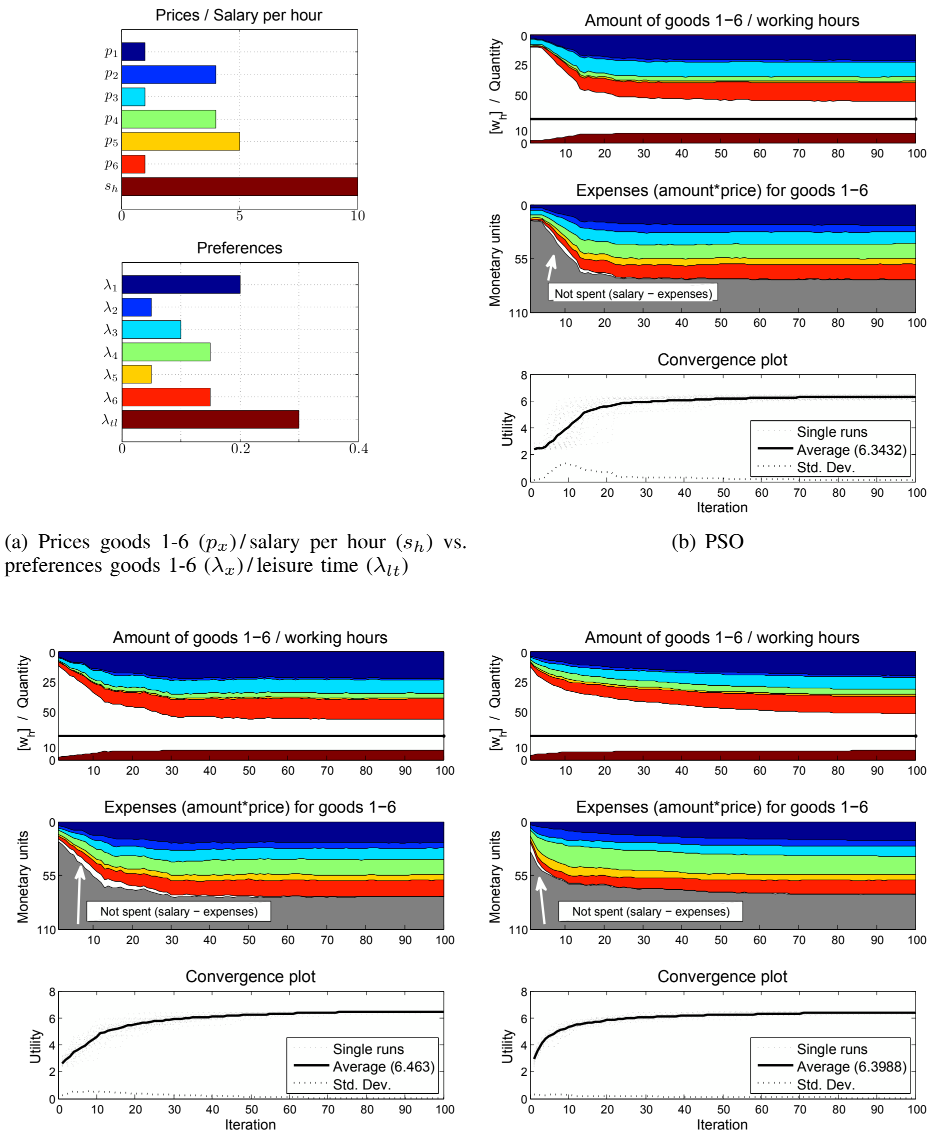
Parameter	Possible interpretation
Particle/fish within allowed range	A possible (*i.e.*, valid and affordable) human decision that might be taken into account
Update	An attempt to find a new decision with better utility (that is valid and affordable)
Velocity (for PSO) Stepsize (for FSS) Diff. weight (DE)	Individuals' curiosity, *i.e.*, the willingness to change his/her current decision. However, it is also connected with experience, since an "inexperienced" consumer is more likely to change its current decision than an "experienced" consumer
Fitness evaluation	Reflection of a specific decision
$\vec{g}^b$ and $\vec{l}_i^b$ (PSO)	A remembered solution with rather high utility; part of the human memory with links to similar decisions (*experience*)
Weight (FSS)	Quality of current movement; knowledge gained from past experience

V. EXPERIMENTAL EVALUATION

All experiments were performed for 100 iterations, using a swarm of 20 particles which resemble a reasonable number of 20 alternative tuples a human might reflect in parallel. The number of goods, prices, preferences, salary per hour s_h and preference for leisure time λ_{lt} are fixed (cf. Figure 1(a)) but can be varied easily, although a classification of goods is necessary due to the tremendous amount of possible goods. The maximum number of work hours w_{max} was set to a rather conservative value of 11 hours per day[4]. For PSO we used the *gBest* topology, $c1$ and $c2$ were set to standard values of 2.05, while ω decreased linearly with the number of iterations from 0.9 to 0.4. For DE we set CR to 0.8, while F decreased linearly with the number of iterations from 1.2 to 0.1. For FSS we used the same parameters as in [6].

Initialization. In [26] we compared two different initialization strategies to simulate the starting situation of a single agent. The results in [26] indicate that these initialization strategies do not have a significant influence on the simulation process for PSO. We found that this holds also for DE and FSS. Hence, due to space limitations we present the results only for the initialization at low work, *i.e.*, we consider that agents start their search at a low work position. All initial particles/fish reflect consumer decisions with only a small amount of working hours and therefore only low consumption — since expenses cannot extend the salary.

[4]Preferences, prices and w_{max} are fictive values, since a realistic setting for those parameters is not required in order to evaluate the performance and general applicability of the algorithms for the given optimization task.



(a) Prices goods 1-6 (p_x) / salary per hour (s_h) vs. preferences goods 1-6 (λ_x) / leisure time (λ_{lt})

(b) PSO

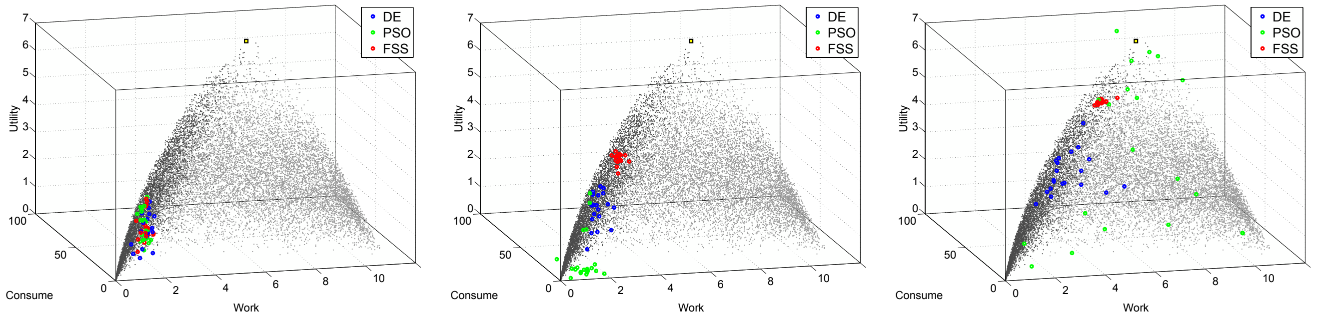
(c) DE

(d) FSS

Fig. 1. Iterative results of the best decision found so far by the whole population by each of the tested optimization algorithms.

Quantitative evaluation. The results for PSO, DE and FSS and the corresponding bounding strategies are presented in Figures 1(b) to 1(d), which show the following information as average result over 30 independent runs: (i) The first graph shows the quantity or amount of each of the six goods (cf. vector $\vec{x}$ in Section III) per iteration, as well as the number of working hours. Note the two different scales on the y-axis which are separated by the solid black line. (ii) The second graph shows the expenses in monetary units for each of the six goods (*i.e.*, $\vec{x} \times \vec{p}$). The limits of the y-axis are set to the maximum possible salary, *i.e.*, $w_{max} \times s_h = 110$. Recall that a decision at maximum possible salary corresponds to a leisure time of 0. (iii) The third graph shows the utility per iteration for all single runs, as well as the average utility and the standard deviation over all 30 runs. Overall parameters for prices and preferences are given in Figure 1(a).

Figures 1(b) to 1(d) indicate that all algorithms reach almost similar levels of utility, but differ in the path of reaching that level. In order to achieve a utility larger than 5 [6], FSS needs on average 8 [28] iterations, DE needs 12 [35] iterations, and PSO needs 14 [35] iterations. During the first 3 iterations, PSO is not able to improve its initial best decision, while DE and FSS improve already during the very first iterations. Until iteration 15, PSO is able to significantly improve its best decision, which is indicated by the steep convergence plot and fast change in consumption / work decision. DE shows similar improvements, although the changes are not as steep as for PSO. It is very interesting to notice that such abrupt changes are not present in FSS, which steadily increases its best decision without a significant alteration between iterations but with continuous improvements. The behavior of DE can be seen as an intermediate between



(a) After initialization (b) After iteration 3 (c) After iteration 10

Fig. 2. Results after given number of iterations

PSO and FSS, since convergence and decisions change more slightly compared to PSO, building less sharp edges in amount of goods and expenses diagrams. Another interesting fact that is revealed refers to the amount of monetary units that are not spent. While for DE, a relatively large amount of monetary units is not spent during the first 30 to 40 iterations, this amount is significantly smaller for PSO and almost negligible for FSS. The main reason for this behavior is the no-invalid-move bounding strategy of DE. In order to investigate the behavior of the three optimization algorithms in more detail we evaluate the distribution of the swarm during different iterations in the next section.

Swarm distribution. The distribution of particles of each algorithm after different numbers of iterations is shown in Figure 2. The search space for the agents decision consists of infinite combinations of consumption of different goods and offered hours of work. Gray and black points outline randomly created valid solutions for the given setting of salary and prices, black points present those valid solutions that incorporate combinations of work and consumption with a full spending of available salary, every point beyond that border is an invalid solution. The optimum is marked as yellow square with black border. Figures 2(a)-2(c) give deeper insight in how the algorithms move through the search space, which can be used to better understand the algorithms' behavior and allows for analyzing the suitability of each algorithm for simulating the human decision making process.

As can be seen in Figure 2(a), all three algorithms start with very similar, valid decisions which are randomly distributed in an area of low work supply. Figure 2(b) shows the distribution of the swarm after three iterations. PSO is still mainly spread in a low-work and low-utility region, while all particles (fish) in FSS already increased their decision and are collectively located at decisions close to the validity limit with significantly higher utility level compared to the initial solutions. DE particles feature a spreading and localization of particles in between PSO and FSS. After 10 iterations (Figure 2(c)) the PSO particles are widely spread across the search space, including particles in invalid areas. FSS appears extremely concentrated on a intermediary utility level and remains close to a position where the entire income is spend for consumption. In comparison, DE solutions are more concentrated than PSO solutions but more widespread than FSS, occupying the lowest utility level of the three algorithms. After 50 iterations (not shown) all swarms are more concentrated and close to the optimum. However, FSS still has the most compact swarm, while PSO continues the search also in low utility areas and in areas without valid decisions.

VI. Discussion and Conclusion

In this paper we have adopted three different optimization methods for simulating the decision finding process of consumers, namely Particle Swarm Optimization (PSO), Differential Evolution (DE) and Fish School Search (FSS). Implemented with slightly different bounding strategies to cater for feasibility of decisions, all three algorithms appeared to be suitable for the underlying problem. It became clear that the different searching behavior and distribution of swarm particles throughout the search space of the three algorithms predestine them for simulating different types of individuals. The primary goal of this paper is neither fast convergence of the optimization algorithms nor the optimality of the found solution, but rather the evaluation of adequacy for simulating human decision making.

- In this regard, PSO seems to be suited for simulating risk affine, visionary characters, indicated by the broad distribution of particles throughout the search space. This behavior, that can be observed especially during the first iterations, can be understood as an individual that considers very different consumption-work possibilities at the same time. This behavior implies that the agent also considers combinations of consumption-work that are similar to solutions that were unfeasible for him in the past.

- On the contrary, FSS appears more appropriate for cautious, systematic individuals, as the variation in considered solutions as well as the change in decisions from one iteration to the other is relatively small.

- DE arises as a solution for simulating intermediate behavior, as the behavior of DE appears to be in between PSO and FSS. In the case of DE, agents are equipped with an average risk affinity, as they are neither very flexible nor very close-minded to a single decision.

As pointed out before, the three algorithms also differ in terms of finding solutions that incorporate less consumption spending than income. Since this approach does not include inter-temporal utility, these savings cause a decrease of utility for the respective solutions. However, models dealing with savings might consider this characteristics of the algorithms

Scope of the presented study. This work has covered solely the performance of the swarm techniques for a single agent – the performance in an environment with interacting agents has not been examined yet. A more detailed adjustment of parameters and operators of the algorithms to individual decision behavior and the calibration with statistical data may contribute to a more accurate simulation. Another open question is the inter temporal impact of savings. Furthermore, the examined algorithms use heuristics for the search, but not for the evaluation of solutions. Though, in a situation of scarcity of time or processing capacity, humans may use heuristics also for the latter [30] and even for choice of applied heuristics [31]. Several interesting questions are raised for future studies: the influence of more elaborated bounding strategies that include the importance of distinct items (e.g., basic vs. luxury goods), the influence of different settings for the swarm and their contributions to a desired search behavior, the influence of different topologies, and the investigation of differently formulated fitness functions that could simulate heuristics, e.g., in accordance with research on consumer decision making in [8] or bounded rationality [30]. Moreover, the simulation of societies with social dynamics or changing parameters, the sharing of experiences of group leaders, and the application of the algorithms to real data in order to verify the reality of the obtained simulation results are further interesting research direction.

VII. Acknowledgments

This research was partially funded by Andrea von Braun Stiftung.

References

[1] N. Gilbert, "Computer simulation of social processes," *Social Research Update*, no. 6, 1994.
[2] J. Kennedy, "Swarm intelligence," *Handbook of nature-inspired and innovative computing*, pp. 187–219, 2006.
[3] J. L. Holland, *Adaption in natural and artificial systems.* MIT Press, 1992.
[4] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of IEEE International Conference on Neural Networks*, vol. 4, 1995, pp. 1942 –1948.
[5] R. Storn and K. Price, "Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
[6] C. Bastos Filho, F. B. De Lima Neto, A. Lins, A. Nascimento, and M. Lima, "Fish school search: An overview," in *Nature-Inspired Algorithms for Optimisation*, ser. Studies in Computational Intelligence, 2009, vol. 193, pp. 261–277.
[7] G. J. Stigler, "The development of utility theory. i," *Journal of Political Economy*, vol. 58, no. 4, pp. pp. 307–327, 1950. [Online]. Available: http://www.jstor.org/stable/1828885
[8] J. Bettman, E. Johnson, and J. Payne, "Consumer decision making," *Handbook of Consumer Behavior*, pp. 50–79, 1991.
[9] T. C. Schelling, "Models of segregation," *The American Economic Review*, vol. 59, no. 2, pp. 488–493, 1969.
[10] G. Deffuant, F. Amblard, and G. Weisbuch, "How can extremism prevail? a study based on the relative agreement interaction model," *Journal of Artificial Societies and Social Simulation*, vol. 5, no. 4, 2002.
[11] M. A. Janssen and W. Jager, "An integrated approach to simulating behavioural processes: A case study of the lock-in of consumption patterns," *Journal of Artificial Societies and Social Simulation*, vol. 2, no. 2, 1999.
[12] P. Davidsson, "Agent based social simulation: a computer science view," *Journal of Artificial Societies and Social Simulation*, vol. 5, 2002.
[13] M. Barbosa and F. B. de Lima Neto, "Distributed agent-based social simulations: An architecture to simulate complex social phenomena on highly parallel computational environments," in *IEEE Symposium on Intelligent Agent*, 2011, pp. 1–8.
[14] H. S. Barbosa-Filho, F. B. de Lima Neto, and W. Fusco, "Migration, communication and social networks an agent-based social simulation," in *Complex Networks*, ser. Studies in Computational Intelligence. Springer, 2013, vol. 424, pp. 67–74.
[15] T. Jordan, P. Cordeiro, F. B. de Lima Neto, A. Presse, and M. Pita, "Computer simulations of small societies under social transfer systems," in *In Proceedings of the 1st BRICS Countries and 11th CBIC Brazilian Congress on Computational Intelligence.*, 2013, in press.
[16] M. Meyer, I. Lorscheid, and K. G. Troitzsch, "The development of social simulation as reflected in the first ten years of jasss: a citation and co-citation analysis," *J. Artificial Societies and Social Simulation*, vol. 12, no. 4, 2009.
[17] D. Colander, P. Howitt, A. Kirman, A. Leijonhufvud, and P. Mehrling, "Beyond DSGE models: Toward an empirically based macroeconomics," *The American Economic Review*, vol. 98, no. 2, pp. pp. 236–240, 2008.
[18] L. Testfatsion, "Agent-based computational economics: Growing economies from the bottom up," *Artificial Life*, vol. 8, no. 1, pp. 55–82, 2002.
[19] E. Bonabeau and C. Meyer, "Swarm intelligence: a whole new way to think about business," *Harvard business review*, no. 5, 2001.
[20] J. Nenortaite and R. Butleris, "Application of particle swarm optimization algorithm to decision making model incorporating cluster analysis," in *Conference on Human System Interactions*. IEEE, 2008, pp. 88–93.
[21] S. Devi and S. K. Panigrahi, "Intelligent decision making using particle swarm optimization for optimizing product-mix model," *J. Comp. Sc. & Inf.*, vol. 1, 2011.
[22] M. Kampouridis, "Computational intelligence in financial forecasting and agent-based modeling: Applications of genetic programming and self-organizing maps," PhD Thesis. University of Essex, Tech. Rep., 2011.
[23] Z. Xue, "A particle swarm optimization based behavioral and probabilistic fire evacuation model incorporating fire hazards and human behaviors," 2006, m.S. Thesis, State University at Buffalo/NY.
[24] X. Cui and T. Potok, "A particle swarm social model for multi-agent based insurgency warfare simulation," in *Software Engineering Research, Management Applications, 2007. SERA 2007. 5th ACIS International Conference on*, 2007, pp. 177–183.
[25] M. Noto, H. Kannabe, T. Morizumi, and H. Kinoshita, "Agent-based social simulation model for analyzing human behaviors using particle swarm optimization," *International Journal of Computer and Information Technology*, vol. 02, pp. 850–859, 2013.
[26] A. Janecek, T. Jordan, and F. Lima-Neto, "Agent-based social simulation and pso," in *Advances in Swarm Intelligence*, ser. Lecture Notes in Computer Science, Y. Tan, Y. Shi, and H. Mo, Eds. Springer Berlin Heidelberg, 2013, vol. 7929, pp. 63–73.
[27] K. V. Price, R. M. Storn, and J. A. Lampinen, *Differential Evolution A Practical Approach to Global Optimization*. Springer-Verlag, 2005.
[28] C. Bastos Filho, F. B. de Lima Neto, A. Lins, A. Nascimento, and M. Lima, "A novel search algorithm based on fish school behavior," in *Systems, Man and Cybernetics, 2008. SMC 2008. IEEE International Conference on*. IEEE, 2008, pp. 2646–2651.
[29] W. Meeusen and J. van Den Broeck, "Efficiency estimation from cobb-douglas production functions with composed error," *Int. economic review*, pp. 435–444, 1977.
[30] P. M. Todd and G. Gigerenzer, "Bounding rationality to the world," *Journal of Economic Psychology*, vol. 24, no. 2, pp. 143–165, 2003.
[31] W. B. Arthur, "Inductive reasoning and bounded rationality," *The American economic review*, vol. 84, no. 2, pp. 406–411, 1994.