Comparison of Desynchronization Methods for a Decentralized Swarm on a Logistical Resupply Problem

Extended Abstract

Joseph P. Giordano[1], Annie S. Wu[1], Arjun Pherwani[1], and H. David Mathias[2]
[1]University of Central Florida, Orlando, FL
[2]University of Wisconsin – La Crosse, La Crosse, WI
{joseph.giordano,arjunp}@knights.ucf.edu,aswu@cs.ucf.edu,dmathias@uwlax.edu

ACM Reference Format:
Joseph P. Giordano[1], Annie S. Wu[1], Arjun Pherwani[1], and H. David Mathias[2]. 2021. Comparison of Desynchronization Methods for a Decentralized Swarm on a Logistical Resupply Problem: Extended Abstract. In *Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), Online, May 3–7, 2021*, IFAAMAS, 2 pages.

1 INTRODUCTION

In this work, we investigate the impact of two approaches to agent desynchronization on task allocation in decentralized swarms: variation in response threshold and variation in response duration. We focus on swarms consisting of simple threshold-based stimulus-response agents and examine how they respond to dynamically changing task demands in a logistics re-supply problem.

Ideally, a swarm will be able to divide the labor of its workers appropriately such that task demands are satisfied in a timely manner. In general, agent resources should be conserved to minimize waste resulting from overdelivery. Because task switching may incur physical or time costs, stable distributions of agents are desirable. We predict that agent performance will be more consistent when agent desynchronization is highest and that the need for sufficient desynchronization is more critical for more difficult problems. To provide grounds for this investigation, we examine primary effects for each method of desynchronization in combination with different types of logistic schedules on agent performance.

2 IMPLEMENTATION

Problem: Our testbed problem is based on the logistics of material resupply. We consider a scenario with materials $m_i,\ i \in \{1, \ldots, M\}$ defining a task s_i. Every agent can respond to global stimuli and perform any of the M tasks. Task demands are specified in two schedules. The original schedule, S_O, is our simulation input consisting of static demands for each material in each timestep. The working schedule, S_W, represents updated task demands based on quantities of materials actually delivered during a simulation. Schedules consist of *sessions*; contiguous sequences of fixed non-zero demand for a given material. In S_O, sessions have defined intervals, whereas S_W contains dynamic start and end times to reflect deliveries. Session end times are triggered when the total requested material amount has been delivered. Subsequently, start times may be delayed, since sessions for the same material must remain disjoint in time. Additionally, start times cannot be decreased.

Altogether, we include 3 benchmarks for measuring swarm performance: Timesteps to completion, sum of agent over-delivery, and average number of task switches per agent - all for a given schedule. The domain goals are to minimize each respective benchmark.

System Description: A response threshold defines a minimum environmental stimulus to trigger a response from an agent. Task selection involves the agent evaluation of current task demands and selection of tasks to address them, or otherwise, to remain idle. An agent uniformly selects, at random, from candidate tasks which exceed its threshold. *Response threshold*, τ_i, for a given task d_i is a value in $[0..1]$. System parameter `Scaling_factor` allows response threshold values to be applied to problems with significantly different demands or in different domains. Let $\sigma_i(t)$ represent the stimulus (demand) for material m_i at time t. Then, an agent may activate for task d_i in timestep t if $\sigma_i(t) \geq \tau_i \cdot$`Scaling_factor`. System parameter `Thresh_init` defines the method by which threshold values are determined. If `Thresh_init` $\in (0..1]$ then $\forall\ i \in [1, \ldots, M]\ \tau_i =$ `Thresh_init`. In other words, the threshold values are a single constant for all tasks. `Thresh_init` > 1, dictates heterogeneous response thresholds generated according to one of several probability distributions. The `Thresh_init` value determines the probability distribution used. In this work, the distributions we explore are uniform, Gaussian, and Poisson. Response threshold values are static, remaining unchanged throughout a simulation. *Response duration* is a measure of how long an agent remains on one task before switching to another. In our model, response duration is probabilistic. System parameter `prob_check` $\in (0..1]$ defines the probability that in a given timestep an agent will evaluate stimuli and select a task. Thus, lower `prob_check` values indicate higher time on task.

3 EXPERIMENTS

Design: Our study is divided into response threshold and duration segments, exploring factors related to each respective form of inter-agent variation. For response thresholds, we examine 7 probability distributions for generating values: constant, uniform, Gaussian ($\mu = 0.50, 0.25$), and Poisson ($\lambda = 3, 5, 7$). For response duration, we test homogeneous `prob_check` values from [0.1,1.0], in increments of 0.1. Every experiment contains 10 runs with distinct input schedules. We define `Stress_index` as a measure of agent resources relative to task demands. It is inversely proportional to the number of agents (`Popsize`) because a smaller swarm will be under higher stress than a larger one given identical task demands. This metric is directly proportional to the `Scaling_factor` because higher values effectively raise thresholds, thus decreasing the number of agents

Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), U. Endriss, A. Nowé, F. Dignum, A. Lomuscio (eds.), May 3–7, 2021, Online.

Table 1: Eight experiments, varying stress index values along two axes: Fixed population with decreasing scaling factor and fixed scaling factor with increasing population.

Expt	Popsize	Scaling_factor	Stress_index
A	100	100	1.00
B	100	50	0.50
C	100	25	0.25
D	50	100	2.00
E	150	100	0.67
F	200	100	0.50
G	400	100	0.25

Table 2: Two-way ANOVA PR(>F) for timesteps to completion averages on response threshold experiments.

Expt	Stress index	Main effect: Init_thresh	Main effect: Sched_type	Interaction effect
A	1.00	6.01E-108	7.93E-14	2.20E-08
B	0.50	4.46E-89	1.03E-26	5.03E-03
C	0.25	5.56E-60	8.06E-31	3.55E-01

that activate for a task. We run eight experiments, partitioned into two sections for our analysis, examining decreasing stress index values along two axes: Fixed population size with decreasing scaling factor and fixed scaling factor with increasing population size. In both instances, progressively lowering `Stress_index`.

Table 1 outlines the parameters for each experiment, which are performed for the response threshold distributions and response duration values above. The Two-way ANOVA encompasses the independent categorical variables (threshold, schedule type), with effects of *stress index* on interaction and within group variance to analyze forms of desynchronization and schedule design.

Broad Analysis of Swarm Behavior: System performance metrics are used for our Two-way repeated ANOVA. Here, we show results for timesteps and task switches. The experimental design is motivated by an assessment of the main effects given by both categorical variables, and their interaction. Experimental p-values are consolidated into a single table, for each metric. Low p-values ($p \leq 0.05$) signify rejection of null hypotheses, and thus, a statistically significant effect for the given parameter.

Table 2 gives timesteps results for response threshold experiments. We see that interaction between threshold and schedule type is weakened, as stress decreases via decreasing scaling factor. Table 3 shows the same results for task switches. With decreasing scaling values, a similar trend occurs for interaction effects, only this time, we stop rejecting once a stress index of 0.50 is reached.

We can see, generally, that placing higher demands on our swarm leads to greater interaction between parameters. When the threshold scaling factor is larger, the load on our agents is more intense, since they can't detect incoming task demands as quickly. One would expect to see a similar effect take hold when population size is decreased, as, in that instance, there are more agents to deal with task demands. Evidently, this is not the case.

Table 3: Two-way ANOVA PR(>F) for agent task-switching averages on response threshold experiments.

Expt	Stress index	Main effect: Init_thresh	Main effect: Sched_type	Interaction effect
A	1.00	1.05E-58	3.67E-07	1.64E-18
B	0.50	5.60E-16	1.00E+00	9.99E-01
C	0.25	0.789445	0.589522	0.942792

Table 4: Two-way ANOVA PR(>F) for timesteps elapsed averages on response duration experiments.

Expt	Stress index	Main effect: Prob_check	Main effect: Sched_type	Interaction effect
D	2.00	8.66E-257	1.07E-121	1.08E-105
A	1.00	1.23E-168	6.84E-08	1.92E-13
E	0.67	3.15E-163	1.86E-05	4.61E-02
F	0.50	8.35E-144	1.03E-04	4.09E-01
G	0.25	2.89E-148	1.44E-15	7.73E-01

Lower `Scaling_factor` corresponds to homogeneity in performance, spanning multiple factors. In the case of response threshold, Tables 2 and 3 show null hypothesis rejection ($p \leq \alpha = 0.05$) more often when our scaling values are higher. This is due to a wider spread of values across our range of possible response thresholds. When this is the case, changing threshold quantities or distributions clearly has a greater burden or weighted impact on performance - especially in the case of timesteps to completion. Experiment A acts as a baseline for other experimental outcomes, since it contains ideal task demands for a population size of 100.

Table 4 shows Two-way ANOVA timesteps results for response duration, with a decreasing `Stress_index` by way of increasing `Popsize`. Observe that higher population sizes play a role in diminishing the effects of changing response duration. With an overabundance of agents, task demands are almost always satisfied. In this case, we base our analysis on changes to task switching due to increasing population. In comparison with response threshold results, it is clear that population size has a greater impact on duration of response, independent of threshold distribution.

4 DISCUSSION

This simulation provides an extensible model for complex task demands for decentralized multi-agent systems. We find that both mechanisms for desynchronization impact swarm behavior, but have different effects. Variable response duration diversifies the frequency with which agents re-evaluate their actions and affects how quickly agents respond to changing task demands, and variable response thresholds allow agents to respond differently to the same material demands, effectively desynchronizing task acceptance for any given material. From our analysis, we broadly observe that interaction between threshold distribution and schedule variant diminishes under low-stress environments.

Acknowledgements: This work was supported by the National Science Foundation under Grant No. IIS1816777.