IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 9, 2020, accepted September 24, 2020, date of publication September 28, 2020, date of current version October 9, 2020.

*Digital Object Identifier 10.1109/ACCESS.2020.3027355*

# A Hierarchical Computational Model Inspired by the Behavioral Control in the Primate Brain

**DONGQING SHI[1,2], (Member, IEEE), JERALD KRALIK[2,3], AND HAIYAN MI[1]**

[1]School of Mechatronics and Information Technology, Yiwu Industrial and Commercial College, Yiwu 322000, China
[2]Department of Psychological and Brain Sciences, Dartmouth College, Hanover, NH 03755, USA
[3]Department of Bio and Brain Engineering, Program of Brain and Cognitive Engineering, College of Engineering, Korea Advanced Institute of Science and Technology (KAIST), Daejeon 34141, South Korea

Corresponding author: Dongqing Shi (shi.dongqing@ywicc.edu.cn)

This work was supported in part by the Yiwu Industrial and Commercial College under Grant ZD2020JD602-04, in part by the Key Project of Jinhua Science and Technology Bureau under Grant 2020-1-001, and in part by the U.S. ONR under Grant N00014-08-1-0693.

**ABSTRACT** The basic cognitive architecture of our brain is still unknown. However, scientists have found evidence for existence of distinct behavioral control systems shared by humans and nonhumans. Inspired by the problem solving systems of the behavioral control in the primate brain, a hierarchical computational model is presented. We focus on the integrative performance of brain substructures, each of which is represented by a problem solver that is further modeled by a certain algorithm. Different levels of brain substructures, as well as the corresponding algorithms, are hierarchically organized both in structure and in function, including how and when higher-order solvers control lower-order ones. Different problem solvers share a same slice of working memory. This novelty is claimed since most of existing brain models emphasize on the neural network structure even though the neuron dynamics of brain is still very controversial. And we compare its performance to three other computational models in the face of a challenging foraging problem. Agents are examined in foraging environment with different sizes, and/or transparent barriers. The experimental results show that our model performed the best outright in most scenarios. Further, the results discover that the virtues of our primate brain lie not only in the heights of thinking it can reach, but also in its range and versatility.

**INDEX TERMS** Cognitive hierarchical architecture, computational model, reinforcement learning, local and global planning.

## I. INTRODUCTION

Recently, more and more robotic research has been inspired by anatomical and psychological studies. Scientists have found there exist two neural systems in mammals controlling their behaviors. One is a model-based goal-directed system based in the ventromedial prefrontal cortex (vmPFC), and the other is a model-free habit system based in the striatum [1], [2]. It has been argued that a model-based decision-making system has potentials for high-order cognitive functions such as mental simulation, planning, and reasoning, which usually leads to better solutions to many problems. However, research shows that pure model-based systems are notoriously brittle and therefore often break under real-world conditions due to either inaccuracy of the model itself or uncertainty of the real-world [3]. A model-free habit system usually relies on a stimuli-reward mechanism, where a positive or negative reward responds to certain stimuli. Experiments have shown that such a simplicity is very efficient for dealing with uncertainties. That is, a fallback to a model-free habit system would be a solution to the brittleness and break of a model-based system. Another approach would be to have a better system that can fix the models when they fail, enabling to solve these harder problems. In the paper, a problem whose features can be properly modeled by a model-based problem-solving system is called as an apparent problem. Otherwise, the problem is called as a non-apparent problem, i.e., one that breaks the model and requires an individual to infer a hidden cause and create a non-perceptual concept to model it when confronting an unknown event or consequence.

The associate editor coordinating the review of this manuscript and approving it for publication was Stavros Souravlas.

In primates, evidence shows there exists a distinct region in prefrontal cortex (PFC), called granular PFC. Some researchers believe that the granular PFC enables primates to perform unconventional behaviors, such as looking away from a salient visual stimulus when necessary [4]. In accordance with this view, we hypothesize that the granular PFC is the base of solving non-apparent problems. Moreover, we believe that a detailed analysis of the region will shed light on the mechanisms that underpin creative problem solving in people.

To analyze the apparent versus non-apparent problem solving, we first focus on the classic detour problem in the literature [5], where subjects must circumvent a barrier to obtain a reward item. Most research [6], [7] has focused on scenarios with opaque barriers as obstacles, showing that agents can solve the problem by taking paths away from a goal item to reach it. However, the detour problem is found extremely challenging when the barrier is transparent. As shown in Fig.1, many nonhuman animals and human infants have difficulties to solve the problem, as they repeatedly attempt to reach directly for the reward item, even in face of strong and negative feedback [8]. Psychologists have tended to explain this insensitivity to negative feedback as an inability to inhibit a lower-level behavioral control system, for example, a Pavlovian system [2]. Very interestingly, when first given experience with an opaque barrier, nonhuman primates succeed to solve the transparent barrier problem, which suggests that the major difficulty does not stem from a lack of self-control and the difficulty may be when and how to activate the self-control. Furthermore, experiments on rhesus monkeys show their losing the ability of solving the transparent detour problem due to lateral PFC lesions [9], which in turn proves the granular PFC is the key to non-apparent problems.

In the transparent experiments, subjects fail because they do not readily see the transparent barrier. Although the response is negative, there is no apparent reason for it, and so they continue to attempt the most efficient solution for the goal item. This would be an example showing that a problem-solving system sees a clear solution and is therefore overriding contrary feedback. Therefore, we further concluded that the transparent obstacle detour problem requires a non-apparent solution. And the problem-solving system must reformulate the problem by including the transparent obstacle via inferring from the effect of being blocked instead of seeing the obstacle directly. And we believe that many mammals don't have a mechanism to solve such a non-apparent problem. In this article, we use the detour problem with a transparent barrier as an example of a non-apparent problem.

Inspired by the hierarchical behavior control system of the primates, this article proposes a hierarchical computational model to solve hard and non-apparent problems. The model consists of four basic levels of behavioral control in primates. The first is a problem-solving system based off the hypothalamus, the first main system in the vertebrate brain. The control system of the first level is responsible for attaining the goal when it is perceived. Thus, it is not explicitly modeled in our computational model. The second level is a model-free problem-solving system based off the striatum located in front of the thalamus. The system is essentially an action-selection mechanism. And we specifically model the level as a reinforcement learning system [10], which is consistent with other research [11], [12]. The third level represents a model-based goal-directed problem-solving system, which solves apparent problems by using a model built from well-defined environments. The fourth level is another model-based problem-solving system, which solves non-apparent problems. The highest level is able to fix the model of the third level by inferring from the effect of the non-apparent feedback, which usually is not directly perceived. The four levels are hierarchically organized and our initial results have been published in a conference [13]. To clarify, the extension of the work mainly includes the following. First, the model is multi-level and hierarchical. The hierarchy lies in different levels of abstraction in states, therefore the extended work is scalable to much larger problems. Second, a network among levels is theoretically presented and studied experimentally at the first time, showing how and when higher levels control lower levels. Last, transparent obstacles are included in the experiments and much more results have been conducted to show the advantages of the model compared with others. To clarify, a direct modeling of neural dynamics in a brain is beyond the study of the work.

In literature, there are many brain-inspired computational systems. In [14], a hierarchical model inspired by rodent medial prefrontal cortex was developed. The model studies how the anterior cingulate cortex (ACC) determines and motivates what tasks to perform. The model also reveals that a patient with the ACC lesion is less interested in engaging in creative activities. The model is essentially implemented in a hierarchical reinforcement learning framework and uses three levels of abstraction of choices. By comparison, our method modeled more high-order cognitive functions, such as planning and reasoning, which has a potential to solve more difficult problems. In [15], researchers present an agent-based computational framework where a specific brain area is modeled by an autonomous agent, mimicing the special features of the area. And each agent is further modeled by a neural network system. A hierarchical co-evolutionary method is used to train the agents. Some researchers [16] developed a hierarchical multi-timescale recurrent neural network model to study how higher-order cognitive mechanisms may emerge. These models are both studied in a level of neurons, instead, we study the brain cognition in a functional level as the neural dynamics and the pathways among brain areas still remain controversial. In [17], [18], researchers reviewed neurocomputational models of working memory and concluded that computational models are very helpful to explore various cognitive mechanisms.

The rest of the paper is organized as follows. In section II we formally propose the computational model. Various levels of the model, connectionism among levels, and hierarchy are

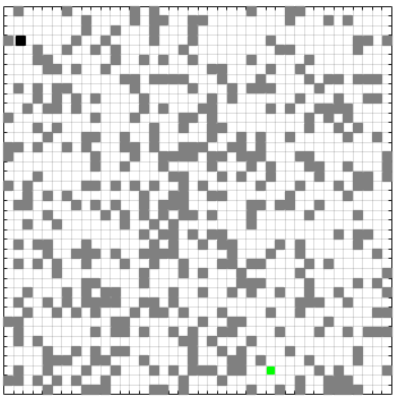


**FIGURE 1.** Example grid world, with the 'Initial state' denoted as the black block (Start) and the 'Goal state' in green. Obstacles are denoted in black.

detailed in the section. In section III we present experimental scenarios for evaluating the performance of the model. Discussions and conclusions are made in the last section.

## II. COMPUTATIONAL MODEL

We present a four-level computational model and each level could be viewed as an independent decision-making system. The four-level system represents the low to high cognition in the brain. Low level systems normally only need few knowledge of the world to make a decision, thus they are fast. Higher level systems require much more knowledge and are likely to have better solutions. The performance of the model is evaluated for a *foraging problem* in a *2D grid world*, where a testing agent must find a path from its start location to a goal location as illustrated in Fig. 1. The detailed model description is stated below, so are switching mechanisms among levels.

### A. MODEL DESCRIPTION

**Level 1** enables actual *goal attainment* once in view. For foraging, it represents the act of food assumption. Thus, it is not explicitly modeled in the scenarios. In the brain, it is the first behavioral control system based on hypothalamus.

**Level 2** represents a model-free problem-solving system based on striatum. Level 2 is modeled using a *Markov Decision Process* and *Reinforcement Learning (RL) framework* [11]. Thus, the world is represented by a set of states $S$, where $s_t \in S$ and $s_t$ is agent's state at the time step $t$; An agent in the world selects an *action*, $a_t \in A(s_t)$, where $A(s_t)$ is the set of possible actions available in state $s_t$. At next time $t+1$, as a result, the agent receives an immediate *reward*, $r_{t+1}$, and transits it to a new state, $s_{t+1}$. A mapping from a state $s_t$ to each possible action $a_t$ is called the agent's policy, $\pi(s_t, a_t)$, representing the probability of taking action $a_t$ when in state $s_t$. The agent seeks to find an optimal policy $\pi^*$ that leads to the maximum expected discounted future reward. For the foraging problem described above, (x,y) coordinates represent states of the world. In a state except the goal state, there are four actions: move up, move down, move left, and move right. We must point out that the agent may not able to identify the coordinate. Level 2 uses the following Q-learning algorithm [19]:

$$Q_{t+1}(s_t, a_t) = Q_t(s_t, a_t) + \alpha[r_{t+1} + \gamma * max_a Q(s_{t+1}, a) - Q_t(s_t, a_t)] \quad (1)$$

where $Q(s_t, a_t)$ is the learned action-value function of taking action $a_t$ when in state $s_t$; $\alpha$ is the learning rate $\alpha \in [0, 1]$ (the higher value $\alpha$ is, the faster the agent learns; however, a larger $\alpha$ would lead to a suboptimal solution.); $\gamma$ is the discount rate $\gamma \in [0, 1]$ and it determines the present value of future rewards (if $\gamma = 0$, the agent is only concerned with maximizing immediate rewards; as $\gamma$ approaches 1, the agent considers future rewards more strongly–the agent becomes more farsighted); and $max_a Q(s_{t+1}, a)$ is the maximum $Q$ value of taking the action $a$ when in the next state $s_{t+1}$. This one-step Q-learning approximates the optimal action-value function $Q^*$, independent of the policy used. A softmax action selection approach, Boltzmann distribution, is used to balance exploration and exploitation in Q-learning. It chooses action $a$ on the time $t$ with probability

$$\pi(a|s_t) = \frac{e^{Q_t(s_t,a)/\tau}}{\sum_b e^{Q_t(s_t,b)/\tau}} \quad (2)$$

where $\tau$ is a positive parameter called the *temperature*. After the learning procedure converges, the optimal policy is achieved by following the maximum $Q^*$ sequence. As you can see, rather than having an explicit model of the world, i.e., an understanding of how the states relate to each other in the grid world, level 2 sees the states independently, making decisions mainly based on the action values $Q(s, a)$ at each state.

**Level 3** represents model-based apparent problem-solving. In a grid world, apparent problems mean that information in the world is well defined and can be directly perceived without confusion. For any novel problem in the grid world, the problem solver cannot see the entire problem immediately — the world is too large — and so a cognitive model must be developed via initial experience with each state. The problems are considered as *multiagent problems* or stochastic games, where there are three types of agents: self, others and goals [20], [21]. Formally, the multiagent problem is defined as a 4-tuple $\{S, A, P, R\}$. $S$ is a set of states of the world; $A$ is a finite set of agent actions $i$; $P(s\prime|s, a_i, a_{-i})$ is the state transition probability function meaning the probability of moving from state $s$ to state $s\prime$ by taking action $a_i$ by agent $i$ and by taking actions $a_{-i}$ by all other agents; $R$ is the reward function for agent $i$. This model thus has a clear understanding of the relationships among the states which Level 2 cannot see. For the foraging problem in the grid world, the cognitive model consists of four components: (1) (x,y) coordinates of grid world that can be perceived and identified by the agent represent states of the world; (2) the set of available actions; (3) the state transition probability; (4) the identification of apparent obstacles. For current study, obstacles are static and there is only one action of obstacles: blocking. As stated

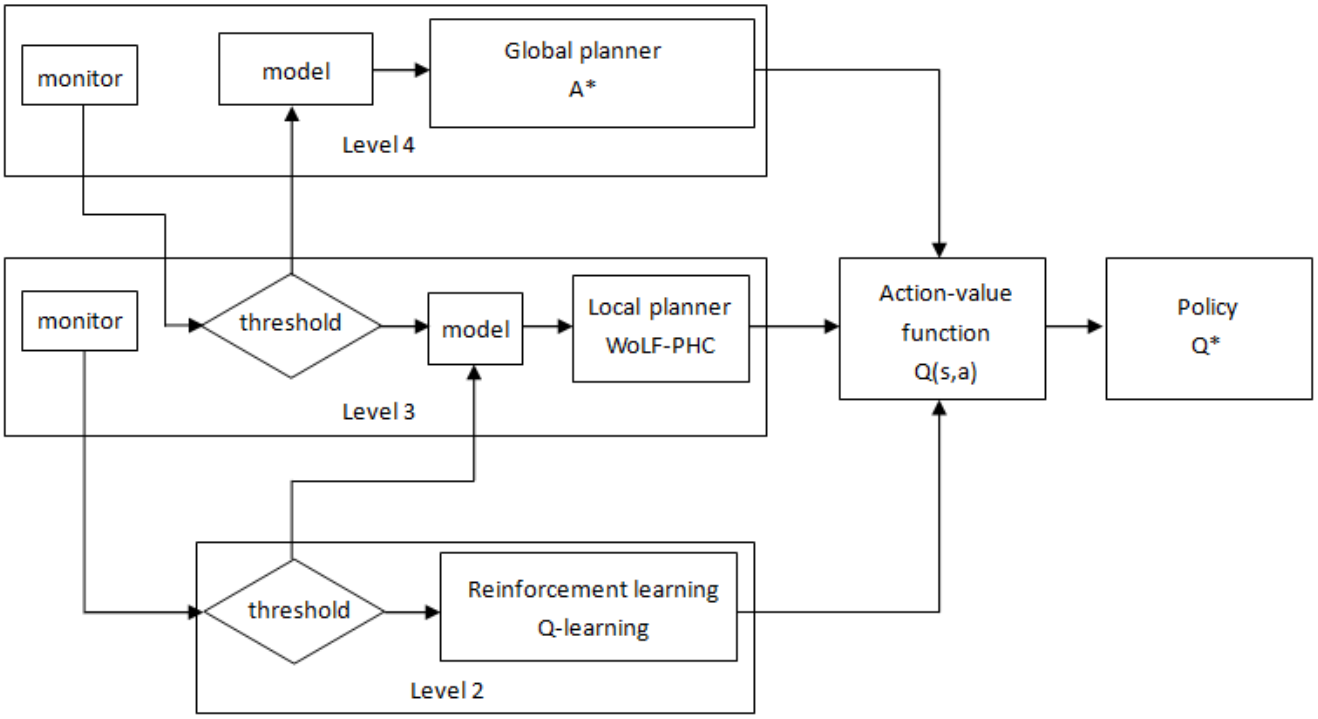



**FIGURE 2.** The network among levels.

in previous section, Level 3 can only see opaque obstacles. Level 3 uses a Win-or-Learn-Fast Policy Hill-Climbing (WoLF-PHC) algorithm that updates Q-functions with the Q-learning rule as in equ. 1 and policies with a WoLF rule [22]–[24] as follows.

$$\pi_{i,t+1}(s_t, a_t) = \pi_{i,t}(s_t, a_t) + \begin{cases} \delta_{i,t} & \text{if } a_t = argmax_a Q_{i,t+1}(s_t, a) \\ -\dfrac{\delta_{i,t}}{|A| - 1} & \text{otherwise} \end{cases} \quad (3)$$

where $|A|$ denotes set cardinality; $\delta_{i,t}$ denotes the learning rate and is updated as follows.

$$\delta_{i,t} = \begin{cases} \delta_w & \text{if winning} \\ \delta_l & \text{if losing.} \end{cases} \quad (4)$$

The agent is winning if $\pi_{i,t}(s_t, a_t) > \bar{\pi}(s_t, a)$ and losing otherwise, where $\bar{\pi}(s_t, a)$ is the average policy of the agent $i$ at the state $s_t$. And $\delta_l > \delta_w$. By using the variable learning rates, it allows the agent to learn quickly when losing.

**Level 4** represents the system that attempts to find these non-apparent solutions. For the current study, a hidden cause occurs, i.e., a transparent obstacle, blocks the direct path to the goal. Such an obstacle is literally non-apparent to Level 3. A non-apparent problem could be generalized as a partially observable Markov decision process (POMDP) [25]. Similar to Level 3, the agent builds the cognitive model of the world via its experience with the world. Differently, it learns hidden causes by inferring the feedback from the environment. For example, when it is blocked by invisible obstacles, it uses this information conceptually, inferring that there is an ''agent'' doing the blocking. In this way, it can build a fairly complete cognitive model of the world. Level 4 currently uses the planning algorithm $A^*$ (called 'A star') to find an optimal path around the obstacles to the goal [26]. In the paper, words, non-apparent, transparent and invisible, are interchangeable.

### B. LEVEL SWITCHING MECHANISM

Four levels work concurrently and cooperatively. Fig. 2 shows the connection among different levels. In a whole system, the control flows from up to down. A higher level can inhibit a lower level. Level 4 monitors level 3 to determine if it needs to take control; Level 3 monitors level 2 to determine if it needs to take control as well. Level 2 acts as an actor via interacting with the world and as a result updates action-value function Q(s,t) under certain policy. Q values are the working memory of the overall model. In a case in which level 3 or 4 needs to take over the system, the working memory will be updated on the fly as a consequence. After that, the control is returned back to level 2 and the updated Q values will be used. Q-learning in level 2 is able to converge an optimal solution for a single-agent MDP, however, it doesn't guarantee to converge for multiagent cases, for each state is stochastic and determined by joint actions of agents. As shown in Fig. 2, level 3 monitors the 'overflow' threshold which accumulates when other agents' actions prevent current agent from converging the optimal policy. Whether other agents' actions confuse the current agent can be told by equ.5

$$n_{s_t} = n_{s_t} + 1, \text{ if } r_{t+1}(s_t, a_t) \neq r_t(s_t, a_t) \quad (5)$$

**Algorithm 1** Algorithm of Our Four-Level Computational Model

**Input:** learning rates: $\alpha, \delta_w, \delta_l$; thresholds: $n_{T_2}, n_{T_3}$; initialize, $Q(s,t) \leftarrow 0, \forall s, n_{s_t} \leftarrow 0$

**Output:** optimal policy: $\pi^*(s,t)$

set start and goal: $s_{start}$ and $s_{goal}$;
activate level 2: $i_{lv} = 2$;
**repeat**
    switch($i_{lv}$);
    case 2: update $Q(s,t)$ using Q-learning equ. 1;
        Check-if-overflow($n_{T_2}$, 2);
    case 3: build or update a map of the world;
        update $Q(s,t)$ using Q-learning equ. 1;
        update $\pi(s,t)$ using WoLF-PHC equ. 3;
        Check-if-overflow($n_{T_3}$, 3);
    case 4:
        build or update a map of the world;
        run $A^*$ and update $Q(s,t)$ on the path;
        switch to level 2: $i_{lv} = 2$;
**until** ($s_{goal}$ is reached)
Execute level 1.
**procedure** Check-if-overflow($n_{T_j}, n_{level}$)
    $i_{lv} = 2$;
    $n_T = n_{T_j}$;
    **if** $r_{t+1}(s_t, a_t) \neq r_t(s_t, a_t)$ **then** $n_{s_t} = n_{s_t} + 1$
    **end if**
    **if** $\sum_S n_{s_t} \geq n_T$ **then**
        $i_{lv} = n_{level} + 1$;
        reset: $\forall s, n_{s_t} \leftarrow 0$;
    **end if**
    return $i_{lv}$
**end procedure**

where, $n_{s_t}$ denotes the number of confusion; $r_{t+1}(s_t, a_t)$ denotes the reward of the current agent received at state $s_t$ at time $t+1$ by taking action $a_t$; and $r_t(s_t, a_t)$ is the reward at time $t$. Level 3 takes over control of level 2 if equ. 6 is satisfied, where $\sum_S n_{s_t}$ denotes the number of states confused and $n_T$ denotes the value of threshold.

$$\sum_S n_{s_t} \geq n_T \tag{6}$$

When there appears non-apparent obstacles, level 3 gets confused. Level 4 takes control of level 3 just as level 3 takes control of level 2, more specifically, level 3 gets confused when it gets trapped due to non-apparent obstacles or local minima [27] that prevents the agent finding an effective path to the goal. The overall algorithm is presented in Algorithm 1.

### C. HIERARCHY

In a large grid world, the state number could be huge. The action space also contributes to learning complexity. For an action space of size $m$, the number of state-action pairs is $m$ times of the state number. An efficient approach to reduce computational complexity is to represent states abstractly.

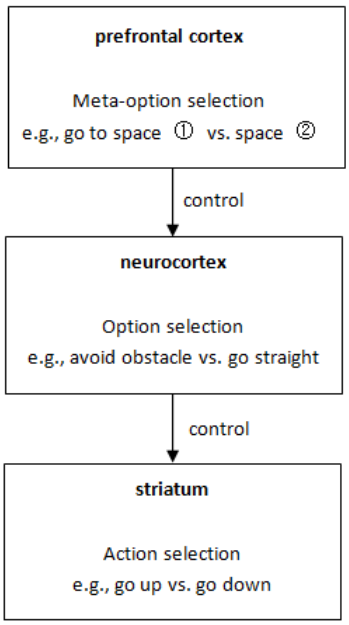



**FIGURE 3.** Levels of action abstraction.

For the foraging problem, state space abstraction using cell decomposition [28], [29] is used. Level 4 runs on the reduced state space. Another hierarchy lies in various levels of action abstraction. As shown in Fig. 3, in prefrontal cortex, metaoption selection is implemented. An example metaoption action in the foraging environment is to go to certain space extracted based on the state space abstraction rules. In neurocortex, option selection is made-so as to maintain the higher metaoption selection. For example, going to space ① may require an option of going around certain obstacle. In striatum, primitive actions specific to the option will be selected. To complete an option of going around obstacles requires actions such as going right, going up, and so on.

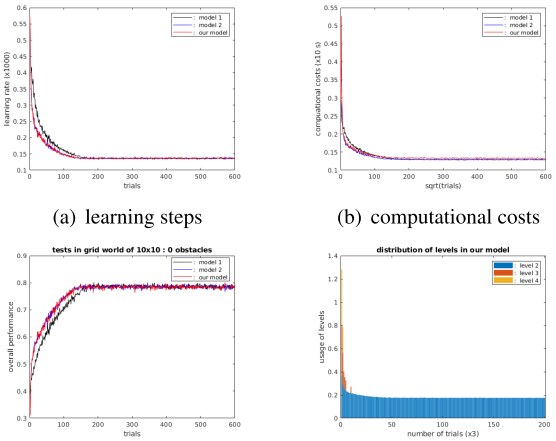
Perception is also hierarchically processed in the brain [30], [31]. Perception signals are processed in low-level brain areas to extract low-level features such as dot, line, and etc., which are inputs of models of levels 1 and 2. High-order brain areas are able to extract high-order non-apparent signals that are inputs to Level 4.

## III. EXPERIMENTAL RESULTS

We compared our four-level model to the following models:

1) Model 1: consisting of levels 1 & 2
2) Model 2: levels 1, 2, & 3
3) Our model, model 3: levels 1, 2, 3, & 4, that is, Model 2 plus Level 4

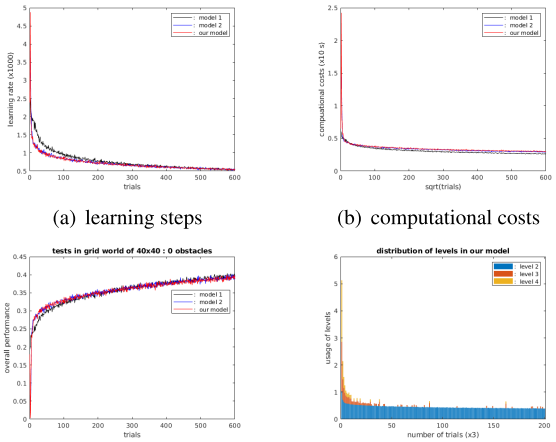
All models assume the existence of Level 1. Model 1 simply uses model-free reinforcement learning, probably representing an ancestral vertebrate. Model 2 combines model-free RL with the ability to solve apparent problems, perhaps representing the ancestral mammalian brain. Model 3 is our four-level model of the primate brain. The models are simulated and compared in grid worlds as shown in Fig. 1. We examined the effects of (1) grid world size, (2) number of obstacles, and (3) invisible obstacles. Two measures




(a) learning steps (b) computational costs

(c) overall performance (d) usage of levels

**FIGURE 4.** A world of a size of 10 by 10, with zero obstacles. Three models were tested in the small world with zero obstacles and each figure was plotted by taking the average of 50 runs. (a) the learning steps of the three models, (b) the computational costs, (c) the overall performance of three models as a function of learning steps and computational costs, (d) the usage of different levels for our four-level model.




(a) learning steps (b) computational costs

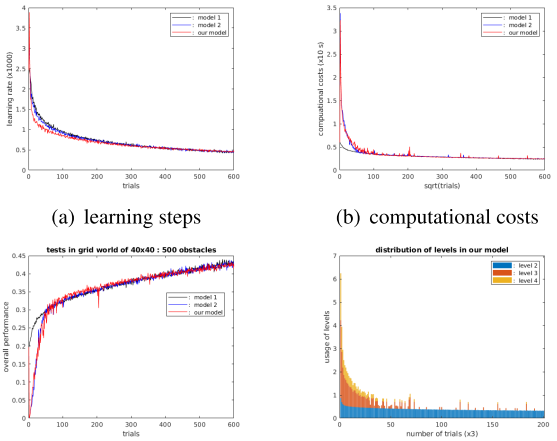
(c) overall performance (d) usage of levels

**FIGURE 5.** A large world of a size of 40 by 40, with zero obstacles.

were used: (1) cumulative number of steps to reach the goal, $N_r$, and (2) cumulative computational cost, $C_r$. They were combined as the overall performance score, $P$, and $P = k_1 N_r + k_2 C_r$, where $k_1$ and $k_2$ are trial-and-error parameters, and $k_1 + k_2 = 1$, $k_1 > k_2$. Each run has 600 trials and a complete experiment includes 50 runs. The experimental data is the average of the 50 runs.

### A. GRID WORLD SIZE

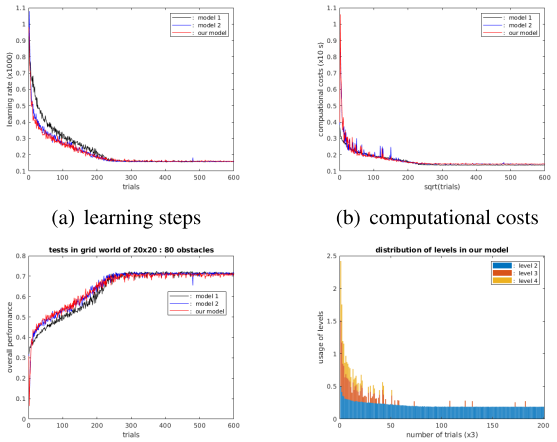
The dimension is used to test how the size of the world may affect the performance of different models. Three different grid world sizes are tested in our experiments, a small world of $10 \times 10$, a medium world of $20 \times 20$, and a large world of $40 \times 40$. As shown in Fig. 4, model 1 initially spent a lot of time to explore and learn the world, thus converged to the optimal solution slower than models 2 and 3 which planned paths with the model learned during exploration.




(a) learning steps (b) computational costs

(c) overall performance (d) usage of levels

**FIGURE 6.** A large world of a size of 40 by 40, with 500 obstacles.




(a) learning steps (b) computational costs

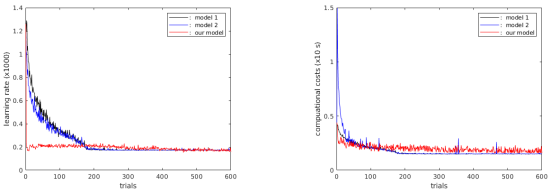
(c) overall performance (d) usage of levels

**FIGURE 7.** A grid world of the size of 20 by 20, with 80 obstacles.

Since the world is small and free of obstacles, level 3 works in the very early trials and level 4 barely works at the whole run as in Fig. 4 (d). Thus, performance of model 2 and 3 is very close to each other. As we know, the faster the agent finds the solution, the better chance it has to survive. These characteristics hold same even in a larger obstacle-free world as shown in Fig. 5. Of course, the larger world makes models longer time to converge.

### B. NUMBER OF OBSTACLES

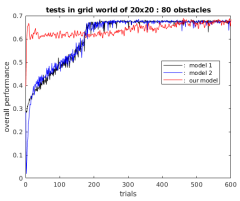
To examine how number of obstacles could affect the performance, we included 500 obstacles in the above large world. A picture of the world is shown in Fig. 1. In Fig. 5 (d), our model shows that Level 4 plays a very important role in the first 100 trials and helps it converge much faster as shown in Fig. 5 (a). That is, Level 3 of model 2 got troubled in the early stages. We found a major reason is that the cognitive map of the world built in the early stages is incomplete, and the incomplete map together with large amount of obstacles forms some local minima which trap Level 3 and make it downgraded to Level 2 instead. When the map becomes



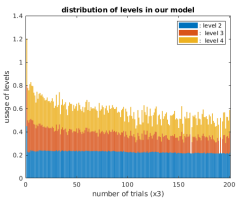

(a) learning steps



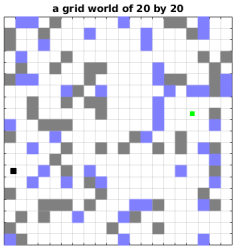

(b) computational costs




(c) overall performance




(d) usage of levels




(e) the medium-size world

**FIGURE 8. The same grid as in Fig. 7: 80 apparent obstacles and 50 invisible obstacles. The gray blocks are apparent obstacles. The light blue blocks are invisible obstacles.**

complete at later stages, the optimal Q values on the path planned by models 2 and 3 are updated. After that, Level 2 takes the control with few helps from higher levels.

### C. INVISIBLE OBSTACLES

To further examine how non-apparent obstacles–invisible obstacles–could affect the performance. We compared tests on a medium size world with 80 apparent obstacles with tests on the same world but with 50 invisible obstacles more. The results for the former are shown in Fig. 7, and the later in Fig. 8. In the former case, Level 3 shows the validness to solve it, so Model 2 and Model 3 are almost equivalent (as Level 4 in Model 3 is inactive in most of time.). As a comparison, in Fig. 8 (d), Level 4 is activated almost at the whole 600 trials, indicating that Level 3 is completely not valid to solve the transparent problem. Hence, our model showed much better performance again.

## IV. CONCLUSION

Multiple levels of computational algorithms are used to represent the complexity of the various levels of brain areas. The experiments show our model is able to improve the problem solving of the agent, especially in non-apparent scenarios. Our model does so because it can update the broken model by inferring hidden cause. And this may give us a hint of how humans might solve difficult problems. Our research also shows that a pure system, either a model-free method or a model-based method, is not an efficient way to solve most problems. Besides, Various levels of problem solvers in our model share a same slice of working memory, which is also consistent with the biological facts. However, we never meant to conclude that the different brain areas only rely on the computational model stated in the paper. The complexity of a primate brain is far more than what is presented in the paper. The major purpose is to present a way in which the primate brain might be organized.

## ACKNOWLEDGMENT

Great thank goes to my previous colleague, Omar A. El-Shroa, at Dartmouth College.

## REFERENCES

[1] A. Rangel, C. Camerer, and P. R. Montague, "A framework for studying the neurobiology of value-based decision making," *Nature Rev. Neurosci.*, vol. 9, no. 7, pp. 545–556, Jul. 2008.
[2] G. F. Striedter, *Principles of Brain Evolution*, Sinauer Associates, Sunderland, MA, USA, 2005.
[3] R. C. Arkin, *Behavior-Based Robotics*. Cambridge, MA, USA: MIT Press, 1998.
[4] R. E. Passingham and S. P. Wise, *The Neurobiology of the Prefrontal Cortex: Anatomy, Evolution, and the Origin of Insight*. Oxford, U.K.: Oxford Univ. Press.
[5] L. Regolin, G. Vallortigara, and M. Zanforlin, "Object and spatial representations in detour problems by chicks," *Animal Behaviour*, vol. 49, no. 1, pp. 195–199, Jan. 1995.
[6] D. Shi, E. G. Collins, and D. Dunlap, "Robot navigation in cluttered 3-D environments using preference-based fuzzy behaviors," *IEEE Trans. Syst., Man, Cybern., B (Cybern.)*, vol. 37, no. 6, pp. 1486–1499, Dec. 2007.
[7] M. F. Selekwa, D. D. Dunlap, D. Shi, and E. G. Collins, "Robot navigation in very cluttered environments by preference-based fuzzy behaviors," *Robot. Auto. Syst.*, vol. 56, no. 3, pp. 231–246, Mar. 2008.
[8] J. D. Wallis, R. Dias, T. W. Robbins, and A. C. Roberts, "Dissociable contributions of the orbitofrontal and lateral prefrontal cortex of the marmoset to performance on a detour reaching task," *Eur. J. Neurosci.*, vol. 13, no. 9, pp. 1797–1808, May 2001.
[9] A. Diamond, "Developmental time course in human infants and infant monkeys, and the neural bases of, inhibitory control in reaching," *Ann. New York Acad. Sci.*, vol. 608, no. 1, pp. 637–676, Dec. 1990.
[10] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 1998.
[11] R. Granger, "Engines of the brain: The computational instruction set of human cognition," *AI Mag.*, vol. 27, pp. 15–32, Jun. 2006.
[12] P. Dayan and N. D. Daw, "Decision theory, reinforcement learning, and the brain," *Cogn., Affect., Behav. Neurosci.*, vol. 8, no. 4, pp. 429–453, 2008.
[13] J. D. Kralik, D. Shi, and O. A. El-Shroa, "From low to high cognition: A multi-level model of behavioral control in the primate brain," in *Proc. 38th Annu. Meeting Cognit. Sci. Soc.*, Philadelphia, PA, USA, Aug. 2016.
[14] C. B. Holroyd and S. M. McClure, "Hierarchical control over effortful behavior by rodent medial frontal cortex: A computational model," *Amer. Psychol. Assoc.*, vol. 122, no. 1, pp. 54–83, 2014.
[15] M. Maniadakis and P. Trahanias, "Modelling robotic cognitive mechanisms by hierarchical cooperative coevolution," *Int. J. Artif. Intell. Tools*, vol. 16, no. 6, pp. 935–966, Dec. 2007.
[16] F. Alnajjar, Y. Yamashita, and J. Tani, "The hierarchical and functional connectivity of higher-order cognitive mechanisms: Neurorobotic model to investigate the stability and flexibility of working memory," *Frontiers Neurorobot.*, vol. 7, pp. 1–13, 2013.
[17] D. Durstewitz, J. K. Seamans, and T. J. Sejnowski, "Neurocomputational models of working memory," *Nature Neurosci.*, vol. 3, no. S11, pp. 1184–1191, Nov. 2000.
[18] J. E. Lisman, J.-M. Fellous, and X.-J. Wang, "A role for NMDA-receptor channels in working memory," *Nature Neurosci.*, vol. 1, no. 4, pp. 273–275, Aug. 1998.
[19] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, nos. 3–4, pp. 279–292, 1992.

[20] D. Shi, M. Sauter, and X. Sun, "An extension of Bayesian game approximation to partially observable stochastic games with competition and cooperation," in *Proc. Int. Conf. Artif. Intell.*, 2010, pp. 1–7.
[21] G. Weiss, *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. Cambridge, MA, USA: MIT Press, 1999.
[22] M. Bowling and M. Veloso, "Multiagent learning using a variable learning rate," *Artif. Intell.*, vol. 136, no. 2, pp. 215–250, Apr. 2002.
[23] S. Singh, M. Kearns, and Y. Mansour, "Nash convergence of gradient dynamics in general-sum games," in *Proc. 16th Conf. Uncertainty Artif. Intell.*, San Mateo, CA, USA, 2000, pp. 541–548.
[24] L. Busoniu, R. Babuska, and B. De Schutter, "A comprehensive survey of multiagent reinforcement learning," *IEEE Trans. Syst., Man, Cybern., C (Appl. Rev.)*, vol. 38, no. 2, pp. 156–172, Mar. 2008.
[25] N. Roy, G. Gordon, and S. Thrun, "Finding approximate POMDP solutions through belief compression," *J. Artif. Intell. Res.*, vol. 23, pp. 1–40, Jan. 2005.
[26] H. Choset, K. M. Lynch, and S. Hutchinson, *Principles of Robot Motion: Theory, Algorithms, and Implementations*. Cambridge, MA, USA: MIT Press, 2005.
[27] C. Ordonez, E. G. Collins, M. F. Selekwa, and D. D. Dunlap, "The virtual wall approach to limit cycle avoidance for unmanned ground vehicles," *Robot. Auto. Syst.*, vol. 56, no. 8, pp. 645–657, Aug. 2008.
[28] J. C. Latombe and J. Barraquand, "Robot motion planning: A distributed representation approach," *Int. J. Robot. Res.*, vol. 10, no. 6, pp. 628–649, 2005.
[29] R. Siegwart and I. R. Nourbakhsh, *Introduction to Autonomous Mobile Robots*. Cambridge, MA, USA: MIT Press, 2004.
[30] D. H. Hubel and T. N. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," *J. Physiol.*, vol. 160, no. 1, pp. 106–154, 1962.
[31] D. J. Felleman and D. C. Van Essen, "Distributed hierarchical processing in the primate cerebral cortex," *Cerebral Cortex*, vol. 1, no. 1, pp. 1–47, Jan. 1991.

**DONGQING SHI** (Member, IEEE) received the B.S. degree in mechanical engineering from Harbin Engineering University, Harbin, China, in 1999, the M.S. degree in mechanical engineering from Zhejiang University, Hangzhou, China, in 2004, and the Ph.D. degree in mechanical engineering from Florida State University, Tallahassee, FL, USA, in 2006.

From 2007 to 2009, he worked as a Control Engineer with Florida A&M University. From 2009 to 2011, he was a Senior Research Associate with the Dartmouth College, Hanover, NH, USA. From 2011 to 2017, he worked as the Vice Manager with Zhejiang Xibeihu Special Vehicle Company Ltd. After that, he worked as a Chief Engineer with Zhejiang YAT Electrical Appliance Company Ltd. He is currently an Associate Professor with the Yiwu Industrial and Commercial College. His research interests include autonomous robots navigation and control, artificial intelligence, and machine learning.

**JERALD KRALIK** received the B.S. degree in zoology from Michigan State University and A.M. and the Ph.D. degrees in psychology from Harvard University. He is currently a Visiting Professor with the Department of Bio and Brain Engineering, Korea Advanced Institute of Science and Technology (KAIST). Before the position, he was an Assistant Professor with the Department of Psychological and Brain Sciences, Dartmouth College. He also completed post-doctoral positions in behavioral neuroscience with the Duke University Medical Center and the National Institute of Mental Health. His research interests include animal cognition and behavior, cognitive neuroscience, and brain engineering.

**HAIYAN MI** received the B.S. degree in polymer materials and engineering from Zhejiang SCI-Tech University, Hangzhou, Zhejiang, in 2001, and the M.S. degree in business management from Florida State University, in 2006. She is currently a Lecturer with the Yiwu Industrial and Commercial College. She has published more than ten articles. Her research interests mainly include mathematical modeling and optimal control in financial systems.