A Classification Framework for Time Stamp Stochastic Assignment Problems

Kees Kooiman[1], Frank Phillipson[2][a] and Alex Sangers[2]
[1]*Erasmus University, Rotterdam, The Netherlands*
[2]*Netherlands Organisation for Applied Scientific Research (TNO), Den Haag, The Netherlands*

Keywords: Assignment, Time-stamp Stochastic Assignment Problem, Classification Framework, Simulation Algorithms.

Abstract: In this paper Time-stamp Stochastic Assignment Problems are studied. These problems occur in places where an incoming request or task has to be connected to a resource immediately. A definition and framework for these problems is given, in which the different Time-stamp Stochastic Assignment Problems can be categorised. An explicit notation is introduced to distinguish the different categories. Several solution methods for Time-stamp Stochastic Assignment Problems are listed and their advantages and disadvantages are discussed.

1 INTRODUCTION

A problem that often occurs in manufacturing and logistics is assigning resources (machines, stock, employees, etc.) to tasks whose arrival time or time order of the occurrence is uncertain. When the task appears, an immediate choice has to be made which resource to assign to that task, no queues or buffers are allowed. The assignment restricts future assignments to upcoming tasks whose arrival times and types of which are unknown. This type of real-time assignment of scarce resources to an ongoing stream of stochastically appearing tasks defines a so-called Stochastic Assignment Problem (SAP). Areas of application are:

- Maintenance or upgrade of networks
- Mechanic work (electricity, printers, coffee machines)
- Real-time assigning of customers to (multi-skilled) servants
- Real-time assignment of tasks to machines
- Assignment of available kidneys to patients on a waiting list

Finding optimal or near optimal solutions to an SAP can be hard in practical applications. Planners may have a good feeling for short term effects of their decisions, however the often complex long term effects are hard to grasp for a human mind. Indeed, the hard part of optimal decision making in an SAP is to properly take into account how current assignments influence the opportunity set for the assignment of resources to uncertain future tasks appearing. Sophisticated mathematical tools might therefore be useful to help making more efficient real-time assignment decisions. These have to take into account the probability distribution of future demands on available resources.

A distinction between two types of Stochastic Assignment Problems is the following:

- Sequential Stochastic Assignment Problem (SSAP): For these problems only the order of events matter, not their exact times.
- Time-stamp Stochastic Assignment Problem (TSAP): For these problems the event times are an essential part of the specification of the problem instance.

For Sequential Stochastic Assignment Problems the exact event times are not important, but only the order of occurrence of events. One can think of patients on a waiting list for kidneys, where the kidneys sequentially become available. The assignment depends on the patient and kidney type. The assignment of a kidney to a patient is made immediately upon arrival of the kidney and an assigned kidney never re-enters the pool of available kidneys. Under these assumptions, exact times are irrelevant.

In the early study of (Derman et al., 1972), the Sequential Stochastic Allocation Model is introduced, which is used to solve a Sequential Stochastic Assignment Problem. This model will be called the DLR (named after Derman, Lieberman and Ross) model in

[a] https://orcid.org/0000-0003-4580-7521

Kooiman, K., Phillipson, F. and Sangers, A.
A Classification Framework for Time Stamp Stochastic Assignment Problems.
DOI: 10.5220/0008871701290137
In *Proceedings of the 9th International Conference on Operations Research and Enterprise Systems (ICORES 2020)*, pages 129-137
ISBN: 978-989-758-396-4; ISSN: 2184-4372

the remainder of this section. The model addresses the assignment of *n* available men (resources) to perform *n* jobs. The jobs arrive in a sequential order and the job types are independent and distributed identically. The objective is to find an allocation between the men and the jobs that maximises the total reward. An assignment of the jobs to the available men is done by creating intervals based on the distribution function of the type of the incoming jobs. The type of the new incoming job falls in a certain interval and is assigned to the corresponding man. After each assignment the intervals are recalculated. In (Derman et al., 1975) the DLR model is used for an investment problem.

Next, (Albright, 1974) investigates different arrival distributions and discount functions for the DLR model. In (Baharian Khoshkhou, 2014) practical variations and extensions of the DLR model are given. It introduces the TSSAP, a target-dependent SSAP under the threshold criterion, which attempts to minimise the probability of the total reward failing to achieve a specified target value. A practical application of the DLR model is the Aviation Security Problem described in (Nikolaev et al., 2007) and (McLay et al., 2009). In the second stage of their sequential stochastic security design problem (SSSDP) they model a policy of screening passengers that arrive at a security station.

Another well known SSAP is the kidney allocation problem. This is described using the DLR model by (David and Yechiali, 1995) and (Su and Zenios, 2005). A match has to be made between available organs (e.g., a kidney) that will arrive sequentially and the transplant patients on the waiting list. The type of the patients are known, the type of the kidney is revealed upon arrival. The reward of assigning a kidney to a particular patient depends on both the type of the kidney and the type of the patient.

The other category is the Time-stamp SAP, where time plays an important role. The planning of nurses in maternity care by (Phillipson, 2015) is an example in this category. Here a maternity care agency (MCA) has to plan nurses to help the mother when a baby is born. The challenge is to assign the right nurse to each upcoming demand, while taking into account what the impact is for future decisions, under uncertainty. To illustrate the difference between a sequential and time-stamp problem we compare this problem with the kidney allocation problem. In the maternity care problem time plays an important role. Here a time interval of a day is used. The delivery dates follow a probability distribution based on days and the mother needs care during the first 10 days after birth. Then the resource (here nurse) re-enters the pool or resources.

Other examples are the planning of known elective patients and unexpected emergency patients to Surgery Rooms (SRs) as done in (Lamiri et al., 2009) and assigning nurses to patients under stochastic scenarios as discussed in (Punnakitikashem et al., 2008). Another example is the Online Ambulance Dispatching problem in, e.g., (Jagtenberg et al., 2015) and (Ji et al., 2019), where it has to be decided which ambulance to send when an incident occurs. (Jagtenberg et al., 2015) look at a policy of sending an ambulance, out of the subset of available ambulances, that are within a target time of the incident. The core of the Ambulance dispatching problem can be seen as a Truckload Motor Carriers problem as discussed in (Powell, 1996). It concerns the problem of assigning drivers to pick up a load while the location and magnitude of the demand is random. Other types of TSAPs are weapon target assignment (Murphey, 2000), multi-robot task-allocation (Liu and Shell, 2011; Amato et al., 2015; Buckman et al., 2019), and project portfolio management (Gutjahr and Reiter, 2010). Also the stochastic task assignment problems that are a special case of the dynamic vehicle routing problem (DVRP) can fall under the Time-stamp SAP, if no buffering occurs and the allocation has to be done immediately on arrival. Examples here are ride sharing (Simonetto et al., 2019), dial-and-ride problems (Ho et al., 2018; Schilde et al., 2011) and task assignment in spatial crowdsourcing services (like Uber) (Chen et al., 2019; Cheng et al., 2017).

A slightly different type of problem is the Stochastic Dynamic Generalised Assignment Problem (SDGAP) as described in (Kogan et al., 2005). An example is the assignment problem in a copy centre at a university bookstore, having multiple machines of different types that can copy predefined ranges of page sizes. Each machine can carry out only one task at a time so the agent resource is equal to one. The task resources are defined in the problem instance. The SDGAP can be extended by dealing with a multi-resource constraint. It differs from the generalised assignment problem in that an agent consumes not just one but a variety of resources in performing the tasks assigned to him, as shown in (Gavish and Pirkul, 1991). An extension to this Multi Resource problem occurs when the resources are not being individually capacitated per agent but collectively for all agents, as can be found in (Toktas et al., 2004). As described by (Albareda-Sambola et al., 2006) Stochastic Generalised Assignment Problems can also be modelled with a two stage recourse model. In stage 1 a planning is made without stochastic information. In stage 2 resources are reassigned to jobs when tasks come in stochastically and incur penalty costs for certain reassignments.

In this paper we will focus on Time-stamp Stochastic Assignment Problems. From the brief literature review we notice that SAPs can have many different forms, but a general representation for various problems is lacking. To find out what solution method to apply, it is useful to have a categorised framework that contains the different types of problems.

In the remainder of this paper we first give a more precise description of the TSAP. Then in Section 3 we zoom in on one of those two forms and propose a framework for TSAP. Next, in literature used solution methods are presented in Section 4. We finish with some conclusions.

2 DEFINITION OF TSAP

The Time-stamp Stochastic Assignment Problem is a stochastic assignment problem in which time is important. Because of this, not only the incoming sequence of demands is important but also the time lapse between two incoming demand items. Next to the time of arrival the demand type can also be known or unknown. TSAPs can have a new incoming demand item while one or multiple previous demand items are still being processed and are still active in the problem, i.e., parallel processing of demand items. The serving time and/or arrival time of the items could be stochastic and unknown. A good prediction of the arrival time of incoming demand items and serving time of serving items is desirable. Because stochastic assignment problems have a very broad scope we redefine TSAPs as used in this paper. A Time-stamp Stochastic Assignment Problem as elaborated in this paper has the following definition. Let **demand items** i be items with different types o_i, which can be known beforehand or revealed on arrival with a stochastic arrival time τ_i and let **serving items** j be items with different known types q_j with a deterministic serving time s_{ij} (with i a demand item), where at least one type of serving item has finite availability. Then a Time-stamp Stochastic Assignment Problem is an assignment starting at $t = 0$ and ending at $t = T$ of **demand items** to **serving items**.

The following assumptions are made:

Assumption 1. *Finite time horizon* $0 \leq t \leq T$

Although this may not be entirely realistic we assume that demand items arriving after T have negligible influence on the optimal allocation at $t = 0$.

Assumption 2. *Assignments can not be retaken.*

Choices made in the past cannot be undone. However after having completed its task, the serving item may return in the pool of available serving items if this is allowed in the problem. This re-entering process is described in section 3.1.4.

Assumption 3. *At least one serving item needs to be available to serve an incoming demand.*

The assignment of demand items to serving items will be hard if there are insufficient serving items to fulfil the demand. The demand will get stacked and has to wait until serving items become available. When this happens the serving items need to be assigned to the demand items. This comes down to a queueing system and as mentioned in the introduction this is outside the scope of this study and that is why the following collateral assumption is made.

Assumption 4. *Infinite serving items are available at a penalty cost.*

In a stochastic assignment problem the main problem is to make an assignment such that it is optimal in the future; using/assigning a serving item now should be optimal afterwards. If a serving item is assigned, it can not be used during its serving time. A newly arriving demand item, that would have been a better match with an already assigned serving item, may result in a suboptimal solution. An important requirement for the stochastic assignment problem is that the cardinality of at least one of the types of serving items is finite. If this is not the case, and the number of available serving items per type are infinite the problem is trivial and for each demand item the best corresponding type of serving item can be chosen. This is translated in the following:

Assumption 5. *The cardinality of at least one of the types of serving items is finite.*

A change in serving time has a minor influence on the assignment problem. In a TSAP there is a low occupancy of serving items and there are no queues, deterministic serving times is a fair assumption. This is why we assume the serving time of the serving items are known and predefined in the problem instance.

Assumption 6. *The serving time is deterministic.*

Another important aspect concerns the possibility of a match between a specific type of serving item and a specific type of demand item. Let M be a binary matrix with rows corresponding to the different types of serving items and the columns corresponding to the different types of demand items, where $M_{ij} = 1$ if demand item type j can be served by serving item type i. Matrix M should satisfy:

Assumption 7. *Each column and row of M should contain at least one* 1.

The demand item type should at least be able to be served by one of the serving item types. On the opposite each serving item type should at least be able to serve one of the demand item types.

Assumption 8. $\sum_j M_{ij} \geq 2$*, for at least one i*

If this is not the case each type of serving item only has one possible type of demand item to serve and the problem becomes trivial.

3 A CLASSIFICATION FRAMEWORK

In this section, we derive a generic framework to classify TSAP applications in a uniform way. We introduce an explicit notation to distinguish the different categories. This not only allows us to classify TSAPs unambiguously, but it also makes clear which characteristics are common, where two problems look totally different at first.

The framework consists of four building blocks:

1. Demand arrival time;
2. Demand type;
3. Resource type;
4. Serving item feature.

Each block has a number of different options; in the framework a specific TSAP is characterised by listing the values these options take within each of the four blocks.

First we discuss the different options observed in practical TSAPs for each of the four building blocks. Subsequently we introduce notation for these options and present the framework.

3.1 Framework Elements

In this section a detailed description is given of the four components that define the framework. The different options that can occur within these components are defined.

3.1.1 Demand Arrival Time

A TSAP consists of tasks, jobs, orders or other demand items. Demand items arrive sequentially and need to be assigned immediately to a serving item. The arrival process is deterministic when the arrival times are known beforehand. Deterministic arrival times are scarce in TSAP applications but we do address them in our framework. If the arrivals are unknown or uncertain we have a stochastic arrival process. In this case we need an arrival distribution and its corresponding parameters (neglecting non parametric statistics). The distribution and its parameters can be based on theory or estimated from historical data. Ideally parameters could be updated during the process by using Bayesian methods of parameter estimation, see, e.g., (Zellner, 1996).

3.1.2 Demand Type

The demand item type can either be known beforehand or can be unknown or uncertain, in which case it is treated as stochastic.

3.1.3 Resource Type

In a classical assignment problem an assignment is a one-on-one assignment between a demand item and a serving item. Based on the type of the demand item and the type of the serving item an optimal allocation is made. Another common assignment problem is the generalised assignment problem. The difference with the classical assignment problem is that the demand item needs a specific number of resources and the serving item has a specific number of resources to contribute. If, for example, a demand item needs 8 pieces to be served, serving item *a* can provide 5 and serving item *b* the other 3. This is called an assignment problem with single item resource. It is also possible the demand items need different items to be served. As an example a demand item needs 8 blue pieces and 6 red pieces, serving item *a* can provide 4 blue pieces and 4 red pieces and serving item *b* provides the other 4 blue pieces and 2 red pieces. This is called a assignment problem with multi-item resource. In these generalised assignment problems the serving items can individually or collectively be capacitated. In the latter example it could be the case that serving item *a* is capacitated to only supply a maximum of 4 red and 4 blue pieces. It could also be that the serving items are collectively capacitated meaning that all serving items together can supply a capacitated number of resources of a specific type. In our previous example this can mean that both serving item *a* and *b* together have 8 blue pieces to supply. Although both *a* or *b* could supply 8 blue pieces, a combination is still possible.

3.1.4 Serving Item Feature

In a classic assignment problem a serving item can be used once, and after the assignment it is no longer available for further assignments. Another serving item feature that is often part of a Time-stamp Stochastic Assignment Problem is that after serving, the serving item returns in the pool of ready to use serving items and is again ready to be allocated to a demand item. In these problems the serving time plays an important role. In a few problems a serving item can switch to the new incoming demand item while already serving another demand item as in the maternity care problem, another serving item should then take over from the switched serving item.

3.2 Classification Scheme

In this section a classification framework is provided TSAPs. The basis of the framework is taken from the classification scheme of (Graham et al., 1979) for scheduling problems. The framework established in this paper uses the characteristics as the building blocks defined in the previous sections. Given an application of a TSAP the components of the modelling system specify the problem in the framework. To indicate the qualitative characteristics of a Time-stamp Stochastic Assignment Problem we propose a four-position framework of the form $\alpha \mid \beta \mid \gamma \mid \delta$. The corresponding options are given below:

Demand Arrival Time Distribution. Options for α are:

- *det* for deterministic arrival time
- *stoch* for stochastic arrival time

Demand Type Distribution. Options for β are:

- *det* for deterministic demand type
- *stoch* for stochastic demand type

Resource Specification. Options for γ are:

- *one* for one on one allocation
- *si-in* for single item and individually capacitated
- *si-co* for single item and collectively capacitated
- *mu-in* for multi item and individually capacitated
- *mu-co* for multi item and collectively capacitated

Serving Item Feature. Options for ε are:

- *once* for when a serving item can only be used once
- *reass* for serving items being re-assignable again after service
- *swit* for when a serving item can switch to another demand item during service

Part of the TSAP literature studied and summarised earlier can be framed in the four position scheme as in Table 1.

4 SOLUTION METHODS

In this section several solution methods for TSAPs are elaborated and their advantages and disadvantages are discussed. First we introduce the meaning of online and offline optimisation and their relevance to the study of TSAPs.

4.1 Optimisation Paradigms

In the field of operations research a well-known and commonly used distinction is made between *offline* and *online* optimisation. In *offline* optimisation all relevant information is known when solving the optimisation problem involved, i.e. there is no uncertainty about any input data relevant to the problem. In *online* optimisation the input data come in sequentially; decisions have to be taken while part of the relevant information is still lacking, since it will only become available after the decision has been made. So in *online* optimisation uncertainty about (part of) the relevant parameters of the optimisation problem is essential. Only *ex post*, when all information has become available, the truly optimal solution can be computed *offline*. This solution forms a bound for the *online* solution, based on the incomplete information available when the decision had to be taken.

To account for the missing information, *online* optimisation is based on an assessment of the possible future outcomes of the missing items. Actual outcomes might deviate from these imputed expected outcomes, rendering the *online* optimisation solution sub optimal *ex post*. Indeed, only in the unlikely case that all future realisations of the missing input items happen to coincide with the imputed expectations, the *online* solution will match the *offline* solution.

TSAP is a good example of an online optimisation problem. The associated *ex post* offline optimisation solution serves as a benchmark to check how well different solution methods of the TSAP perform.

4.2 Online Solution Methods

The literature review (summarised earlier) made clear that a lot of different methods have been used in finding a solution for TSAPs. Because of the different characteristics of TSAPs involved different solution methods were suitable for solving these problems. In this section an overview is given of the most common solution methods used. We start with (two stage) stochastic programming followed by Markov chain optimisation. Then a simulation method is explained and finally a description is given of rule based decision making.

Table 1: Classification of different TSAP applications.

Reference	α	β	γ	δ
Project portfolio (Gutjahr and Reiter, 2010)	det	stoch	si-in	once
Maternity care (Phillipson, 2015)	stoch	det	one	reass/swit
Job Shop Scheduling (Weber, 1982)	stoch	det	one	reass
Truckload Motor Carriers (Powell, 1996)	stoch	det	one	reass
Airport Gate Assignment (Yan and Tang, 2007)	stoch	det	one	reass
Pre-media printing (Kogan et al., 2005)	stoch	det	mu-in	reass
Container Shipping (Braekers et al., 2011)	stoch	det	si-in	once
Container Shipping (Kooiman et al., 2016)	stoch	det	si-in	once
Call Center (Tica et al., 2011)	stoch	stoch	one	reass
Surgery Planning (Lamiri et al., 2009)	stoch	stoch	one	reass
Ambulance dispatching (Jagtenberg et al., 2015)	stoch	stoch	one	reass
Nurse Assignment (Punnakitikashem et al., 2008)	stoch	stoch	one	reass
Location based mobile advertising (Spentzouris and Koutsopoulos, 2017)	stoch	stoch	one	once
Bin Packing (Coffman et al., 1980)	stoch	stoch	si-in	once

4.2.1 (Two Stage) Stochastic Programming

Mathematical programming with uncertain data in the objective function or constraints is called stochastic programming. The uncertainty is usually translated into a probability distribution on the parameters. The uncertainty can in practice be characterised by a precisely defined probability distribution or just a few scenarios (possible outcomes of the data with corresponding probability). An obvious way of dealing with this problem is using a recourse model, as in (Kools and Phillipson, 2016) and (Leenman and Phillipson, 2015). The recourse model requires that a decision is made now such that it optimises the expected objective value of the consequences of that decision. As an example we take x to be a vector of decisions that we must take now, and $y(\omega)$ is a vector of (later) decisions that correspond to the reactions to the decisions of x or the consequences of x. This way we can model a problem with the following Two Stage formulation.

The optimisation problem is:

$$max\ f(x) + E[G(y(\omega),\omega)], \qquad (1)$$

subject to

$$q_i(x) \leq 0, \qquad i = 1,...,m, \qquad (2)$$

$$h_j(x,y(\omega)) \leq 0 \text{ for } \omega \in \Omega \qquad j = 1,...,k, \qquad (3)$$

$$x \in X, y \in Y. \qquad (4)$$

The constraints (3) are the links between the decisions x for the first stage and the decisions $y(\omega)$ in the second stage. It is required that all constraints hold for each possible $\omega \in \Omega$. The model can be extended in numerous ways, e.g. by making it a multistage problem, in effect make one decision now, wait till new data is observed, and make a next decision. The implementation of a recourse model is already complicated for two stages, but the method described by (Haneveld et al., 2006) can be used to solve the optimisation problem.

In stochastic programming used for TSAPs, the optimisation solution is based on a probability average to deal with the uncertainty. Because a high number of demand items become observable one by one in a TSAP, a large number of decisions need to be made. This requires multistage programming with a large number of stages, which is even more complex to solve than the two-stage case. Because of this characteristic of TSAPs, stochastic programming does not work well. Stochastic programming is more apt for stochastics with infrequent decision making rather than frequent decision making as in TSAPs. An advantage of stochastic programming is that after each stage a future allocation or planning is made. Although in the future there may be a deviation from the planning, it could be useful as an indication of future assignments.

4.2.2 Markov Chain Optimisation

A commonly used way of tackling uncertainty in operations research problems is using Markov Chains and Markov Decision Processes. Markov Chains are widely used in queueing theory, what is a form of a TSAP, but due to earlier mentioned characteristics is outside the scope of this study. In a Markov chain all states of the problem/system are considered and their corresponding transition probabilities. With known mathematics an optimal allocation of demand and serving items can be obtained, see (Tijms, 2003). The downside of Markov chains is the computational burden of the process. If there is a rather large number of demand items and/or serving items and there is a reasonable large time window, then the number of states can get very large as this grows exponentially in these parameters. Here Machine Learning is used as solution method more and more, see for examples (Rahili et al., 2018; Spentzouris and Koutsopoulos, 2017).

4.2.3 Simulation

By repeatedly simulating the uncertain factors different scenarios are considered. When the decision is based on these simulations a robust choice can be made. With the following steps, derived from the simulation method used in (Phillipson, 2015), a simulation algorithm for a TSAP can be designed for making an *online* assignment between a demand item and one of the available serving items.

- Step 1. Observe demand item i.
- Step 2. Consider each serving item j available that is able to serve demand item i.
- Step 3. For each considered serving item j draw a set of dates (and types) for all future demand items within the time horizon, based on the corresponding distribution.
- Step 4. For each draw find the impact of assigning serving item j to demand item i by assigning the remaining simulated demand items to the remaining serving items, the impact is translated in a score for serving item j.
- Step 5. Repeat step 3 and 4 a predefined number of times for each considered serving item j.
- Step 6. The demand item i is assigned to the serving item j with the best average score.

With this method possible scenarios of yet to come demand items are considered. For each draw of future demand items the problem becomes an *offline* optimisation problem and a(n) (near) optimal assignment can be made. Repeating the simulation a large number of times results in a large set of possible scenarios being examined and a robust assignment can be made. Because the simulation is repeated multiple times a fast but accurate *offline* assignment method is needed. The simulation algorithm is a good solution method for TSAPs as it recalculates the current situation each time a demand item becomes observable. This simulation approach can be seen as a part of a Digital Twin environment for the system (Boschert and Rosen, 2016).

4.2.4 Rule based Decision Making

An easy to use but rather simple solution method is using a decision rule. With this rule the uncertainty in future events is neglected and a decision is made upon the current state of the problem. To assign an incoming demand item a rule is used based on the characteristics of this demand item, the available serving items and assignments made in the past. This method neglects uncertain information about future demands all together and is likely to be sub-optimal therefor. It can be implemented fairly easily and this is why it is used often in practice. It can be used as a benchmark to assess the gain in performance when using more sophisticated solution methods. Note the also more complex (greedy) heuristic approaches that are only based on the current, known, situation, can be placed under this category of rule based solutions.

5 SUMMARY AND CONCLUSIONS

The topic of Stochastic Assignment Problems is addressed in this paper with the intention of defining and classifying SAPs and examining its solution methods.

From the literature review we concluded that there exist a lot of different applications of SAPs but that there is no good representation of the problem characteristics. First we introduced the Time-stamp Stochastic Assignment Problem.Starting from a literature review of Stochastic Assignment Problem we have developed a comprehensive and versatile framework for classifying Time-stamp Stochastic Assignment Problems by assessing four characteristics of the problem at hand: demand type distribution, demand arrival time distribution, resource specification and serving item feature. The framework distinguishes the different types of TSAPs and their applications in a clear cut way by typing them with an explicit notation.

As to the available solution methods, four methods were identified: (two stage) stochastic programming, Markov chain optimisation, simulation, and rule based decision making. Which of these solution methods is advisable depends strongly on the characteristics of TSAPs involved. We concluded that Markov chain methods are only feasible when the number of states is small, whereas simulation methods seem to work rather well even in large problem instances, provided there is useful stochastic information about future events to be exploited.

REFERENCES

Albareda-Sambola, M., Van Der Vlerk, M. H., and Fernández, E. (2006). Exact solutions to a class of stochastic generalized assignment problems. *European Journal of Operational Research*, 173(2):465–487.

Albright, S. C. (1974). Optimal sequential assignments with random arrival times. *Management Science*, 21(1):60–67.

Amato, C., Konidaris, G., Cruz, G., Maynor, C. A., How, J. P., and Kaelbling, L. P. (2015). Planning for decentralized control of multiple robots under uncertainty. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1241–1248. IEEE.

Baharian Khoshkhou, G. (2014). *Stochastic sequential assignment problem*. PhD thesis, University of Illinois at Urbana-Champaign.

Boschert, S. and Rosen, R. (2016). Digital twin—the simulation aspect. In *Mechatronic Futures*, pages 59–74. Springer.

Braekers, K., Janssens, G. K., and Caris, A. (2011). Challenges in managing empty container movements at multiple planning levels. *Transport Reviews*, 31(6):681–708.

Buckman, N., Choi, H.-L., and How, J. P. (2019). Partial replanning for decentralized dynamic task allocation. In *AIAA Scitech 2019 Forum*, page 0915.

Chen, Z., Cheng, P., Zeng, Y., and Chen, L. (2019). Minimizing maximum delay of task assignment in spatial crowdsourcing. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)*, pages 1454–1465. IEEE.

Cheng, P., Lian, X., Chen, L., and Shahabi, C. (2017). Prediction-based task assignment in spatial crowdsourcing. In *2017 IEEE 33rd International Conference on Data Engineering (ICDE)*, pages 997–1008. IEEE.

Coffman, E. G., So, K., Hofri, M., and Yao, A. (1980). A stochastic model of bin-packing. *Information and Control*, 44(2):105–115.

David, I. and Yechiali, U. (1995). One-attribute sequential assignment match processes in discrete time. *Operations Research*, 43(5):879–884.

Derman, C., Lieberman, G. J., and Ross, S. M. (1972). A sequential stochastic assignment problem. *Management Science*, 18(7):349–355.

Derman, C., Lieberman, G. J., and Ross, S. M. (1975). A stochastic sequential allocation model. *Operations Research*, 23(6):1120–1130.

Gavish, B. and Pirkul, H. (1991). Algorithms for the multi-resource generalized assignment problem. *Management Science*, 37(6):695–713.

Graham, R. L., Lawler, E. L., Lenstra, J. K., and Rinnooy Kan, A. (1979). Optimization and approximation in deterministic sequencing and scheduling: a survey. *Annals of Discrete Mathematics*, 5:287–326.

Gutjahr, W. J. and Reiter, P. (2010). Bi-objective project portfolio selection and staff assignment under uncertainty. *Optimization: A Journal of Mathematical Programming and Operations Research*, 59(3):417–445.

Haneveld, W. K. K., Stougie, L., and Van der Vlerk, M. H. (2006). Simple integer recourse models: convexity and convex approximations. *Mathematical programming*, 108(2-3):435–473.

Ho, S. C., Szeto, W., Kuo, Y.-H., Leung, J. M., Petering, M., and Tou, T. W. (2018). A survey of dial-a-ride problems: Literature review and recent developments. *Transportation Research Part B: Methodological*, 111:395–421.

Jagtenberg, C., Bhulai, S., and van der Mei, R. (2015). An efficient heuristic for real-time ambulance redeployment. *Operations Research for Health Care*, 4:27–35.

Ji, S., Zheng, Y., Wang, W., and Li, T. (2019). Real-time ambulance redeployment: A data-driven approach. *IEEE Transactions on Knowledge and Data Engineering*.

Kogan, K., Khmelnitsky, E., and Ibaraki, T. (2005). Dynamic generalized assignment problems with stochastic demands and multiple agent–task relationships. *Journal of Global Optimization*, 31(1):17–43.

Kooiman, K., Phillipson, F., and Sangers, A. (2016). Planning inland container shipping: a stochastic assignment problem. In *International Conference on Analytical and Stochastic Modeling Techniques and Applications*, pages 179–192. Springer.

Kools, L. and Phillipson, F. (2016). Data granularity and the optimal planning of distributed generation. *Energy*, 112:342–352.

Lamiri, M., Grimaud, F., and Xie, X. (2009). Optimization methods for a stochastic surgery planning problem. *International Journal of Production Economics*, 120(2):400–410.

Leenman, T. and Phillipson, F. (2015). Optimal placing of wind turbines: Modelling the uncertainty. *Journal of Clean Energy Technologies*, 3(2):91–105.

Liu, L. and Shell, D. A. (2011). Assessing optimal assignment under uncertainty: An interval-based algorithm. *The International Journal of Robotics Research*, 30(7):936–953.

McLay, L. A., Jacobson, S. H., and Nikolaev, A. G. (2009). A sequential stochastic passenger screening problem for aviation security. *IIE Transactions*, 41(6):575–591.

Murphey, R. A. (2000). An approximate algorithm for a weapon target assignment stochastic program. In *Approximation and Complexity in Numerical Optimization*, pages 406–421. Springer.

Nikolaev, A. G., Jacobson, S. H., and McLay, L. A. (2007). A sequential stochastic security system design problem for aviation security. *Transportation Science*, 41(2):182–194.

Phillipson, F. (2015). Planning nurses in maternity care: a stochastic assignment problem. *Journal of Physics: Conference Series*, 616(1):012006.

Powell, W. B. (1996). A stochastic formulation of the dynamic assignment problem, with an application to truckload motor carriers. *Transportation Science*, 30(3):195–219.

Punnakitikashem, P., Rosenberger, J. M., and Behan, D. B. (2008). Stochastic programming for nurse assignment. *Computational Optimization and Applications*, 40(3):321–349.

Rahili, S., Riviere, B., Olivier, S., and Chung, S.-J. (2018). Optimal routing for autonomous taxis using distributed reinforcement learning. In *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 556–563. IEEE.

Schilde, M., Doerner, K. F., and Hartl, R. F. (2011). Metaheuristics for the dynamic stochastic dial-a-ride problem with expected return transports. *Computers & operations research*, 38(12):1719–1730.

Simonetto, A., Monteil, J., and Gambella, C. (2019). Real-time city-scale ridesharing via linear assignment problems. *Transportation Research Part C: Emerging Technologies*, 101:208–232.

Spentzouris, P. and Koutsopoulos, I. (2017). A stochastic optimization framework for personalized location-based mobile advertising. In *2017 15th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, pages 1–8. IEEE.

Su, X. and Zenios, S. A. (2005). Patient choice in kidney allocation: A sequential stochastic assignment model. *Operations Research*, 53(3):443–455.

Tica, S., Radojičić, V., Marković, G., and Marković, D. (2011). Modelling for evaluations of call center for public traffic and transport systems. *Technological and Economic Development of Economy*, 17(1):116–132.

Tijms, H. C. (2003). *A first course in stochastic models*. John Wiley and Sons.

Toktas, B., Yen, J. W., and Zabinsky, Z. B. (2004). A stochastic programming approach to resource-constrained assignment problems. Technical report, Stochastic Programming E-Print Series.

Weber, R. R. (1982). Scheduling jobs with stochastic processing requirements on parallel machines to minimize makespan or flowtime. *Journal of Applied Probability*, 19(1):167–182.

Yan, S. and Tang, C.-H. (2007). A heuristic approach for airport gate assignments for stochastic flight delays. *European Journal of Operational Research*, 180(2):547–567.

Zellner, A. (1996). *An introduction to Bayesian inference in econometrics*. John Wiley and Sons.