Independent Scheduling System in Online Classrooms with Simple Multi-agent Temporal Networks

R Perwiratama

Program Studi Sistem Informasi, Universitas Atma Jaya Yogyakarta

E-mail: rangga.perwirama@uajy.ac.id

Abstract. Because of the growing popularity of online classes, flexible scheduling solutions that accommodate a wide range of user schedules and preferences are required. When schedule flexibility is one reason a person chooses online classes, it becomes clear that good scheduling algorithms are a basic requirement so that users get the flexibility they seek. The goal of this research is to create and develop a scheduling algorithm based on a multi-agent temporal network to solve the scheduling restrictions of teachers and students in both classroom and distant situations. This study uses a multi-agent temporal network method to create algorithms that offer independent scheduling solutions for teachers and students. These algorithms consider a variety of restrictions, providing successful scheduling within and outside of typical classroom situations. The idea that was put forward demonstrates effective scheduling outcomes, allowing students as well as teachers freedom. The multi-agent strategy effectively controls numerous restrictions, providing customizable scheduling solutions for a wide range of user requirements. Teachers and students can independently form scheduling solutions with algorithms that will solve all internal and external constraints of teachers and students.

Keywords: independent scheduling; online class; simple multi-temporal network

Abstrak. *Karena semakin populernya kelas daring, diperlukan solusi penjadwalan fleksibel yang mengakomodasi berbagai jadwal dan preferensi pengguna. Ketika fleksibilitas jadwal menjadi salah satu alasan seseorang memilih kelas daring, maka menjadi jelas bahwa algoritma penjadwalan yang baik merupakan kebutuhan dasar agar pengguna mendapatkan fleksibilitas yang mereka cari. Tujuan dari penelitian ini adalah untuk membuat dan mengembangkan algoritma penjadwalan berdasarkan jaringan temporal multi-agen untuk mengatasi batasan penjadwalan guru dan siswa baik dalam situasi kelas maupun jarak jauh. Penelitian ini menggunakan metode jaringan temporal multi-agen untuk membuat algoritma yang menawarkan solusi penjadwalan independen untuk guru dan siswa. Algoritme ini mempertimbangkan berbagai batasan, sehingga menghasilkan penjadwalan yang berhasil di dalam dan di luar situasi kelas pada umumnya. Ide yang dikemukakan menunjukkan hasil penjadwalan yang efektif, memberikan kebebasan bagi siswa dan guru. Strategi multi-agen secara efektif mengontrol berbagai batasan, memberikan solusi penjadwalan yang dapat disesuaikan untuk berbagai kebutuhan pengguna. Guru dan siswa dapat secara mandiri membentuk solusi penjadwalan dengan algoritma yang akan menyelesaikan semua kendala internal dan eksternal guru dan siswa.*

Kata Kunci: *penjadwalan mandiri; kelas daring; jaringan multitemporal sederhana*

1. Introduction

Over the past few years, even before the coronavirus pandemic forced a temporary shift to emergency remote learning, there was a boom in online enrollment pattern and there has been a huge growth in online class programs [1]. The Babson Survey Research Group found that in 2016, online class registrations at educational institutions that provided online class options increased for fourteen consecutive years. So that between 2012 and 2016, the number of students studying on campus (traditional classes) fell by more than one million students [2]. Learning that is supported by technology through ICT (or e-learning) is becoming increasingly important for academic and business education. In addition, e-learning has the potential to be one of the most important developments in the field of information and communication technology [3]. Previous research shows that the most common reason for students to choose the online format is flexibility or ease of scheduling. For some students, flexibility is a big problem, due to work schedules, home activities, or distance from the campus. For others, although it is possible to take traditional classes, it is more convenient to take online classes [4]. Online courses frequently need a greater dedication to two-way communication, clear expectations, and student autonomy [5]. As a result of these circumstances, both teachers and students who switch to online classrooms typically feel alienated since online programs need more discipline, time management, and collaboration, making them more challenging. In online classes, planning assignments and independent activities can be a complicated and time-consuming process. To overcome this challenge, a system with independent scheduling that consider Simple Multi-Agent Temporal Network as method of solution will be proposed and developed. This system uses a multi-agent approach, where each agent represents a task or activity in the online class.

This system uses a Simple Multi-agent Temporal Network to model temporal constraints and dependencies between tasks. By encoding problem and scheduling constraints into the underlying Simple Temporal Network, the system ensures that the start and end times of different tasks are within the allowable range [6].

This enables efficient and flexible scheduling of independent assignments in an online classroom environment, providing students with a smooth and organized learning experience. Following resources will be used as needed and set up a complex production network, and a dynamic scheduling algorithm based on multi-layer network statistics was used to complete the scheduling of multiple independent resources and tasks [7]. Jiang's work on complex production networks and dynamic scheduling algorithms based on multi-layer network metrics can be a valuable reference for developing similar systems for independent scheduling in online classrooms with Simple Multi-agent Temporal Networks to limit the range of allowed values for the start and end times of different methods in a given agent schedule, all issues and schedules.
To ensure flexibility and exceed the problems that have been raised, a good scheduling algorithm is needed. So that the human element involved in the environment can interact optimally. Humans interact with each other both directly and through the system will always resemble a dilemma. Humans will calculate long-term social benefits and short-term personal benefits for themselves [8]. Therefore, the results of behavior in the dilemma can be very dependent on how successful the participants are in calculating risk, related to uncertainty in future rewards and anticipating the opponent's choices. Humans behave differently depending on the level of risk they face, which requires this study to understand how humans perceive and predict risk, and the effect of their actions on their opponents and partners [9]. With that the best solution is to create a system to design scheduling that understands and respects these external constraints with intelligent algorithms, so that the system can schedule independently.

Regarding algorithms, many algorithms that can be used are Bayesian models. Where in previous studies there has been an approach wherein, Bayesian methods are used to represent the flow of information on a schedule and how information on that schedule can vary according to place and perception of humans and robots. Based on this model, a computational model can be determined to understand where and how the human process informs the schedule to a robot [10]. Another algorithm that can be used is an evolutionary algorithm that is proven to be able to maintain several solutions for each iteration and it can work well resolving problems with noise [11].

However, many scheduling problems can be represented using simple temporal problems (STP), where events are represented as variables which domain is the best implementation time. The best context in this case can be represented as a possibility. Besides that, STP must also have limitations, where the boundary limits the time between events which are represented as the boundary of the difference in values between variables [12]. Simple Temporal Problem (STP) arguably is the most famous quantitative temporal representation framework in the world of artificial intelligence. STP considers time points as variables and represents time information by a series of unary or binomial boundaries, each defining an interval on the real number line. Since its introduction in 1991, STP has become an important part of problem planning and scheduling. While STP was initially introduced for a single scheduling agent and was completed with a centralized algorithm, many STP applications in the real world involved several agents who interact with each other [13]. To solve the problem of representation of schedules in STP, all STP cases can be represented naturally in graphical form called simple temporal network (STN). STN offers a way to efficiently maintain the temporary limitations of an STP. A simple temporal network (STN) is a weighted directed graph in which nodes represent variable points in time, usually associated with the start or end of an activity, and edges represent lower and upper bounds on the distance between two connected points in time [14]. Each STN is associated with a distance graph, originating from the upper and lower limits, where the constraints between a pair of X and Y time points are represented as two different sides [15].

STN has played a central role in many planning systems and scheduling systems, for example in coordinating military assistance efforts on natural disasters, Mars explorer missions, health care operations, and manufacturing tasks. It can be concluded that most are applied to solve scheduling problems between humans and tools (machines / robots) [16]. In the domain of planning and scheduling between humans who in fact have the awareness to calculate their interactions with each other as mentioned above, a single STN is not effective to use. Humans must coordinate with other humans and simultaneously manage their own temporal constraints efficiently when scheduling things [17]. So, it can be concluded that there will be several STNs that can be mapped in the Multi-agent STN, which allows agents who in all cases are humans to understand their temporary constraints and then interact in a decentralized manner [18]. This representation allows each agent to maintain its local portion largely independently of other agents, leading to increased concurrency and privacy.

2. Methods

The first thing to do was look for a scheduling dilemma in an online class dataset. In this case, a dataset was used from an Indonesian language e-learning company called cakap.com. The steps to find a dilemma can be seen in Figure 1.

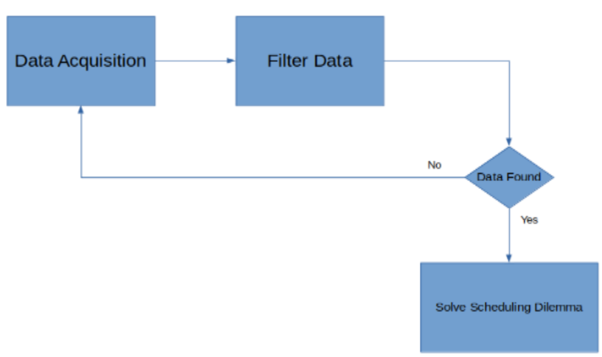

Figure 1. Data Preparation

The practice of refining a data set by selecting, deleting, or eliminating specific data points or subsets based on established criteria is known as data acquisition and filtering. This method is used to isolate, extract, or remove specific portions of a larger data set with the goal of focusing on relevant information or preparing data for subsequent analysis or modeling.

The first step is to take the scheduling data needed. Using query directly to the database that will produce data. The example can be seen in Table 1. This table appears to represent a schedule involving different agents (teachers and students) and their assigned tasks or lessons at specific times.

- **Agent** : Refers to a teacher or a student.
- **Schedule** : Represents the date and time a particular task or lesson is scheduled.
- **Task** : Describes the nature or title of the task or lesson an agent must handle at the specified time.

Each agent (teacher or student) has a schedule with specific tasks or lessons assigned to them at different times and dates. The tasks related to lessons, possibly with specific content such as vocabulary, dialogue or discussing weather conditions in lesson 8. In addition, one of the students is assigned a lesson in a language other than English ("lesson 7 第 7 课 "), possibly indicating a multilingual setting. This table appears to show a schedule of different agents (teachers and students) and their assigned tasks or lessons at specific times.

Table 1. Schedule Data

Agent	Schedule	Task
Teacher 1	09/09/14 05:00 PM	Lesson 8 Vocab & Dialogue
Teacher 1	09/10/14 05:00 PM	LESSON 8 - What's the Weather Like?
Student 1	09/12/14 07:00 PM	LESSON 8 - What's the Weather Like?
Student 2	09/17/14 07:00 PM	lesson 7 第 7 课

The next step was to do the data filtering which tends to produce a dilemma. One of the simplest ways to filter the correct data to be used as experiment was to do a comparison with the complaint data from students and teachers, that can be seen in Figure 2. Complaint data produced by direct query will look like Table 2. Look for 1 set of data that causes scheduling complaints, then the data is feasible to be used as trial material in this study. The core goal in this phase is to identify cases in the data set that trigger complaints from students or teachers. The cases, which often reflect problems, anomalies, or discrepancies, have intrinsic value for the exploratory analysis of this research. If 1 set of data that raises complaints is found, it is considered potentially suitable for inclusion as test material in the research.

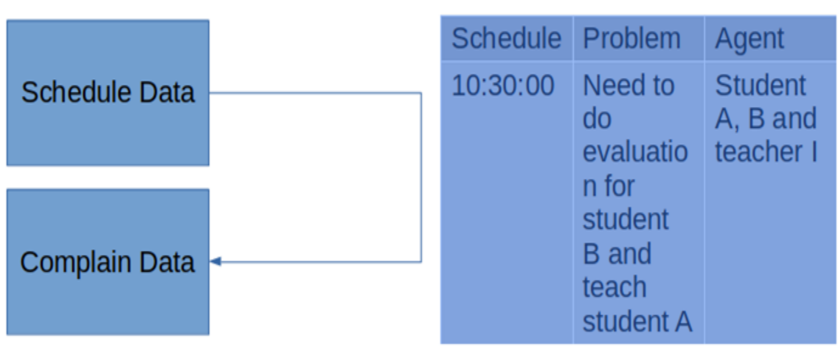

Figure 2. Data Filtration (Example)

The importance of this approach lies in its ability to not only sift through data sets, but also its potential to extract valuable insights from real complaints or concerns expressed by stakeholders – students and teachers. By adapting the research focus to these complaints, this ensures that the material studied is not only relevant, but also addresses real-world problems or observations.

Additionally, the process of selecting data that generates complaints provides a unique opportunity to analyze and address potential problems in an educational context. This provides a lens through which research can delve deeper into the complexity of problems students face or the challenges felt by educators, thereby contributing to a more holistic understanding of the subject.

This approach to data screening and selection, based on identifying complaint-triggering cases in the data set, serves as a foundation in building a strong foundation for subsequent analysis. This not only helps collect material for research, but also ensures that the selected dataset covers real-world scenarios, thereby increasing the applicability and relevance of the research.

Table 2. Complaint Data

Schedule	Problem	Agent
09/12/14 07:00 PM	Conflicting schedule with Student A	Teacher A
09/12/20 10:30 AM	Need to do evaluation for Student B and teach student A	Teacher A

In the dataset that has been taken, a dilemma can be implemented. There are three agents, 2 students (A, B) and a teacher (H) in the online classroom environment which can be seen in Figure 3, which will then be shown as STN S = <V, E> each temporal variable is represented by a node (vertex) $v_i \in V$, and each obstacle is represented by an edge (edge) $\{v_i, v_j\} \in E$.

The students must perform three tasks D, F, and G. Task D which were learning activities such as attending another class or doing homework. Teacher H is responsible for the task of testing competency I, and for conducting routine tests on student B: assignment M. In this case, each agent has a variety of local obstacles when activities can occur, including the duration of the assignment and the time of transition between tasks.

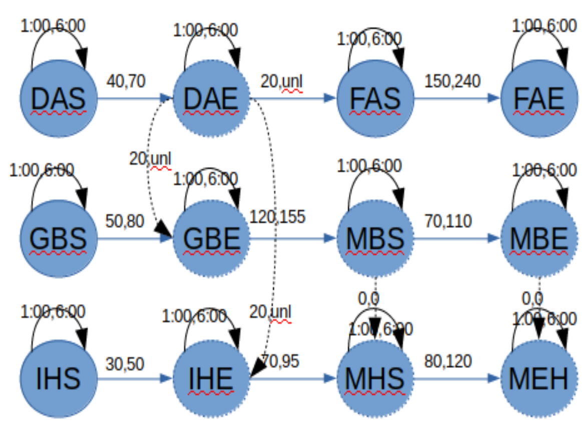

Figure 3. MaSTN Implementation

Each individual (agent) has different constraints on when activities may occur, such as how long tasks take and how lengthy transition times between them. These constraints are represented as a network with subsets for each individual (sub-STNs) and relationship between these subsets (EX). Several factors are required to describe a sub-STN (let's name it STN A):

- VLA (Vertex Local Agent) : These are time points associated with an agent.
- ELA (Edge Local Agent) : These are connections between agents that are based on shared edges.

Constraints then are represented as a network consisting of N number of sub-STNs, one for each agent in a set of {1, . . ., N}, and a set of edge EX that builds relationships between different agent sub-STNs. To describe STN A for a MaSTN, several elements are needed, including VLA and ELA. VLA is defined as a local vertex A, which corresponds to all time point variables specified by agent A. Then, ELA is defined as a series of agents of the same edge or series of edge that can be considered a local edge, where the local edge {vi, vj} ∈ ELA connects two local vertices {vi, vj} ∈ VLA. Both VLA and ELA combined will form agent A local sub-STN, that can be defined as SLA = <VLA, ELA>.

To connect the sub-STN to each other there is an EX relationship, each external edge in the EX set connects the sub-STN of distinct agents and the event to two different agents' local vertices. Or can be formulated as vi ∈ VLA and vj ∈ AVLB. If agent A is aware of EX, then agent A will set the external edge that involves one of the local vertices A.

For example, the initial and final events of task D are represented as vertices named D A S and D A E; D constraints take between 40 to 70 minutes represented as lines from D A S to D A E. Besides, there are external boundaries represented by dashed lines that form relationships between agents. In this example, when doing assignment D, student A interferes with the M student B test; thus, student A must be permitted 20 minutes after the completion of task D to immediately provide space for student B and the teacher to enter the next online class which has several steps assuming login, verification, then waiting for loading to join the telephones. Furthermore, in the local representation of the teacher and student B, the start and end times must be exactly the same. The collection of time points known by student A includes vertices in the row above G B E, and I H E; and edge sequences that are known to cover all sides between these time points.

3. Result and Discussion

3.1. Result

Result will be calculated by performance and variance in schedule. Performance will be compared between 3 algorithms using web browser response time counter. Those algorithms will be served on python web platform called Django.

Evaluation of algorithm performance in web-based applications is an important aspect that influences user experience and system efficiency. This evaluation aims to investigate the performance of three algorithms – MaSTN, Bayesian and Evolutionary Models – in the context of the Python-based web framework, Django. Evaluation metrics include response time measured through web browser interactions. This procedure focuses on assessing the computational efficiency and schedule compliance variance that these algorithms exhibit when applied in a web environment.

Preliminary observations suggest variations in computational efficiency among the three algorithms, with possible implications for their suitability in web-based applications. In particular, considerations such as browser variability, server load, and algorithm optimization influence the interpretation of results. Performance comparison can be seen in Table 3.

Analysis of variance in data selection between the algorithms – MaSTN, Bayesian and Evolutionary Models – revealed interesting patterns in their respective methodologies. Variance in this context shows the number of choices made by each algorithm during the execution process.

MaSTN shows relatively higher variance with two choices, indicating the potential for an iterative or multi-step approach in the decision-making process. This behavior indicates that MaSTN may use a strategy involving multiple data subsets or iterations to obtain the final result. The complexity of the MaSTN decision-making process may contribute to variations in data selection, which may indicate different adaptability or exploration of the input space.

Table 3. Performance

Algorithm	Response
MaSTN	121.47ms
Bayesian Model	847.11ms
Evolutionary	143.31ms

In contrast, the Bayesian model showed lower variance with just one choice. This shows a more deterministic or single selection process compared to MaSTN. The Bayesian model's reliance on a single choice may imply the existence of an efficient decision-making mechanism and support a particular subset or approach of the data, perhaps driven by probabilistic or statistical inference. Like MaSTN, Evolutionary also exhibits a two-choice variant, indicating a decision-making process involving an iterative or evolutionary strategy. Evolutionary algorithms typically explore multiple solutions through successive iterations, with the goal of optimization or adaptation. The observed variance is consistent with these characteristics and indicates some selection of the data during implementation.

The variance in data selection between these algorithms highlights the diverse strategies used by each algorithm in processing and selecting data during its execution. MaSTN and Evolutionary, with their higher variance, suggest an iterative and exploratory decision-making process, whereas the lower variance of the Bayesian model implies a more deterministic or focused approach. Understanding the variance in data selection provides insight into the decision-making dynamics of the algorithm and its potential adaptation to different input scenarios. Further research into how these different selection strategies affect overall algorithm performance and decision quality could provide valuable implications for algorithm design and implementation.

These observations highlight the importance of considering variance in data selection as an important aspect in understanding algorithmic behavior and its implications in various problem-solving scenarios. Variance of schedule calculated by how many schedule selections that each algorithm can produce. Schedule selections can be seen in Table 4.

Table 4. Variance

Algorithm	Variance
MaSTN	2 selections
Bayesian Model	1 selection
Evolutionary	2 selections

3.2. Discussion

Knowing that this algorithm can be used to overcome the flexibility problems scheduling in an online classroom environment is enough to prove this theory can be used. Although it can be admitted that this proof is only a proof on paper, it will be very helpful for future development. The algorithm used is very repetitive and simple, so it is very suitable to be developed to a large scale with the help of computers, for example to solve STP in managing thousands of teachers and students simultaneously or regulating not only teachers and students but along with academic advisers and laboratory use schedules. The schedule produced by this algorithm is very clear and orderly so that teachers and students can be followed comfortably. In addition, in this study it was found that MaSTN is the right algorithm to be used to regulate scheduling between humans, the possibility of using these algorithms used in other schemes is very large.

In some cases, it is likely that this algorithm can be implemented directly without the need for other developments. For example, it is used for scheduling doctors and patients, scheduling shareholder meetings in a company, or being implemented in the daily lives of a family. In the future, this research can be expanded to investigate how teachers and students can react to the dynamics of multi-agent STP like this. These dynamics can include addition or deletion of boundaries or agents, which the possibility of future research can take advantage of the same approach. The algorithm to achieve the expected reaction will be the focus of the research. The important evolution that will be revealed in this research revolves around the critical need for new algorithms. This algorithm is key to determining how the Spanning Tree Protocol (STP) updates its constraints, thereby allowing MaSTN – a complex system – to efficiently deploy additional updates consistently and iteratively for each case that arises. This transformative initiative aims to increase the effectiveness of STP while aligning it with the adaptable MaSTN framework.

The essence of this research lies in extending the applicable evaluation paradigm to more complex and multi-agent STP examples. Often handled through subroutines for scheduling purposes, STP core routines tend to lean toward human intervention and manual oversight. This study serves as a basis for a comprehensive investigation into the development of alternative algorithms intended to be integrated into multi-agent frameworks to address complex temporal problems.

At the heart of this effort is an attempt to go beyond the limitations of manual planning. Today's manual planning processes require significant time investments and constant adjustments, which often hinder operational efficiency. By introducing innovative algorithms, the ultimate goal is to eliminate dependence on manual intervention, increase efficiency across various aspects of the business, and go beyond the online classroom into the broader workflow of the organization.

This research aims to bring about a paradigm shift in the way temporal problems are approached and resolved, especially those packaged in STP scenarios. By investigating the complex layers of MaSTN interactions with STP updates, it aims to not only improve the efficiency of this system, but also lay the foundation for future developments in multi-agent algorithmic solutions.

The results go beyond mere algorithmic development; it promises to usher in a new era where manual planning becomes an artifact of the past. The broader implications are not just limited to academia or single use cases; this involves a paradigmatic shift in operational methodology across business sectors, which has the potential to revolutionize organizational efficiency on a global scale.

In short, this research serves as a catalyst for innovation, aiming not only to optimize the MaSTN-STP relationship, but also to redefine the way temporal problems are approached in a multi-agent framework. The integration of new algorithms is poised to reshape the operational landscape, resulting in major savings and efficiencies across multifaceted business ecosystems, surpassing the limitations of conventional manual planning practices.

4. Conclusion

The proposed algorithm covers a wide range of variables, carefully addressing the critical elements for optimal planning, while maintaining one important element: flexibility. These algorithms demonstrate extraordinary versatility in accommodating the dense network of constraints that exist in both traditional classrooms and the ever-expanding field of remote learning. The results of its application prove the effectiveness of this multi-agent method. They show not only how to overcome various obstacles efficiently, but also how to provide personalized scheduling solutions. Lastly, this strategy empowers users by providing an individualized and effective scheduling mechanism for the ever-changing world of online education. The ability to face adversity and provide flexible responses is an important step to meeting the diverse demands of today's students and teachers.

References

[1] A. C. Hachey, K. M. Conway, C. Wladis, and S. Karim, “Post-secondary online learning in the U.S.: an integrative review of the literature on undergraduate student characteristics,” J Comput High Educ, vol. 34, no. 3, pp. 708–768, Dec. 2022, doi: 10.1007/s12528-022-09319-0.
[2] G. Blau, S. Jarrell, A. Seeton, T. Young, K. Grace, and M. Hughes, “Proposing an Expanded Measure for Comparing Online/Hybrid to Face-to-face Courses,” Journal of Education and Development, vol. 2, no. 2, p. 1, Jun. 2018, doi: 10.20849/jed.v2i2.400.
[3] S. S. Al-Gahtani, “Empirical investigation of e-learning acceptance and assimilation: A structural equation model,” Applied Computing and Informatics, vol. 12, no. 1, pp. 27–50, Jan. 2016, doi: 10.1016/j.aci.2014.09.001.
[4] T. Daymont, G. Blau, and D. Campbell, “Deciding Between Traditional and Online Formats: Exploring the Role of Learning Advantages, Flexibility, and Compensatory Adaptation,” 2011.
[5] C. E. Shepherd and D. U. Bolliger, “Institutional, program, and professional community: a framework for online higher education,” Educational Technology Research and Development, vol. 71, no. 3, pp. 1233–1252, Jun. 2023, doi: 10.1007/s11423-023-10214-3.
[6] S. F. Smith, A. Gallagher, and T. Zimmerman, “Distributed Management of Flexible Times Schedules,” in Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems, in AAMAS ’07. New York, NY, USA: Association for Computing Machinery, 2007. doi: 10.1145/1329125.1329215.
[7] F. Zhang, Y. Mei, S. Nguyen, and M. Zhang, “Multitask Multiobjective Genetic Programming for Automated Scheduling Heuristic Learning in Dynamic Flexible Job-Shop Scheduling,” IEEE Trans Cybern, vol. 53, no. 7, pp. 4473–4486, 2023, doi: 10.1109/TCYB.2022.3196887.
[8] A. R. Morse et al., “What could we do differently next time? Australian parents’ experiences of the short-term and long-term impacts of home schooling during the COVID-19 pandemic,” BMC Public Health, vol. 22, no. 1, Dec. 2022, doi: 10.1186/s12889-022-12495-4.
[9] M. Hausknecht and P. Stone, “Deep Recurrent Q-Learning for Partially Observable MDPs,” Jul. 2015, [Online]. Available: http://arxiv.org/abs/1507.06527
[10] S. Bhat, J. B. Lyons, C. Shi, and X. J. Yang, “Effect of Adapting to Human Preferences on Trust in Human-Robot Teaming,” Sep. 2023, [Online]. Available: http://arxiv.org/abs/2309.05179
[11] A. Tabrez, M. B. Luebbers, and B. Hayes, “Descriptive and Prescriptive Visual Guidance to Improve Shared Situational Awareness in Human-Robot Teaming; Descriptive and Prescriptive Visual Guidance to Improve Shared Situational Awareness in Human-Robot Teaming,” 2022. [Online]. Available: www.ifaamas.org
[12] J. C. Boerkoel and E. H. Durfee, “Decoupling the Multiagent Disjunctive Temporal Problem.” [Online]. Available: www.aaai.org
[13] S. Kong, J. H. Lee, and S. Li, “Multiagent Simple Temporal Problem: The Arc-Consistency Approach,” 2018. [Online]. Available: www.aaai.org
[14] J. Cui, P. Yu, C. Fang, P. Haslum, and B. C. Williams, “Optimising Bounds in Simple Temporal Networks with Uncertainty under Dynamic Controllability Constraints.” [Online]. Available: www.aaai.org
[15] K. Osanlou et al., “Solving Disjunctive Temporal Networks with Uncertainty under Restricted Time-Based Controllability using Tree Search and Graph Neural Networks,” Mar. 2022, [Online]. Available: http://arxiv.org/abs/2203.15030
[16] A. Bit-Monnot, F. Leofante, L. Pulina, E. Abraham, and A. Tacchella, “SMarTplan: a Task Planner for Smart Factories,” Jun. 2018, [Online]. Available: http://arxiv.org/abs/1806.07135
[17] J. C. Boerkoel, L. R. Planken, R. J. Wilcox, and J. A. Shah, “Distributed Algorithms for Incrementally Maintaining Multiagent Simple Temporal Networks.” [Online]. Available: www.aaai.org

[18] G. Casanova GuillaumeCasanova, C. Lesire-Cabaniols CharlesLesire, and onerafr Cédric Pralet, "Managing Dynamic Multi-Agent Simple Temporal Network." [Online]. Available: www.ifaamas.org