1. Ravi Raj[1], 2. Andrzej Kos[2]

(1), (2) AGH University of Science and Technology, Krakow, Poland.
ORCID: 1. 0000-0001-8073-1812; 2. 0000-0003-3089-241X

doi:10.15199/48.2023.09.09

Study and Analysis of Discrete Event-Driven Autonomous System with a Case Study for a Robotics Task

***Abstract**. Highly probabilistic, potential, and dynamic domains are only relatively known to contribute classical techniques for complex system establishment impossible. Currently available technologies and strategies do not adequately address these novel needs. Thus, by enabling autonomous systems to adapt, decision-making and learning abilities, we can empower them with sufficient and appropriate expertise to recognize and address such issues. To address these requirements, discrete event-driven systems (DEDS) have been developed. This system can help the technologists of future autonomous systems by simulating the effect of auxiliary designs on the performance of the autonomous system. For modeling regular feedback of performance that is influenced by traditional techniques and depends on trust, the discrete event-driven method is most suited. This paper describes the DEDS system, the modeling of this system, and as well as the supervisory control system by explaining the supervisor, and partial supervisor. A comprehensive literature survey has been carried out in this article to explain the controllability, diagnosability, and observability potential of the DEDS system for various applications. Some of the major areas of applications such as healthcare, logistics, robotics, and banking sectors, have been discussed. Also, we have explained this system with the help of modeling a discrete event system for a queuing problem associated with robotics tasks as an example by the simulation with MATLAB 2022a. Lastly, the possible future research directions in the DEDS advancement have been provided.*

***Streszczenie.** Wysoce probabilistyczne, potencjalne i dynamiczne domeny są znane z tego, że niemożliwe jest wniesienie klasycznych technik do tworzenia złożonych systemów. Obecnie dostępne technologie i strategie nie zaspokajają odpowiednio tych nowych potrzeb. Zatem, umożliwiając autonomicznym systemom adaptację, podejmowanie decyzji i zdolność uczenia się, możemy wyposażyć je w wystarczającą i odpowiednią wiedzę fachową, aby rozpoznawać i rozwiązywać takie problemy. Aby spełnić te wymagania, opracowano dyskretne systemy sterowane zdarzeniami (DEDS). System ten może pomóc technologom przyszłych systemów autonomicznych, symulując wpływ podsystemów pomocniczych na wydajność systemu autonomicznego. Do modelowania regularnych informacji zwrotnych na temat wyników, na które wpływają tradycyjne techniki i które zależą od zaufania, najbardziej odpowiednia jest metoda dyskretnych zdarzeń. W artykule opisano system DEDS, modelowanie tego systemu, a także system kontroli nadzorczej poprzez opisanie nadzorcy i kierownika częściowego. W tym artykule przeprowadzono obszerny przegląd literatury w celu wyjaśnienia sterowalności, diagnozowalności i potencjału obserwowalności systemu DEDS w różnych zastosowaniach. Omówiono niektóre z głównych obszarów zastosowań, takich jak sektor opieki zdrowotnej, logistyki, robotyki i bankowości. Wyjaśniliśmy również ten system za pomocą modelowania systemu zdarzeń dyskretnych dla problemu kolejkowania związanego z zadaniami robotyki na przykładzie symulacji z MATLAB 2022a. Na koniec przedstawiono możliwe przyszłe kierunki badań w zakresie rozwoju DEDS. (**Badanie i analiza dyskretnego systemu autonomicznego sterowanego zdarzeniami ze studium przypadku zadania robotyki**)*

Keywords: Autonomous system, discrete event system (DES), logical controlled, Petri nets, supervisory control.
Słowa kluczowe: System autonomiczny, system zdarzeń dyskretnych (DES), sterowanie logiczne, sieci Petriego, sterowanie nadzorcze.

1. Introduction

Autonomous systems contain an extensive collection of technologies: from drones that can empower scanning social, physical, and natural environments; to machines empowered with artificial intelligence (AI) that might out game human beings in challenges or games such as chess; to mobile robots that work independently; to self-driving enabled vehicles. If any unanticipated events are occurring during the operation of an autonomous system, then the system might change its behavior [1]. These systems potentiality and their area of application have significantly augmented in recent decades, with tremendous successes in both military and civilian implications. These systems are applicable in remote sensing performance to monitor the Antarctic area changes which mean to keep eye on the changes in the area of land having ice due to climate change problems. The area of AI and autonomous systems has already experienced significant expansion, and we can observe the recent development in almost every sector, including healthcare, gaming, finance, robotics, education, space exploration, e-commerce, social media, monitoring for security and safety, and numerous others [2]. The recent changes and current trends in the application of automation in businesses, logistics, industries, securities, health sectors, environmental sciences, and many more ways lead to novel, complex, and moderately conflicting demands on the planning of these sectors and control systems [3]. Several analytical and empirical techniques have been advanced to brief the decision of design for novel systems that grasp higher degrees of automation. For complex systems which drive automation, but as well as depend on the interaction of humans for contingency and guidance management, integrated models are required that yield an individual human understanding and elements of computer and confront the crucial interactions of that type of complex systems [4].

Dynamic systems are known as discrete event-driven systems whose nature is described by instant fluctuations in their state values, that gather discrete values from a set of infinite values. The evolution of the state is due to the occurrence of events. A discrete-event-driven system (DEDS) is a state of a discrete, event-driven technique in which the evolution of the state depends totally on the appearances of non-synchronous discrete events across time. However equivalent to continuous-variable driven systems, DEDS consists exclusively of spaces of discrete state and state of event-driven transition mechanisms. DEDS are dynamic in nature with two characteristics of definition: Their state and space are potentially infinite and discrete, and they have event-driven dynamics type. That means there are not any differences or differential equations in this type of system. Events that generally happen non-synchronously cause an upgradation from one state to another for the state space model. These sets of events are abbreviated as E, which is a discrete set. The behavior of a DEDS is thus occupied by a structure of discrete transition, where the states are represented by nodes, and the arcs adjoining all the nodes represent all the feasible transitions between states which are created due to the occurrence of events [5]. These structures of discrete transition can be abbreviated in various forms, depending on the type of attributes, which are connected to them for quantitative or logical analyses.

2. Modelling of discrete event-driven system

Modeling of discrete event-driven system formalisms must represent the transitions of event-driven between the multiple discrete conditions of a DEDS. The universal method for the formalism of the modeling of DEDS is that of automation, which has a better relationship with general linguistic theory. A discrete event-driven automation system can be abbreviated as, G [6]:

$$G = (X, E, f, \Gamma, x_o, X_m) \tag{1}$$

Where, X is the group of states, which might be infinite; E is the set of events, which is finite, related to adaptations in G; $f: X \times E \rightarrow X$ is the function of adaptation; $f(x, e) = y$, which means here adaptation is labeled by event ***e*** between x to y-states. $\Gamma: X \rightarrow 2^E$ is the possible function of events; $\Gamma(x)$ is the collection of every event e for $f(x, e)$ has been defined; the initial state is x_o, and $X_m \subseteq X$ is the collection of marked states.

Supervisor (S) can control the transition function (G) in the regard that the events that are controllable of transition function G might be enabled dynamically or disabled by S. Generally, a supervisor (S) is a function gathered through the generated language by G to the power sets of E's and which is abbreviated in Equation (2) by the following way [6]:

$$S_P: L(G) \rightarrow 2^E \tag{2}$$

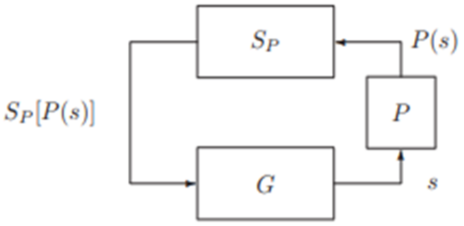

Fig. 1. The feedback control of supervisory control for a partial observation case [6]

Fig. 1 represents a feedback control system loop for the supervisory control during partial observation, which includes a logical projection *P* among *G* and the supervisor. The partial observation supervisor is denoted by S_P to show the presence of *P*. The actions carried out by subordinated level or if the systems in equation (1) are integrated with other DEDS without their own cost function is called supervisor, which might be depended upon the events occurring in the system by following the feedback law. Because of the *P* presence, the possible supervisors cannot differentiate between two strings s_1 and s_2 due to the projections between s_1 and s_2 are similar, which means, $P(s_1) = P(s_2)$; where $s_1, s_2 \in L(G)$, and the same control action will be carried out by the supervisors, $S_P[P(s_1)]$. For the validation of these discussions, the partial observation supervisor can be defined and abbreviated as [6]:

$$S_P: P[L(G)] \rightarrow 2^E \tag{3}$$

Where S_P is the partial observation supervisor, *P* is the natural projection, *L(G)* is the possible language set in the group of events *G,* and *E* is the events. Equation (3) shows that as a result, the control action might only be altered once an observable event has occurred, which means when *P(s)* fluctuates. Whenever there are some changes occurring instantaneously in the control action then there must be some observable event occurring.

Events in a DEDS are described as generally connected state pairs [7]. The adoption of a sequence of operations might be directed to get an optimal collection of the performance index. The ambitions of both levels of control are generally conflicting since the higher satisfaction of the goals of the supervisor, the less allowable are control patterns, and the less freedom remains for subordinated decision makers. For example, if the problem of traffic control only allowed in right and forward directions are permitted then the fluency and safety of the traffic as a completely might enhance although these conditions might not be convenient for every participant in traffic problems. DEDS has become more popular in the last three decades among researchers who are working in the field of autonomous systems. Researchers are mainly focused on the diagnosability, controllability, and observability of DEDS. Several terms such as realization, stability, state reduction, discrete control, etc. belong to this system. Petri nets are the most important networks playing a vital role in the configuration of DEDS and it is also applicable for diagnosability, controllability, and observability.

3. Literature survey

In the last two decades, several researchers have been working on DEDS. Wenbin Qiu *et al.* [8] proposed a decentralized methodology to diagnose system failure using multiple diagnoses, every diagnosis having its collection of sensors, without incorporating any communication between coordinators and diagnosis. The decentralized diagnosis notion is estimated by presenting the co-diagnosability notion which needs that a diagnosis detects one failure within a delay time. They describe algorithms of complexity polynomial according to the non-fault specification and the system size is illustrated for individual diagnosis of offline synthesis, testing co-diagnosability, calculating the area in diagnosis delay, and online diagnosis. A. Ramirez-Trevino *et al.* [9] present an interpreted Petri nets (IPN) model for the diagnosis of faults of discrete event systems. The proposed model contains the normal and faulty states of the system. The major role of this research work is a methodology of bottom-up modeling. It illustrates the action of system elements utilizing the necessary variable of states and designating a range to the individual variables of state. All individual variables of state are denoted by a model of IPN, which is known as a module. The proposed research paper results allow for making the model of DEDS in which faults might occur, estimating the model's diagnosability, and implicating an online diagnosis. Tae-Sic Yoo *et al.* [10] present algorithms of polynomial time for determining I-diagnosability and diagnosability. The proposed algorithms in this article are dependent on the illustration of a verifier of nondeterministic automation. S. Hashtrudi Zad *et al.* [11] present an approach for the online diagnosis of passive fault in systems, which has been modeled as automata of finite-state. The fault detection system and the modeled system do not need to be started at a similar time. Even more, for starting the diagnosis process, information about the fault status and state of the system is not required. The scheme for model reduction with the complexity of polynomial time is defined to reduce the complexity of computational design in this paper. Shengbing Jiang *et al.* [12] study the failure diagnosis of DEDS with LTL (linear-time temporal logic) specifications. The formulas of LTL are utilized for failure specification in the system. The specifications based on LTL make the specification more user-friendly and specifying process easier than the normal automata/language-based specifications; and they might diagnose the failures indicating the opposition of both safety and liveness properties, whereas the previous normal automation/language-based specifications might gather the failures indicating the violation of safety properties only.

They defined the diagnosability and pre-diagnosability of DEDS in the setting of temporal logic. Finally, the testing algorithm of diagnosability and pre-diagnosability, and the mixture of a diagnosis is obtained. D. Thorsley *et al.* [13] discuss the diagnosability of stochastic DEDS. They describe the notions of AA- and A-diagnosability for stochastic automation. This paper defines offline conditions as sufficient and necessary to assure A-diagnosability and enough to assure AA-diagnosability. This article takes two examples of high-voltage AC systems as an illustration of this approach. F. Basile *et al.* [14] present a new technique for the diagnosis of faults in a discrete event system (DES). This technique is different than the usual methodology of diagnosis for faults in the DES because it depends upon the estimation of possible events of fault through online mode needed to explain the final observed event. The efficiency of this system is very high due to its ability of mathematical programming based on the modeling of the system by Petri nets. Finally, this Petri nets representation in the form of a graph allows the diagnosis agent to estimate off-line parts of the net to enhance the efficiency of computation. M. V. Moreira *et al.* [15] present a technique for the verification of polynomial time of decentralized diagnosability of DES. The initial step in the failure occurrence diagnosis for DES is the verification of the diagnosability of the system. Various works have been focused on this issue utilizing either verifiers or diagnoses for both decentralized and centralized architectures. This algorithm needs polynomial time for the number of events and states of the DES and has smaller complex computation than every other technique discussed earlier. Z. Chen *et al.* [16] discuss the active diagnosability of DES and its implication for the diagnosis of battery fault. There are too many batteries present in the battery system, and each battery has different operating and faulty modes. This keeps the battery system status very complex. This article describes the active diagnosis of DES. This approach is illustrated by the help of four batteries system and two temperature sensors.

V. Chandra *et al.* [17] present a novel formalism of modeling for creating appropriate architectures of complex systems based on event rules and precedence of occurrence. The applied class in this system consists of binarized signals. The presented technique is very less error-prone, less cumbersome, and user-friendly for the modeling of tasks. Also, the size of the model and the number of states for the model of the proposed system are polynomial and exponential consecutively in the number of the binarized signals. This article formalized the modeling of systems utilizing examples drawn from process control and manufacturing systems. L. Feng *et al.* [18] present an architecture of supervisory control for the DED system. The proposed architecture is hierarchical and flexible and decentralized to reduce the effort of computation in the optimal designing of nonblocking supervisors for the system. They make a system into modular subsystems which acquire dependencies of internal interactions. This paper presented appropriate conditions to authorize those modular supervisors and coordinator results in optimally nonblocking and permissive control. K. Cai *et al.* [19] proposed an approach for the distributed control of DES in the framework of the theory of supervisory control. The main issue estimated is how to evaluate local controllers for agents to the final controlled performance is equal to that attained by global supervision. The main objective of this paper is to study the design of distributed control for DES at the local and global levels. K. W. Schmidt *et al.* [20] discuss the framework and application of fuzzy DES for multi-objective control in the navigation of mobile robots. The selection of a fuzzy system delineation is explained by the supposition in a realization of the controller which is based on several potentially imprecise measurements of the sensor. The proposed scheme follows three steps; a sensor evaluation is responsible to determine the current fuzzy state, the fuzzy state for the future instant of sampling is supposed for every possible control reaction of the system, and a weighting strategy of the original multi-objective is presented to evaluate the control works to be implied at the present instant of sampling. Y. Chen *et al.* [21] proposed a compact supervisory control of DES by Petri nets integrated with arcs of data inhibitor. The technique used in this paper is for designing a supervisor of optimal Petri net with arcs of data inhibitors to obstruct a system from approaching the markings of illegal with respect to specifications control. Here, two methods are developed to decrease the structure of the supervisor by constricting all the control places. A. Raman *et al.* [22] proposed a technique for fault-tolerant control of DES with failure of controllability. They take supervisory control of this system in the availability of an irrelevant fault that contributes an irrational subset of controllable events for short-term disturbance. The fault is recognized when controllable events first occur which are crippled by the supervisors. This paper presents an important and appropriate condition for the availability of a supervisor that lays out a specification of the desired language. This research proves that such a supervisor can always be described in the specification and the language of a plant is regular.

U. Wikborg *et al.* [23] present noncyclic programming for timed DES with implications to the tool of unit-armed cluster utilizing optimization of Pareto-optimal. This methodology is to convert the issue into a multi-objective issue by observing the primed times of the assets as objectives to reduce. Only possible states form a multi-objective smallest track problem and provide us with a top around in the number of states. These problems are solved with absolute enumeration by creating their proposed decisions depending on every state and their Pareto maximum solutions. The technique can potentially represent the manufacturing system state and utilizes them to estimate a better schedule to enhance productivity. The most important benefit of this technique is that it might very fast estimation of an optimal schedule dependent upon the present state of the system. K. Cai *et al.* [24] present a relative observability of DES and their corresponding supreme sub-languages. A novel algorithm has been implemented for the computation of relatively controllable and observable sub-language of a given language. This article illustrates the new concept of observability and algorithms with autonomous guided vehicles and a guideway example. X. Yin *et al.* [25] present an approach for illustrating supervisors of property-enforcing for partially observed DES. This methodology applies to various properties such as opacity, diagnosability, anonymity, safety, attract-ability, and detectability. This approach utilizes all enforcement structures to make an algorithm of synthesis that develops a supervisor which is maximally permissive and provably property enforcing. The application of this uniform approach has been demonstrated to the enforcement of all possible properties. S. Tripakis *et al.* [26] present a decentralized observation of DES. If agents are truly able to take their own decisions depending upon their observations, without dependence on a fusion rule or centralized coordinator is known as an ideal decentralized system. A condition taken in this research which is called at least one can tell is responsible to characterize the behavior of the decentralized agents are bad or good. This article constructs the relationship between this condition and decentralized DES. P. Xu *et al.* [27] present an approach of delay co-observability verification for DES and their language of specifications. The complexity of computation is polynomial for this proposed algorithm in comparison to the number of events and states of the DES and exponential by the number of the upper bounds and local supervisors on the delays. The language specification is delayed and co-observable if verifiers have no bad states.

4. Applications of a discrete event-driven system

Discrete event simulation, logic controllers, performance evaluation and control, and supervisory control are demonstrated. Automata acquire the sequential and logical behavior of the system. In recent decades most automation companies are utilizing DEDS such as programmable logical controlled (PLC) and supervisory controlled and data acquisition (SCADA) which are reducing the workloads in different fields of industries. Discrete event dynamic systems and DES are two approaches of modeling widely utilized as tools of decision support in supply chain and logistics management. DES is mainly utilized at the tactical/operational level, whereas a discrete event dynamic system is mainly utilized for problems of the model at the level of strategy. This system can be used to monitor the waste-fire management system by keeping an eye on the pattern of the waste fire. DEDS is applicable in the health care, logistic system, robotics, autonomous system, and banking sector.

4.1. Application in Healthcare

In the area of health care services, modeling of simulation has been extensively recognized and is nowadays the best methodology for studying problems of health care management. Although various qualitative and quantitative techniques might be utilized to analyze the systems of health care, simulation is comprehended as the best direction and is playing an efficient vital role in the procedures that aid managerial decisions of health care. This technique is an effective tool to reach various types of issues related to health care. For the modeling of disease progression, DES is generally implied to transparently construct and conceptualize the disease courses, transition of health states, and events of disease-related patients will pass from different interventions [28]. Models of discrete events simulate procedures over time and accompany individuals, dynamic entities that interface with the resources of systems. In the modeling of health care, individual patients are generally considered as these entities, although medicines or receipts are simulated by the available models. Entities fluctuate with time, and they capture and release the resources of systems such as beds, diagnostic equipment, nurses, operating theaters, and doctors. Simulation of discrete events plays an important and better role in making decisions about health care. The main important advantage of DES is its capacity to affect interventions, risks, and activities with patients that are followed by their judgment. Simulation techniques are described in several ways, but the most famous are divided into four categories: system dynamics, Monte Carlo, agent-based simulation, and discrete event simulation [29].

4.2. Applications in the Logistic System

Growing competition between conventional and emerging techniques has brought novel attention to very fast innovation and regular differentiation in each feature of the logistic system, from the initial stage of production to the last steps of distribution. To fill the gap between excellent plans and outstanding initiatives of business, leading industries implement DES and DEDS in the logistics and supply chain system. Logistic and supply chain management is a crucial part of any business, nowadays discrete event simulation and driven systems or dynamic systems are playing a vital role to make the best management system for logistics by integrating all business entities together. Application of system dynamics and discrete event simulation to support decisions at the operational/tactical or strategic level, there is a small proof of any difference within the logistics system [30]. Fig. 2 demonstrates the ordering of the issues related to the logistic system.

Fig. 2. Ordering of logistics system problems into strategic and operational/tactical platform

4.3. Applications in the Robotics and Autonomous Systems

In recent decades, the significant evolution of new technology in computer science has brought the rapid growth of dynamic systems that are highly complex and mostly man-made. The performance of dynamic systems is described by the manifestation of asynchronous events, due to this aspect they are known as DES. The basic mathematical illustration of complex manufacturing and robotics autonomous systems is not able to generate a set of models that efficiently acquires the system dynamics over the complete range of the operation of the system. The mathematical models which have a closed-control format are almost limited to the models of linear systems. Computer simulation of discrete-event and non-linear hybrid models provides an online layout of robotic control systems. Hybrid and discrete event-driven systems have been utilized in the automation and manufacturing domains to state changes in the designing system within a process. Untimed and timed Petri nets and automata of state integrated into Markovian-stochastic perturbation and other models have been utilized significantly to build and control automated systems of manufacturing [31]. DEDS controllers of high level can guide the robot’s behavior based on the outputs of the sensor. These system models and evaluate outcomes of systems that are deterministic when events occur continuously. During this work, every simulation is also responsible for constraints, uncertainties, and interdependencies that affect the condition. Like whenever interactions between operations and objects are dependent on time then DES can be considered.

4.4. Application in the Banking Sector

This system can be used in the banking sector to enhance the quality of customer services and reduce the load of bank employees. It can be utilized for solving problems related to the queuing structure such as multiple types of documents organized in a queuing pattern and calculations of the interest rates for different types of saving accounts. It can also be implemented in more complex transactions such as clearing cheques, making demand drafts, and many more operations carried out daily in the banking system. DEDS system can also be utilized for the purpose of making teller lines in front of automatic teller machines (ATM) to deliver cash to the customers with higher accuracy and reduced time of operation. Therefore, this system is playing one of the major driven forces for the functioning of banks with very high efficiency.

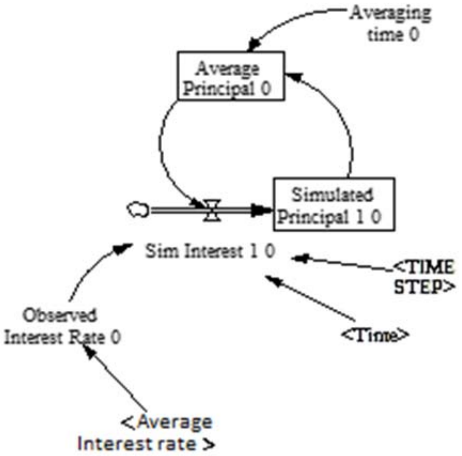

Fig. 3. Discrete and deterministic bank account model [32]

Fig. 3 represents the discrete model of the bank account with deterministic interest capital and Fig. 4 illustrates the discrete model of the bank account with stochastic interest capital.

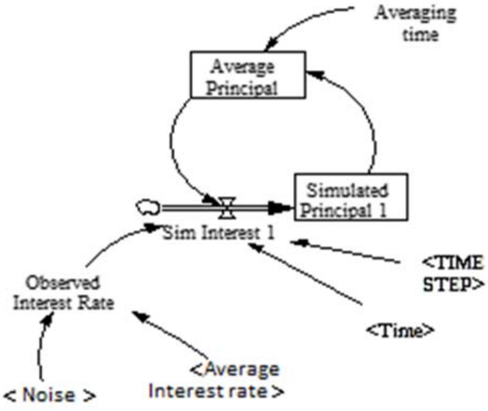

Fig. 4. Discrete and stochastic bank account model [32]

5. Modelling of a robotic task based on the DEDS

Most of the models of available robotic tasks are not based on traditional methods, are significantly customized to the task, and are associated with very few numbers of behaviors at hand. P. Lima *et al.* [33] present a modeling approach based on system theory for common robotic tasks which allows a structured technique to design, model, and analysis, scaling up to applications in the real world, providing techniques for design from specifications, logical verification, and stochastic performance, as well as the implementation of development through learning over time. This method of modeling is based on using DEDS models, mainly finite state automata and Petri nets, for the planning of robot representation. This specific representation allows utilizing every available DES design and analysis tool to control the task of robotic general design and analysis.

A robot plan is constituted of various actions, running sequentially or concurrently. For example, sequences of action, loops, and sub-plans directing distinguish robots from a group of robots. There are three entities such as propositions that estimate with expressed arguments, events, and primitive actions that might have arguments, but should be confined to some real objects are defined for the modeling of robotic tasks [34]. Those subsets of events that are responsible for the primitive action initiation are controllable and those responsible for the existence of the robot might not stop or start are uncontrollable. When we use representations of finite state automata, primitive actions, and propositions are gathered in states, and transitions occur due to events. Robotic tasks modeling by Petri Net distributes the state around the places of Petri nets. Some places represent primitive actions, others represent propositions. The plan of Petri net representations is sufficient to represent the cooperative robot’s plan.

Let’s consider a factory that has autonomous robots to perform operations. Robots are performing work in a queue on the operating belt. That means firstly one robot performs the work, after that second robot performs the work, and so on. Thus, just one robot can do work at a time, which means the operation belt has just one robot's capacity. But the robots are more than the specified number of capacities at the operation belt. Thus, we are considering that robots must be in a queue in front of the entry gate of the operation belt to wait for their term to enter and perform their work. The consuming time for each robot to perform work at the operation belt is considered for 2 seconds per unit. Where the Entity generator is responsible to create entities for the modeling of robot arrival. Entity Queue is responsible for entities in queue storage to model the robots queuing which are waiting to perform their task.

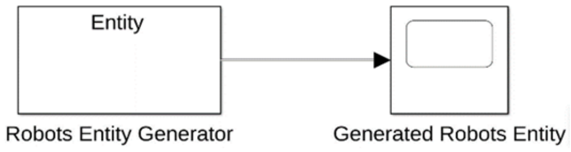

Fig. 5. Simulink model for the generation of robot’s entity

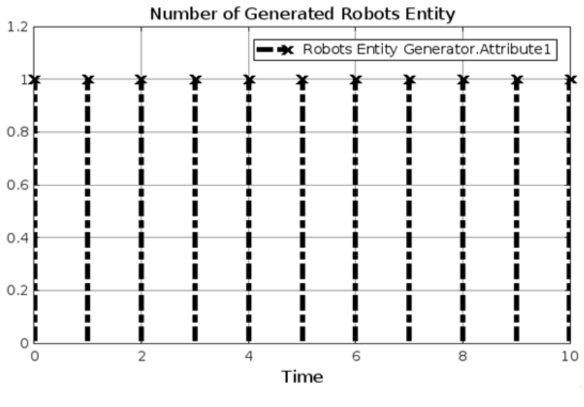

Fig. 6. Number of robot entities generated per second.

This case study has been taken as an experiment related to the MATLAB simulation. Fig. 5 represents the model of robot generation at every second. Fig. 6 presents the corresponding generation of entity of robot per second from the Simulink model described in Fig.5 and Fig. 7 represents the Simulink model for the functionality of the autonomous system in the factory organized in the way of DEDS. The Simulink model will provide the real-time functionality of the robot on the operation belt.

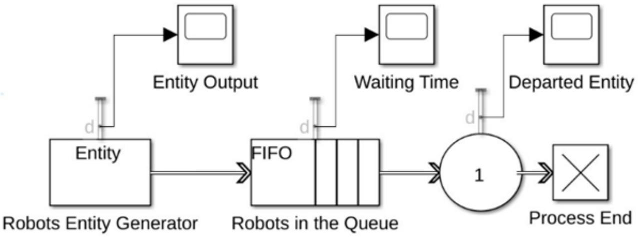

Fig. 7. Simulink model for the robotics work in queueing tasks

Fig. 8 represents the average time taken by the robots to deliver their corresponding work, which is in the queue, and Fig. 9 represents the average time taken by the robots departing from the operation belt.

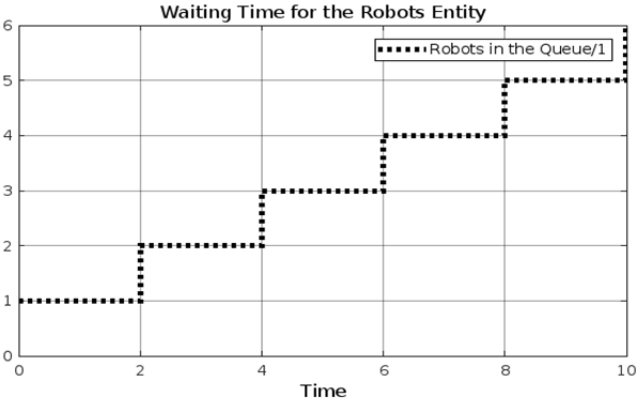

Fig. 8. Waiting time for robots in the queue

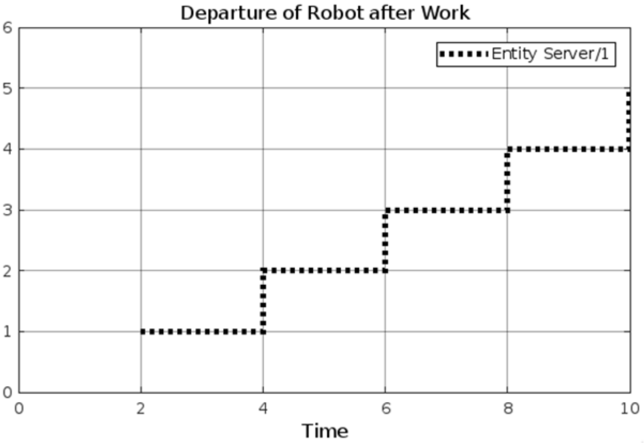

Fig. 9. Average time taken by robots to departure after work.

6. Future Research Trends

In recent years discrete event-driven systems is applying in various fields such as air traffic control, automated manufacturing, transportation, and computer networks, as well as in novel emerging fields such as information and communication processing, healthcare, management, and allocation of technical, financial, and human resources. In recent decades a huge number of practitioners, scientists, and researchers have been focused on DEDS to encounter a series of complex issues of combinational property to optimize, analyze, and control DEDS. Researchers can be devoted to providing solutions for a series of drawbacks that emerge in every system such as fault diagnosis, supervisory control, state estimation, optimal resource allocation, recovery, and deadlock prevention with a very high degree of automation. However, various theoretical problems in DEDS are still available, or this system has only limited application to studies of small cases, while DEDS system application at a very large scale for the problems of the industrial background remains a big challenge [35]. Additionally, a series of prominent fields such as distribution of power and healthcare, deserve the concentration of the DEDS researchers since various features in their framework might be formulated with higher accuracy in terms of DEDS, taking benefit of the huge amount of methodological and theoretical outcomes in this area. Finally, researchers must focus on novel models for abbreviating DEDS, such as models of fluid based on event-driven. There are various models of these types have already been described by the Petri-nets or queuing networks fluidification, and utilized to solve optimization, control, and simulation problems such as estimation of state or fault diagnosis, which has been demonstrated by a smaller number of researchers, and the application of DEDS based models of fluid to novel emerging fields is generally unexplored. This system can be used to combat various types of climate change issues to gather data from historical changes in the environmental parameters to enhance accuracy.

7. Conclusion

Discrete event-driven systems play a crucial role in getting higher accuracy of autonomous systems for different types of applications such as health care, marketing management, human-computer interactions, manufacturing industries, industrial robots, etc. with a real-time domain. We have discussed various challenges and problems associated with this system. Also, the methodology for the modeling of the robotic task based on this system has been discussed completely in this research work. This paper provides a comprehensive introduction to the DEDS, a wide range of surveys for three main properties of diagnosability, controllability, and observability of DEDS, some major applications of this system, an overview of a robot task modeling based on DEDS, and future research perspective of this system. This paper provides an introduction to the supervisor and partial supervisor for the supervisory control case. This article is best for early-stage researchers to get a comprehensive idea of DEDS from it and start their careers in this field. We hope that a similar type of research will be carried out extensively in this area to deal with the problems discussed in this research work.

Acknowledgments

The authors wish to thank for the financial support from subvention No. 16.16.230.434 paid for by the AGH University of Science and Technology, Krakow, Poland.

***Authors**: Ravi Raj, Faculty of Computer Science, Electronics, and Telecommunications, AGH University of Science and Technology, Aleja Adama Mickiewicza 30, 30-059 Kraków, Poland, E-mail: raj@agh.edu.pl; prof. dr hab. inż. Andrzej Kos, Faculty of Computer Science, Electronics, and Telecommunications, AGH University of Science and Technology, Aleja Adama Mickiewicza 30, 30-059 Kraków, Poland, E-mail: kos@agh.edu.pl.*

REFERENCES

[1] D. P. Watson, and D. Scheidt, "Autonomous systems", Johns Hopkins APL Technical Digest (Applied Physics Laboratory), vol 26, issue 4, pp. 368-376, Jan. 2005.
[2] R. Raj, A. Kos, "Artificial Intelligence: Evolution, Developments, Applications, and Future Scope", Przegląd Elektrotechniczny, vol. 2023, no. 2, Feb. 2023. doi:10.15199/48.2023.02.01.
[3] R. Raj, A. Kos, "A Comprehensive Study of Mobile Robot: History, Developments, Applications, and Future Research Perspectives", Applied Sciences, vol. 12, no. 6951, July 2022. https://doi.org/10.3390/app12146951
[4] M. L. Cummings, and A. S. Clare, "Holistic modelling for human-autonomous system interaction", Theoretical Issues in

Ergonomics Science”, vol. 16, issue 3, March 2015, doi: 10.1080/1463922X.2014.1003990.
[5] S. Lafortune, “Discrete Event Systems: Modeling, Observation, and Control”, Annual Review of Control, Robotics, and Autonomous Systems, vol. 2, pp. 141-159, 2019. doi: 10.11146/annurev-control-053018-023659.
[6] C. G. Cassandras, and S. Lafortune, “Introduction to Discrete Event Systems”, Springer New York, NY, Ed. 2, 2007. https://doi.org/10.1007/978-0-387-68612-7
[7] D. Huang, X. Hong, and M. Gerla, "Situation-aware trust architecture for vehicular networks," in IEEE Communications Magazine, vol. 48, no. 11, pp. 128-135, November 2010, doi: 10.1109/MCOM.2010.5621979.
[8] Wenbin Qiu and R. Kumar, "Decentralized failure diagnosis of discrete event systems," in IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, vol. 36, no. 2, pp. 384-395, March 2006, doi: 10.1109/TSMCA.2005.853503.
[9] A. Ramirez-Trevino, E. Ruiz-Beltran, I. Rivera-Rangel, and E. Lopez-Mellado, "Online Fault Diagnosis of Discrete Event Systems. A Petri Net-Based Approach," in IEEE Transactions on Automation Science and Engineering, vol. 4, no. 1, pp. 31-39, Jan. 2007, doi: 10.1109/TASE.2006.872120.
[10] Tae-Sic Yoo and S. Lafortune, "Polynomial-time verification of diagnosability of partially observed discrete-event systems," in IEEE Transactions on Automatic Control, vol. 47, no. 9, pp. 1491-1495, Sept. 2002, doi: 10.1109/TAC.2002.802763.
[11] S. Hashtrudi Zad, R. H. Kwong, and W. M. Wonham, "Fault diagnosis in discrete-event systems: framework and model reduction," in IEEE Transactions on Automatic Control, vol. 48, no. 7, pp. 1199-1212, July 2003, doi: 10.1109/TAC.2003.814099.
[12] Shengbing Jiang and R. Kumar, "Failure diagnosis of discrete-event systems with linear-time temporal logic specifications," in IEEE Transactions on Automatic Control, vol. 49, no. 6, pp. 934-945, June 2004, doi: 10.1109/TAC.2004.829616.
[13] D. Thorsley and D. Teneketzis, "Diagnosability of stochastic discrete-event systems," in IEEE Transactions on Automatic Control, vol. 50, no. 4, pp. 476-492, April 2005, doi: 10.1109/TAC.2005.844722.
[14] F. Basile, P. Chiacchio and G. De Tommasi, "An Efficient Approach for Online Diagnosis of Discrete Event Systems," in IEEE Transactions on Automatic Control, vol. 54, no. 4, pp. 748-759, April 2009, doi: 10.1109/TAC.2009.2014932.
[15] M. V. Moreira, T. C. Jesus, and J. C. Basilio, "Polynomial Time Verification of Decentralized Diagnosability of Discrete Event Systems," in IEEE Transactions on Automatic Control, vol. 56, no. 7, pp. 1679-1684, July 2011, doi: 10.1109/TAC.2011.2124950.
[16] Z. Chen, F. Lin, C. Wang, Y. Le Wang, and M. Xu, "Active Diagnosability of Discrete Event Systems and its Application to Battery Fault Diagnosis," in IEEE Transactions on Control Systems Technology, vol. 22, no. 5, pp. 1892-1898, Sept. 2014, doi: 10.1109/TCST.2013.2291069.
[17] V. Chandra and R. Kumar, "A discrete event systems modeling formalism based on event occurrence rules and precedences," in IEEE Transactions on Robotics and Automation, vol. 17, no. 6, pp. 785-794, Dec. 2001, doi: 10.1109/70.975991.
[18] L. Feng and W. M. Wonham, "Supervisory Control Architecture for Discrete-Event Systems," in IEEE Transactions on Automatic Control, vol. 53, no. 6, pp. 1449-1461, July 2008, doi: 10.1109/TAC.2008.927679.
[19] K. Cai and W. M. Wonham, "Supervisor Localization: A Top-Down Approach to Distributed Control of Discrete-Event Systems," in IEEE Transactions on Automatic Control, vol. 55, no. 3, pp. 605-618, March 2010, doi: 10.1109/TAC.2009.2039237.
[20] K. W. Schmidt and Y. S. Boutalis, "Fuzzy Discrete Event Systems for Multiobjective Control: Framework and Application to Mobile Robot Navigation," in IEEE Transactions on Fuzzy Systems, vol. 20, no. 5, pp. 910-922, Oct. 2012, doi: 10.1109/TFUZZ.2012.2189219.
[21] Y. Chen, Z. Li, K. Barkaoui, N. Wu and M. Zhou, "Compact Supervisory Control of Discrete Event Systems by Petri Nets with Data Inhibitor Arcs," in IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 47, no. 2, pp. 364-379, Feb. 2017, doi: 10.1109/TSMC.2016.2521833.
[22] A. Raman and R. S. Sreenivas, "Fault-Tolerant Control of Discrete-Event Systems with Controllability Failures," in IEEE Control Systems Letters, vol. 4, no. 3, pp. 674-679, July 2020, doi: 10.1109/LCSYS.2020.2991192.
[23] U. Wikborg and T. Lee, "Noncyclic Scheduling for Timed Discrete-Event Systems with Application to Single-Armed Cluster Tools Using Pareto-Optimal Optimization," in IEEE Transactions on Automation Science and Engineering, vol. 10, no. 3, pp. 699-710, July 2013, doi: 10.1109/TASE.2012.2217128.
[24] K. Cai, R. Zhang, and W. M. Wonham, "Relative Observability of Discrete-Event Systems and Its Supremal Sublanguages," in IEEE Transactions on Automatic Control, vol. 60, no. 3, pp. 659-670, March 2015, doi: 10.1109/TAC.2014.2341891.
[25] X. Yin and S. Lafortune, "A Uniform Approach for Synthesizing Property-Enforcing Supervisors for Partially-Observed Discrete-Event Systems," in IEEE Transactions on Automatic Control, vol. 61, no. 8, pp. 2140-2154, Aug. 2016, doi: 10.1109/TAC.2015.2484359.
[26] S. Tripakis and K. Rudie, "Decentralized Observation of Discrete-Event Systems: At Least One Can Tell," in IEEE Control Systems Letters, vol. 6, pp. 1652-1657, 2022, doi: 10.1109/LCSYS.2021.3130887.
[27] P. Xu, S. Shu, and F. Lin, "Verification of Delay Co-Observability for Discrete Event Systems," in IEEE Transactions on Control of Network Systems, vol. 7, no. 1, pp. 176-186, March 2020, doi: 10.1109/TCNS.2019.2913562.
[28] M. Bozena, “Review of modelling approaches for healthcare simulation”, Operations Research and Decisions, vol. 26, no. 1, 2016, pp. 55-72, doi: 10.5277/ord160104.
[29] S. C. Brailsford, P. R. Harper, B. Patel, and M. Pitt, “An analysis of the academic literature on simulation and modelling in health care”, Journal of simulation, vol. 3, issue 3, 2009, pp. 130-140, doi: 10.1057/jos.2009.10.
[30] A. A. Tako, and S. Robinson, “The application of discrete event simulation and system dynamics in the logistics and supply chain context”, Decision Support Systems, vol. 52, 2012, pp. 802-815, doi: 10.1016/j.dss.2011.11.015.
[31] T. M. Sobh, “Discrete Event Hybrid Systems in Robotics And Automation”, Nova Science Pub. Inc, UK ed. Edition, July 18, 2005, ISBN: 978-1594544637, Available online: https://www.amazon.com/Discrete-Hybrid-Systems-Robotics-Automation/dp/1594544638 [accessed on May 15, 2022].
[32] M. Carvalho, and L. Luna, “Discrete and Continuous Simulation”, PAD 824-Advanced Topics in System Dynamics, 2002, Available online: https://www.slideshare.net/chienql/discrete-and-continuous-simulation-presentation [accessed on June 8, 2022].
[33] P. Lima, H. Gracio, V. Veiga and A. Karlsson, "Petri nets for modeling and coordination of robotic tasks," SMC'98 Conference Proceedings. 1998 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No.98CH36218), 1998, pp. 190-195 vol.1, doi: 10.1109/ICSMC.1998.725407.
[34] “Discrete Event Systems Based Robotic Task Modeling”, Intelligent Robots and Systems, Available online: https://irsgroup.isr.tecnico.ulisboa.pt/irsg_research/discrete-event-systems-based-robotic-task-modeling/ [accessed on May 16, 2022].
[35] “Discrete Event Systems for Automation”, IEEE Robotics and Automation Society, Available online: https://www.ieee-ras.org/publications/t-ase/special-issues-t-ase/discrete-event-systems-for-automation [accessed on May 16, 2022].