Exploring social theory integration in agent-based modelling using multi-objective grammatical evolution

Tuong Manh Vu [1] **Charlotte Buckley** [2] **João A. Duro** [2] **Robin C. Purshouse** [2]

Abstract

In Generative Social Science, modellers design agents at the micro-level to generate at the macro-level a target social phenomenon. In the Inverse Generative Social Science (iGSS), from a target phenomenon, the goal is to search for possible explanatory model structures. This model discovery process is a promising tool to improve the explanatory capability and theory exploration of computational social science. This paper presents a framework for iGSS and applies Grammatical Evolution to an empirically-calibrated agent-based model of alcohol use. Results of the model discovery process find many alternative rules for agent behaviours with different trade-offs. Future work should involve domain experts to evaluate the discovered structures in terms of theoretical credibility and knowledge contribution.

1. Introduction

1.1. Inverse generative social science

Agent-based modeling (ABM) is a bottom-up methodology that models a complex system as a collection of heterogeneous agents and their interactions. ABM has become an established tool for *Generative Social Science* (Epstein, 1999). If an agent-based model with a defined set of mechanisms can produce the target social phenomenon, the model is a candidate explanation for the phenomenon. However, there can be other models that can generate the same target phenomenon. In the inverse problem, from a target phenomenon, the aim is to find possible explanatory agent-based models. The process is known as *Inverse Generative Social Science* (iGSS) or *model discovery*. In the development of agent-based models, modellers usually *manually* experiment with different sets of mechanisms, often derived from subjective interpretations of theory. The iGSS methodology allows modellers to explore the space of model structures *automatically* with the aim of better explanatory capability. iGSS usually utilizes techniques from Artificial Intelligence and provides a powerful tool for social scientists to explore combinations of social theories or even develop new social theories.

1.2. Existing works on iGSS

The nascent field of iGSS has limited literatures. Smith (2008) used a genetic algorithm to identify simplified rules of behavior that could reproduce the observed social assortativity of birds. Zhong et al. (2014) used gene expression programming to identify a individual reward function to reproduce observed human crowd dynamics. Gunaratne & Garibay (2017; 2020) used genetic programming to revise agents' farm selection rules to reproduce the archeological population demography. Vu et al. (2019) used Genetic Programming to identify alternative situational mechanisms for a social norms model of alcohol use. In these works, the model discovery limits to a single rule in agent decision making, usually manipulating a single equation. Recently, Greig & Arranz (2021) extends that by working with a series of equations, basic mathematical and conditional operations in two examples of flocking and opinion dynamics.

Finally, instead of focusing on a single mechanism, Vu et al. (2020) took a step further by manipulating multiple mechanisms of agent drinking decision within a social roles model of alcohol use. A limitation of this work is that it includes only few constants in the model discovery process. This means that, for every discovered alternative structure, there is a limit to the ability of the model discovery process to improve the fitness further. Ideally, for each discovered structure, a parameter calibration should be carried out to accurately evaluate its fitness. In this paper, we want to propose a solution to this issue by including parameter distribution selection in the model discovery process.

---

[1] School of Health and Related Research, University of Sheffield, Sheffield, United Kingdom [2] Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield, United Kingdom. Correspondence to: Tuong Manh Vu <t.vu@sheffield.ac.uk>, Robin C. Purshouse <r.purshouse@sheffield.ac.uk>.

AI for Agent-Based Modelling (AI4ABM) Workshop, 2022. Copyright 2022 by the author(s).

1.3. Aim and organisation of the paper

This paper aims to demonstrate a bi-level model discovery process that explore both model structures and parameter distributions to explain a social phenomenon. As a case study, the paper uses an established agent-based model based on social norm theory.

Section 2 briefly introduces the existing model of social norm theory for alcohol use. Section 3 provides background on multi-objective Grammatical Evolution. In Section 4, the model discovery methodology is explained. Results and discussion of the process are provided in Section 5. Lastly, Section 6 concludes the paper and suggests possible future works.

2. A calibrated norm-theory agent-based model of alcohol use

This paper uses an agent-based model of alcohol use described in detail in Buckley et al. (2022). The individual decision making was based on frequently used psychological models of behaviour and uses social norm theory to explain alcohol use. The agents in the model are individuals, and are assumed to be representative of adults in the US. Agent drinking behaviors are typologized using five schema representing different daily drinking practices: abstaining, light drinking, moderate drinking, heavy drinking and very heavy drinking. The following explains the agent drinking decision, including several important concepts that will be used later on in the model discovery process.

The decision process begins with a probabilistic selection process between two pathways: habitual or intentional pathway. The trigger of a pathway is based individual *automaticity*. Automaticity ranges from 0 (always intentional) to 1 (always habitual). If a sampled random number between 0 and 1 is lower than an individual's automaticity, they will behave according to their previous drinking patterns on that day. Otherwise, the intention pathway is triggered, and intentions will be calculated to determine the probability of different drinking decisions.

In the intention pathway, social norm theory was implemented. The drinking decision is affected by descriptive norms, injunctive norms and autonomy. The injunctive norm refers to the perceived acceptability of drinking in society for an individual like them (based on membership of socio-demographic subgroups). The descriptive norm is an individual's appraisal of actual drinking behavior for an individual like them. Autonomy refers to the individual's desire to ignore the norms. An autonomy of 0.6 would indicate that the agent only pays attention to norms 40% of the time.

The model was calibrated to the US Behavioral Risk Factor Surveillance System (BRFSS) survey data for New York, adjusted to per-capita New York State alcohol sales data for each year using a method described in Rehm et al. (2010). Three alcohol use targets were defined for each year describing (1) prevalence — the overall proportion of individuals consuming alcohol at least once during the previous year, (2) quantity — the average grams of alcohol consumed per day among drinkers, and (3) frequency — the average number of drinking days per month among drinkers. Targets were calculated separately for each year and split by sex. Models were calibrated using data for the years 1984-2010 and validated using reserved data from 2011-2015.

3. Multi-objective Grammatical Evolution

Genetic programming (GP) (Koza, 1992) is used to generate computer programs[1] automatically with the help of genetic operators. It is usually applied to problems where there is an underlying requirement for structural optimization, besides the need (or not) to find the optimal parameters of the problem. An example is the design of a digital filter where besides having to find a set of optimal filter parameters, one also needs to determine the order of the filter. Although traditional *genetic algorithms* (GAs) are well suited for finding the optimal parameters of an optimization problem, they are not able to represent the structural requirements of a solution in chromosomes. A chromosome in GP besides supporting constants, it has provision for variables and functions (including algebraic operators, such as $+$ and $-$), which can be combined in different ways to generate a *syntax tree*. In a similar fashion as it is done for GAs, a GP algorithm evolves a population of computer programs (or candidates) over many generations where selection and variation operators (e.g. crossover and mutation) take turns to find the fittest individuals. Traditionally in GP, Lisp prefix notation was used to represent a syntax tree in a chromosome but it has several shortcomings, such as the chromosome can have variable length, and that it is very easy for crossover and mutation operators to generate illegal offsprings. To mitigate these shortcomings, this paper uses Grammatical Evolution (GE) (O'Neill & Ryan, 2001; 2003) which relies on the Backus–Naur form (BNF) syntax. This ensures chromosomes with a fixed length, restricts the search space in a way that it is possible to apply standard crossover and mutation operators more freely, and prevents the generation of illegal offsprings.

Multi-objective evolutionary algorithms (MOEAs) are very popular nowadays for dealing with multi-objective optimization problems. An MOEA evolves a set of solutions (also known as a population) over several generations by applying operators based on the principles of natural evolution

---

[1] A computer program in this context could be an algorithm, a machine or even a brain.

such as selection, crossover, and mutation, until some termination criterion is satisfied (e.g. a maximum number of generations has been exceeded). There are many MOEAs in the literature and existing ones can be categorised as Pareto based, decomposition based and indicator based. For a more detailed discussion about the different types of MOEAs, including their strengths and weaknesses, the reader is referred to a recent tutorial by Emmerich & Deutz (2018). In this study we employ a GP version of a popular Pareto based MOEAs known as NSGA-II (Deb et al., 2002), and the implementation can be found in the PonyGE2 toolkit (Fenton et al., 2017). NSGA-II relies on a non-dominated sorting algorithm to assign ranks to solutions, and on a diversity preservation mechanism that ensures a good spread across the Pareto frontier.

To evaluate the performance of a non-dominated solution set obtained by a multi-objective optimization algorithm there are several performance indicators in the literature that could be used. The hypervolume indicator (Zitzler & Thiele, 1998) is very popular in the evolutionary multi-objective optimization community and will be used in this study. This is also known as the Lebesgue measure and it is determined by quantifying the region in the objective space enclosed by the front of the non-dominated solutions and an upper bounded reference point (assuming minimisation). Improving the hypervolume means increasing the area between the non-dominated solutions and the reference point, and this could be achieved by either improving the convergence (i.e. solutions with better performance with respect to both objectives) or improving the spread across the front (i.e. generating more evenly spread non-dominated solutions). To determine the exact hypervolume value we use a dimension-sweep algorithm (Fonseca et al., 2006).

4. Model discovery method

4.1. Overview of model discovery process

A schematic of the overall model discovery process is shown in Figure 1. Step 0 represents a pre-condition for the model discovery process. We defined a library of theory building blocks implemented as model components and a grammar to guide the search process. Step 1 to 6 is the Grammatical Evolution process. In Step 1, a initialized population of models is generated. In Step 2, variation operators, e.g. crossover and mutation, are applied to produce new candidate models. In Step 3, the models in the current population are evaluated for generative sufficiency. Step 4 selects the models to retain in the population. If convergence is not achieved, go back to Step 2. If the convergence is achieved, the Pareto optimal model structures will be assessed for their theoretical contribution. Afterward, if improvement is needed, the whole process can be restarted by adjusting the grammar or the library of components.

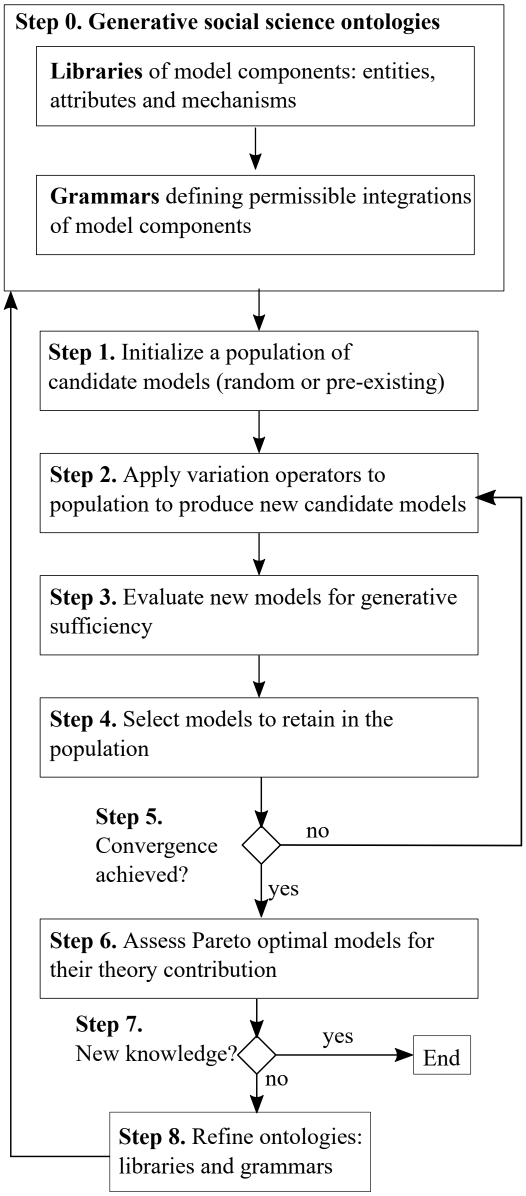

Figure 1. Model discovery process

To search for the combinations of social theories that offer the best explanation to the macro-level phenomenon (drinking behaviors in New York State), we make use of GE with *bi-level multi-objective optimization*. The model structure and the model parameters are separated in a bi-level formulation, where for each model structure identified by the GE algorithm at the upper-level, a separate calibration process is conducted to identify the best parameters at the lower-level. For clarification, all model parameters were calibrated in the original model (Section 2), only some parameters relevant to the alternative structure will be *re-calibrated* in the model discovery process. This is known as a *nested approach* in the bi-level optimization literature (Sinha et al., 2018), which is commonly used in conjunction with evolutionary computation algorithms, such as genetic algorithms, differential evolution, and swarm intelligence. However, to our knowledge, this is the first time that this approach is used in conjunction with a GE algorithm.

4.2. Objectives

We use the following three objectives in this study:

First, we consider the overall goodness-of-fit between the model output and the target data, accounting for sampling uncertainty in both the output and target measurement. Here, our objective function for model structure $\mathbf{x}$ is an *implausibility* metric (Vernon et al., 2010) that measures the error between the mth simulated output $y^{\star}_{m}$, averaged over N model replications, and equivalent empirical target data y_m over a sequence of temporal observations k defined by:

$$z_1 = \frac{1}{KM} \sum_{k=1}^{K} \sum_{m=1}^{M} \frac{\left| \left(\frac{1}{N} \sum_{n=1}^{N} y^{\star}_{m}[k]_n \right) - y_m[k] \right|}{\sqrt{(s_m[k])^2 + (d_m)^2}} \quad (1)$$

where M is the number of output measures, K is the number of observations, $s_m[k]$ is the observed standard error for output m at time point k, and $(d_m)^2$ is the variance of the model discrepancy for output m, which is taken as 10% of the possible output range for each output. 'Model discrepancy' is the error in a model output that arises because the model is not a perfect representation of reality.

Since we are interested in whether the iGSS process might work better to fit to targets based on male drinking patterns than female drinking patterns, and vice versa, we also decompose this first objective into two sub-objectives:

$$z_{1,\text{male}} = \frac{1}{KM_{\text{male}}} \sum_{k=1}^{K} \sum_{m=1}^{M_{\text{male}}} \frac{\left| \left(\frac{1}{N} \sum_{n=1}^{N} y^{\star}_{m}[k]_n \right) - y_m[k] \right|}{\sqrt{(s_m[k])^2 + (d_m)^2}} \quad (2)$$

$$z_{1,\text{female}} = \frac{1}{KM_{\text{female}}} \sum_{k=1}^{K} \sum_{m=1}^{M_{\text{female}}} \frac{\left| \left(\frac{1}{N} \sum_{n=1}^{N} y^{\star}_{m}[k]_n \right) - y_m[k] \right|}{\sqrt{(s_m[k])^2 + (d_m)^2}} \quad (3)$$

where M_{male} and M_{female} are the subset of targets relating to men and women respectively.

The final objective we consider is a complexity measure, which aims to promote model parsimony for the purposes of interpretability and avoiding over-fitting. This type of approach was introduced to genetic programming by (Rodríguez-Vázquez et al., 2004) to combat the issue of *bloat*—where the tree lengths in the genetic program tend to drift upwards over time.

We calculate interpretability therefore as the number of nodes in the grammar-based tree:

$$z_2 = \texttt{nodes}(\mathbf{x}), \quad (4)$$

where `nodes(.)` calculates the number of nodes in the tree encoding model structure $\mathbf{x}$.

4.3. Bi-level optimization

In this study, our interest is to find *model structures* and *parameters* that offer the best trade-offs between the complexity of the model and the goodness-of-fit with respect to multiple phenomenon targets. In total there are three objectives in this GE problem that has to be dealt with simultaneously, and these are: patterns of female alcohol use; patterns of male alcohol use, and complexity. When dealing with multiple conflicting objectives the problem does not contain a single optimal solution, which is often the case with single-objective problems. Besides the existence of a single optimal solution for each objective function, there is also a set of trade-off solutions where an improvement in one objective is earned at the expense (deterioration) of another objective(s). The best trade-offs in terms of Pareto-optimality can be captured by the concept of dominance. Consider two solutions $\mathbf{a} \in \mathcal{R}^M$ and $\mathbf{b} \in \mathcal{R}^M$ where M is the number of objectives, $\mathbf{a}$ is said to dominate $\mathbf{b}$ if $\mathbf{a}$ is not worse than $\mathbf{b}$ in all objectives (i.e. $a_i \leq b_i \ \forall i = 1, \ldots, M$), and $\mathbf{a}$ is strictly better than $\mathbf{b}$ in at least one objective (i.e. $\exists_{i \in \{1,\ldots,M\}} : a_i < b_i$). From a set of solutions, any subset that contains only solutions that are not dominated by any other solution in the set constitutes a non-dominated set, and the non-dominated set with respect to the entire search space is known as the Pareto-optimal set. The next section will describe in detail how the GE incorporates both model structures and parameters via the use of grammar.

4.4. Grammar

The grammar describes what components in agent behavioural rules will be included and how they can be combined together. This study focuses on agents' daily decision of the number of alcoholic drinks to consume (Section 2). The structure of agents' intention pathway will be exposed to the model discovery process. It is a function that decides the number of drinks based on agent desire to drink and social norm theory concepts (injunctive norm, descriptive norm).

The grammar also includes the individual parameters of the agents that are relevant to the number-of-drinks decision: the *automaticity* that control the trigger between habitual and intentional pathway and the *autonomy* which refers to individual desire to ignore the norm. For these two parameters, the grammar will have several options for their distributions in all agents.

Listing 1 describes the grammar that guides the model discovery process. Each candidate (a program $< p >$) contains 10 expressions: 1 for the log odds intention, 3 distribution options for the automaticity of three different drinker types, and 6 distribution options for the autonomy of three different drinker types split by male and female. Each expression can be formed only by a defined combination of expressions, variables, constants, and distribution options. This hierarchical grammar captures the complexity of different expressions.

Listing 1. Grammar for the model discovery process

```
<p> ::= LogOddsIntention=<lo_exp>;
    automaticity.low=<automaticity_opt>;
    automaticity.med=<automaticity_opt>;
    automaticity.high=<automaticity_opt>;
    autonomy.male.abstainer=<autonomy_opt>;
    autonomy.male.moderate=<autonomy_opt>;
    autonomy.male.heavy=<autonomy_opt>;
    autonomy.female.abstainer=<autonomy_opt>;
    autonomy.female.moderate=<autonomy_opt>;
    autonomy.female.heavy=<autonomy_opt>;

<lo_exp> ::= <vc>*log(<odds>) |
    -<vc>*log(<odds>) |
    (<lo_exp> + <lo_exp>)

<odds> ::= odds(<exp>)

<exp> ::= Desire | Descriptive | Injunctive |
    (<exp> + <exp>) | <vc>*(<exp>) |
    sqrt(<exp>) | pow(<exp>,2)

<vc> ::= <v_autonomy> | <c_01> |
    <c_01>*<v_autonomy> |
    sqrt(<vc>) |
    pow(<vc>,2)

<v_autonomy> ::= Autonomy | (1-Autonomy)

<c_01> ::= 0.<d0><d0> | 1.<d1><d1>

<d0> ::= 0|1|2|3|4|5|6|7|8|9

<d1> ::= 0

<automaticity_opt> ::= skewed_right |
    centre | skewed_left | bi_modal

<autonomy_opt> ::= skewed_right |
    centre | skewed_left | bi_modal
```

The intention pathway uses a log odds function for each schema because, in the model, the probabilities of instantiating each drinking schema are represented using a multinomial logit equation. The log odds expression `<lo_exp>` can be modified by multiplying log odds with a variable/constant expression `<vc>`, or by summing two log odds. The variable/constant expression `<vc>` can be constructed from `Autonomy`, `(1-Autonomy)`, constant between 0 and 1 `<c_01>`, taking a square root, or raising to the power of 2. For the expression within the odds `<exp>`, it can be constructed with norm-theory concepts `Desire`, `Descriptive`, `Injunctive` and several operation (sum, multiplication, square root, raising to the power of 2). For simplification, automaticity and autonomy distributions can be selected from four distribution options `<autonomy_opt>`: skewed-right, centre, skewed-left, bi-modal.

5. Results and Discussion

The model discovery evolutionary setup was as follows: 500 candidates per population for 50 generations, 75% subtree crossover, 25% subtree mutation, and other default settings of PonyGE2. It is computationally intensive to do a complete run of GE; the process took about 60 hours on an Intel i9 9980XE processor with 36 cores.

The hypervolume plot, Figure 2, shows the convergence of the model discovery process over 50 generations. There are 97 non-dominated model structures in the Pareto front of the last generation. However, some structures are duplicated (in terms of the three fitness) because they have the equivalent functional form of log odds intention, and the simulation was run with the same seed. After removing the duplicated structures, there are 82 unique structures. Figure 3 shows the trade-offs between three objectives (male goodness-of-fit, female goodness-of-fit, and complexity) of 82 non-dominated structures in a parallel coordinates plot. Because of the trade-offs between objectives, model selection is not a straightforward process.

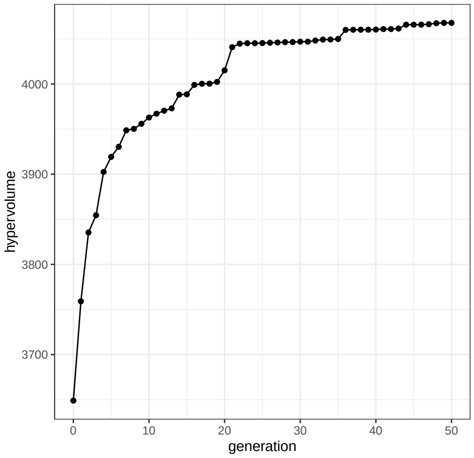

Figure 2. Hypervolume convergence plot

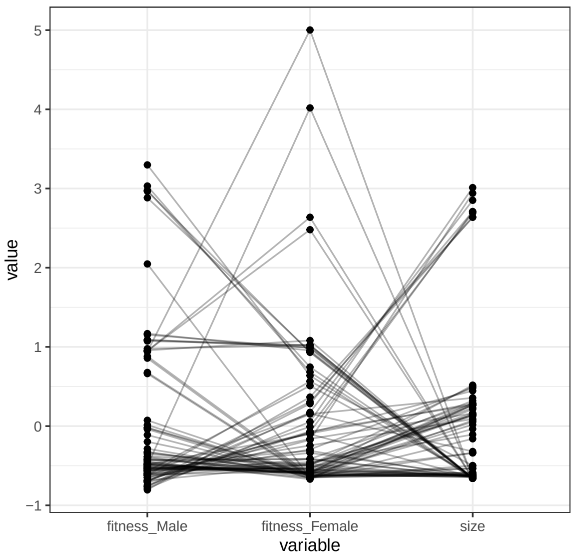

Figure 3. Parallel coordinates plot of the structures on the Pareto front

Let us discuss the model structures in detail. There are two parts the log odds intention and the parameters. The discussion will focus on the structure of the log odds intention. In the non-dominated structures, there are structures with the same log odds intention but different parameters which leads to a slightly different goodness-of-fit. So when showing an example of a log odds intention, the model structure with the least average goodness-of-fit of male and female will be presented.

The simplest structure has a complexity measure of 26, and the most complex one is 232. Bringing attention to the norm-theory concepts. For structures with a complexity of less than 35, most structures only contain Desire (for example, Listing 2). The only different case is that Injunctive appears by itself in the simplest structure of 26 (Listing 3). These Injunctive-only structures result in unbalanced goodness-of-fit: low male goodness-of-fit but high female goodness-of-fit (0.888 and 2.255 on average). On the other hand, Desire-only structures have a balance goodness-of-fit between male and female (1.578 and 1.470 on average). In the context of social norm theory, when the log odds intention consists of only Desire and not Injunctive and Descriptive norm, agents follow their past behaviour initialised at baseline.

As the complexity increase, injunctive and descriptive norms start to appear. In complexity from 44 to 71, there is either Injunctive or Descriptive in the log odds intention. From complexity 72, both appear at the same time in most structures. This means that to achieve better male and female fitness (as the complexity is higher), injunctive and descriptive norms must be included. It is noted that Desire is still a necessary component since it is present in all structures with size greater than 27.

Listing 2. Non-dominated structure id 95

```
LogOddsIntention=Autonomy*log(odds(Desire));
automaticity.low=skewed_right;
automaticity.med=skewed_left;
automaticity.high=skewed_left;
autonomy.male.abstainer=skewed_left;
autonomy.male.moderate=skewed_left;
autonomy.male.heavy=skewed_left;
autonomy.female.abstainer=skewed_left;
autonomy.female.moderate=skewed_left;
autonomy.female.heavy=bi_modal;

Fitness: [1.577748, 1.381777, 26]
```

Listing 3. Non-dominated structure id 84

```
LogOddsIntention=(1-Autonomy) *
    log(odds(Injunctive));
automaticity.low=bi_modal;
automaticity.med=bi_modal;
automaticity.high=centre;
autonomy.male.abstainer=centre;
autonomy.male.moderate=skewed_right;
autonomy.male.heavy=skewed_right;
autonomy.female.abstainer=skewed_right;
autonomy.female.moderate=skewed_right;
autonomy.female.heavy=bi_modal;

Fitness: [0.8686032, 1.668882, 26]
```

Table 1 summarizes the count of Desire, Injunctive, and Descriptive in the log odds intention function, averaged over the number of structures in different ranges of complexity. The count of all three concepts increases as the complexity increases. Desire concepts are always greater than the other two concepts. Injunctive and Descriptive counts are not greatly different.

Table 1. Average count of norm-theory concepts over different ranges of complexity

	AVERAGE OF COUNT		
COMPLEXITY	DESIRE	INJUNCTIVE	DESCRIPTIVE
20-39	0.77	0.23	0.00
40-59	2.00	0.20	0.60
60-79	4.05	1.85	0.50
80-99	4.86	2.29	1.29
100+	9.13	3.13	3.88

Models with high complexity are extremely difficult to interpret. For example, Listings 4 shows a model structure with a complexity of 72. The log odds intention is complex but still manageable. It shows intention as a function of Desire, the interaction of Desire and Injunctive, and the interaction of Desire and Descriptive. However, looking at

Listing 4. Non-dominated structure id 28

```
LogOddsIntention=(-1.00*(1-Autonomy)*
    log(odds(Desire)) +
    ((0.71*log(odds((sqrt(Desire) +
    (Injunctive + Injunctive)))) +
    Autonomy*log(odds(Autonomy*
    ((Descriptive + Descriptive))))) +
    Autonomy*log(odds(Desire))));
automaticity.low=bi_modal;
automaticity.med=skewed_left;
automaticity.high=skewed_left;
autonomy.male.abstainer=skewed_left;
autonomy.male.moderate=skewed_left;
autonomy.male.heavy=skewed_right;
autonomy.female.abstainer=skewed_left;
autonomy.female.moderate=skewed_right;
autonomy.female.heavy=skewed_left;

Fitness: [0.3398256, 0.3983111, 72]
```

the most complex structure in Listing 5, it is near impossible to interpret the function. Thus, if the modellers only want the best goodness-of-fit and does not interest in theoretical interpretability, complex structures are selected. However, to interpret the agent behaviour and underlying social theory, it is better to pick simpler structures. Other strategies can be used during model discovery process for theoretical interpretability include simplifying the function before evaluation, limiting the tree depth in GE configuration, or restricting the grammar. With the grammar, modellers have to be careful to balance exploration capability and interpretability.

6. Conclusion

This paper presents a model discovery method that utilizes multi-objective GE to explore different combinations of social theory concepts and parameter distributions to search for alternative agent behavioural rules. The case study of alcohol use modeling has shown that different realizations of theoretical mechanisms can result in trade-offs between different metrics. Our model discovery method offers a promising approach to generating novel combinations of social theory components. In the research frontier of inverse generative social science, our work shows the feasibility of performing a bi-level optimization, i.e. joint structure-parameter model discovery. Future research should involve domain experts to interpret the discovered model structures. This work also highlights the challenge of meaningful model discovery, requiring the involvement of the domain expert during grammar design as well as the theoretical credibility assessment.

Listing 5. Log odds intention of the most complex structure id 1

```
LogOddsIntention=(((((-0.35*(1-Autonomy)*
    log(odds(1.00*(1.00*
    ((pow((sqrt(Descriptive) + Desire),2) +
    sqrt(pow(Desire,2))))))) +
    pow(1.00*Autonomy,2)*
    log(odds(pow(pow(1.00,2)*
    (Descriptive),2)))) +
    -(1-Autonomy)*log(odds(Injunctive))) +
    (pow(Autonomy,2)*log(odds(Desire)) +
    ((pow(pow((1-Autonomy),2),2)*
    log(odds(Desire)) +
    (-1.00*Autonomy*log(odds(sqrt(Desire))) +
    -pow(Autonomy,2)*log(
    odds(pow(sqrt(Autonomy),2)*
    (sqrt(pow(Desire,2))))))) +
    (-sqrt(Autonomy)*log(odds(
    pow((Desire + Descriptive),2))) +
    sqrt(sqrt(0.49*Autonomy))*
    log(odds(pow(0.27*Autonomy,2)*
    (sqrt(sqrt(pow(Desire,2)))))))))) +
    (1.00*log(odds(Injunctive)) +
    (Autonomy*log(odds(
    pow(sqrt(Descriptive),2))) +
    pow(pow(pow(0.60,2),2),2)*
    log(odds(Desire))))) +
    sqrt(sqrt(pow(sqrt(1.00*Autonomy),2)))*
    log(odds((Injunctive + Desire))))
```

References

Buckley, C., Field, M., Vu, T. M., Brennan, A., Greenfield, T. K., Meier, P. S., Nielsen, A., Probst, C., Shuper, P. A., and Purshouse, R. C. An integrated dual process simulation model of alcohol use behaviours in individuals, with application to us population-level consumption, 1984–2012. *Addictive Behaviors*, 124:107094, 2022. ISSN 0306-4603. doi: 10.1016/j.addbeh.2021.107094.

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2): 182–197, 2002. doi: 10.1109/4235.996017.

Emmerich, M. T. M. and Deutz, A. H. A tutorial on multiobjective optimization: fundamentals and evolutionary methods. *Natural Computing*, 17:585–609, 2018.

Epstein, J. M. Agent-based computational models and generative social science. *Complexity*, 4(5):41–60, 1999. doi: 10.1002/(SICI)1099-0526(199905/06)4:5⟨41::AID-CPLX9⟩3.0.CO;2-F.

Fenton, M., McDermott, J., Fagan, D., Forstenlechner, S., Hemberg, E., and O'Neill, M. Ponyge2: Grammatical evolution in python. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, GECCO '17, pp. 1194–1201, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450349390. doi: 10.1145/3067695.3082469.

Fonseca, C. M., Paquete, L., and López-Ibáñez, M. An improved dimension–sweep algorithm for the hypervolume indicator. In *Proceedings of the 2006 Congress on Evolutionary Computation (CEC'06)*, pp. 1157–1163. IEEE Press, 2006. doi: 10.1109/CEC.2006.1688440.

Greig, R. and Arranz, J. Generating agent based models from scratch with genetic programming. In *ALIFE 2021: The 2021 Conference on Artificial Life*. MIT Press, 2021.

Gunaratne, C. and Garibay, I. Alternate social theory discovery using genetic programming: towards better understanding the artificial anasazi. In *GECCO 2017 - Proceedings of the 2017 Genetic and Evolutionary Computation Conference*, pp. 115–122, Berlin, Germany, 2017. ACM Press. doi: 10.1145/3071178.3071332.

Gunaratne, C. and Garibay, I. Evolutionary model discovery of causal factors behind the socio-agricultural behavior of the ancestral pueblo. *Plos one*, 15(12):e0239922, 2020.

Koza, J. R. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA, 1992. ISBN 9780262111706.

O'Neill, M. and Ryan, C. Grammatical evolution. *IEEE Transactions on Evolutionary Computation*, 5(4):349–358, 2001. doi: 10.1109/4235.942529.

O'Neill, M. and Ryan, C. *Grammatical Evolution: Evolutionary Automatic Programming in an Arbitrary Language*. Springer US, 2003. doi: 10.1007/978-1-4615-0447-4.

Rehm, J., Kehoe, T., Gmel, G., Stinson, F., Grant, B., and Gmel, G. Statistical modeling of volume of alcohol exposure for epidemiological studies of population health: the us example. *Population Health Metrics*, 8(1):1–12, 2010.

Rodríguez-Vázquez, K., Fonseca, C., and Fleming, P. Identifying the structure of nonlinear dynamic systems using multiobjective genetic programming. *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans.*, 34(4):531–545, 2004. doi: 10.1109/TSMCA.2004.826299.

Sinha, A., Malo, P., and Deb, K. A review on bilevel optimization: From classical to evolutionary approaches and applications. *IEEE Transactions on Evolutionary Computation*, 22(2):276–295, 2018. doi: 10.1109/TEVC.2017.2712906.

Smith, V. Evolving an agent-based model to probe behavioral rules in flocks of cowbirds. In *Artificial Life XI: Proceedings of the 11th International Conference on the Simulation and Synthesis of Living Systems, ALIFE 2008*, pp. 561–568, 2008.

Vernon, I., Goldstein, M., and Bower, R. Galaxy formation: A Bayesian uncertainty analysis. *Bayesian Analysis*, 5 (4):619–670, 2010. doi: 10.1214/10-BA524.

Vu, T. M., Probst, C., Epstein, J. M., Brennan, A., Strong, M., and Purshouse, R. C. Toward inverse generative social science using multi-objective genetic programming. In *Proceedings of the genetic and evolutionary computation conference*, pp. 1356–1363, 2019.

Vu, T. M., Buckley, C., Bai, H., Nielsen, A., Probst, C., Brennan, A., Shuper, P., Strong, M., and Purshouse, R. C. Multiobjective genetic programming can improve the explanatory capabilities of mechanism-based models of social systems. *Complexity*, pp. 1–20, 2020. doi: 10.1155/2020/8923197.

Zhong, J., Luo, L., Cai, W., and Lees, M. Automatic rule identification for agent-based crowd models through gene expression programming. In *13th International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2014*, volume 2, pp. 1125–1132, 2014.

Zitzler, E. and Thiele, L. Multiobjective optimization using evolutionary algorithms — a comparative case study. In *Parallel Problem Solving from Nature — PPSN V*, pp. 292–301. Springer Berlin Heidelberg, 1998.