Advances in Cognitive Systems 9 (2018) 211–230 Submitted 6/2018; published 9/2018

The Role of Goal Ranking and Mood-Based Utility in Dynamic Replanning Strategies

Bart Gajderowicz BARTG@MIE.UTORONTO.CA
Mark S. Fox MSF@MIE.UTORONTO.CA
Michael Grüninger GRUNINGER@MIE.UTORONTO.CA
Department of Mechanical and Industrial Engineering, University of Toronto, 5 King's College Road,Toronto, ON M5S 3G8, Canada

Abstract

In classical AI planning, replanning strategies are used to reevaluate a plan during execution. For human-like agents, goal preferences and emotions play an important role in evaluating a plan's progress. However, most existing systems rely on predefined goal ordering and a static association between an event and its emotion-based utility calculation. During execution, utility of individual actions are used to trigger the replanning process. This approach assumes that a complete sequence of actions can be generated, preferences are known, are transitive, the mood-based utility of every action's outcome is known, and a replanning condition is well defined. This paper presents an alternative approach, one that does not make assumptions about the agent's or observer's omniscience about factors influencing decision making. Our approach recognizes the bounds that limit the agent and observer equally. To accommodate these limits, first, human-centric goal ranking are grounded in a domain-specific mapping to Maslow's hierarchy. Second, a new replanning condition is proposed with dynamically changing mood-based utility during plan execution.

1. Introduction

The objective of our work is to reproduce behavior of individuals who interact with social service programs for the purpose of evaluating such programs. Towards this end, the framework proposed here strives to reproduce behavior exhibited by social service clients as captured by data about them. This paper proposes a human-centric replanning condition based on basic human needs and a mood-based evaluation of a plan by a human-like agent. In traditional economic theory, the agent is assumed to be rational during the planning phase, relying on optimal goal preferences for a given task and the neoclassical utility function that increases expectation of success at each time step (Camerer & Lovallo, 1999; Marnet, 2005). However, due to bounded rationality, a human-like agent would not able to consider all information required to make globally optimal choices that maximize utility (Simon, 1955). Instead, the chosen plan is an approximation of a *potentially* executable plan. During plan execution, the deviation of the executed plan from the chosen plan takes a toll on the agent (Camerer et al., 2004; Kensinger & Schacter, 2008; Marnet, 2005). Depending on the resilience of the agent, a sufficiently pessimistic evaluation of the executed plan triggers a replanning process. In our framework, rather than relying on a static assignment of valence for each action and

© 2018 Cognitive Systems Foundation. All rights reserved.

outcome state *a priori* that impacts mood, valence is assigned dynamically during execution and creates a dynamic mood-based replanning trigger.

This paper extends the BRAMA framework, described elsewhere (Gajderowicz et al., 2017a) and summarized in the next section, that reproduces decisions of cognitively bounded and seemingly "irrational" agents, focusing on the homeless population. The main contributions of this paper are:

- the extension of expected utility function by incorporating Maslow's (1943) hierarchy (MH) for ranking goals;
- the extension of expected utility function for a dynamic mood theory called the Emotional Cycle of Change (ECOC)[1] (Kelley & Connor, 1979);
- a replanning condition based on pessimistic and optimistic moods of the agent; and
- a simulation environment in which the agent executes and revises its plans.

Section 2 introduces the BRAMA framework. Section 3 describes our approach to bounded plan generation and human-like goal ranking, while Section 4 explains how BRAMA incorporates expected utility functions and Section 5 presents the replanning process. We finish with related work and concluding remarks in Sections 6 and 7.

2. Review of the BRAMA Framework

The **B**ounded **R**ational **A**gent **M**otiv**A**tions framework (BRAMA) supports the construction of bounded agent models and simulates their interactions with the environment (Gajderowicz et al., 2017b). A BRAMA agent mimics the human decision making and reasoning processes such that its execution within a simulation replicates how the person being modeled would make decisions. Our objective is to replicate behavior of homeless clients captured by data as they interact with service providers while participating in an intervention program. The simulation environment controls plan generation, execution, and monitoring to reproduce the effects dynamic and constrained social services have on the client's replanning.

The core theoretical tenets of this work are based on ideas from artificial intelligence (AI), economics, sociology, and psychology to reproduce behavior of human-like agents (Gajderowicz et al., 2017a). Economists and AI practitioners focus on understanding the internal processes of decision making, which we call the "reasoning view": an objective understanding of choices (Etzioni, 1988; Russell, 1997). Within psychology and sociology, rationality is a reference point and researchers focus on interpreting observed behavior, which we call the "behavioral view": a subjective understanding of choices (Simon, 1996; Etzioni, 1988). This division has been characterized as being between the subjective rationality of an emotional agent and the objective rationality of an observer (Zafirovski, 2005). The objective of our work is to reproduce seemingly "irrational" behavior using a rational reasoner by combining these two views.

In BRAMA, the two views are split between two phases of behavior, planning and execution. During the planning phase, the agent is assumed to be rational in the neoclassical sense, to know

1. Kelley and Connor (1979) described an agent's "mood" as being optimistic or pessimistic but used "emotion" for the overall theory. The ECOC function used here is a modified version the first author observed while working at a homeless shelter for this research, as described in (Gajderowicz et al., 2017b). It was used to help clients participating in a program recognize as their own optimistic and pessimistic moods.

required information, be nonemotional, and plan its actions based on its own preferred ranking of goals. During the execution phase, however, the agent must adjust to the real and unforeseen consequences of its actions. The responses may be expected and included in the original plan or unexpected, causing the executed plan to deviate from the original plan.

The target domains for a BRAMA agent model are those for which the motivations and means of individual agents are not well understood. This paper targets social service clients, specifically the homeless population. To replicate true homeless clients, the agent model configurations are based on a data set provided by the Calgary Homeless Foundation[2] (CHF) that contains information about clients as they participated in a "Housing First" (HF) intervention program administered by CHF. The CHF-HF data set contains information on approximately 4,000 unique clients who participated in the HF program in Calgary, Canada from 2009 to 2015. Gajderowicz et al. (2018b) provide a complete description of the data and analysis. Participants were surveyed at program intake with follow-up interviews every three months until exiting the program. Amongst other information, the questionnaire captured client demographics and requests for goods and services. The 763 different types of requests provided by clients were combined into 58 different need categories.

To provide a more objective ranking of goals and calculate utility, BRAMA relies on domain-specific mappings of agent goals to levels of Maslow's (1943) hierarchy that grounds the rankings in basic human needs. There is some consensus among researchers that behavior models can rely on theories like Maslow's Hierarchy (MH), given appropriate adjustments for specific domains (Kenrick et al., 2010; Sumerlin, 1995; Henwood et al., 2015). This hierarchy categorizes goals into five levels of short-term and long-term needs. BRAMA utilizes homeless-specific mapping described by Gajderowicz et al. (2018b) to link requests reported in the CHF-HF data set to one or more levels of the hierarchy. The Ontology of Social Service Needs was developed to map each need category to an MH level based on goal definitions and 15 key client demographics (Gajderowicz et al., 2018a).

Emotions also play an important part in calculating expected utility. Many existing models assume that emotions can be assigned statically to all observed situations, and that such assignments are known *a priori* (Lin et al., 2012; Steunebrink et al., 2007). However, an agent will respond differently to similar situations depending on its mood. Such dynamism is not captured by existing models of emotion-based utility functions due to statically assigned emotions for specific events and scenarios. BRAMA relies on the Emotional Cycle of Change (ECOC) theory (Kelley & Connor, 1979), as adopted by the social service domain and described by Gajderowicz et al. (2017b). ECOC recognizes that, in response to the same event or scenario, behavior changes according to stages of pessimism and optimism. In BRAMA, a replanning condition is triggered when the agent's pessimism threshold is reached. Once triggered, goals are reranked and a new plan is generated.

3. Human-Centric AI Planning

As discussed in Section 2, BRAMA adopts two perspectives of rationality, the presumed "rational view" and the observed "behavioral view", which considers environmental and internal constraints that may not be known. This section introduces a bounded agent's representation of goals and actions for generating plans, and how goal ranking is incorporated to calculate utility.

2. The Calgary Homeless Foundation: `http://calgaryhomeless.com/`.

3.1 Bounded Plan Generation

According to Simon (1955), bounded rationality limits our ability to be completely rational when making decisions. The BRAMA planner, STRIPS-BR, uses a STRIPS-style planner (Fikes & Nilsson, 1971) that selects one goal at a time, selects actions that satisfy it, or generates subgoals and required actions if the action's preconditions are not satisfied. Planners that use this type of heuristic, guided by preconditions to reduce the difference between the current state and the goal state, are said to perform means-ends analysis. In STRIPS-BR, three bounds limit the creation of a search tree (Gajderowicz et al., 2017a). The time bound $BR(T)$ limits the number of states that can be visited during the planning phase while creating the search tree. Cognitive limitation set by $BR(C)$ limits the depth of a search tree that can be evaluated. A third factor, $BR(I) = \{S_{BR,t}, G_t, AS_{BR}\}$ as described below, impacts how goals are stored and reranked in memory during planning by limiting the amount of correct information an agent can store in memory, and is discussed here in detail.

In bounded AI planning, the knowledge bound limits the agent's creation of its search tree to only include those states of the world the agent knows about ($S_{BR,t}$) at time step t, an action schema of known actions and their characteristics (AS_{BR}), and what goals an agent has (G_t) at time step t. As the plan is generated, interim goals are included to satisfy preconditions of actions in the plan. The final search tree is made up of branches representing a set of possible plans (P^x) from which the agent can select.

S_t provides grounded facts that are known to be true at time step t. $S_{BR,t}$ are states the agent believes are true at time step t. G are basic achievement goals an agent wants to be true and are mapped to an MH level. G^I are interim goals an agent wants to be true that are not mapped to an MH level but identified as maintenance goals during the planning phase to satisfy preconditions of actions that satisfy existing unsatisfied goals. Three other important terms are

$$G_t^U \subseteq G \cup G^I \text{ , where } \varnothing = G_t^U \cap S_{BR,t} \tag{1}$$

$$G_t^S \subseteq G \cup G^I \text{ , where } G_t^S \subseteq S_t \tag{2}$$

$$G_t^{U+S} = G_t^U \cup G_t^S \text{ , where } G \subseteq G_t^{U+S} \tag{3}$$

where G_t^U is a set of unsatisfied goals at time step t, as per Equation 1, the goal set G_t^S is a set of satisfied goals at time step t, as per Equation 2, and the goal set G_t^{U+S} is a set of all goals for an agent at time step t, as per Equation 3.

An agent's bounded memory to store knowledge in $BR(I)$ limits the number of goals, states, and actions the client can retain at any one time. The actions an agent can use to create a search tree are stored in the bounded action schema AS_{BR}. Each action a in AS_{BR} is defined along with its preconditions (PRE) and postconditions ($POST$). PRE are states that must be true before a is executed while $POST$ are actions that must be true after a is executed. Some action definitions in AS_{BR} are correct, while other definitions may be incorrect. Incorrectly defined actions in AS_{BR} are referred to as incorrect actions with partially defined preconditions and postconditions, as per Equation 5. The set of incorrect actions is AS_{inc}, while the set of correct actions is AS_{cor}. The union of these actions make up the set AS of all possible actions, as defined in

$$AS = AS_{cor} \cup AS_{inc} \text{ ,} \tag{4}$$

where $\varnothing = AS_{cor} \cap AS_{inc}$. The function $inc(a)$ transforms a correct action a to an incorrect action a^*, as in

$$a^* = inc(a) \text{ , where } (PRE \neq PRE^*) \text{ or } (POST \neq POST^*) \text{ ,} \tag{5}$$

where the inverse $inc^-()$ converts an incorrect action to its correct equivalent. The agent's knowledge about actions then is defined as AS_{BR}, where

$$AS_{BR} \subseteq AS_{cor} \cup AS_{inc} \text{ .} \tag{6}$$

States at time step t that are true before an action is executed are defined as $s_i \in S_t$, while states true after execution at time step t+1 are defined as $s_j \in S_{t+1}$. Finally, the agent's reasoner relies on actions available in AS_{BR} to satisfy goals in G^U, given what the client knows about the world in S_{BR}. The final bounded knowledge the agent uses is $BR(I)$.

3.2 Human-Centric Goal Ranking

Human-centric goal ranking is based on ranking of concrete goals mapped to basic needs as identified by Maslow's hierarchy. As discussed in Section 2, while this preference may not be used in the planning phase, it is assumed to be true during the plan execution phase. Goal utility is based on two types of preferences, nominal goal ranking and cardinal goal ranking (Wold et al., 1952).

Nominal goal ranking identifies the order in which goals are preferred, whether by the agent or based on an MH level. Agent preferred ranking is represented by the $\succ_A$ relation, where $s_i \succ_A s_j$ indicates the agent (A) prefers goal s_i over s_j. MH goal ranking relies on MH levels where each level has a different rank. For example, the original hierarchy ranks any goals mapped to the physiological level the highest, while goals mapped to the self-actualization level are ranked the lowest. Hence, MH is captured by the ordering relation $\succ_{mh}$, where

$$physiological \succ_{mh} security \succ_{mh} social \succ_{mh} esteem \succ_{mh} self\text{-}actualization \text{ .} \tag{7}$$

The $rank(pref, s_i)$ function returns the numerical rank for goal s_i according to its nominal mapping, as in

$$rank(pref, s_i) < rank(pref, s_j) \iff s_i \succ_{pref} s_j \text{ where } pref \in [MH, A] \text{ .} \tag{8}$$

For an agent's preferred ordering, $rank(A, s_i)$ returns the index the agent assigned to the goal s_i, where $rank(A, s_i) \in \{1, 2, \ldots, n\}$ for n goals in G. For Maslow's hierarchy where $pref$=MH, $rank(MH, s_i) \in \{1, 2, 3, 4, 5\}$ for each of the five MH levels and n=5. For example, the goal for food is mapped to the physiological MH level, where s_i=$food$ and $rank(MH, s_i)$=1. The goal for having friends is mapped to the social MH level, where s_j=$friends$ and $rank(MH, s_j)$=3. If a goal is mapped to two different levels, it is represented by two separate goals at different MH levels. For example, receiving "addiction support" generates two separate goals, say s_m for the physiological level and s_n for the self-actualization level, resulting in $rank(MH, s_m)$=1 and $rank(MH, s_n)$=5. Once all goals in G are mapped, the appropriate goal ordering can be applied. For a complete discussion on goal mapping see Gajderowicz et al. (2018a).

Cardinal goal ranking indicates a degree of importance in relation to other goals. The nominal ranking $rank(pref, s_i)$ is used to calculate whether there are preferred goals that should be satisfied before s_i. For example, assuming that the physiological goal is most important, and some physiological goal s_j is still outstanding, any unsatisfied goal s_i mapped to higher MH-levels should have a lower utility. The degree to which the utility of s_i is lower is relative to its distance from the unsatisfied and higher ranked physiological level of goal s_j, as defined in

$$min(G^U) = rank(pref, s_i), \text{ where for all } s_j \in G^U, rank(pref, s_i) \leq rank(pref, s_j) \qquad (9)$$

and

$$u(pref, s_i) = 1 - \left(\frac{rank(pref, s_i) - min(G^U)}{n - 1} \right)^{1/e}. \qquad (10)$$

To calculate cardinal utility for $pref$=MH, BRAMA relies on the MH level of goal s_i in relation to the lowest outstanding MH-level goal. For $pref$=A, BRAMA relies on the index of goal s_i in relation to the index of the lowest goal outstanding. First, the function $min(G^U)$ defined in Equation 9 returns the minimum $rank(pref, s_i)$ from outstanding goals. Second, $u(pref, s_i)$ defined in Equation 10 returns the cardinal utility of the goal s_i. Here, the difference between $rank(pref, s_i)$ and $min(G_t^U)$ is divided by four and taken to the power of $1/e$. This reflects logarithmic declining utility of goals at higher levels of the hierarchy,

$$u(pref, s_i) \geq u(pref, s_j) \iff s_i \succ_{pref} s_j \ , \qquad (11)$$

as originally observed by Bernoulli about a declining marginal utility, and adopted by economists like von Neumann (1944), Savage (2012), and others. The resulting $u(pref, s_i)$ function is a cardinal ranking for one goal relative to another that allows us to compare goal preferences using goal utility. Suppose s_i and s_j are goals in a set ordered by $\succ_{pref}$, then Equation 11 defines their relation based on their utilities.

4. Expected Utility Functions for Bounded Agents

In subjective expected utility, an action's utility represents the probability that an action will successfully satisfy the intended goals (Savage, 2012; Jeffrey, 1990). Unless probabilities are known *a priori*, a bounded agent must infer them from its perception of success. In this section we introduce two approaches, first based on neoclassical expected utility and another on the changing mood of the agent using ECOC.

4.1 Relative Expected Utility of Actions

The calculation of an action's expected utility is based on the probability of successfully achieving a goal that action satisfies. Since goals closer to the current time step have higher probability of success their weight is relative to the action's position in the plan. Also, it is possible that one action can satisfy goals at different time steps of a plan. This scenario occurs when alternative actions exist in the same plan, at different time steps. For example, imagine a homeless shelter offers takeaway meals, where the agent receives a hot meal and sandwiches. Both require the precondition the agent is at the shelter, and both satisfy the same physiological need for food. The agent is free to eat the hot meal first and sandwiches later in the day, and vice versa. This type of scenario is possible when alternative actions have overlapping postconditions and preconditions for some plan P^x.

Generally speaking, if an action a_t^x originally at time step t in plan P^x can be executed at other time steps k, the action's utility increases. For each basic goal s_i^x, there is a subplan $P_t^x(s_i^x)$ of actions required to satisfy goal s_i^x from the plan's initial state S_p. First, to calculate the distance between arbitrary goals s_i^x and s_j^x in Plan P^x,

$$dist(s_i^x, s_j^x) = j - i \tag{12}$$

returns the number of actions required to transition from goal s_i^x to state s_j^x. Second, the weight

$$aw_k^x = 1 - \frac{dist(s_t^x, s_k^x)}{dist(S_p, s_i^x)} \tag{13}$$

relies on this distance to calculate the contribution action a_k^x makes to plan P^x, starting with the plan's initial state S_p to when goal s_i^x is true. A higher weight aw_k^x means action a_k^x is closer to the target goal s_i^x in sequence x, and contributes more. Finally, action utility that captures all time steps action a_t^x can be executed in is calculated as the product of mean goal utility for goals satisfied by a_k^x and aw_k^x in

$$u(pref, a_t^x) = \frac{\sum_i u(pref, post_{t,i}^x)}{|POST_t^x|} \times \sum_k aw_k^x \ . \tag{14}$$

The mean goal utility considers utility in Equation 11 for all postcondition states that satisfy unsatisfied goals, where for each $post_{t,i}^x \in G^U \cap POST_t^x$. For each time step k, the mean is multiplied by the action's weight aw_k^x to produce the action's utility $u(pref, a_t^x)$.

4.2 Neoclassical and Mood-Based Expected Utility for Actions

As mentioned in Section 1, according to neoclassical theories of utility, the rational expectation of success is always increasing, assumed to be improving from one action to another. While different methods capture this type of "improving" pattern, we normalize all neoclassical theories by settling on the "expected value" function

$$exp(t) = \frac{|G_t^S|}{|G^{U+S}|} \ . \tag{15}$$

This expected utility is a simple ratio of satisfied goals (G^S) to all goals (G^{U+S}), as depicted in Figure 1 (a). The final neoclassical utility for action a_t^x in plan P^x at time step t is

$$u_{exp}(pref, a_t^x) = u(pref, a_t^x) \times exp(t) \ , \tag{16}$$

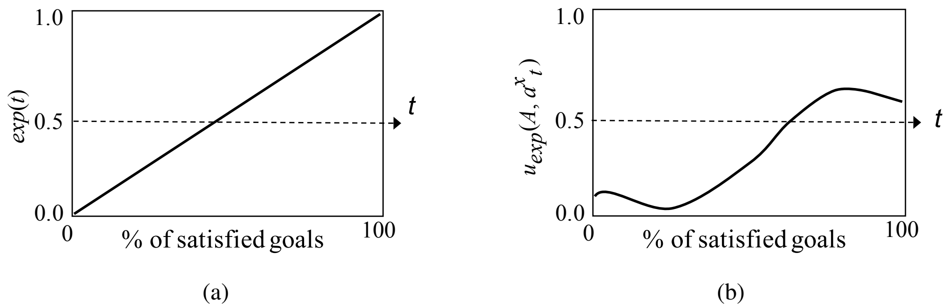

Figure 1: Neoclassical expected utility at time step t using (a) function $exp(t)$ and (b) average action utility $u_{exp}(A, a_t^x)$ for action a in plan P^x.

where for each $post_{t,i}^x \in G^U \cap POST_t^x$. It is a product of $exp(t)$ and postcondition states that satisfy unsatisfied goals, as depicted in Figure 1 (b).

A key limitation of the $exp(t)$ method and its increasingly monotonic characteristic is that it does not reflect how people's expectations actually change over time (Campbell, 2006; Raiffa, 1961). Time itself is a factor, where our perception of risk, preferences, reward, or available information changes over time in a way that not always increases action utility. Instead, people go through optimistic and pessimistic stages that influences how they perceive expected utility of their choices (Kelley & Connor, 1979; Gajderowicz et al., 2017a). Recall from Section 2 that according to the ECOC theory, as individuals begin a task, they are overly optimistic about success, become pessimistic once true efforts becomes apparent, and again become optimistic if sufficient gains towards completing these tasks are made. We can describe these stages in terms of gains and losses of probability of success. During the initial optimistic stages, the probability of success is high without any evidence to justify the optimism. During the pessimism stage, probability of success falls when constraints become apparent. Finally, if constraints are removed, ECOC resembles the $exp(t)$ function where the probability of success again rises based on new evidence.

The $ecoc(x)$ utility function in Equation 17 produces the non-monotonic graph in Figure 2 (a), approximating the ECOC graph. The function $ecoc(x)$ takes $exp(t)$ as its only parameter. The result is an adjusted expectation of success according to the ECOC theory:

$$ecoc(x) = \begin{cases} 0.6 - \dfrac{sin(8x-1) + cos(8x)}{x-2}, & \text{if } x \leq 0.8\,; \\ x, & \text{otherwise}\,. \end{cases} \tag{17}$$

$$u_{ecoc}(pref, a_t^x) = u(pref, a_t^x) \times ecoc(exp(t)) \tag{18}$$

The final mood-based utility of action a_t^x takes $exp(t)$ as a parameter passed to $ecoc(x)$ as x, multiplied by goal utility assigned to the action's postconditions, as defined in Equation 18.

4.3 Plan Utility for Planning and Execution

The utility of plan P^x combines the contribution of each action towards goals being satisfied. During the planning phase, the agent assumes it is objective and rational in the neoclassical sense. It

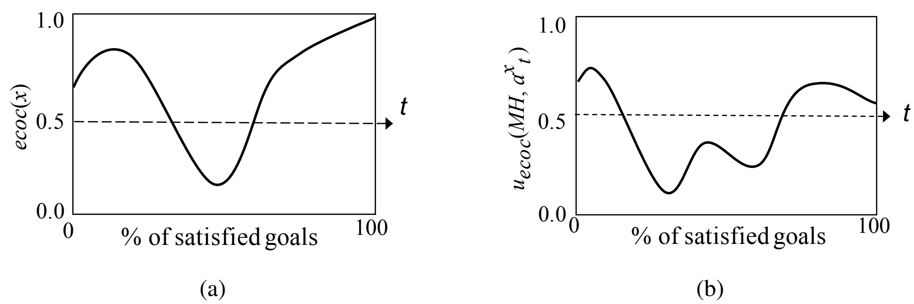

Figure 2: ECOC expected utility at time step t using (a) an approximation of the ECOC function $ecoc(x)$ and (b) average action utility $u_{ecoc}(MH, a_t^x)$ for action a in plan P^x.

assumes no biases are used to evaluate plans in the search tree it created. Hence, during the planning phase the agent relies on $exp(t)$ to calculate plan utility. It also uses a custom goal preference ordering $\succ_A$ for achievement goals mapped to MH needs, relying on $rank(A, s_i)$ for goal ranking.

Neoclassical plan utility combines goal and action utility as the mean utility of all actions in the plan up to time step t, as in

$$U_{exp}(pref, P_t^x) = \frac{\sum_k u_{exp}(pref, a_k^x)}{|P_t^x|} \ , \tag{19}$$

and illustrated in Figure 3 (a). The plan with the highest utility for the entire plan is selected by the agent for execution, as discussed in the next section.

During the execution phase, realistic consequences of planned actions become apparent. First, the agent's preferred order of basic needs is not used. Rather, MH ranking according to domain-specific mapping to MH levels is used, as defined in the ordering Equation 7. This ensures that, regardless of the order the agent preferred during planning, true preferences during plan execution are represented by Maslow's order. Second, if the agent was too optimistic during planning, it will not improve its utility after each action, as implied by the neoclassical theory. Instead, ECOC is used to capture a pessimistic evaluation of the agent's planned actions in response to the true consequences of its executed actions.

If the agent was too optimistic during planning, action utility is calculated using $u_{ecoc}(MH, a_t^x)$, as defined in Equation 18. As illustrated in Figure 2 (b), the trend of action utility is influenced by the ECOC graph in Figure 2 (a) and discussed in Section 4.2. The initially high utility represents an optimistic evaluation of actions. As the plan is executed, the evaluation is more pessimistic resulting in a lower utility. Over time, the plan's utility again rises.

Finally, plan utility combines ECOC-based goal and action utility as the mean utility of all actions in the plan up to time step t, as in

$$U_{ecoc}(pref, P_t^x) = \frac{\sum_k u_{ecoc}(pref, a_k^x)}{|P_t^x|} \ , \tag{20}$$

and illustrated in Figure 3b. Unlike the planning phase, plan utility is not used to select the best plan. Rather, it is used to trigger the replanning process, as discussed in the next section.

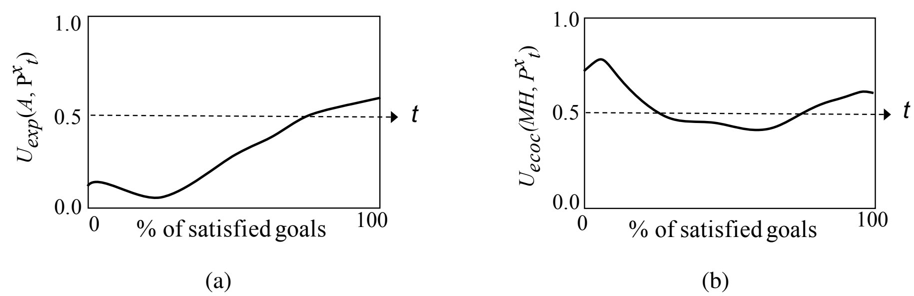

Figure 3: Average plan utility as goals in G^U are being satisfied in plan P^x using (a) neoclassical function $U_{exp}(A, P_t^x)$ and (b) ECOC function $u_{ecoc}(MH, P_t^x)$.

5. Human-Centric Replanning Phase

An important part of replicating a bounded agent's plan generation and monitoring is the environment that forces the agent to reevaluate its plan during plan execution. This section introduces the simulation environment that controls these tasks. As an agent's plan is executed, plan monitoring evaluates the "true" plan utility. A substantial difference between the planned and "true" utilities triggers the replanning process. BRAMA incorporates a discrete-event simulation, a popular simulation architecture for bounded agents in social sciences (Harpring et al., 2014). In such a simulation, the execution of complex systems is represented as an ordered sequence of events. Using forward chaining, an action is executed as a discrete event at time step t, rather than continuously over time. The actual time taken between each event may vary in length.

In BRAMA, each executed plan, whether partially or completely, represents a cycle in the life of the agent. For a "homeless" agent, the cycle may be a 24-hour period in which it must carry out tasks to satisfy as many goals as it can. The goals satisfied by the executed portion of a plan are not included in plan generation during future cycles. The agent is assumed to have mastered these tasks and can execute them without planning. While this plan-reuse resembles case-based approaches, portions of plans are not reasoned about explicitly, as in case-based planning architectures (Lee et al., 2008). Once a partially executed plan is complete, the current cycle ends. At the beginning of the next cycle, the agent retrieves deferred goals and generates a new plan. The time between cycles is also domain or situation specific. For example, an agent may take one day, week, or a month to move from one cycle to another. Matching cycles to actual time duration is addressed during the evaluation of the model, which is beyond the scope of this paper.

The simulation environment controls when plan generation, monitoring, and execution occur with several modules that perform specific functions, as listed in Table 1. The simulation process begins with $simulate(S_t, G_t)$, which takes two parameters, the current state of the world S_t and the agent's unsatisfied goals G_t^U at time step t. At the start of the simulation, $S_{BR,t}$ and G_t are used to initialize the process. The procedure returns a set of final plans that were executed and the resulting world states, PL_{Final} and S_{Final} respectively. Another module, $plan(S_t, G_t, AS_{BR})$, generates a plan P_t^x using BRAMA's STRIPS-based planner (Gajderowicz et al., 2017a). The third parameter AS_{BR} ensures the plan is generated using the agent's bounded action schema in $BR(I)$.

Table 1: Simulation environment modules.

Module	Descriptions
$simulate(S_t, G_t)$	Begins the simulation process.
$plan(S_t, G_t)$	Generates a plan P_t^x using BRAMA's STRIPS-based planner.
$next\_action(P^x)$	Returns the next action a_t^{*x} in plan P_t^x.
$exec(a_t^x, S_t, G_t)$	Executes action a_t^x, given true states S_t and goals G_t. Returns the new states S_{t+1}.
$retain(G_x)$	Returns a set of goals G_R that will be retained for replanning, as per Equation 21.

The function $next\_action(P_t^x)$ returns the next action a_t^x in plan P^x to be executed. The action must be a correct action in AS_{cor} to ensure realistic preconditions and postconditions are enforced on the agent during the execution phase. To ensure it is correct, the inverse of equation 5 is used, mainly a_t^x=$inc^-(a_t^{*x})$ where $a_t^x \in AS_{cor}$ whether a_t^{*x} is in AS_{cor} or not. Next, the procedure $exec(a_t^x, S_t, G_t)$ executes the action, given the agent's current unsatisfied goals in G_t and the true current state S_t. During execution it transitions the state S_t to the new state S_{t+1}.

The agent monitors plan execution by comparing plan utility to $ecoc\text{-}th$, its mood-based threshold. If the utility is above the threshold execution continues, but if the utility falls below the threshold the replanning process is triggered. During the replanning process, G_R=$retain(G_t)$ returns a subset of goals in G_t to be used in planning. For each goal $s_i \in G_t \cap G$ and each action $a_t^x \in P_t^x$ that satisfies that goal,

$$G_R = retain(G_t) : \{s_i \in G_t \cap G | s_i \in POST_t^x \wedge U(a_t^x) \geq act\text{-}th\} \tag{21}$$

where $s_i \in POST_t^x$. The subset of original goals in G (excluding interim goals) not yet satisfied are returned in the goal set G_R, and used to generate a new plan. Goals not in G_R are deferred until a future cycle, as discussed in Section 5.2.

5.1 BRAMA Agent Model

The BRAMA agent model $\mathbb{M}$ provides properties used to define an agent that generates and monitors plans while the simulation executes them and recalculates utility. The structure

$$\mathbb{M} = \{demo(), BR(I), BR(C), BR(T), executil, G_0, pref, planutil, ecoc\text{-}th, act\text{-}th\} \tag{22}$$

represents a particulate type of individual and his or her characteristics. The function $demo()$ categorizes the agent as some cohort of a population based on its demographics, such as age, gender, or income. The agent's bounds ($BR(I), BR(C), BR(T)$) indicate its cognitive limitation during the plan generation process. Its expected utility function during plan generation is $exp(t)$, and its initial goals at time step t=0 are G_0. During the execution phase, the utility function ($executil$) can be either $exp(t)$ or $ecoc(x)$. During the planning phase, the agent uses its goal preferences ($pref$=A). During the execution phase the agent can be configured to use its preferred ranking ($pref$=A) or Maslow's ranking ($pref$=MH). During the planning phase, the agent always maximizes its utility. During execution, it can either maximize its utility ($planutil$=$planutilswap$) or not ($planutil$=$none$). Finally, to trigger the replanning process, the agent has two thresholds $ecoc\text{-}th$ and $act\text{-}th$, as discussed in the next section.

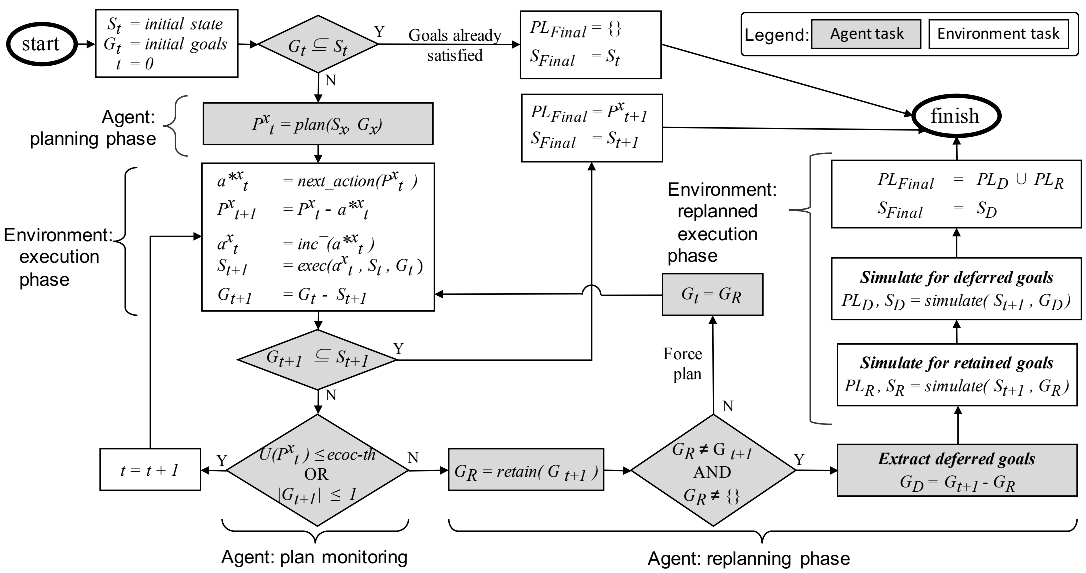

Figure 4: Agent simulation flowchart for $simulate(S_t, G_t)$.

5.2 BRAMA Simulation Environment

Figure 4 presents the simulation flowchart, which relies on the agent model $\mathbb{M}$ properties to control how the agent interacts with the external world. The simulation environment attributes and modules are listed in Tables 1 and 2. If the agent is configured to evaluate plan utility during execution using ECOC then $executil$=$ecoc$ and the simulation may trigger the replanning process.

Once a plan is generated and plan execution begins, plan utility is recalculated at each time step t. While monitoring execution, the agent compares the new utility to its $ecoc$-th threshold. If $U_{ecoc}(pref, P^x_t)$>$ecoc$-th, plan execution continues at time step t=t+1. Otherwise, goals are retained using G_R=$retain(G_t)$ according to the agent's act-th threshold, as per Equation 21. The deferred goals are added to the set G_D. If $U_{ecoc}(pref, P^x) <$ $ecoc$-th but either no goals can be removed or all goals are removed, a new forced plan is created and executed without considering $ecoc$-th. Any goals satisfied by a "forced" plan are accumulate for all forced plans in one cycle. If a "forced" plan cannot satisfy its goals it becomes a "failed" plan. Goals of a "failed" plan persist until they are satisfied after replanning or remain until the end of the simulation. After a "forced" plan completes or becomes a "failed" plan, the agent generates a new plan for the remaining goals. This plan's initial state and goals are one from the previous time step, where $U_{ecoc}(pref, P_x)$>$ecoc$-th. The cycle continues until all goals are satisfied or the current plan fails.

Once retained goals are successfully satisfied the state of the world is represented as S_R and plans required to satisfy retained goals is PL_R. The simulation is executed again for all deferred goals G_D starting at state S_R. Once deferred goals are retrieved and satisfied, the state of the world is represented as S_D and the set of plans required to satisfy all deferred goals is PL_D. The simulation ends when all retained and deferred goals are satisfied. The final state of the world is S_{Final}. The set of plans used to satisfy retained and deferred goals is PL_{Final}, where PL_{Final}=$PL_D \cup PL_R$.

Table 2: Attributes of the BRAMA simulation environment.

Attribute	Descriptions
t	Simulation time unit.
S_t	Actual state of the world at time step t.
G_t	Agent goals at time step t, where $G_t \subseteq G^U$.
P^x	Plan at index x.
P_t^x	Executed portion of plan P^x from start to time step t.
a_t^x	Current action being executed.
G_R	Retained goals, where $G_R \subseteq G_t \cap G$ and G_R=$retain(G_t)$.
G_D	Deferred goals, where G_D=$G_t - G_R$.
S_R	State after retained goals are satisfied.
S_D	State after deferred goals are satisfied.
PL_R	Set of partially executed plans after retained goals are satisfied.
PL_D	Set of partially executed plans after deferred goals are satisfied.
S_{Final}	Final state of the world, returned by $simulate()$.
PL_{Final}	Final set of executed plans, returned by $simulate()$.

5.3 Replanning Example

Figures 5 to 7 illustrate the search trees an agent creates using STRIPS-BR, their goals and how the replanning process reranks them. Consider an agent denoting a homeless client whose goals are to obtain food (physiological), meet with friends (social) and meet a housing worker (security). Some actions have preconditions that must be satisfied first. These include the subgoals of being "at shelter", "at store", or the "at street". The preferred order is $[friends, housing, food]$. To achieve its goals, the agent creates a search tree with several plans to satisfy them. According to the correct action schema AS_{cor}, food can be obtained by going to the shelter when the local soup kitchen is open at 11:00 or 12:00. An agent can also purchase food at the store any time at a cost of \$10.00 and panhandle for more money. The agent can visit friends after the 12:00 lunch at the common area where clients socialize. Finally, it can book an appointment with a housing worker and wait until it is called for an appointment. The agent's bounded action schema AS_{BR} is a subset of $AS_{cor} \cup AS_{inc}$. In AS_{inc}, food costs \$3.00, which the agent believes, rather than the true cost of \$10.00, as defined in AS_{cor}.

5.3.1 Step 1: Planning

In Figure 5, the agent begins the planning process in the "Start" node at time step t=0. With a cognitive bound of $BR(C)$=4 it can look four steps ahead. Each action transitions the agent into a new node at time steps t=1, . . . , 4, with each subsequent node labeled as the goal that has been satisfied at time step t. With a $BR(T)$=24, it can only see the first 24 nodes in the search tree, omitting the last two branches that begin with a visit to the housing worker and panhandling. After calculating the utility of each plan using $exp(t)$ and its preferred goal ordering $\succ_A$, the agent chooses plan P^1 as having the highest utility. The order goals are satisfied in remains the preferred order $[friends$, $housing$, $food]$.

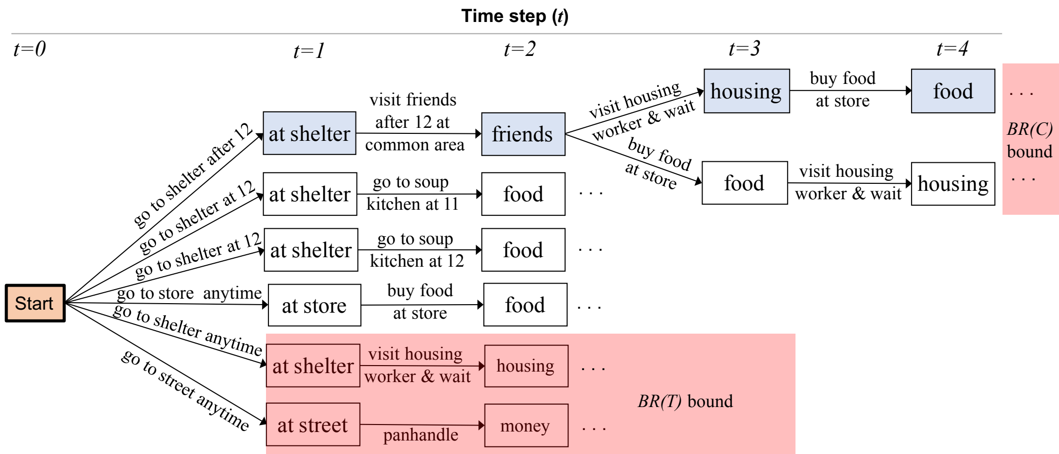

Figure 5: First search tree starting at time step t=0, and selected plan P^1.

5.3.2 *Step 2: Execute Original Plan*

During execution, the agent meets with friends at time step t=2 with the intention of next visiting the housing agent to secure housing, then going to the store to buy food. Once at time step t=2, however, the plan is reevaluated using Maslow's ordering and the $ecoc(x)$ expected utility function. Since housing (security) is ranked lower than food (physiological) according to MH ranking $\succ_{mh}$ but not according the preferred ranking $\succ_A$, the agent becomes unexpectedly hungry. Knowing it will only have one meal today, it becomes worried about spending the entire time waiting for the housing worker without a meal beforehand. In this scenario, the utility of P^1 falls below the agent's $ecoc\text{-}th$ threshold, triggering the replanning process.

5.3.3 *Step 3: Replanning*

The replanning process begins by identifying the actions for which $u_{ecoc}(MH, a_t^x) < act\text{-}th$, and deferring any goals it satisfies. Housing is less important than food in Maslow's order and, say, utility of a "housing" action falls below the threshold, hence $housing$ is deferred. A new search tree is created for the remaining goal, $food$. The agent knows that it is too late for visiting the soup kitchen, as it is after 12:00. There are only two possible plans for which preconditions are true in $S_{BR,2}$ at time step t=2. The first is plan P^{11}, where the agent buys a sandwich at the store for \$3.00 with the \$5.00 it possesses. The second is plan P^{12}, in which the agent panhandles for more money and then purchases a sandwich at the store. Believing it has enough money for a sandwich and having low expectation of making any money panhandling, the highest utility is calculated for plan P^{11}, as Figure 6 depicts.

Starting at time step t=2, the agent generates and chooses plan P^{11} for execution, which then begins at time step t+1 until replanning is triggered or all goals are satisfied. The union of the partially executed plan $P^1_{0,1}$ and the new plan P^{11} produces $P^{1\cup 11}$, with the goal order $[friends, food]$. For the new plan, the $friends$ goal remains satisfied. The $food$ goal is moved up from the third place to the second place in plan P^1, while the $housing$ goal is deferred and omitted from $P^{1\cup 11}$.

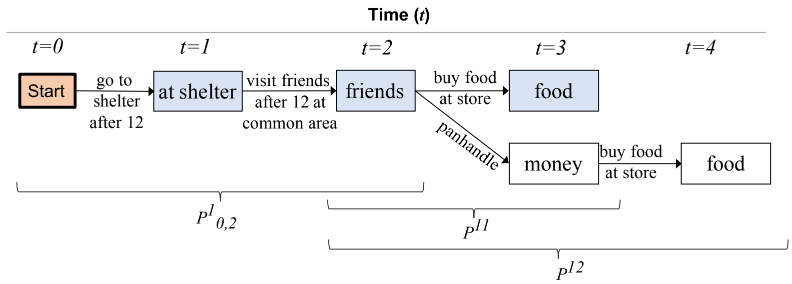

Figure 6: Second search tree after replanning, starting at time step t=2, and selected plan $P^{1\cup 11}$.

5.3.4 *Step 4: Execute New Plan*

During execution of $P^{1\cup 11}$, the agent learns the true cost of food at the store is $10.00 and not $3.00, so it tries to satisfy this precondition. As described in Section 5.1, the agent tries a "forced" plan execution that ignores $ecoc\text{-}th$. However the precondition is a hard requirement, and the agent has no immediate actions that would let it purchase food, so plan $P^{1\cup 11}$ is deemed as "failed", but it can search for another plan. For example, in Figure 6 the agent tries plan P^{12}, to panhandle and then purchase the sandwich, which successfully makes enough money panhandling to buy a sandwich at time step t=4 for $10.00, concluding execution of the plan $P^{1\cup 12}$.

5.3.5 *Step 5: Planning and Execution for Deferred Goals*

Once the $friends$ and $food$ goals have been achieved, deferred goals are retrieved and added to G_t. For this example, the $housing$ goal is retrieved at time step t=4. A new plan is created and executed, adding a new action for visiting the housing worker and waiting for the appointment. Figure 7 illustrates the new plan P^{13} and how it extends the previous plan $P^{1\cup 12}$.

The result is a new plan $P^{1\cup 12\cup 13}$, where $P^{1\cup 12}$ is the executed positions of plans P^1 and P^{12}, and P^{13} is ready for execution. The simulation assumes executed actions are successful with 100% probability. Hence, after execution at time step t=5, the agent successfully satisfies all goals in G^U. Alternatively, this success could depend, with some probability, on the availability of the housing worker before the day is finished. However probabilities for success fall outside the scope of this paper. The final plan is a union of all partially executed plans, resulting in plan $P^{1\cup 12\cup 13}$, stored in PL_{Final}. The final goal order is $[friends, food, housing]$ at time step t=5. Through replanning, goals were reranked from the original order $[friends, housing, food]$ the agent had preferred.

6. Related Work

Our approach to goal ranking and replanning incorporates ideas from a number of fields that aim to reproduce human-like behavior. BRAMA builds on top of this work, but most existing methods for plan generation, execution, monitoring, and replanning adopt the neoclassical approach to agent behavior, making them unsuitable for modeling key aspects of human behavior.

The STRIPS-based planner used by BRAMA to generate plans is limited by its bounded rationality. Most planning systems incorporate methods for overcoming such bounds and improve reaction time during execution, as reviewed by Hendler et al. (1990). For example, to overcome

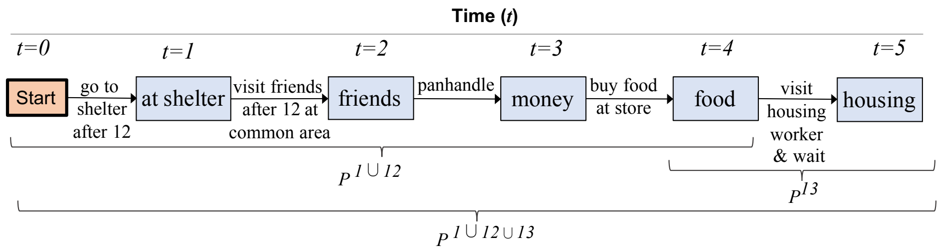

Figure 7: Third search tree after replanning, starting at time step t=4, and selected subplan P^{13}, producing the final plan PL_{Final}=$P^{1\cup11\cup13}$.

time bounds, classic planners like NOAH perform parallel planning to find multiple options for one goal state. NASL interleaves planing and execution, executing one step at a time and replanning. PRIAR uses case-based planning to annotate plans with dependencies between operators, which it used during plan execution to replan more efficiently. A few recent systems explicitly define bounds as part of the planning process and handle discrepancies between generated and executed plans. Similar to BRAMA, PUG/X (Langley et al., 2017) explicitly sets cognitive bounds on the search tree, as well as the number of plans to consider, before stopping search. There has also been some research on reproducing cognitive impairments that cause bounded reasoning. For example, Nuxoll et al. (2010) evaluated Soar's episodic memory to investigate how well it performs with different types of memory.

Goal reasoning in BRAMA focuses on ranking and satisfying the most important goals first while deferring lower ranked ones. An agent's preferred goal ranking, which is provided *a priori*, is used during the planning phase. A domain-specific mapping of a goal to Maslow's hierarchy is used as the basis for goal ranking and utility calculation during the execution phase. Meneguzzi et al. (2013) propose a hierarchical representation of goals expressed as commitments. ICARUS (Langley & Choi, 2006) uses hierarchical goal definitions and a reactive goal management process, with later versions reprioritizing goals as the agent's situation changes (Choi, 2010). Shivashankar et al. (2013) introduced the hierarchical goal network, a set of predefined methods that define the relations among operators, goals, preconditions, and subgoals. ActorSim (Roberts et al., 2016) is a simulator and planner with goal refinement capabilities. It uses hierarchical goal and task networks from which the agent learns to perform sophisticated tasks efficiently. BRAMA can benefit from such hierarchical goal structures, provided they can be grounded in a human-centric representation like Maslow's (1943) framework.

Replanning in BRAMA is based on triggering the replanning process, reranking goals, deferring low ranking goals, and generating a new plan for high ranking goals. Benton et al. (2009) have referred to such concentration on a subset of important goals as *partial satisfaction planning*. This approach can be contrasted with systems that retain goals but modify or repair existing plans, as in the case-based planning literature. Lee et al. (2008) proposed a hierarchical case-based reasoner that selected parts of previous plans to modify the current plan. Hierarchical goal networks similarly repair plans with predefined methods (Shivashankar & Alford, 2013). Rizzo et al. (1999) extend the Prodigy planning architecture to include abstract goals and reactive action packages for execution.

Other related systems rely on goal reasoning to control plan regeneration rather than refinement. Cushing et al. (2008) provide a framework for satisfying abstract goals defined as commitments (required objectives) and opportunities (optional objectives). Replanning selects objectives for goals that must be satisfied to maximize utility and minimize cost. PUG/X (Langley et al., 2017) triggers replanning when one of four anomaly types are detected during execution, at which point it generates given the current state and goal rankings. Some systems like PrefPlan (Brafman & Chernyavsky, 2005) and AltAlt^{PS} (van den Briel et al., 2004) rely on predefined common-sense rules that decide when to modify a plan and reprioritize goals.

Finally, BRAMA relies on a dynamic assignment of utility to goals and actions based on the "mood" of the agent. Similarly, Ojha et al. (2017) propose a replicable, domain-independent computational model for the emotional plan appraisal that generalizes assignment of emotions to events. Such generalization, however, contrasts with most emotion-based planners, like ACRES/WILL, ActAddAct and EM-ONE that rely on predefined associations between emotional appraisals of specific events, as Lin et al. (2012) discusses. Lin also contrasts how systems like FAtiMA, EM, FLAME, Émile, and work by Gmytrasiewicz et al. rely on appraisal theory that associates events with discrete emotional responses and valence. This analysis also describes how AI planners like EMA utilize arousal theory and weighted drives to rate the utility of plans. Emotions have also been used as replanning triggers. Steunebrink et al. (2007) propose a hierarchical representation of emotions that statically link objects, agents, and consequences of events.

7. Conclusion and Future Work

The work presented here is based on the theoretical tenets that, for human-like goal-driven agents, planning and execution stages can diverge and trigger a replanning process. It extends earlier work with complete definitions of expected utility functions for planning and execution phases. This paper defines the utility calculation for goal ranking based on domain-specific mappings of agent goals to Maslow's hierarchy, as well as a dynamic mood-based utility calculation. The divergence between the planning and execution utility functions leads to a novel trigger condition for replanning. The result is a set of partially executed plans that aim to reproduce the changes of seemingly "irrational" agents using a rational reasoner.

However, the BRAMA agent model lacks support for a number of human-like characteristics. First, the planning process is sequential, and would benefit from selecting operators in parallel to pursue multiple goal rankings at once and consider multiple worlds efficiently. Second, the mapping of goals to Maslow's hierarchy could be extended by creating a hierarchical goal network, allowing the ranking of goals at different levels of abstraction. Third, BRAMA would benefit from ranking existing actions and learn about new actions. The current process simply iterates through available actions until it finds one that satisfies outstanding goals. By providing actions with weights and probabilities or the ability to modify them during execution, the agent could rerank actions, abandon incorrect or unused actions, and discover new actions through exploration. Finally, BRAMA does not have the ability to generate new goals. Basic goals are provided *a priori* and reranked dynamically, with interim goals added as required. New goals would let the agent satisfy basic MH needs in new but realistic ways, like superficially satisfying hunger by drinking water or smoking.

The evaluation of BRAMA would benefit from additional data sets that track the same or different intervention programs. By relying solely on the CHF-HF data set, we evaluated our architecture on its ability to replicate CHF-HF participants. By calibrating BRAMA agent models with additional data, a social service program could be evaluated and ranked against alternatives. For example, in addition to participant demographics and a successful or failed program outcome, program participants across two cities could be categorized by BRAMA agent characteristics in $\mathbb{M}$ and grouped into appropriate cohorts. By identifying the same $\mathbb{M}$-based cohorts that failed in one city but succeeded in the other, policy makers in the social service domain may identify city-specific factors that lead to different outcomes for cohorts target by the program.

Acknowledgements

This research was funded by a NSERC Discovery Grant. The authors would like to thank the reviewers for their insightful comments, as well as the Calgary Homeless Foundation for sharing their data set with us, and clarify that our findings do not reflect the views of the Foundation.

References

Benton, J., Do, M. B., & Kambhampati, S. (2009). Anytime heuristic search for partial satisfaction planning. *Artificial Intelligence*, *173*, 562–592.

Brafman, R. I., & Chernyavsky, Y. (2005). Planning with goal preferences and constraints. *Proceedings of the Fifteenth International Conference on Automated Planning and Scheduling* (pp. 182–191). Monterey, CA: AAAI Press.

van den Briel, M., Sanchez, R., & Kambhampati, S. (2004). Over-subscription in planning: A partial satisfaction problem. *Proceedings of the ICAPS 2004 Workshop on Integrating Planning into Scheduling* (pp. 1–8). Whistler, CA.

Camerer, C. F., Ho, T.-H., & Chong, J.-K. (2004). A cognitive hierarchy model of games. *The Quarterly Journal of Economics*, *119*, 861–898.

Camerer, C. F., & Lovallo, D. (1999). Overconfidence and excess entry: An experimental approach. *The American Economic Review*, *89*, 306–318.

Campbell, S. (2006). Risk and the subjectivity of preference. *Journal of Risk Research*, *9*, 225–242.

Choi, D. (2010). Nomination and prioritization of goals in a cognitive architecture. *Proceedings of the Tenth International Conference on Cognitive Modeling* (pp. 25–30). Philidelphia, PA: Drexel University.

Cushing, W., Benton, J., & Kambhampati, S. (2008). *Replanning as a deliberative re-selection of objectives*. Technical report, Department of Computer Science and Engineering, Arizona State University, Tempe, AZ.

Etzioni, A. (1988). Normative-affective factors: Toward a new decision-making model. *Journal of Economic Psychology*, *9*, 125–150.

Fikes, R. E., & Nilsson, N. J. (1971). STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, *2*, 189–208.

Gajderowicz, B., Fox, M. S., & Grüninger, M. (2017a). General model of human motivation and goal ranking. *Proceedings of the 2017 AAAI Fall Symposium Series on Standard Model of the Mind* (pp. 1–6). Arlington, VA: AAAI Press.

Gajderowicz, B., Fox, M. S., & Grüninger, M. (2017b). Requirements for emulating homeless client behaviour. *Proceedings of the AAAI Workshop on Artificial Intelligence for Operations Research and Social Good* (pp. 1–7). San Francisco, CA: AAAI Press.

Gajderowicz, B., Fox, M. S., & Grüninger, M. (2018a). Ontology of social service needs: Perspective of a cognitive agent. *Proceedings of the 2018 Joint Ontology Workshops, Cognition And OntologieS + Explainable AI* (pp. 1–12). Cape Town.

Gajderowicz, B., Fox, M. S., & Grüninger, M. (2018b). Report on the ontology of social service needs: Working paper. From `http://bit.ly/gajderowicz-wp-ossn-2018-pdf`.

Harpring, R., Evans, G. W., Barber, R., & Deck, S. M. (2014). Improving efficiency in social services with discrete event simulation. *Computers & Industrial Engineering*, *70*, 159–167.

Hendler, J. A., Tate, A., & Drummond, M. (1990). AI Planning: Systems and techniques. *AI Magazine*, *11*, 61–77.

Henwood, B. F., Derejko, K.-S., Couture, J., & Padgett, D. K. (2015). Maslow and mental health recovery: A comparative study of homeless programs for adults with serious mental illness. *Administration and Policy in Mental Health and Mental Health Services Research*, *42*, 220–228.

Jeffrey, R. C. (1990). *The logic of decision*. Chicago, Ill: University of Chicago Press.

Kelley, D., & Connor, D. R. (1979). The emotional cycle of change. In J. E. Jones & J. W. Pfeiffer (Eds.), *The 1979 annual handbook for group facilitators*, 117–122. San Diego, CA: Wiley.

Kenrick, D. T., Griskevicius, V., Neuberg, S. L., & Schaller, M. (2010). Renovating the pyramid of needs: Contemporary extensions built upon ancient foundations. *Perspectives on Psychological Science*, *5*, 292–314.

Kensinger, E. A., & Schacter, D. L. (2008). Memory and emotion. In M. Lewis, J. M. Haviland-Jones, & L. F. Barrett (Eds.), *Handbook of emotions*, 601–617. New York, NY: Guilford Press.

Langley, P., & Choi, D. (2006). A unified cognitive architecture for physical agents. *Proceedings of the Twenty-First National Conference on Artificial Intelligence* (pp. 1469–1475). Cambridge, MA: MIT Press.

Langley, P., Choi, D., Barley, M., Meadows, B., & Katz, E. (2017). Generating, executing, and monitoring plans with goal-based utilities in continuous domains. *Advances in Cognitive Systems*, *5*, 1–16.

Lee, C. L., Cheng, K. Y., & Liu, A. (2008). A case-based planning approach for agent-based service-oriented systems. *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics* (pp. 625–630). Singapore: IEEE.

Lin, J., Spraragen, M., & Zyda, M. (2012). Computational models of emotion and cognition. *Advances in Cognitive Systems*, *2*, 59–76.

Marnet, O. (2005). Behaviour and rationality in corporate governance. *Journal of Economic Issues*, *39*, 613–632.

Maslow, A. H. (1943). A theory of human motivation. *Psychological Review*, *50*, 370–396.

Meneguzzi, F., Telang, P. R., & Singh, M. P. (2013). A first-order formalization of commitments and goals for planning. *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence* (pp. 697–703).

Nuxoll, A., Tecuci, D., Ho, W. C., & Ningxuan, W. (2010). Comparing forgetting algorithms for artificial episodic memory systems. *Proceedings of the Remembering Who We Are - Human Memory for Artificial Agents Symposium* (pp. 14–20). Leicester, UK.

Ojha, S., & Williams, M.-A. (2017). Emotional appraisal: A computational perspective. *Poster Collection of the Fifth Annual Conference on Advances in Cognitive System*. Troy, NY.

Raiffa, H. (1961). Risk, ambiguity, and the savage axioms: Comment. *The Quarterly Journal of Economics*, *75*, 690–694.

Rizzo, P., Veloso, M. M., Miceli, M., & Cesta, A. (1999). Goal-based personalities and social behaviors in believable agents. *Applied Artificial Intelligence*, *13*, 239–271.

Roberts, M., Shivashankar, V., Alford, R., Leece, M., Gupta, S., & Aha, D. W. (2016). Goal reasoning, planning, and acting with ActorSim, the actor simulator. *Poster Collection of the Fourth Annual Conference on Advances in Cognitive Systems* (pp. 1–16). Evanston, IL: Cognitive Systems Foundation.

Russell, S. J. (1997). Rationality and intelligence. *Artificial Intelligence*, *94*, 57–77.

Savage, L. J. (2012). *The Foundations of Statistics (3rd ed.)*. Courier Corporation.

Shivashankar, V., & Alford, R. (2013). Hierarchical goal networks and goal-driven autonomy: Going where AI planning meets goal reasoning. *Proceedings of the 2013 Annual Conference on Advances in Cognitive Systems: Workshop on Goal Reasoning* (pp. 95–110). Baltimore, MD.

Simon, H. A. (1955). A behavioral model of rational choice. *The Quarterly Journal of Economics*, *69*, 99–118.

Simon, H. A. (1996). *The sciences of the artificial (3rd ed.)*. Cambridge, MA: MIT Press.

Steunebrink, B. R., Dastani, M., & Meyer, J.-J. C. (2007). *Emotions as heuristics for rational agents*. Technical report, Department of Information and Computing Sciences, Utrecht University, Utrecht, Germany.

Sumerlin, J. R. (1995). Adaptation to homelessness: Self-actualization, loneliness, and depression in street homeless men. *Psychological Reports*, *77*, 295–314.

Von Neumann, J., & Morgenstern, O. (1944). *Theory of games and economic behavior*. Princeton, NJ: Princeton University Press.

Wold, H., Shackle, L. S., & Savage, L. J. (1952). Ordinal preferences or cardinal utility? *Econometrica*, *20*, 661–664.

Zafirovski, M. (2005). Is Sociology the science of the irrational? Conceptions of rationality in sociological theory. *The American Sociologist*, *36*, 85–110.