The Thirty-First International Florida
Artificial Intelligence Research Society Conference (FLAIRS-31)

Specialization versus Re-Specialization: Effects of Hebbian Learning in a Dynamic Environment

Vera A. Kazakova, Annie S. Wu
University of Central Florida
Orlando, FL 32816-2362
kazakova@cs.ucf.edu, aswu@cs.ucf.edu

Abstract

Specializing on a subset of tasks available within a system allows agents to more efficiently fulfill system demands. When demands change, agents need to Re-Specialize. Since Re-Specialization inherently requires undoing some prior Specialization, the opposing effort often results in agents settling on a worse task allocation than after Specialization, even when presented with similar demands. In this work, we demonstrate these task allocation differences by looking at how well demands are fulfilled, as well as how much task switching is happening within the system. We analyze what causes the observed differences and discuss potential approaches to improving Re-Specialization in the future.

1 Introduction

In this work, we show how threshold reinforcement Specialization dynamics can hinder Re-Specialization in decentralized MultiAgent Systems (MAS). We apply a well-known, biologically inspired Specialization model by Theraulaz, Bonabeau, and Deneubourg (1998) (from here on denoted as TBD) to decentralized task allocation within a multi-task enviroment where each task in unlimited supply but has limited and varying demand. We present the differences between the resulting Specialization and Re-Specialization behaviors and investigate the underlying reasons.

In fully decentralized, multi-task, dynamic environments, where each task is in unlimited supply but has limited demand subject to change over time, threshold reinforcement is the dominant approach to Specialization. In this *unlimited-supply-limited-demand* (USLD) domain, each agent can work on up to one task during each time step, but all agents can choose the same task or even no task at all, regardless of system needs. This differs from the domain tackled by market-based approaches, which assume that each task can only be taken up by a single agent, who must win it based on some dominance metric. In USLD, each task's demand can be fulfilled at any point throughout a simulation day, by any number of agents, based solely on the total amount of daily work on the task. Without the added domain-specific costs and task availability constraints, USLD is a more general and complex domain. To minimize switching among tasks, it is most efficient to allow agents to develop an affinity for some tasks and an aversion toward others, i.e. to *Specialize*. Threshold reinforcement uses task stimuli to elicit agent action and develop preferences based on Hebbian learning: when agents act on a task, they becomes more likely to act on it again (Cicirello and Smith 2003; Campos et al. 2000; Price and Tiňo 2004; Kittithreerapronchai and Anderson 2003). When demands change, a commensurate change in agents' affinities and aversions (i.e. agents' task-specific thresholds) is needed. Following the same threshold reinforcement process, previously Specialized agents begin to adjust their preferences again and *Re-Specialize*. The most commonly used threshold reinforcement model is TBD.

Although Specialization and Re-Specialization follow the same threshold reinforcement model, the resulting behaviors differ. When unspecialized agents are first presented with a set of tasks with various demands, agents begin to self task allocate. For agents to Re-Specialize, however, means to unspecialize from existing tendencies, i.e. to develop an aversion to the same tasks the agents had been developing an affinity for during Specialization. As the underlying mechanism is identical throughout, Re-Specialization efforts are at a disadvantage, battling the draw of existing Specialization. As a result, Re-specialized agents can present notably more switching among tasks than Specialized agents, even when the Re-Specialization needs are a mirror image of the Specialization task demands (Wu and Kazakova 2017). Additionally, Re-Specialization may be required for reasons other than changes in demand, such as a change in the total number of agents, a replacement of some agents, or a change in available tasks. Thus, threshold reinforcement can lead to suboptimal task allocation in dynamic environments.

In this work, we first overview the TBD behavioral model used in our system. We then conduct experiments to showcase performance differences between Specialization and Re-Specialization under identical task demands. We discuss the reasons for the observed differences by: (1) overviewing the behavior driving task thresholds, task stimuli, and action probabilities; (2) observing how these values change during Specialization; and (3) how the values change during Re-Specialization. We then take a closer look at behavior near low-stimulus-low-threshold values, which appear to negatively affect Re-Specialization, and discuss some potential avenues for improvement.

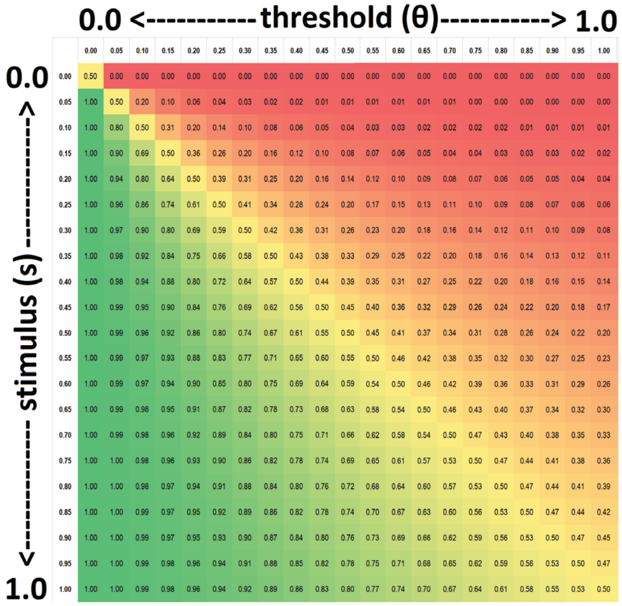

Figure 1: TBD Action Probability: $P_{a,t} = s_t^2/(s_t^2 + \theta_{a,t}^2)$ (Theraulaz, Bonabeau, and Deneubourg 1998)

2 TBD Threshold Reinforcement

Before we analyze the behavioral differences under Specialization vs. Re-Specialization, we first review the main behavioral driver employed in our tests: TBD. In this section, we review the TBD model: its probability formula, the distribution of probabilities it generates, and the redefined discontinuity where $s_t = \theta_{a,t} = 0.0$. We also discuss how TBD agents behave based on the combination of task thresholds and task stimuli, as well as how these thresholds and stimuli are initialized and changed over time through agent actions.

TBD combines task stimulus with agents' affinity for tasks to produce behavior approximating what has been observed in complex natural societies (Theraulaz, Bonabeau, and Deneubourg 1998). Responding to higher stimuli ensures agents prefer to act on the most needed tasks. Responding to lower thresholds ensures agents prefer to act on tasks they've acted on before. Thus, focusing on one or the other represents a trade-off between responsiveness and specialization, respectively. To balance out these tendencies, the current stimulus for a task and an agent's affinity for that task are combined into a *probability* to act on that task:

$$\mathbf{P_{a,t} = s_t^2/(s_t^2 + \boldsymbol{\theta}_{a,t}^2)} \quad \text{where } \mathbf{s} \in [\mathbf{0.0}, \mathbf{1.0}], \boldsymbol{\theta} \in [\mathbf{0.0}, \mathbf{1.0}]$$
$$\mathbf{P_{a,t} = 0.5} \quad \text{where undefined}(\boldsymbol{\theta}_{\mathbf{a,t}} = \mathbf{s_t} = \mathbf{0.0})$$

where $P_{a,t}$ is the probability of agent a to act on task t, given a current stimulus s_t for task t and the agent's affinity threshold $\theta_{a,t}$ for task t. Note the redefinition of $P_{a,t}$ to avoid division-by-zero when s_t and $\theta_{a,t}$ are both zero. The map of resulting probabilities is shown in Fig. 1. In the rest of this work, we refer to the corners of this map as: ⌜ (top-left), ⌝ (top-right), ⌟ (bottom-right), and ⌞ (bottom-left).

On every time step, agents consider the available tasks one at a time, in descending $P_{a,t}$ order, as this is shown to work well for specialization (Wu and Kazakova 2017). When one task is not chosen, the agent moves on to considering the next, and so on until all tasks have been considered, in which case the agent defaults to idling (T0). Given a set of 4 tasks (T1-T4), each agent a compares a randomly generated $value_{a,t}$ to $P_{a,t}$, $t \in [1, 4]$. If this value is below $P_{a,t}$, the task is acted on during this step. Otherwise, the following task is considered. Note that this differs from TBD, where only one task is available for consideration at a time.

Agents are unaware of actual task demands. Their perceived task t need, termed *stimulus* s_t in TBD, can perhaps be more accurately defined as task *deficit*, since s_t remains unchanged when agents do the exactly sufficient amount of work to keep up with demand. Calculated as:

$$\mathbf{deficit_t = s_t = \frac{dailyDemand_t - currentAmount_t}{dailyDemand_t}},$$

truncated as needed, to remain within the range [0.0,1.0].

Specialization in TBD employs threshold reinforcement akin to Hebbian learning: when agent a acts on task t, its $\theta_{a,t}$ decreases, increasing the agent's future probability to act on task t again. When agent a chooses task t, it Specializes toward task t ($\theta_{a,t} \to 0.0$) and against all other tasks ($\theta_{a,t' \neq t} \to 1.0$) according to the following update rules:

$$\boldsymbol{\theta}_{\mathbf{a,t}} = \boldsymbol{\theta}_{\mathbf{a,t}} - \boldsymbol{\xi} \qquad \text{(where } \xi \text{ is the affinity rate)}$$
$$\boldsymbol{\theta}_{\mathbf{a,t' \neq t}} = \boldsymbol{\theta}_{\mathbf{a,t'}} + \boldsymbol{\phi} \qquad \text{(where } \phi \text{ is the aversion rate),}$$

with θ truncated as needed to remain within range [0.0,1.0].

We use $\xi = 0.01$ and $\phi = \xi \ / \ (number\ of\ tasks - 1)$. In TBD, aversion rate is only 0.1*ξ for a setup of 2 tasks. We feel, however, that it is more generally justifiable to assume that aversion increases at the rate of 1/(number of other tasks), such that however much affinity is gained in one task, is simultaneously lost by all the other tasks combined. This increase in aversion rate should make Re-Specialization faster. Consequently, showing that it is still slower than Specialization even with this higher aversion rate is an even more meaningful result. Note that we avoid the terminology of "learning" and "forgetting" used in TBD, as these terms may imply skill acquisition and loss, respectively, while in actuality we are only discussing building a preference toward some tasks and against others.

3 Threshold Reinforcement Performance: Specialization vs. Re-Specialization

The TBD threshold reinforcement model is often used in its original or adapted forms as the driving force behind decentralized Specialization in MAS. We hypothesize that under TBD and similar approaches, agents can generally Specialize faster to a set of demands than they can Re-Specialize given new demands. In this section, we test our hypothesis by comparing how agents behave when presented with different demands under one of two setups: new demands are presented to (1) agents with randomly distributed task thresholds or (2) agents who are each fully Specialized on any one of the tasks ($\theta_{a,i}$=0.0) and fully Specialized away from the other tasks ($\theta_{a,j \neq i}$=1.0). These setups correspond to Specialization and Re-Specialization, respectively.

Daily task completion statistics allow agents to do sufficient work over the course of a day, without regard for when in the day, by how many agents, or over how many of the day's steps. Daily task demand percentage indicates what portion of agents must be working full time on this task or, equivalently, what portion of the total available working hours must be spent on this task. Each day is broken into steps, symbolizing how often agents reconsider the stimuli and select an action. We test with teams of 20 agents, but as task demands scale with number of agents, team size is inconsequential; TBD demonstrates identical behavior with 10 and with 100 agents (Wu and Kazakova 2017)).

Unspecialized agents begin Specialization with uniformly distributed random thresholds $\theta_{a,t}$ assigned to each agent a for every task t. Re-Specializing agents face new demands (or rather the resulting new s_t) while already Specialized on previous demands. Consequently, they must act against existing specialization in order to develop new preferences.

We conduct 6 experiments during every run and repeat these for 32 runs. Each run consists of 300 days of 10 steps each, subdivided into 6 groups of 50 days, one for each experiment. The 6 experiments correspond to 3 sets of task Demand Periods (DP) for Specialization, followed by 3 identical DP for Re-Specialization. Table 1 lists the 6 experiments along with task demands for each. Percentages correspond to how much work must be dedicated to a task throughout the day to achieve 100% task completion (e.g. DP I requires 35% of all daily work possible to be done on T1).

On the first 150 days of each run, we test how unspecialized agents Specialize during each DP. For this, we convert Re-Specialization behavior into Specialization by resetting agents' $\theta_{a,t}$ when demands change, i.e. on days 0, 50, and 100, to simulate identical unspecialized initial conditions. On the following 150 days, we test how pre-specialized agents (each having one $\theta_{a,t} = 0$ and three $\theta_{a,t} = 1.0$) Re-Specialize to new demands. Agents' thresholds are not reset over this period: when Re-Specializing to DP I (day 150), $\theta_{a,t}$ correspond to how they had Specialized over days 100-149; when faced with DP II (day 200), agents keep their $\theta_{a,t}$ from days 150-199, and when entering DP III (day 250), agents begin with $\theta_{a,t}$ developed over days 200-249.

Performance over a DP is measured by daily task completion, daily task switches, and when the last task switch occurred. Daily task completion represents how close agents get to the required amount activity on that task. Specifically, we track *deviations*: how much task completion deviates from 100% each day, calculated as the absolute value of the difference between the achieved task completion and 100%.Daily task switches indicate the number of times agents switch tasks each day, with more switching indicating lower efficiency due to lower specialization. Timing of last time switch within a DP indicates how quickly agents are able to settle on a new stable task assignment.

Demand Period	Days	T1	T2	T3	T4
Specialization I	0-49	35%	20%	30%	15%
Specialization II	50-99	40%	25%	25%	10%
Specialization III	100-149	10%	25%	25%	40%
Re-Specialization I	150-199	35%	20%	30%	15%
Re-Specialization II	200-249	40%	25%	25%	10%
Re-Specialization III	249-299	10%	25%	25%	40%

Table 1: Task Demands for each Demand Period

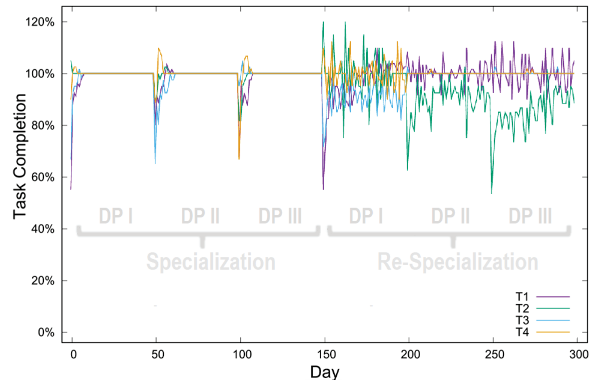

Figure 2: Sample run: tasks' T0-T4 demand completion. T0 has no line, as without a demand, % task completion is n/a.

Fig. 2 shows a representative sample run with task completions (not deviations) plotted in a different color line for each task. Left half of the timeline shows Specialization on DP I, II, and III; right half of the timeline shows Re-Specialization on DP I, II, and III. Specializing agents quickly find a task allocation that fulfills all tasks, stabilizing completion lines to 100% soon after days 0, 50, and 100. Despite identical demands and resources, Re-Specializing agents struggle to fulfill the tasks: completion lines fluctuate above and below 100% throughout all three DP.

Table 2 shows performance averaged over 32 runs: task deviations per DP, daily task switches (t.s.), and percent of DP steps before last task switch. Average deviations are lower for Specialization (blue column) than Re-Specialization (yellow column) across all 3 DP, for most of the tasks, indicating that more time is spent near 100% completion by Specializing agents. For example, the average deviation for T1 for DP I is 2.193±0.443% for Specialization and 6.18±0.267% for Re-Specialization. Note that although for DP II and DP III, T3 and T4 have slightly lower deviations during Re-Specialization, T1 and especially T2 are sufficiently worse so as to keep Specialization in the performance lead (e.g. 1.216% vs. 11.925% for T2 during DP II). Daily task switching is approximately 4 times lower for DP II (3.141% vs. 13.515%) and 10 times lower DP I (3.453% vs. 34.406%) and DP III (2.813% vs. 25.813%) for Specializing vs. Re-Specializing. The last task switch happens around 30-35% into each Specialization period, while during Re-Specialization agents never stop switching tasks (values for last t.s. in yellow column are 94-100%).

For a visual comparison, we plot average daily deviations (Fig. 3) and daily task switch averages (Fig. 4). In Fig. 3, Specialization DPs quickly reach near-zero deviations, while Re-Specialization DPs remain at higher levels. In Fig. 4, Specializing agents quickly settle on some task after a change in demands, while Re-Specializing agents con-

		SPECIALIZING		RE-SPECIALIZING	
		average	(± 95% C.I.)	average	(± 95% C.I.)
		days 0-49		**days 150-199**	
DP I	last t.s.	34.394%	(0.783%)	99.763%	(0.838%)
	daily t.s.	3.453%	(0.795%)	34.406%	(0.614%)
	T1 dev.	2.193%	(0.443%)	6.180%	(0.267%)
	T2 dev.	0.828%	(0.335%)	6.600%	(0.438%)
	T3 dev.	1.955%	(0.743%)	8.407%	(0.640%)
	T4 dev.	1.359%	(0.704%)	4.020%	(0.151%)
		days 50-99		**days 200-249**	
DP II	last t.s.	35.400%	(0.397%)	94.375%	(0.752%)
	daily t.s.	3.141%	(0.820%)	13.516%	(1.802%)
	T1 dev.	1.690%	(0.387%)	2.774%	(0.315%)
	T2 dev.	1.216%	(0.314%)	11.925%	(1.495%)
	T3 dev.	1.558%	(0.661%)	0.908%	(0.512%)
	T4 dev.	1.910%	(0.484%)	0.250%	(0.372%)
		days 100-149		**days 250-299**	
DP III	last t.s.	30.650%	(0.573%)	99.781%	(0.883%)
	daily t.s.	2.813%	(0.535%)	25.813%	(1.733%)
	T1 dev.	1.670%	(0.457%)	4.606%	(0.193%)
	T2 dev.	1.422%	(0.361%)	15.507%	(1.158%)
	T3 dev.	0.995%	(0.310%)	0.780%	(0.435%)
	T4 dev.	1.853%	(0.500%)	0.244%	(0.287%)

Table 2: Specialization vs. Re-Specialization DP Deviations
Averages are over 32 runs; in parenthesis we provide the standard error values for a 95% confidence interval (± 95% C.I.)
daily t.s. = daily task switches (out of 200 = 20 agents*10 steps);
last t.s. = how far into that DP the last task switch occurred.

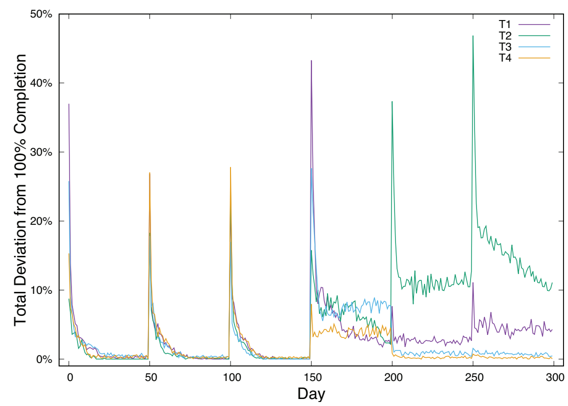

Figure 3: Average deviation from 100% task completion Specializing (day0-149) vs. Re-Specializing (day150-299)

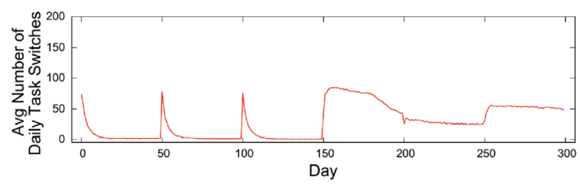

Figure 4: Average Daily Task Switches Specializing (day0-149) vs. Re-Specializing (day150-299)
Lower values are best; max possible: 200 task switches per day.

tinuously switch tasks throughout each 50-day period. These results show that adjusting to demands is significantly easier when agents are not pre-Specialized on different demands.

4 Specialization Over Time

We look at the emergent agent Specialization in an environment with sufficient resources for each agent to focus on one task and given sufficient time to settle into a stable task-assignment based on the given threshold update rates.

Emergent Specialization can be tracked over time by looking at *activations*: $\langle s, \theta \rangle$ value pairings that agents plug into their TBD action probability calculations, given the system's stimuli and agents' affinities. We choose the term "activations" to represent that these are the pairings active within the system. Fig. 5 shows the total activations present on day zero of a single representative simulation run, summed over all 10 steps of that day. We group activations into buckets of 10% increments, from 0.0 to 1.0 for both s_t and $\theta_{a,t}$. Additionally, as many of the activations involve edge values, we add two more rows dedicated to $s_t = 0$ and $s_t = 1$, as well as two columns dedicated to $\theta_{a,t} = 0$ and $\theta_{a,t} = 1$. Thus, we get a 12x12 matrix where rows represent $\theta_{a,t}$ buckets, columns represent s_t buckets, and color represents the number of activations for a given $\langle s, \theta \rangle$ pairing.

All activations are used to calculate $P_{a,t}$ values by each agent to order the tasks prior to action selection. If a task is considered but not chosen for action, the next task with largest $P_{a,t}$ is considered. Only a subset of all activations are considered, because once a task is chosen for action, tasks further down in the ordering are not considered. Similarly, only a subset of the considered activations are ultimately chosen for action. Thus, we can look at activations as three sets: those *"present"* in the system, a subset of these that get *"considered"* during action selection, and a subset of the considered activations which get *"acted on"*. Fig. 6 shows these three sets of activations mapped for DP I. Columns 1-3 represent the present, considered, and acted on activations, respectively, for Specialization (days 0-49). Columns 4-6 represent the same sets for Re-Specialization (days 150-199). Rows correspond to a subset of the fifty days in DP I, due to space limitations. Each large square is a matrix of activation buckets, just like the one in Fig. 5, containing the sum of activations of one type (present, considered, or acted on) over the 10 steps of a single day, with amounts color-coded via a heatmap. Since matrix axes are oriented identically to the probability map axes (Fig. 1), the matrices in Fig. 6 show which activations are happening during each day, while the $P_{a,t}$ values resulting from these activations are shown in the same area on the probability map in Fig. 1, although the 20x20 map has higher $\langle s, \theta \rangle$ pair granularity.

Before Specialization agents have uniformly random initial $\theta_{a,t}$. Every day has 10 steps so, on every step s_t increases by 10% of the tasks' daily demands, growing from 0.0 to 1.0 over the course of a day if task t is left untouched by the agents. Given no demands, activations are uniformly distributed along the *top-edge* of the matrix. As soon as demands are introduced, deficits increase from $s_t = 0.0$ to $s_t = 0.1$ (Fig. 6, day 0, col.1-3).

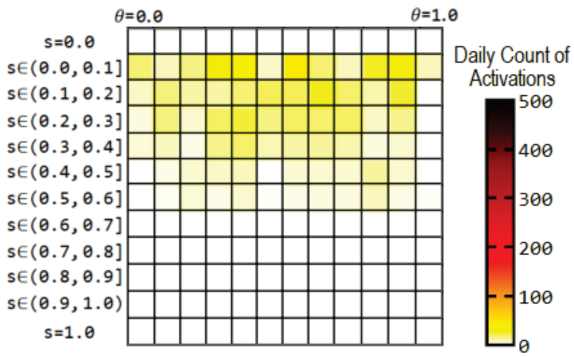

Figure 5: Activations $\langle s, \theta \rangle$ present on day 0 of a single run (omitted $\theta_{a,t}$ bucket labels mirror those of s_t buckets)
Total daily activations: 800 = 1 day * 10 steps * 20 agents * 4 tasks

During Specialization, newly increased s_t cause activations to move down from the *top-edge* ($P_{a,t} = 0.0$) and agents begin to act. Given TBD probabilities (Fig. 1), $\theta_{a,t}$ randomly initialized to lower values lead to more actions, causing further decrease of these $\theta_{a,t}$ ("acted on" pairs quickly shift to the *left-edge* (col.3, Fig. 6). By day 8, agents settle into Specializations, i.e. all work by each agent over each day is on a single task. Thresholds, however, do not yet show full Specialization: activations are spread across multiple $\theta_{a,t}$ columns (day 8, col.1-3, Fig. 6). Four rows of stimuli quickly form, corresponding to the four tasks. Some tasks stay at their initial low s_t (higher rows), while others shift downward over time, taking longer to stop incurring deficit. Once all four tasks get sufficient agents, all deficits s_t stabilize and activations settle into a final four rows (colored rows on days 8-49, Fig. 6). Note that the final s_t depend only on which agents settled on what task quicker and do not reflect demand values, or the quality of the final task distribution.

After Specializing for some time, since agents are remaining on their selected tasks across many days, $\theta_{a,t}$ move toward 0.0 for the preferred task of each agent, and toward 1.0 for every other task. In Fig. 6, the preferred tasks of all agents reach $\theta_{a,t}$=0.0 by day 11. Since building aversion is (number of tasks - 1) times slower than building affinity, unpreferred task thresholds take until day 33 to reach $\theta_{a,t}$=1.0. This maximal Specialization is seen on days 33-49 in that all activations fall within lowest and highest possible θ (first and last columns). There are three times as many activations for $\theta_{a,t}$=1 (right-most column) than $\theta_{a,t}$=0 (left-most column), as agents are Specialized on one task and Specialized against three other tasks (yellow/orange buckets represent 50-100 activations, while red represent 150-300). Despite higher activation counts for the right column and that activations near ⌟ have a 50% chance to be acted on when "considered", none of these $\theta_{a,t} = 1$ activations ever become "considered", and thus none can be "acted on". Since agents consider tasks starting with highest $P_{a,t}$ value, *left-edge* tasks with $P_{a,t}$=1 prevent others from being considered, regardless of s_t, unless it drops to $s_t = 0.0$ (see left edge of Fig. 1). Deficits, however, are now stabilized, so s_t remains unchanged.

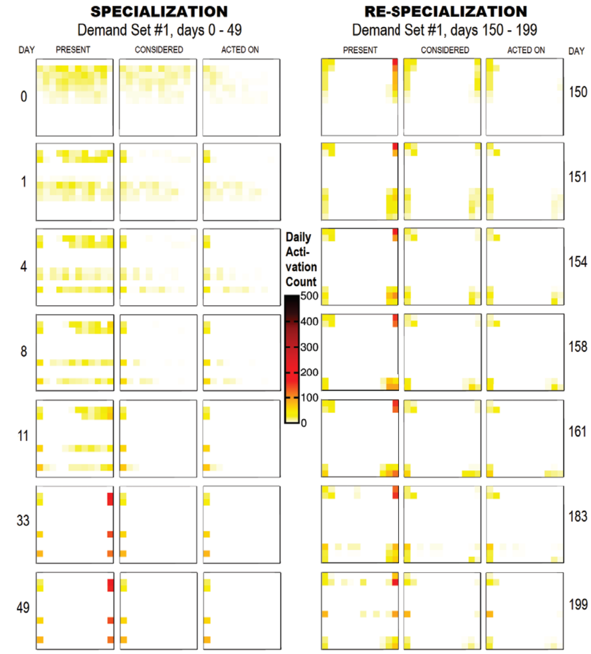

Figure 6: TBD $\langle s, \theta \rangle$ pairings over time during Specialization and Re-Specialization

Each large square is a day containing bucketed activation counts for $\langle s, \theta \rangle$ pairings. For axes and bucket definitions see Fig. 5.

5 Re-Specialization Over Time

The mechanism behind Re-Specialization is identical to that behind Specialization, but the startinfg points are different. In Re-Specialization, instead of uniformly random $\theta_{a,t}$, responses to new demands come from agents with $\theta_{a,t} = 0.0$ for one task and $\theta_{a,t}$=1.0 for the others.

Before Re-Specialization a population of Specialized agents is faced with changed demands. With the introduction of new demands, task deficits are reset, and are incremented by (1/ daily steps)% on every step (in our setup s_t becomes 0.1 on the first step of new demands). We see this in the activations spreading downward from the top row (day 150, Fig. 6). Activations are more frequent near the top row given that all s_t values are lowered, but only some of the s_t immediately begin to increase (move downward) again, indicating insufficient workers. As with Specialization, we do see more activations on the *right-edge*, as we again have a higher total number of agents with an aversion to a task ($\theta_{a,t} = 1.0$)than those with an affinity for it ($\theta_{a,t} = 0.0$).

During Re-Specialization, the tasks with insufficient workers soon reach max $s_t = 1.0$ (day 158). At this point we again see only 4 rows of activations, one per task, indicating that s_t values have stopped changing throughout the day. At $s_t = 1.0$, activations for tasks to which the agents are averse move into the ⌟, causing the most probable action of the entire *right-edge* ($P_{a,t} = 0.5$) (see Fig. 1). Since activations in the ⌜ are only 50% likely to cause action, the next highest $P_{a,t}$ task in ⌟ will have a 50%x50%=25% chance to lead to action on the newly needed tasks with high aversion (see "acted on" activations near ⌟ on days 151-199, Fig. 6).

After Re-Specializing for some time only minor changes are observed. Overall, we see that Re-Specialization takes considerably longer, although the specific pre-existing and new demand needs do play a role in the exact resulting behavior. Since there is a relatively small chance to take on the tasks with ⌟ activations, threshold adjustments are much slower than during Specialization (we see activations shifting across the rows over the period 183-199). In the end, the 50 days are not enough for agents to adjust to this demand change. In fact, we see on days 183-199 that one of the deficits begins to reduce (one of the rows begins moving upward), indicating that extra work is being done. In Fig. 2 we can see that toward the end of period 150-200, the green line of T2 is in fact above 100% completion, indicating excess work on that task, to the detriment of the others.

6 Discussion: Low vs. High Threshold Tasks

We see that Re-Specializing causes agents to arrive at sub-optimal task allocation, marked by decreased task completion and increased task switching. The ability to Re-Specialize hinges on the ability to stop doing what agents have Specialized on doing thus far, a behavior that is problematically opposed to Specialization tendencies. In this section, we generalize the behavior observed through $\langle s, \theta \rangle$ activations and consider possible Re-Specialization improvements in systems with dynamic task allocation needs.

Specialization and Re-Specialization behavioral differences stem from differences in initial agents' task thresholds. When Specializing, agents begin with a random set of task thresholds, corresponding to having random task affinities and aversions, and thus resulting in random action selection. When Re-Specializing, agents' thresholds begin at values near/at 0.0 and1.0, indicating maximal and minimal affinity (i.e. minimal and maximal aversion), respectively. Thus, in order to Re-Specialize, agents not only have to change their existing thresholds, but also to actively overcome how pre-existing thresholds cause them to act.

The issue with threshold-reinforcement models is that they focus on allowing agents to Specialize, which by its very nature hinders Re-Specialization. As agents develop affinities toward certain tasks, they develop aversions for others. This favoritism later slows or prevents agents from adjusting to changes requiring a new task assignment. In TBD, in particular, we see that the old preferred tasks cause high probability activations on the *left-edge* opposing low probability activations for the newly needed tasks all along the *right-edge*, putting new demands at a disadvantage. For tasks with newly increased demands, there are insufficient agents acting on them ($\theta_{a,t}$=1.0), causing s_t to grow, indicating that the previous task-assignment is now unsuitable. For tasks with newly decreased demands, continued actions from previously Specialized agents decrease s_t to 0.0. While every other s_t along the *left-edge* causes certain action, probability to act at $s_t = 0$ is 50%, allowing other tasks to be considered half of the time. These tasks are the ones agents have grown averse to, with activations of $\theta_{a,t}$=1, along the *right-edge* of Fig. 1. Probabilities along this edge are low: $P_{a,t} \in [0.0, 0.5]$), so even when these tasks are considered, defaulting to idling is likely, further preventing much needed actions. Additionally, as soon as the preferred tasks are not acted on for a step, corresponding s_t increases slightly above zero, again causing certain action. Thus, agents cannot intelligently prioritize among the high and low deficit tasks.

One way to improve Re-Specialization is to randomly reinitialize agents $\theta_{a,t}$ when existing task allocation should be re-assessed. However, to make such resets possible, agents must know when to give up their existing Specialization. One possible heuristic is to assume that drastic changes in s_t should trigger a reset in $\theta_{a,t}$. An added benefit is that if s_t values vary throughout the system, threshold resets take place only where warranted. Additionally, the drastic change from $P_{a,t} = 0.5$ for $s_t = 0.0$ to $P_{a,t} = 1.0$ for any infinitesimally larger s_t goes against the intuition that changes in $P_{a,t}$ should be commensurate with changes in s_t. Consequently, a less drastic shift in probabilities near ⌜ may be beneficial.

Alternatively, a new model that places equal value on Specialization and Re-Specialization may be needed. If agents have no explicit signal to Re-Specialize, the model must allow for balancing personal preferences with system needs. In TBD, when $\theta = 0$, any $s_t > 0$ results in $P_{a,t} = 1.0$, causing agents to overvalue their preferences to the detriment of performance. We hypothesize that a model that is commensurately sensitive to both stimulus and threshold variations would be more responsive to Re-Specialization needs.

7 Conclusions

In this work we show that Re-Specialization can be problematic even when agents can easily Specialize to initial demands. The same forces that drive agents to Specialize, can get in the way of Re-Specialization, causing slow adjustment as probabilities along for preferred tasks disregard task deficit levels (except when $s_t = 0.0$). As demands are likely to fluctuate throughout in real-world domains, it is of interest to further investigate the conditions causing agents to fail to Re-Specialize, as well as to devise potential solutions.

References

Campos, M.; Bonabeau, E.; Theraulaz, G.; and Deneubourg, J.-L. 2000. Dynamic scheduling and division of labor in social insects. *Adaptive Behavior* 8(2):83–95.

Cicirello, V. A., and Smith, S. F. 2003. *Distributed Coordination of Resources via Wasp-Like Agents*. Springer. 71–80.

Kittithreerapronchai, O., and Anderson, C. 2003. Do ants paint trucks better than chickens? Markets versus response thresholds for distributed dynamic scheduling. In *Evolutionary Computation, 2003. CEC '03. The 2003 Congress on*, volume 2, 1431–1439.

Price, R., and Tiňo, P. 2004. *Evaluation of Adaptive Nature Inspired Task Allocation Against Alternate Decentralised Multiagent Strategies*. Springer. 982–990.

Theraulaz, G.; Bonabeau, E.; and Deneubourg, J.-L. 1998. Response threshold reinforcement and division of labour in insect societies. *Proc. of the Royal Society of London B* 265:327–332.

Wu, A. S., and Kazakova, V. A. 2017. Effects of task consideration order on decentralized task allocation using time-variant response thresholds. In *Proc. FLAIRS-30 Conference*, 466–471.