Action Selection and Task Sequence Learning for Hybrid Dynamical Cognitive Agents

Eric Aaron[a], Henny Admoni[b]

[a]*Department of Mathematics and Computer Science*
Wesleyan University
Middletown, CT 06459
[b]*Department of Computer Science*
Yale University
New Haven, CT 06520

Abstract

As a foundation for action selection and task-sequencing intelligence, the reactive and deliberative sub-systems of a hybrid agent can be unified by a single, shared representation of intention. In this paper, we summarize a framework for *hybrid dynamical cognitive agents* (*HDCAs*) that incorporates a representation of *dynamical intention* into both reactive and deliberative structures of a *hybrid dynamical system* model, and we present methods for learning in these intention-guided agents. The HDCA framework is based on ideas from *spreading activation* models and *belief-desire-intention* (*BDI*) models: Intentions and other cognitive elements are represented as interconnected, continuously varying quantities, employed by both reactive and deliberative processes. HDCA learning methods —such as *Hebbian* strengthening of links between co-active elements, and *belief-intention* learning of task-specific relationships— modify interconnections among cognitive elements, extending the benefits of reactive intelligence by enhancing high-level task sequencing without additional reliance on or modification of deliberation. We also present demonstrations of simulated robots that learned geographic and domain-specific task relationships in an office environment.

1. Introduction

In a hybrid agent —i.e., a robot or other agent with behavior controlled by a hybrid reactive / deliberative system— if the reactive and deliberative levels share a single representation of intention, the agent's goal-directed behavior

can be distributed over both levels. Reactive-level learning can therefore extend intention-guided intelligence without additional reliance on or alteration of the deliberative system, enhancing the benefits of reactivity in dynamic environments. This paper presents a framework for such agents based on shared *dynamical intention* representations, demonstrating its fundamental capabilities and describing mechanisms by which reactive-level learning can affect task-sequencing intelligence and time-efficiency on navigation tasks.

As a motivating example, consider three hybrid agent robot assistants performing navigation tasks in an office environment. These couriers autonomously navigate from a shared initial position to various target locations in sequence, retrieving or depositing something at each; errands include going to the payroll office (retrieve a check), the supply cabinet (get pens), the administrative office (get pens, as well), or the mailroom (drop off a letter). Each target location is known, although in general, what is to be retrieved at a location may be unknown until a robot arrives. On a cognitive level, along with desires about goals to achieve and beliefs about the world, each robot starts out with dynamical intentions in its cognitive system, one for each task it might perform; each dynamical intention has a cognitive *activation* value, representing task *priority*, i.e., the intensity of commitment to the corresponding task; the tasks are then sequenced by priority, with the highest priority task done first, etc. Cognitive activation values vary continuously during errand runs, reflecting continuous changes in task priorities and, more generally, in the overall cognitive systems. Cognitive states thus change due to both deliberative and sub-deliberative processes: straightforward deliberative re-planning of task sequences; and continuous sub-deliberative evolution of tasks' priorities, which also re-sequences the tasks.

In one of these three robotic agents, A_R (for *Rules*), some geographic and task-specific intelligence arises from explicit deliberative rules. In particular, when deliberating to re-plan its task sequence, A_R employs a sorting-based *distance bias* intended to improve time-efficiency, giving higher priority to tasks with target locations closer to its current position. Moreover, A_R follows the *minimal-effort rule* to avoid redundancies: Because A_R has come to know that the supply cabinet and administrative office tasks both result only in getting pens, it follows a pre-encoded, propositional rule to perform exactly one of the two tasks; the rule has no effect, though, on which task is performed, not affecting cognition or behavior until after one is completed.

By comparison, a second robotic agent, A_{NR} (for *non-Rules*), is identical to A_R except that the deliberative system of A_{NR} does not encode the dis-

tance bias or the minimal-effort rule. Instead, A_{NR} relies simply on reactive priorities, selecting a maximal priority task to perform at every opportunity. Unsurprisingly, although agents A_R and A_{NR} begin at the same position, they take different paths during their errand runs: Robot A_{NR}, not following the minimal-effort rule or the distance bias, sequences tasks differently from A_R, redundantly goes to both the administrative office and the supply cabinet for pens, and takes longer than A_R does to complete its tasks.

The third robot, A_L (for *Learning*), is identical to A_{NR} in its deliberation, with no explicit distance bias or minimal-effort rule, but through the joint application of two learning methods, it has learned reactive-level cognitive associations that affect its task-sequencing intelligence. One learning method strengthens links between intentions corresponding to errands with geographically proximate targets, so that when a task T has high priority, priorities are raised on all tasks T' with target locations near that of T. The other method associates beliefs and intentions, training A_L to negatively associate the belief that one redundant task is completed with the intention to perform the other. Due to these learning methods, even though A_L has no deliberative encodings of the distance bias or the minimal-effort rule, it runs errands similarly to A_R, going to the administrative office but not the supply cabinet, and it sequences errands for greater efficiency than A_{NR}.

In this paper, we summarize a framework for *hybrid dynamical cognitive agents* (*HDCAs*, for short) that supports such dynamical intention-guided action selection and task-sequencing intelligence. The design of HDCAs' cognitive systems is influenced unconventionally by the *belief-desire-intention* (or *BDI*) theory of intention [1]; BDI theory and its implementations (e.g., [2, 3] and successors) suggest that BDI elements (beliefs, desires, and intentions) are an effective foundation for goal-directed intelligence, explicitly recognizing the separate roles of desires and intentions in cognitive agents. Unlike conventional BDI agent implementations, HDCAs' cognitive models interconnect BDI elements in a continuously evolving system inspired by *spreading activation* frameworks [4]. Each BDI element in an HDCA is represented by an activation value, indicating its salience and intensity "in mind" (e.g., intensity of commitment to an intention), and cognitive evolution is governed by differential equations in which elements' activation values affect rates of change of other elements' activations. HDCAs employ these cognitive representations on both reactive and deliberative levels, and demonstrations in this paper illustrate the resulting hybrid task-sequencing intelligence: Our example HDCAs perform reactive task re-sequencing due to continuous cognitive

evolution in addition to deliberative task re-sequencing due to proposition-based rules, with straightforward integration of the two kinds.

HDCAs' reactive cognitive models are also influenced by *distinguishing properties* of intention (noted in [1]). For example, in HDCAs, an intensely committed intention I diminishes impacts of other intentions on the intensity of I; strongest intentions (i.e., with the most intense commitment) need not correspond to strongest desires; and intentions, not desires, govern HDCAs' task priorities. HDCAs' dynamical intentions are therefore specifically designed to function as conventional BDI-based intentions. This preserves the relevant, separable roles of beliefs, desires, and intentions that have supported deliberative BDI agent models, which not only enables the reactive intelligence of HDCAs but also suggests the ready feasibility of smooth integration with deliberative levels based on BDI elements that extend the straightforward deliberation employed for demonstrations in this paper.

In addition, we introduce the first learning methods for HDCAs, the kinds of learning employed by A_L above: *Hebbian* learning, which strengthens associations based on co-active elements; and *belief-intention* (BI) learning, which encodes effects of beliefs on intentions. We present learning-based simulations of agents in the office environment described above, demonstrating that the level of abstraction in dynamical intention-based reactive intelligence supports meaningful learning: Demonstrations in navigation task domains, in which goals consist of navigating to target locations in an order that might require dynamic adjustment, show that HDCAs' reactive systems can learn intelligent task re-sequencing behavior that may be more conventionally encoded in deliberative rules. Such learned behavior, our demonstrations further suggest, need not directly affect the deliberative system, so the results of such learning can retain robustness, flexible autonomy, and other benefits typically associated with reactive intelligence.

2. Model Structure and Foundations

The underlying HDCA model draws upon several conceptual frameworks, including *hybrid automata*, *BDI* theory, and *spreading activation* models.

- At its foundation, an HDCA model is a finite state machine. The states, called *modes*, correspond to continuous actions or behaviors; in each mode, differential equations govern behavior, and transitions between states are instantaneous. HDCAs are thus modeled as *hybrid automata*, as described in section 2.3.1.

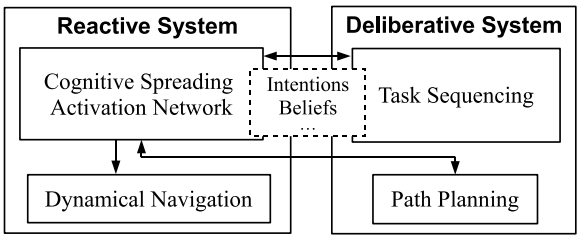

Figure 1: System-level architecture of an HDCA, showing deliberative and reactive levels. Representations of cognitive elements such as intentions and beliefs are shared by the sub-deliberative spreading activation network and the deliberative task sequencing process.

- Cognitive elements of HDCAs are primarily *BDI* elements —beliefs, desires, and intentions— represented by continuously evolving *activation* values. In each HDCA mode, evolution of activation values is governed by differential equations in that mode, as described in section 2.2.

- Cognitive elements are interconnected in an unconventional *spreading activation* framework: Elements serve as variables in differential equations, so activations of cognitive elements affect rates of change of activations of other cognitive elements, as described in section 2.2.

In this section, we further illuminate these foundational ideas, with descriptions at two levels: the HDCA framework, generally; and details of the particular instances of HDCAs illustrating these ideas for this paper.[1]

2.1. Hybrid and Deliberative Structure

The reactive / deliberative structure of HDCAs is illustrated in Figure 1, showing sub-deliberative cognitive and dynamical navigation processes; deliberative task sequencing and path planning processes; and cognitive representations shared across levels. Each level employs cognitive representations in its own manner, but hybrid integration is straightforward, with shared representations as the full cognitive foundation for both levels (see section 2.3.3

[1]To distinguish the levels, we formulaically use a phrase such as "for this paper" to signal that text is about the more specific level. This is not intended to limit applicability strictly to immediate context; indeed, we may intend to suggest applicability to other instances, but not to the level of full generality for HDCAs.

for discussion). For this paper, HDCAs' deliberative planners straightforwardly derive "utility" values for each option, each task or path segment, based on geographic information, task-specific knowledge, and cognitive activations; plans, then, are sequences of options in decreasing order of utility.

Shared cognitive representations enable HDCAs to emphasize reactive intelligence, such as *reactive task re-sequencing*—changing task ordering due only to continuous evolutions of intention activations (task priorities). To embody this emphasis, for this paper, HDCAs are designed to invoke deliberation in only two circumstances: if the current task is unexpectedly interrupted; or if the agent is called upon to change its current task —due to completing the previous task, evolutions of intention activations, or any other cause— and must select from multiple candidates with essentially equivalent intention activations. When these HDCAs deliberate, deliberative task sequencing explicitly incorporates constructs such as the distance bias and the minimal-effort rule (unless forbidden to do so by experimental conditions). Deliberation also re-evaluates an agent's entire task sequence, adjusting cognitive activations so that, e.g., tasks earlier in the sequence have higher-active corresponding intentions, and precluded tasks have highly negative intention activations. After deliberation, an agent simply continues with its new cognitive activations in its new highest priority task. This straightforward system suffices for the immediate demonstrations in this paper; for different applications, HDCAs could readily be designed to rely differently on deliberation —perhaps for rigid assurances of performance, without the flexibility of reactive autonomy— without excessive complications to either the deliberative or reactive system, due to shared cognitive representations.

2.2. Reactive Structure

A navigating HDCA's physical state (position and direction) continuously varies as it moves; for this paper, HDCAs' navigation is based on methods similar to [5, 6], although other approaches could be equally effective. This cleanly integrates with an HDCA's cognitive system, which is based on continuously evolving activations of BDI elements (beliefs, desires, intentions); differential equations govern evolutions of all elements, physical and cognitive. (Element values can also be changed discretely, as effects of mode transitions (see section 2.3.1). For this paper, for example, after a task-completion, a mode transition sets the corresponding intention activation to the minimum value and the belief that the task has been completed to its maximum activation value.) Figure 2 shows BDI elements (and abbreviations

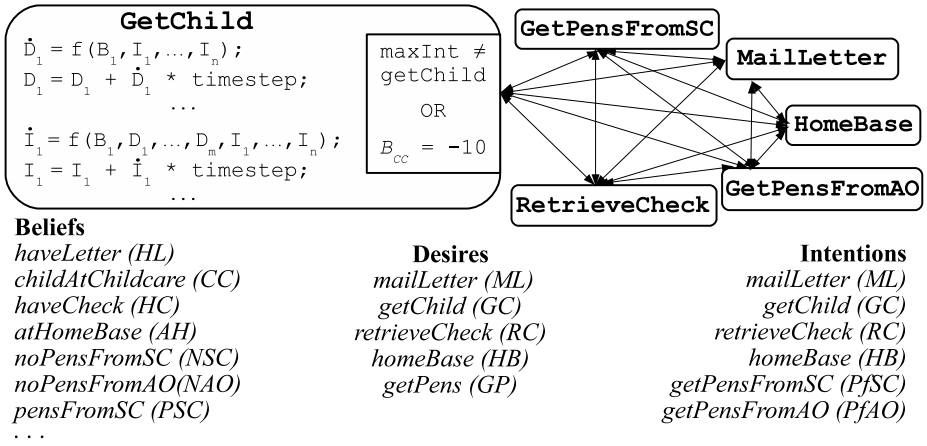

Figure 2: Hybrid dynamical system modes and BDI elements (including abbreviations for names) for HDCAs in this paper.

for their names) and the mode transition model for HDCAs for this paper, which is based on a one-to-one correspondence between intentions and tasks.

The particular BDI elements in an HDCA's cognitive system are predetermined for its domain. For this paper, HDCAs navigate to target locations and retrieve or deposit items there, and those locations and possible items are pre-coded in the HDCAs, along with associated BDI elements: one intention for each target location; appropriate beliefs and desires about items and task completion; etc. Activation values of BDI elements are restricted to the range $[-10, 10]$, where near-zero values indicate low salience and greater magnitudes indicate greater salience and intensity of associated concepts; thus, e.g., more active intentions represent more urgency of the related tasks. Negative values indicate salience of the opposing concept, such as, for intentions, intention not to perform the related task. In this paper, we restrict beliefs to only two values, -10 (*false*) and 10 (*true*), although the system could in principle express intermediate degrees of belief.

Cognitive activations are interconnected in differential equations, guided by ideas from a standard BDI control loop [7]—i.e., desires depend on beliefs and intentions; intentions depend on beliefs, desires, and intentions; etc. Equation 1 is a partial cognitive system (with many elements omitted), where beliefs, desires, and intentions are represented by variables beginning with B, D, and I, and time-derivative variables are on the left in each equation:

$$\begin{aligned} \dot{D}_{RC} &= a_1 B_{HC} + a_3 I_{RC} + a_5 I_{GC} + \cdots \\ \dot{I}_{RC} &= b_1 B_{HC} + b_3 D_{RC} + b_6 D_{HB} + \\ & \quad b_8 I_{RC} + b_{10} I_{GC} + \cdots . \end{aligned} \tag{1}$$

This illustrates interconnectedness: Elements exert *excitatory* or *inhibitory* influence by increasing or decreasing derivatives, with *positive* or *negative connections*, i.e., positive or negative values of the relevant coefficients. Variables stand for activations of cognitive elements (e.g., desire to retrieve a check, D_{RC}), and coefficients encode impacts of connections. For this paper, all coefficients have scalar components, and many also contain components encoding, e.g., *distinguishing properties of intention* (section 3) or learning-specific structure (section 4). Apt scalar values for coefficients in the cognitive system may not be known in advance, but learning methods such as those in this paper can refine initial guesses to meet desired performance criteria.

In general, perception partially determines which BDI elements impact differential equations (i.e., which BDI elements have non-zero coefficients). For example, it may not always be known *a priori* that going to the supply cabinet would result in getting pens, so it cannot be known *a priori* whether or not a desire to get pens should apply to the evolution of the activations of I_{PfSC}. The coefficient on D_{GP} in the equation for $\dot{I}_{PfSC}$, therefore, includes a term t that encodes a perceptual trigger: $\dot{I}_{PfSC} = \ldots + k_{PfSC,GP} \cdot t_{PfSC,GP} \cdot D_{GP} + \ldots$, where $k_{PfSC,GP}$ is a scalar and $t_{PfSC,GP}$ takes value 1 exactly if the HDCA perceived that the *getPens* desire actually applies to the supply cabinet task, and value 0 otherwise. In this paper, to focus on task sequencing rather than perception, we pre-determine when such terms should affect various activations, and we employ names in Figure 2 that make such connections clear (e.g., the names *getPens* and *getPensFromSC*).

For this paper, we assume any two intentions mutually conflict, and from that, the signs of cognitive connections are generally intuitive—each intention is negatively connected to other intentions; each belief that a task is completed is negatively connected to the corresponding intention; desires are positively connected to corresponding intentions (e.g., desire *getPens* to both intentions *getPensFromSC* and *getPensFromAO*); etc.

2.3. Foundations and Background

This section discusses the hybrid dynamical system, BDI, and reinforcement learning foundations underlying HDCA intelligence.

2.3.1. Hybrid Dynamical Systems and Formal Hybrid Structure

In addition to being a hybrid reactive / deliberative system, an HDCA is a *hybrid dynamical system* (*HDS*, for short), a combination of continuous and discrete dynamics, modeled by a *hybrid automaton* [8]. A hybrid automaton is a finite state machine in which each discrete state (or *mode*) is a continuous behavior, containing differential equations governing system evolution in that mode. Transitions between modes are instantaneous, based on *guard* conditions, and may have discontinuous *side effects*, encoding discrete dynamics. Hybrid dynamical systems can be apt models for navigating robots or animated agents (e.g., [9, 10]), and HDCAs' reactive and deliberative structures naturally correspond to HDS elements: Each task of an HDCA is a reactive behavior, implemented as an HDS mode; deliberation in HDCAs occurs during mode transitions. HDS modeling thus facilitates a desirable formal structure for system design, discouraging *ad hoc* approaches.

2.3.2. BDI and Reactive Intention

Ours is not the only agent model with dynamical systems-based elements that can be construed as intentions. For example, the dual dynamics framework [11, 12] explicitly represents *activation dynamics* distinct from *target dynamics*, which are analogous to HDCAs' dynamics of intention evolution and navigation dynamics, respectively. *Dynamic neural field* approaches [13, 14] also associate activations of cognitive entities with actuations of behaviors in a way that can be readily seen as representing intention. These dynamic neural field approaches are based on neuroscientific principles, related to low level phenomena of human intelligence, as opposed to the BDI-influenced approach in HDCAs, which relates more directly to high-level behavior.

HDCA foundations embrace the BDI-theoretic notion [1] that not every element influencing behavior selection is an *intention*. For example, both desires and intentions influence action selection, but BDI theory distinguishes them: Desires (i.e., desired goal states) may conflict; intentions, in contrast, are *conduct-controlling* elements, reflecting commitment to behaviors and resisting conflict. By separating and enabling the contributions of both cognitive elements, BDI theory both encourages structured agent design and allows a rich expression of factors for goal-directed behavior. BDI elements are also established, in philosophy and applications [1, 2, 3], as effective foundations for goal-directed intelligence; the HDCA approach thus derives philosophical and pragmatic benefits from having dynamical features that serve and preserve distinct BDI functions. It is this functionality that mo-

tivates our approach: Any dynamics-based or behavior-based approach that employed functionally identical structures *would* have BDI elements in its framework and thus share potential benefits with the HDCA framework.

The HDCA framework integrates BDI-based intention with differential equations, whereas some other approaches integrate BDI-based deliberation with probabilistic structures such as POMDPs [15, 16, 17] or Bayesian networks [18]. Indeed, from some perspective, HDCAs might be viewed as probabilistic, with intention activations construed as probabilities (modulo normalization) for action selection. HDCA structure emphasizes dynamics close to the low-level control pertinent for navigating agents, however, instead of probabilities conventionally used by deliberative methods, although such methods might be compatible with the underlying HDCA framework (if, e.g., HDCA beliefs were continuum-valued to represent probabilities, etc.).

2.3.3. Learning and Cognitive Architectures

The HDCA learning methods in this paper encode a kind of *policy search*, based on *reinforcement learning* (*RL*, for short), which seems more naturally adaptive and appropriate for our applications than other BDI-based or HDS-based learning methods [19, 20]. Generally, RL trains agents through experience to perform a series of actions that completes a task sequence, without needing *a priori* specification of how tasks should be selected [21]: Given set S of world *states* and set A of possible *actions*, an agent in a state $s \in S$ receives input i from the environment, based upon which it selects an action $a \in A$; this transforms s to a new state $s' \in S$, and the agent receives a numerical reinforcement signal r that indicates the value of the transition to s' [22]. The mapping of states to actions is a *policy* π: $S \rightarrow A$, and policy search trains agents to learn a policy that maximizes the sum of reinforcement signals, with optimization based on searching a parameter space rather than the state space. This is how HDCAs learn: Training alters coefficients of the cognitive system, which determines action selection and thus embodies policy π, to improve performance. As with other RL methods, HDCA learning does not require explicit encodings of a "correct" action or policy, but it does require environmental feedback mechanisms (e.g., to indicate the value of an action in a state). For this paper, to simplify explanations and focus on task-sequencing behavior, we presume such general feedback mechanisms exist, but we fix their applications for our methods and demonstrations.

With its broadly applicable and cognitively inspired foundations —an activation-based cognitive network; BDI-based intentions; and a reinforce-

ment learning paradigm— the HDCA framework is intended to support potential extensions beyond task sequence applications, including aspects of a full *cognitive architecture* [23]. For this task-sequencing-focused paper, only limited (simulated) perception is fully implemented in HDCAs, enabling proximity to items (e.g., an office, a mail cart) to affect cognitive activations; in our demonstrations, more sophisticated knowledge that might be acquired through perception is pre-coded in HDCAs (see sections 4 and 5). Moreover, HDCAs in this paper have no significant autonomous mechanism for discretizing or categorizing perceptual or cognitive values. A full cognitive architecture could integrate richer perception and autonomous categorization into inference and learning. (Parameters for categorizations could themselves potentially be learned.) Such additions, if sufficiently sophisticated, could support an even more potent connection between reactive and deliberative levels than is demonstrated in this paper, potentially converting continuous elements into discrete representations for applications including BDI-based multi-agent systems approaches and cognitive and developmental robotics.

3. Properties of Intention

By employing representations of BDI elements in HDCAs' cognitive systems, we implicitly invite comparison of our application with the philosophical foundations of BDI agents in [1]. In particular, it might initially seem possible that entities called intentions in HDCA cognition are not *genuinely* BDI intentions: HDCA intentions might be inconsistent with properties noted in [1] that distinguish intentions from other cognitive elements (e.g., desires). We carefully implement HDCA intention, however, to be consistent with theoretical *distinguishing properties* that apply to our dynamical account of intention: Intentions are *conduct-controlling* elements that, when salient, *resist reconsideration* and *resist conflict* with other intentions.[2]

For *reconsideration resistance*, we encode two criteria: any *high-active* intention I_a (i.e., with high activation magnitude) tends to minimize impacts on I_a from other intentions; and the strength of this effect grows as the activation (magnitude) of I_a grows. To enable this, for intentions I_a and I_b ($a \neq b$), the differential equation for $\dot{I}_a$ includes the following:

[2]These are not the only properties of intention that are emphasized in [1]; they are, however, properties that can apply to reactive-level intention, not requiring, e.g., future-directedness incompatible with reactive implementations.

$$\dot{I}_a = \ldots \ + \sigma_n \cdot PF(I_a) \cdot I_b + \ \ldots . \tag{2}$$

For example, in equation 1, the coefficient of I_{GC} in the equation for $\dot{I}_{RC}$ has the form $b_{10} = \sigma_{10} \cdot PF(I_{RC})$, with *persistence factor* PF defined as

$$PF(I_a) = 1 - \frac{|I_a|}{\sum_i |I_i| + \epsilon}, \tag{3}$$

where very small $\epsilon > 0$ prevents division by 0, and i ranges over all intentions. For $b \neq a$, $PF(I_a)$ multiplies every intention I_b in the equation for $\dot{I}_a$, so as $PF(I_a)$ nears 0 (i.e., as I_a grows in magnitude relative to other intentions), contributions of every such I_b are diminished, and when $PF(I_a) = 1$ (i.e., $I_a = 0$), such contributions are unaffected. The denominator encodes that I_a is less reconsideration-resistant when other intentions are highly active.

Demonstrations of PF and similarly activation-oriented mechanisms for *conduct control* and *conflict resistance* establish that dynamical intentions are consistent with the distinguishing properties of intention noted above; see [24] for discussion about HDCA intention beyond the scope of this paper. Supported by our results, we treat dynamical intentions as conventional BDI intentions; this is an important, if subtle, part of the HDCA framework.

4. Learning

This paper presents two *policy search*-based HDCA learning methods (see section 2.3.3) that modify cognitive connections: *Hebbian* learning, which strengthens associations based on concurrently salient cognitive elements; and *belief-intention* (*BI*) learning, which encodes task-specific effects of beliefs on intentions. In this section, we describe both the general methods and particular applications of each to the goal-directed intelligence discussed in section 1: Hebbian learning trains agents to approximate the *distance bias*; and BI learning trains agents to approximate the *minimal-effort rule*.

4.1. Hebbian Learning

Hebbian learning, inspired by ideas of synaptic plasticity and neuronal interconnections in [25], enhances connections between co-active (i.e., concurrently high-active) elements in a cognitive system. It requires mechanisms for determining when elements are co-active, and pre-specified *stopping criteria* so that connections do not grow arbitrarily strong, but is otherwise general,

in principle. Stopping criteria may be general, and HDCA designers need not even know in advance if an HDCA would ever meet the stopping criteria, only that if they were met, learning should stop. Moreover, Hebbian learning could theoretically apply to any cognitive elements, but for this paper, it is applied only to enhance connections between intentions corresponding to geographically proximate target locations.

Training for Hebbian learning simply consists of the agent navigating in its environment, using any navigation and path planning methods (the standard methods for this paper are noted in section 2.2); while navigating, learning affects cognitive connections until stopping criteria are met. For this paper and our demonstrations of task sequence-related functionality, training consists specifically of an HDCA taking a pre-specified route through its office that passes in proximity to all task-target locations (e.g., mail room, supply cabinet); the stopping criterion is simply the conclusion of that route. (Different HDCA training routines or stopping criteria —perhaps based on some criteria for optimal performance— could result in different values learned, but this choice suffices for our present demonstrations.) During this training, an agent represents task-targets as *ground concepts*, cognitive elements corresponding to entities perceived in its environment (see section 2.3.3 for discussion of the limited perception for HDCAs in this paper); there is a one-to-one correspondence between task-targets and tasks, and thus between task-target ground concepts and intentions. Ground concepts have baseline activation values of 0, but when an agent is within its *radius of perception* of a task-target, activation on the associated ground concept instantaneously rises; except in extraordinary cases, those activations do not rise above the initial-boost level. When the agent moves outside its radius of perception from a target, the target loses salience and the corresponding activation gradually drops to zero. (For implementation details, see supplementary website [24].) In this way, targets are co-active —with activations both greater than 0— only when geographically proximate, with greater co-activation when targets are perceived closer to each other during training.

Based on these concept activations, agents associate intentions corresponding to co-active (i.e., proximate) task-targets. At every timestep during a training session, for any task-target ground concepts a and b ($a \neq b$) with activations greater than 0, and corresponding intentions I_a and I_b, the following adjustment is made:

$$k_{a,b} = k_{a,b} + \frac{\beta}{c_1}. \tag{4}$$

In this equation, $k_{a,b}$ is the scalar part of the coefficient on the I_b term in the differential equation for $\dot{I}_a$, β is the activation of ground concept b, and c_1 is a pre-specified scaling constant. Thus, this process strengthens connections between intentions corresponding to co-active targets; the extent to which the effect of I_b on $\dot{I}_a$ is changed, moreover, is proportional to the activation of b, so more highly active concepts have stronger effects. Therefore, e.g., positive activation on intention I becomes less inhibitory to intentions corresponding to task-targets near that of I, and the extent of the effect corresponds to perceived proximity of the targets.

4.2. Belief-Intention Learning

Belief-intention (BI) learning alters cognitive connections between beliefs and intentions, training HDCAs to relate intentions to perform tasks with task-completion beliefs about other tasks—relationships that might naturally be encoded in propositional rules (e.g., the minimal-effort rule). To enable BI learning, all coefficients relating beliefs to intentions in HDCAs' cognitive systems are implemented to have the general form

$$\begin{aligned} IC(I_a, B_b) &= k_{a,b} \cdot [r_{a,b} \cdot C_{a,b} + (1 - r_{a,b})] \\ IC(I_a, B_{\bar{b}}) &= k_{a,\bar{b}} \cdot [r_{a,\bar{b}} \cdot C_{a,\bar{b}} + (1 - r_{a,\bar{b}})]. \end{aligned} \tag{5}$$

In these equations, a and b ($a \neq b$) range over all tasks, so I_a is the intention for task a, B_b ($B_{\bar{b}}$, respectively) is the belief that task b has been completed (not completed), and $IC(I_a, B_b)$ ($IC(I_a, B_{\bar{b}})$, respectively) is the coefficient for term B_b ($B_{\bar{b}}$) in the differential equation for intention I_a. Values $k_{a,b}, k_{a,\bar{b}}$ are scalars. The $r_{a,b}$ and $C_{a,b}$ terms specify the desired effects of beliefs on intentions; these terms might represent different functions in different applications, but the particular motivating application in this paper —learning consistency with the minimal-effort rule, avoiding redundant tasks but not affecting cognition until one of the redundant tasks is completed— both illuminates their general purposes and presents a specific application.

For this paper, the $r_{a,b}$ term takes value 1 exactly when pre-specified conditions determine that belief B_b should affect intention I_a, otherwise $r_{a,b} = 0$ (and similarly for term $r_{a,\bar{b}}$, belief $B_{\bar{b}}$); for our minimal-effort rule example, $r_{a,b} = 1$ exactly when a, b correspond to redundant tasks—here, the pen-related tasks *GetPensFromSC* and *GetPensFromAO*. The $C_{a,b}$ term specifies how belief B_b should affect $\dot{I}_a$ when $r_{a,b} = 1$; for this example, $C_{a,b} = C_{a,\bar{b}} = \frac{B_{\bar{b}} - 10}{-20}$, which encodes that $B_b, B_{\bar{b}}$ have no impact on intention I_a when task

b is not completed ($B_{\bar{b}} = 10$), but when task b is completed ($B_{\bar{b}} = -10$), I_a is affected by learning (see equation 6), which alters the coefficient so that activation on I_a drops rapidly, preventing the redundant errand.

In general, the value of $r_{a,b}$ itself could be determined by environmental feedback (a necessary element for all RL methods); perception and knowledge representation could, for instance, determine redundancy by detecting that going to the administrative office and the supply closet resulted in getting the same object. For this paper, to focus on task-sequencing intelligence rather than perception, we presumed correct redundancy detection, pre-coding the appropriate $r_{a,b}$ values. Rule-like behavior then arises from dynamical intention-based intelligence, as long as the scalar parts k of the coefficients have appropriate values. To learn those values for our minimal-effort rule example, for this paper, training consists of an agent autonomously running errands in its office. Stopping criteria are met when the agent completes a run having performed exactly one pen-related task; otherwise, inhibitory links are strengthened between beliefs that pens were obtained and intentions to obtain pens, and another training run is made from the same initial position. Specifically, after each training run that did not meet stopping criteria, scalar parts k of all coefficients (for $a \neq b$) are altered as follows:

$$k_{a,b} = k_{a,b} \cdot [1 + r_{a,b}(\gamma_{a,b} - 1)] \tag{6}$$

(and similarly for $k_{a,\bar{b}}$), where pre-specified $\gamma_{a,b} > 1$ encodes the extent of the modification. Thus, for this paper, when $r_{a,b} = 0$, there is no change to $k_{a,b}$, but when learning should affect intentions —i.e., $r_{a,b} = 1$— inhibitory links are strengthened, leading to minimal-effort rule-like behavior: Before either task is completed, beliefs have no enhanced effect on intentions, but after one is completed, activation on the complementary intention rapidly decreases.

This presentation of BI learning presumes HDCA attributes such as precoded stopping criteria, appropriate cognitive beliefs about task completion, and, more subtly, that because BI learning effects are multiplicative, signs of coefficients relating beliefs and intentions should not be changed by the BI learning process. Different implementations, however, could be productively based on different application-appropriate assumptions within the same general HDCA-BI learning framework.

4.3. Integrating Hebbian and BI Learning

Hebbian and BI learning alter disjoint sets of cognitive connections, and HDCAs can employ both methods together; indeed, for demonstrations in

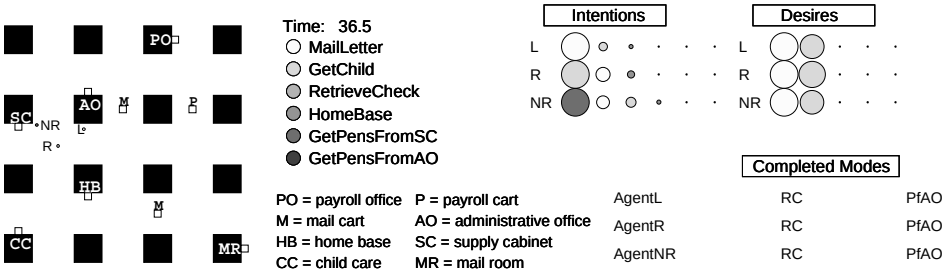

Figure 3: Screen display of a simulation in progress. A map of the office environment, left, shows offices and obstacles (black squares), targets (white squares abutting offices), and three robotic agents (L, R, and NR). Visual representations of agents' desire and intention activations are on the right, beneath which are lists of tasks completed by each agent.

this paper, *integrated Hebbian-BI* (*HBI*) learning employs exactly the mechanisms in sections 4.1 and 4.2. A training run for HBI learning consists of running errands in the office environment along an autonomously determined path. Stopping criteria are met when the most recent training run meets conditions similar to the distance bias and minimal-effort rule—i.e., the agent performs exactly one of the two pen-related tasks, suggesting adequate minimal-effort rule learning; and the entire errand run takes no less time than the previous run did, suggesting adequate distance bias learning. (Because HDCAs in our simulations move at identical, constant speed, time and distance are equivalent measures).

5. Demonstrations and Experiments

To demonstrate HDCA learning, we created MATLAB simulations of office-assistant robot HDCAs in the environment of Figure 3. In each simulation, one or more robots (which did not interact with each other; there were no multi-agent dynamics) navigated to the targets in Figure 3, completing multiple tasks (listed on the display as *MailLetter*, ..., *GetPensFromAO*). For comparison, some agents had deliberative systems that explicitly encoded the distance bias and minimal-effort rule, while other agents did not, instead approaching such performance from reactive-level learning; tests compared performance on autonomous errand runs, evaluating task sequences and times of task completion. We compactly summarize results here; supplementary website [24] has additional information and implementation details.

5.1. Hebbian Learning

To establish that Hebbian learning improves performance, a *reactive task sequencing agent* —an HDCA with deliberative task sequencing disabled— learned to approximate the distance bias, with training as described in section 4.1: Agent A_H (for *Hebbian*) made a single training run with radius of perception $r_p = b + i$, where b and i are the lengths of a hallway block and an intersection in the office; its path passed within r_p of each target location. At each timestep, cognitive coefficients were updated as in equation 4.

Agent A_H was then tested by comparing its errand-running performance to that of agent A_{NH} (*non-Hebbian*), which was identical to the original, pre-training A_H. Both robots made single errand runs from the same position with the same initial cognitive activations; the initial location and cognitive activations were those for the training of A_H. Each robot's cognitive system and task priorities evolved individually during the test run, but because the initial activations were highest on intentions corresponding to remote target locations —e.g., $I_{MailLetter}$, because the mail room is not near any other target— the effects of learning were not immediately apparent: With no high-priority targets proximate to other targets, both agents began their runs on similar paths, completing their first two tasks simultaneously. After that, however, the robots' behavior diverged, showing that reactive-level learning of A_H affected navigation in apparent accord with the distance bias: After both agents completed *MailLetter* and *RetrieveCheck*, agent A_{NH} next went to complete *GetChild*, whereas A_H went to the administrative office and the supply cabinet, the next task-targets in order of geographic proximity.

5.2. Belief-Intention Learning

To establish that BI learning improves HDCA performance, a reactive task sequencing agent A_{BI} learned to approximate the minimal-effort rule, with training as described in section 4.2: Robot A_{BI} autonomously ran errands, adjusting cognitive coefficients as in equation 6 and training until a training run included exactly one of the pen-related tasks; each of its 7 training runs began with the same cognitive element activations and from the same position near the supply cabinet on the left of the office environment.

After its training had concluded, A_{BI} was tested by comparison to robot A_{NBI}, which was identical to the original, pre-training A_{BI}. Tests were run from 16 starting locations, hallway intersections in the office world, which did not include the training location; for each test run, agents autonomously ran errands with identical initial cognitive element activations (the same as

those for the training of A_{BI}) from their shared starting point. Robot A_{BI} completed exactly one pen-related task on all 16 test runs, whereas A_{NBI} did so on 8 test runs, showing that BI learning enables reactive-level changes to encode behavior in apparent accord with the minimal-effort rule.

5.3. Integrated Hebbian and BI Learning

A demonstration similar to the three-agent errand-running example in section 1 of this paper illustrated the integration of Hebbian and BI learning, showing that HDCAs can learn behaviors consistent with the distance bias and minimal-effort rule without explicit deliberative encoding of either. Training of learning agent A_L was in accord with the description in section 4.3; each of its 18 training runs began with the same cognitive element activations and from the same position near the supply cabinet on the left of the office environment. (See supplementary website [24] for relevant parameter values and other implementation details.)

After training, A_L was compared to two other agents: A_{NR}, which was identical to the pre-training A_L; and A_R, which was identical to A_{NR} except with explicit deliberative encodings of the distance bias and minimal-effort rule. Tests were run from 16 starting locations, hallway intersections in the office world, which did not include the training location; for each test, robots autonomously ran errands with identical initial cognitive element activations (those for the training of A_L) from their shared starting point.

On every run, A_R retrieved pens from the administrative office but not from the supply cabinet, due to its explicitly encoded minimal-effort rule; by comparison, A_L went to exactly one of those two locations on every run, and untrained agent A_{NR} went to both locations on every run. Additionally, A_L always finished the run in less time than A_{NR}, though later than A_R; average completion times for A_R, A_L, and A_{NR} were 68.8, 69.6, and 73.1 simulated seconds, respectively (see supplementary website [24] for additional data and analysis). Indeed, on 15 of the 16 test runs, A_L and A_R performed exactly the same task sequence, and on 12 of those, A_L finished less than 0.71 seconds behind A_R, with the difference seemingly due to the time immediately after completing *GetPensFromAO* in which the activation on I_{PfSC} in A_L decreased as an effect of BI learning. Together, these results support our results about individual Hebbian and BI learning methods, suggest the effectiveness of integrated Hebbian-BI learning in this task domain, and suggest the potential for learned behavior to successfully generalize beyond a training set.

6. Discussion

The general framework for HDCAs —combining HDS design, spreading activation evolutions, and BDI-based reactive cognition— seems apt for cognitive modeling, learning, and deliberation extensions beyond the applications in this paper. As an example, in this paper, we conflate salience and cognitive intensity or commitment "in mind"—one activation value represents both qualities— but if future applications required agents to be very aware (high salience) of a mild desire (low intensity), the HDCA framework could readily adopt multiple activations for each cognitive element. The HDCA framework's emphasis on low-level intelligence also suggests potential compatibility with more general cognitive or developmental robotics approaches: Concept or behavior formation, for instance, might arise organically from environmental feedback and primitive factors (e.g., desires to have a pen, finish errands quickly, and not carry needless weight). Furthermore, for this paper, deliberation is modeled as occurring during HDS mode transitions, but alternative designs could encode specific deliberation modes that directly represent time spent during deliberation, which could be incorporated without altering reactive cognitive representations. Such cognitive expansions could, in expected ways, correspondingly increase computational burden; our MATLAB-based demonstrations of HDCA functionality did not comprehensively investigate time efficiency.

Learning methods in this paper affect only beliefs and intentions, which seem most relevant for task-sequencing intelligence (see [16] for a related observation), and they do not result in HDCAs learning *explicit propositional rules*; instead, HDCAs learn *tendencies* that approximate rules—behaviors that generally obey a rule but retain autonomy to diverge in exceptional circumstances. The scope of mechanisms in our HDCA learning methods, however —spanning excitation and inhibition, due to links from multiple kinds of elements, across multiple training and link-alteration protocols— suggests the feasibility of extensions that could incorporate, e.g., desires, or other mechanisms. Moreover, it might be possible to construct an HDS model in which guards and mode transitions are parameterized, for which parameter values could be learned to encode true propositional rules in the HDS structure; this is related to the problem of learning new *behaviors*, i.e., reshaping a mode-transition system to include new modes, transitions, and guards. Formal HDS structure thus illuminates how unconventional HDS models might potentially support such guard-level or mode-level learning.

7. Conclusion

This paper introduces the HDCA framework and *dynamical intention*-based reactive systems, demonstrating task-sequencing intelligence and other fundamental capabilities of HDCAs in simulations of courier robots in an office environment. HDCA cognition is based on continuously varying cognitive representations that are shared by deliberative and reactive processes, distributing intention-guided intelligence over both levels and enabling reactive-level enhancements to affect overall behavior without additional complication of or reliance on deliberation. In our demonstrations, these dynamical intention-based representations are coupled with a straightforward deliberative level, supporting reactive task re-sequencing due to continuous cognitive evolution, deliberative task re-sequencing due to proposition-based rules, and the smooth integration of the two levels. In addition, we present two HDCA learning methods, demonstrating that dynamical intention-based cognition supports meaningful reactive-level learning: Demonstrations in navigation task domains show that HDCAs' reactive systems can learn task re-sequencing intelligence that may be more conventionally encoded in deliberative rules. We further describe how dynamical intention meets distinguishing properties of true BDI intentions, which both supports HDCAs' reactive intelligence and facilitates the possibility of smooth integration with more general, deliberative BDI-based methods. Ultimately, dynamical intention-based learning could extend the benefits of reactive intelligence, retaining the strengths of deliberation while minimizing reliance on it, potentially making agents more robust, autonomous performers in dynamic or incompletely known environments.

Acknowledgments

The authors greatly appreciate the effort and insights of Juan Pablo Mendoza during the later stages of paper preparation. Thanks also to Jim Marshall, Tom Ellman, Michael Littman, and reviewers of previous versions of this paper for their insightful and helpful comments. The second author is supported by a National Science Foundation Graduate Research Fellowship.

References

[1] M. Bratman, Intentions, Plans, and Practical Reason, Harvard University Press, Cambridge, MA, 1987.

[2] M. Georgeff, A. Lansky, Reactive reasoning and planning, in: AAAI-87, 1987, pp. 677–682.

[3] A. Rao, M. Georgeff, Modeling rational agents within a BDI-architecture, in: Proc. of Principles of Knowledge Representation and Reasoning, 1991, pp. 473–484.

[4] P. Maes, The dynamics of action selection, in: IJCAI-89, 1989, pp. 991–997.

[5] S. Goldenstein, M. Karavelas, D. Metaxas, L. Guibas, E. Aaron, A. Goswami, Scalable nonlinear dynamical systems for agent steering and crowd simulation, Computers And Graphics 25 (6) (2001) 983–998.

[6] E. Aaron, F. Ivančić, D. Metaxas, Hybrid system models of navigation strategies for games and animations, in: HSCC 2002, pp. 7–20.

[7] M. Wooldridge, Reasoning About Rational Agents, MIT Press, Cambridge, MA, 2000.

[8] R. Alur, T. Henzinger, G. Lafferriere, G. Pappas, Discrete abstractions of hybrid systems, Proc. of the IEEE 88 (7) (2000) 971–984.

[9] E. Aaron, H. Sun, F. Ivančić, D. Metaxas, A hybrid dynamical systems approach to intelligent low-level navigation, in: Proceedings of Computer Animation, 2002, pp. 154–163.

[10] H. Axelsson, M. Egerstedt, Y. Wardi, Reactive robot navigation using optimal timing control, in: American Control Conference, 2005.

[11] H. Jaeger, T. Christaller, Dual dynamics: Designing behavior systems for autonomous robots, Artificial Life and Robotics (2) (1998) 108–112.

[12] J. Hertzberg, H. Jaeger, P. Morignot, U. Zimmer, A framework for plan execution in behavior-based robots, in: Proceedings of ISIC/ISAS, 1998.

[13] G. Schöner, M. Dose, C. Engels, Dynamics of behavior: Theory and applications for autonomous robot architectures, Robotics and Autonomous Systems 16 (1995) 213–245.

[14] W. Erlhagen, E. Bicho, The dynamic neural field approach to cognitive robotics, Journal of Neural Engineering (3) (2006) R36–R54.

[15] G. Rens, A. Ferrein, E. van der Poel, A BDI agent architecture for a POMDP planner, in: Proceedings of Commonsense 2009, pp. 109–114.

[16] M. Schut, M. Wooldridge, S. Parsons, Reasoning about intentions in uncertain domains, in: Proceedings of ECSQARU-2001, pp. 84–95.

[17] R. Nair, M. Tambe, Hybrid BDI-POMDP framework for multiagent teaming, Journal of Artificial Intelligence Research (23) (2005) 367–420.

[18] M. Fagundes, R. Vicari, H. Coelho, Deliberation process in a BDI model with Bayesian networks, in: Proceedings of PRIMA 2007, Vol. 5044 of Lecture Notes in Artificial Intelligence, pp. 207–218.

[19] B. Subagdja, A. H. Tan, Planning with iFALCON: Towards a neural-network-based BDI agent architecture, in: IEEE/WIC/ACM International Conference on Intelligent Agent Technology, 2008, pp. 231–237.

[20] H. Kawashima, T. Matsuyama, Multiphase learning for an interval-based hybrid dynamical system, IEICE Trans. Fund. of Electronics, Communications and Computer Sciences E88-A (11) (2005) 3022–3035.

[21] R. Sutton, A. Barto, Reinforcement Learning: An Introduction, MIT Press, 1998.

[22] L. Kaelbling, M. Littman, A. Moore, Reinforcement learning: A survey, Journal of Artificial Intelligence Research 4 (1996) 237–285.

[23] D. Vernon, A survey of artificial cognitive systems, IEEE Trans. Evolutionary Computation 11 (2) (2007) 151–180.

[24] E. Aaron, H. Admoni, Supplementary material for this paper, available at http://eaaron.web.wesleyan.edu/ras10_supp.html.

[25] D. O. Hebb, The Organization of Behavior, John Wiley & Sons, Inc., New York, NY, USA, 1949.