Online Re-Planning and Adaptive Parameter Update for Multi-Agent Path Finding with Stochastic Travel Times

Extended Abstract

Atsuyoshi Kita
Panasonic Holdings Corporation
Osaka, Japan
kita.atsuyoshi@jp.panasonic.com

Nobuhiro Suenari
Panasonic Holdings Corporation
Osaka, Japan
suenari.nobuhiro@jp.panasonic.com

Masashi Okada
Panasonic Holdings Corporation
Osaka, Japan
okada.masashi001@jp.panasonic.com

Tadahiro Taniguchi
Ritsumeikan University and Panasonic Holdings Corporation
Shiga, Japan
taniguchi@ci.ritsumei.ac.jp

ABSTRACT

This study explores the problem of Multi-Agent Path Finding with continuous and stochastic travel times whose probability distribution is unknown. It is often the case with real-world applications (e.g., automated delivery services in office buildings) that the time required for the robots to traverse a corridor takes a continuous value and is randomly distributed because pedestrians and a wide variety of robots coexist, and the prior knowledge of the probability distribution of the travel time is limited. We propose 1) online re-planning to update the action plan of robots while it is executed and 2) parameter update to estimate the probability distribution of travel time using Bayesian inference as the delay is observed. Through simulations, we empirically compare the performance of our method to those of existing methods.

KEYWORDS

Autonomous Robot Management System; Multi-Agent Path Finding; Stochastic Travel Time; Bayesian Inference

ACM Reference Format:

Atsuyoshi Kita, Nobuhiro Suenari, Masashi Okada, and Tadahiro Taniguchi. 2023. Online Re-Planning and Adaptive Parameter Update for Multi-Agent Path Finding with Stochastic Travel Times: Extended Abstract. In *Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), London, United Kingdom, May 29 – June 2, 2023*, IFAAMAS, 3 pages.

1 INTRODUCTION

This study focuses on delivery services in buildings. In a corridor with many people passing by, robots have to slow down or temporarily stop and take longer time than initially expected. Consequently, the time required to traverse a corridor is stochastically distributed (aleatoric uncertainty). In addition, it is usual that the information on the probability distribution of the travel time of each corridor is unavailable or limited (epistemic uncertainty). Our focus is on making a collision-free path for automated delivery robots in the corridor of buildings under both aleatoric and epistemic uncertainty.

Although Multi-Agent Path Finding (MAPF) has been studied and applied to warehouse [5] and airport operations [3], aleatoric and epistemic uncertainty of the traffic condition in buildings make it difficult to automate the robot operation system fully. Recent studies on the p-Robust CBS [1] and STT-CBS [4] successfully addressed the aleatoric uncertainty by modeling the delay of travel time using probability distributions. The STT-CBS models the delay using a gamma distribution and creates the travel paths of agents whose conflict probability is less than a certain value. However, in practice, the parameters of the distribution are not available.

Such situations motivate the need to estimate the parameters from observations instead of using predefined distributions and to re-calculate the plan using updated parameters. The solution method presented in this paper (GSTT-CBS with PU and OR) is an extension of STT-CBS [4] by 1) adding a greedy heuristic [1]; and by 2) parameter update (PU) with Bayesian inference; and by 3) online re-planning (OR) of the travel paths of robots.

2 PROPOSED METHODS

We consider a connected bidirectional graph $G = (V, E)$ consisting of a set E of edges and a set V of vertices. The default travel time $w(e)$ of edge e equals the edge length. We have N agents, and agent $i \in \{1 \ldots N\}$ moves from a start vertex $start_i \in V$ to a goal vertex $goal_i \in V$. Agent i is assigned a path $p_i = \{c_1, \ldots, c_{n_i}\}$ consisting of n_i commands $c_j = (u_j, v_j, d_j)$ for $j \in \{1 \ldots n_i\}$. A command is either a *move* command such that

$$u_j \neq v_j, (u_j, v_j) \in E, d_j = w(u_j, v_j)$$

or a *wait* command such that

$$u_j = v_j, u_j \in V, d_j \in R^+.$$

A path for agent i is valid when $u_1 = start_i$, $v_{n_i} = goal_i$, and $v_{j-1} = u_j$ for $j \in 2 \ldots n_i$.

Let us model a stochastic travel time using gamma distributions. Each edge $e \in E$ has a shape parameter $a(e)$ and a scale parameter $b(e)$. The actual travel time of the edge is $w(e) + x$ with $x \sim \mathrm{Gamma}(a(e), b(e))$.

Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), A. Ricci, W. Yeoh, N. Agmon, B. An (eds.), May 29 – June 2, 2023, London, United Kingdom.

Table 1: Average number of conflicts during simulation of 100 tasks. Underlines indicate the best results.

	GSTT-CBS				STT-CBS
	with PU, OR (ours)	with PU	with OR		[4]
Map1	1.81	1.97	2.68	2.92	3.56
Map2	0.87	0.96	2.05	1.97	2.38
Map3	0.05	0.05	0.12	0.14	0.31
Map4	0.15	0.22	0.91	0.93	1.32

2.1 Parameter Update

We assume that the true parameters of delay distribution, $a(e)$ and $b(e)$ for $e \in E$, are unknown and we only have prior knowledge, $a_{prior}(e)$ and $b_{prior}(e)$ for $e \in E$. We calculate the maximum a posteriori (MAP) estimator of a and b from m observations of delay $\mathbf{x} = (x_1, \dots x_m)$ obtained as the agents traverse the edge e using Bayesian inference.

2.2 Online Re-planning

The initial plan created using the estimated delay distribution can still cause conflicts during the execution of the plan because of estimation errors. Therefore, to reduce conflicts, we regularly update all agent's plans during the execution phase. We consider a centralized robot operating system. The central controller has information about which command each agent executes, and it can update the plan of all agents at any time; however, the currently running command must remain unchanged, and other commands in the updated path must be consistent with it.

3 EXPERIMENTS

We conduct an experiment to evaluate our approach using random non-grid maps. Random 100 tasks for each map are solved and simulated sequentially, i.e., the plan for a task is created using the probability distribution updated with the observation in the previous tasks. In short, the planning is based on a slightly more accurate model every time. The result in Table 1 shows that our method results in fewer conflicts.

We conduct an additional experiment to analyze the effect of the parameter update using a random map with 50 nodes and 1000 tasks consisting of 10 agents. We compare three cases: (1) the GSTT-CBS with the parameter update; (2) the GSTT-CBS without the parameter update; and (3) the GSTT-CBS with no error, which is a hypothetical setting where the true parameters are obtained from the beginning. After each task, the root-mean-square error (RMSE) of the estimated parameter a_{map} and b_{map}, and the average number of conflicts are calculated and presented in Figure 1. The error rate $E_a = \frac{|a-a_{map}|}{|a-a_{prior}|}$ and $E_b = \frac{|b-b_{map}|}{|b-b_{prior}|}$ of each edge is computed and shown in Figure 2.

The performance was improved by the parameter update reaching their minimum values after about 200 tasks, but the RMSE of parameters does not decrease to 0 even after 1000 tasks. This is because the error of frequently used edges quickly decreases, while some unimportant edges remain inaccurate, even after 1000 iterations (as shown in blue in Figure 2).

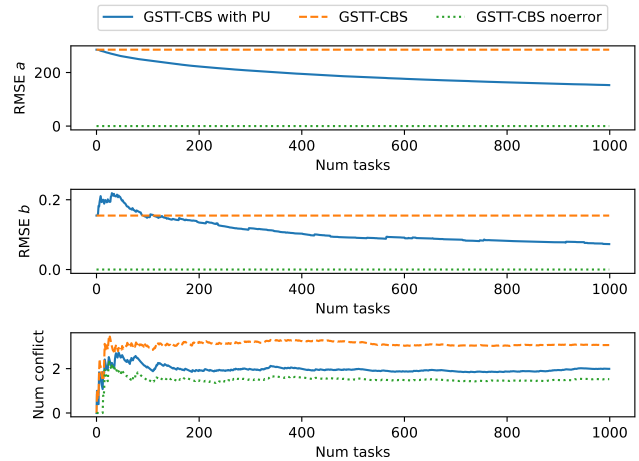

Figure 1: First and second rows: RMSE of the estimated a and b values over all edges after each task execution. Third row: the number of conflicts averaged until each execution of tasks.

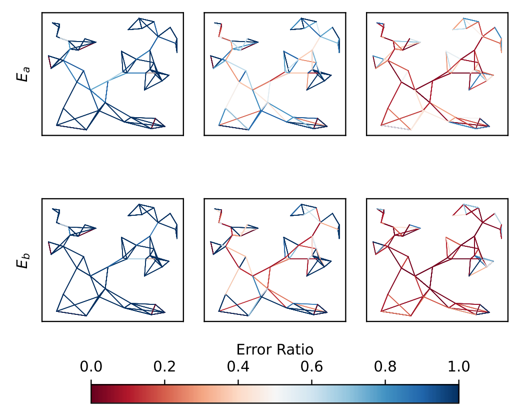

Figure 2: Error ratio of each edge plotted on the map, after 10, 100, and 1000 tasks (left to right).

4 CONCLUSION

In this work, we addressed the MAPF problem with stochastic travel times on non-unit time graphs, where the travel time of agents is stochastically distributed, and the true distribution is unknown and has to be estimated from the data obtained during the plan execution. The experiment result showed that our proposed method could decrease the number of conflicts after 100 to 200 trials by learning the true distribution of frequently used edges. However, there is still some room for improvement compared to the ideal planning (the gap between blue and green lines in the third row of Figure 1). Path planning is affected by edges that have not been passed frequently, and exploration for obtaining information on the delay of these edges is necessary.

Complete details on problem setup, relevant related works, proposed methods, and experiment results are presented in [2].

REFERENCES

[1] Dor Atzmon, Roni Stern, Ariel Felner, Nathan R. Sturtevant, and Sven Koenig. 2020. Probabilistic Robust Multi-Agent Path Finding. *Proceedings of the International Conference on Automated Planning and Scheduling* 30, 1 (Jun. 2020), 29–37. https://ojs.aaai.org/index.php/ICAPS/article/view/6642

[2] Atsuyoshi Kita, Nobuhiro Suenari, Masashi Okada, and Tadahiro Taniguchi. 2023. Online Re-Planning and Adaptive Parameter Update for Multi-Agent Path Finding with Stochastic Travel Times. *arXiv preprint arXiv:2302.01489* (2023).

[3] Robert Morris, Corina S Pasareanu, Kasper Luckow, Waqar Malik, Hang Ma, TK Satish Kumar, and Sven Koenig. 2016. Planning, scheduling and monitoring for airport surface operations. In *Workshops at the Thirtieth AAAI Conference on Artificial Intelligence*.

[4] Oriana Peltzer, Kyle Brown, Mac Schwager, Mykel J Kochenderfer, and Martin Sehr. 2020. Stt-cbs: A conflict-based search algorithm for multi-agent path finding with stochastic travel times. *arXiv preprint arXiv:2004.08025* (2020).

[5] Peter R Wurman, Raffaello D'Andrea, and Mick Mountz. 2008. Coordinating hundreds of cooperative, autonomous vehicles in warehouses. *AI magazine* 29, 1 (2008), 9–9.