Managing Complexities in Theatre Logistics: A Proof-of-Concept Study

Roberto MONTANARI(a), Gino FERRETTI(a), Andrea VOLPI(a), Eleonora BOTTANI(a), Federico SOLARI(a), Letizia TEBALDI(a), Giorgia CASELLA(a), Claudio SUPPINI(a)

a. *Department of Engineering and Architecture, University of Parma, Viale delle Scienze, 181/A 43124 – Parma – Italy (federico.solari@unipr.it)*

Abstract: Logistics plays a vital role in all sectors, even in unconventional areas. This paper demonstrates the benefits of optimizing logistics processes in atypical contexts like the performing arts. A methodological tool was developed and implemented for managing theater productions. Each production consists of a variable number of items, adding complexity to their efficient management across multiple warehouses. The continuous development of new productions and maintenance efforts have strained logistics operations, causing inefficiencies. Valuable artworks require specific care during handling.

A customized algorithm and logistics solution were developed for the performing arts and implemented in a software tool. This latter considers planned demand, scheduling the theater season, and stochastic demand, driven by equipment rental requests, and incorporates operators' expertise and historical data for accurate modeling. A numerical case study tests the algorithm using the software, showcasing its ease of use, effectiveness, and efficiency. This paper aims at providing an hands-on experience, demonstrating the benefits of the tool.

Keywords: theatre logistics, allocation policies, process optimization

I. INTRODUCTION AND LITERATURE REVIEW

Scientific literature widely recognizes the importance of logistics in all industrial sectors and numerous articles demonstrate the technical and economic benefits resulting from the optimization of logistics processes. Supporting that, for instance, [1] presents a case study illustrating a reduction in operator downtime during subsequent picking phases resulting from the application of a decision-making process suitable for optimizing product allocation. Other studies have investigated the product allocation under specific constraints, as demonstrated by [2], and the achievement of greater process efficiencies in terms of energy consumption [3]. Additionally, [4] developed a decision support model to address the challenges of the electronic market's specific demands that make traditional allocation modes within the warehouse difficult to use.

This demonstrates the continuous effort for optimizing logistics processes' effectiveness and efficiency, depending on the context under investigation. However, specific and niche sectors exist, such as the theater industry, in which studies have focused on other, more context-specific aspects and neglected the fundamental role of optimization and management of warehouses used for storing materials and stage setups. In these sectors, the attention is often placed on how to transmit messages to the audience [5], or on the social role that certain theatrical activities own [6] (accordingly, more humanistic studies).

Often, theaters internally manufacture stage setups for the plays they stage, thus facing problems similar to production and warehouse management activities of traditional production systems. Typically, each theatrical production consists of its own setup, including a variable number of components ranging from 200 to 1200, with non-standard shapes and sizes, thus resulting in a random number of containers that must be allocated in a warehouse area.

Then, as the theatrical season changes, the demand for a specific setup will vary, and each warehouse location (container) will have its "accessibility". This means that the allocation problem becomes a significant issue that will affect the movement of containers (for small to medium-sized items) and large products in the warehouse, involving relevant costs of handling. In line with this consideration, this study illustrates the development of an optimization algorithm for container allocation in the intended storage locations.

The algorithm was entirely developed using the MS Excel^{TM} software tool and considers both the demand for use of the setup and the "accessibility" of storage allocations; the underlying logic is to determine (and suggest with priority) those handling operations that allow for the most requested setups to be positioned, at any time, in the most easily accessible allocations, given a specific warehouse layout. The algorithm itself was then tested in an important Italian theater company, implementing it within a proof-of-concept application. Thanks to the variables considered, this study demonstrates how the tool is able to provide extremely rigorous indications for each use and to easily adapt to possible variations to the warehouse layout considered.

The possible implementation of this tool within existing management applications can contribute to the integration of new features that allow for the better monitoring of logistics processes, even in very specific contexts such as theaters and in which these processes are typically perceived as unimportant.

The remainder of the paper is structured as follows: section II, for the sake of clarity and a better understanding, illustrates the nomenclature involved in the model; section III, then, details the developed optimization algorithm, followed by section IV in which the developed software tool capable of implementing the algorithm in an operational manner is presented. Section V deals with the numerical case study, followed by the last section, i.e., section VI, concluding the manuscript and proposing future research directions.

II. NOMENCLATURE

The nomenclature adopted is presented in **Errore. L'origine riferimento non è stata trovata.**.

TABLE I. NOMENCLATURE OF THE MODEL EVALUATED.

Symbol	Description
N_POS	Total number of storage positions
N_TOT	Total number of setups
N_FREE	Total number of free positions
N_OCC	Total number of occupied positions
N_CON	Total number of stored containers
$N\_CRITICITY$	Total number of criticalities detected
EXC_A	Exchange Area
$i\ (i=1, \dots, N\_POS)$	i-th storage position
$j\ (j=1,\dots,N\_TOT)$	j-th setup
$DATA\_ELAB$	Dataset containing data of N_CON allocated in N_OCC positions
$DATA\_POS$	Dataset containing only N_FREE positions
$pr(j)$	Priority of j-th setup
$val(i)$	Accessibility Index of i-th position
Δ	Criticality Index
$\lvert \Delta \rvert$	Absolute value of the Criticality Index
$\lvert \Delta_{USER} \rvert$	Absolute value of the Criticality Index defined by the user
$\lvert \bar{\Delta} \rvert$	The average absolute value of the criticalities associated with the warehouse
Δ_{opt}	Optimal value of Criticality Index used within the algorithm
N_MOV_MAX	Maximum number of movements to be made

III. OPTIMIZATION ALGORITHM

As recalled earlier in the paper, the setups of each production are contained in one or more containers, while the rotation index of each setup is closely related to its demand from the market, i.e., on the number of theaters that will request the rental of that specific setup. The greater the demand, the greater the need to move the containers containing the setup. Based on this premise, a priority value (pr(j)) ranging from 0 to 6 (integer value) can be assigned to each j-th setup. Therefore, from a mathematical point of view, in accordance with the nomenclature illustrated in the previous chapter, we have:

Eq.1 $pr(j) \in [0,\dots,6]\ \forall\ j \in [1,\dots,N\_TOT]$

Similarly, each storage location in the warehouse is characterized by its 'accessibility'. This variable can be evaluated in both subjective and objective terms; indeed, its values are based on aspects related to physical parameters (such as distances), but also on aspects related to the experience of the operators working in the warehouse. Hence, its determination was left to the operators, who had to evaluate it using again a scale from 0 to 6. Using this scale, a location difficult to be accessed will be assigned a score of 0, while a location that is extremely ease to be accessed will be assigned a score of 6. Therefore, from a mathematical point of view, the equation 2 holds:

Eq.2 $val(i) \in [0,\dots,6]\ \forall\ i \in [1,\dots,N\_POS]$

Based on these two variables, when a container is assigned to a specific location, the Δ value can be calculated as equation 3:

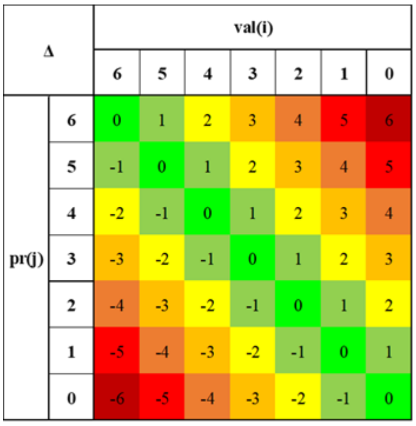
Eq.3 $\Delta(j,i)=pr(j)-val(i)$

where $\Delta(j,i) \in [-6,\dots,+6]$ based on the previous definition of the two variables.

Δ is the key variable that the algorithm takes into account, as it represents the level of consistency achieved by allocating the j-th set design into the i-th allocation. In particular, the more the value of Δ deviates from 0, the more the situation is critical, according to the description below:

- Δ < 0 means that a set with low market demand has been allocated in an easily accessible position (which would have best suited a different, more requested, item);
- Δ > 0 means, on the contrary, that a set with high market demand has been located in a position difficult to access, and hopefully it must be reallocated.

Figure 1 below summarizes the situations that could be observed in practice:



Δ		val(i)						
		6	5	4	3	2	1	0
pr(j)	6	0	1	2	3	4	5	6
	5	-1	0	1	2	3	4	5
	4	-2	-1	0	1	2	3	4
	3	-3	-2	-1	0	1	2	3
	2	-4	-3	-2	-1	0	1	2
	1	-5	-4	-3	-2	-1	0	1
	0	-6	-5	-4	-3	-2	-1	0

Figure 1 - different Δ values allowed.

Once the main parameters have been defined, the dataset DATA_ELAB was generated, including all the positions occupied by containers, with an initial value of Δ assigned. The dataset includes N_OCC rows, sorted in descending order of Δ.

The free positions are instead listed in another dataset, called DATA_POS. The exchange area (EXC_A) represents a position that cannot be occupied; rather, it has to be left empty for allowing exchange operations between two containers allocated in two different positions.

The use of |Δ| is for filtering out critical issues to be resolved within the layout.

Based on both the criticality value $|\Delta_{\text{USER}}|$ defined by M_Users and the maximum number of movements allowed (N_MOV_MAX), the algorithm identifies all the criticality conditions |Δ| equal to or greater than $|\Delta_{\text{USER}}|$ and suggests a number of movements equal to or lower than the N_MOV_MAX, so as to solve as many criticalities as possible. Leveraging (and tuning) the N_MOV_MAX parameter, this algorithm can be used in situations in which constraints exist about the time slot available for operators to carry out handling activities. It should also be emphasized that while the presence of a minimum |Δ| value is mandatory for the algorithm to work well, this is not needed for the case of N_MOV_MAX. More precisely, the algorithm could work even if a maximum number of movements has not been set; in this case, the algorithm will search for a possible solution for all critical situations regardless of the number of movements required to solve the criticalities.

The algorithm follows two distinct logics for solving critical situations, i.e., in order of priority:

1) It searches among the free positions in the warehouse, for a position having Δ=0. This allows for immediately solving the criticality;
2) It searches among the critical situations, to find criticalities that can be solved by exchanging two containers using the EXC_A. This means that both containers will have Δ=0 after the movement.

In case neither of the above logics returns a positive result, the algorithm proceeds by searching for a position that allows for minimizing (although not zeroing) the value of Δ, again following the two abovementioned logics. The process is repeated through iterations until (termination conditions):

- it is no longer possible to get a Δ value lower than the initial one; or
- the number of movements required to solve the criticality reaches N_MOV_MAX.

A graphical representation of the algorithm flowchart is shown in **Errore. L'origine riferimento non è stata trovata.**.

As can be seen in the following Figure 3, at the beginning, the algorithm tries to find a solution by applying logic #1. If this logic fails, logic #2 is implemented.

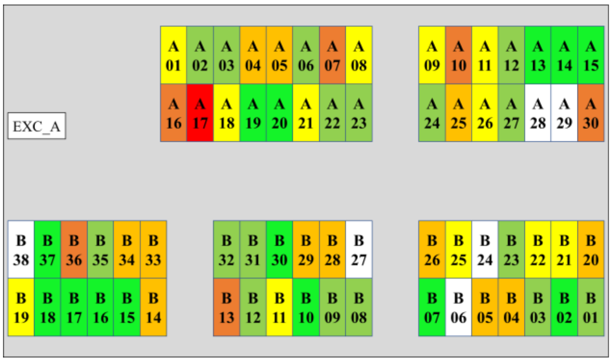


Figure 3 - An hypothetical warehouse floor plant layout in an AS-IS Scenario.

We chose to proceed in this way because prioritizing the usage of logic #1 would always save (at least) two movements. Indeed, with this logic, a container is moved directly to a more consistent position, thus immediately solving the criticality. On the contrary, if logic #2 is implemented, an exchange is made between two positions using EXC_A. EXC_A has to be left empty to allow for the switch of the containers, and thus, three handling operations are performed to resolve two critical situations; one of these operations is mandatory for having the EXC_A empty, but obviously it does not add value to the process, and in any case, does not (by itself) solve the critical situation.

The algorithm described above has been implemented for testing purpose in a proof-of-concept logistics scenario of an important Italian theater. In the following sections, the developed software tool capable of implementing the algorithm in an operational manner is presented. Furthermore, we will demonstrate its application through a numerical case study.

IV. SOFTWARE

The algorithm has been implemented in a proof-of-concept scenario of a warehouse management application, designed specifically for the context of a prominent Italian theater.

First and foremost, two distinct categories of operators have been considered when setting up the application: “managers” (M_Users) and “ordinary operators” (O_Users). M_Users have access to all the functionalities of the application, while O_Users can utilize the software to perform correct movements and update necessary records only. The main features of the application are reported in TABLE II.

TABLE II. SOFTWARE TOOLS

Features	M_User	O_Users
Records management	Yes	No
Warehouse management	Yes	No
What-If Analysis	Yes	No
Movements Management	Yes	Yes

The application package has been entirely developed using MS Excel^{TM}, and the input data used to check its functionality are based on a hypothetical situation that could arise within the warehouses of the targeted theater.

A M_User can modify all the records and has the following functions available: enabling new users to use the tool by registering them; modifying the values of pr(j) for each setup and of val(i) for each position. In the same way, it is also possible to add new positions and setups. To support the M_Users in monitoring the allocations, the application provides the warehouse floor plans that depict the positions and the setups allocated to them. A color-coded scale represents the magnitude of the $|\Delta|$ value for each position setup. Hence, M_Users have the possibility to assess the initial floor plan by visualizing the current situation. Subsequently, upon issuing an order for movement, the software enables the user to observe the resulting TO-BE scenario, which accurately reflects the confirmed changes.

In the previous Figure 3, we could observe a sample warehouse floor plan in a not optimized scenario (AS-IS scenario). It is important to note that the colors used in this figure are consistent with those depicted in Figure 1 and correspond to the same values of Δ.

The optimization algorithm has been integrated into the "Warehouse Management" tool, which serves as a comprehensive solution for warehouse operations. Within the what-if analysis tool, the algorithm is employed to design the TO-BE scenarios (as it will be seen in the section dedicated to the case study, specifically in the future Figure 4), providing insights about the possibility of optimizing the state of the warehouse.

M_Users only can exploit the what-if tool, which allows them to test up to three different optimization conditions simultaneously. They can define the minimum value of $|\Delta|$ as the starting point for addressing critical issues and set the maximum number of allowed movements(N_MOV_MAX).

To assist M_Users, the software provides an overview of the current state of the warehouse, including a frequency table that presents statistical information regarding existing critical issues.

By launching the algorithm for each scenario, an average criticality value is determined, along with a table illustrating the criticalities across the warehouse.

Once the M_Users have identified the most favorable scenario, by the “Movements management” tool the O_Users can convert this final selection into specific movements that need to be executed. Upon confirmation, the system updates the corresponding records accordingly.

Within the "Movements Management" tool, O_Users have the ability to directly execute the movements within a selected scenario, as designated by the manager. This interface is accessible to all the registered O_Users, regardless of their role within the organization.

This tool presents a complete overview of the current state of the warehouse, including the existing allocations and available positions. Moreover, it highlights the average value of |Δ|, thus allowing users to gauge the current level of criticality.

At this stage, O_Users have the possibility to specify a minimum value for |Δ| and N_MOV_MAX. Leveraging the scenario determined by M_Users, the software provides suggestions for the movements to be executed, prioritizing those with the highest level of criticality.

Once the operator makes a selection, the intended movements can be confirmed. By clicking the confirmation button, the software issues the movement orders in a paper-based format, as explicitly requested. After having executed these movements, the operator can mark their completion within the software, triggering automatic updates to the data. This ensures that the information system accurately reflects the current state of the warehouse. All these activities are recorded in a backup copy to keep a complete log of the operations and facilitate the resolution of discrepancies, if necessary.

V. CASE STUDY

The case study consists of two allocation zones, called A and B, and an exchange station (EXC_A). Zone A hosts 30 allocations, while Zone B 38, totaling 68 available storage locations. Out of these, 62 positions are occupied and in the AS-IS scenario exhibit high criticality. This can be observed from TABLE III and in the color-coded representation of Figure 3.

The level of criticality can also be quantified using the average absolute value of the criticalities, defined as $|\bar{\Delta}|$, which equals 1.71, as calculated using equation 4, illustrated below

Eq. 4 $$|\bar{\Delta}|=\frac{\sum_{1}^{N\_POS}|\Delta|}{N\_POS}$$

TABLE III - FREQUENCY AND PROBABILITY DISTRIBUTION OF Δ INTO THE TWO DIFFERENT SCENARIOS.

	AS-IS		TO-BE	
Δ	**Frequency**	**Pdf (Δ) [%]**	**Frequency**	**Pdf(Δ) [%]**
6	*0*	*0,00*	*0*	*0,00*
5	*1*	*1,60*	*0*	*0,00*
4	*3*	*4,80*	*0*	*0,00*
3	*7*	*11,30*	*0*	*0,00*
2	*4*	*6,50*	*6*	*9,70*
1	*6*	*9,70*	*8*	*12,90*
0	*14*	*22,60*	*27*	*43,50*
-1	*11*	*17,70*	*13*	*21,00*
-2	*8*	*12,90*	*8*	*12,90*
-3	*5*	*8,10*	*0*	*0,00*
-4	*3*	*4,80*	*0*	*0,00*
-5	*0*	*0,00*	*0*	*0,00*
-6	*0*	*0,00*	*0*	*0,00*
Total	*62*	*100,00*	*62*	*100,00*

To reduce the warehouse's criticality by addressing the most severe conditions, the developed software was utilized with the algorithm parameters set to |ΔUSER| = 3 and N_MOV_MAX = 30. This trained the system to minimize criticalities with |Δ| greater than 3 while limiting the maximum number of movements to 30.

By employing the developed tool, it was possible to identify and manage 19 highly critical issues, through 27 movements: 13 issues were eliminated and their Δ was set at 0, while in the remaining 6 issues a strong reduction of Δ was obtained. This resulted in a significant improvement, both qualitatively (as evident by the color-coded representation in Figure 4, in the next page) and quantitatively, as indicated by the values in TABLE III (TO-BE configuration).

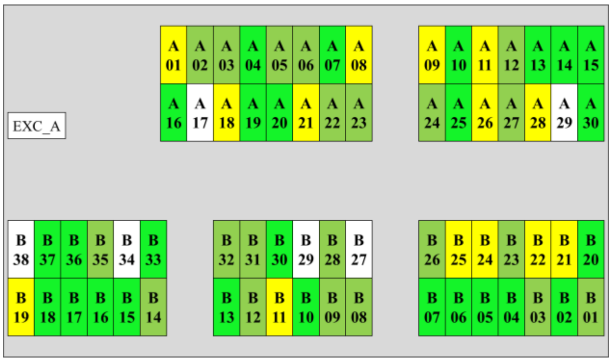


Figure 4 - Warehouse layout of the TO-BE Scenario.

Overall, the criticalities were reduced by 53.8%, leading to an $|\bar{\Delta}|$ value of 0.79 in the TO-BE scenario.

VI. CONCLUSION

Logistics has consistently played a crucial role in various industrial fields, including seemingly unrelated sectors like that of theater and entertainment. This study aims to develop and implement an algorithm as a proof-of-concept within a software platform dedicated to managing a diverse portfolio of one hundred theatrical productions for a prominent company.

By assigning dynamic priority values to each production based on market demand and proportional values to storage locations considering their accessibility during handling operations, the research demonstrates the feasibility of devising an allocation policy that minimizes warehouse transfers (containers) and optimizes storage positions for each production.

The algorithm has been successfully integrated into a proof-of-concept warehouse management software tool. In addition to its primary function of triggering movement orders, the algorithm also facilitates preliminary what-if analyses, providing valuable insights to support decision-making processes. This versatility underscores the potential benefits of utilizing the same tool for multiple purposes.

To exemplify the efficacy of the developed software, a numerical case study was presented, showcasing its practical application in a realistic scenario.

As we move forward, it would be beneficial for future research endeavors to delve into quantifying the economic advantages achieved through the utilization of the proposed tool for optimizing logistical processes. This investigation would enable a more detailed assessment of the tangible benefits that can derive from employing the tool in practice. Furthermore, exploring the effects of diverse layout management strategies on the performance of the software would provide additional valuable insights, enhancing our knowledge of its capabilities and identifying potential areas for refinement.

VII. REFERENCES

[1] Alqahtani, A. Y. (2023). Improving order-picking response time at retail warehouse: a case of sugar company. *SN Applied Sciences*, *5*(1). https://doi.org/10.1007/s42452-022-05230-6

[2] Mendes, N. F. M., Bolsi, B., & Iori, M. (2022). An Iterated Local Search for a Pharmaceutical Storage Location Assignment Problem with Product-cell Incompatibility and Isolation Constraints. *International Conference on Enterprise Information Systems, ICEIS - Proceedings*, *1*. https://doi.org/10.5220/0011039000003179

[3] Roshan, K., Shojaie, A., & Javadi, M. (2019). Advanced allocation policy in class-based storage to improve AS/RS efficiency toward green manufacturing. *International Journal of Environmental Science and Technology*, *16*(10). https://doi.org/10.1007/s13762-018-1921-6

[4] Shah, B., & Khanzode, V. (2018). Designing a lean storage allocation policy for non-uniform unit loads in a forward-reserve model: An enterprise information management with an e-decision support system. *Journal of Enterprise Information Management*, *31*(1). https://doi.org/10.1108/JEIM-01-2017-0018

[5] Finch, A. (2021). Representing victims and offenders in contemporary performance: the ideal and the complex in Shakespeare’s Measure for Measure. *Law and Humanities*, *15*(2). https://doi.org/10.1080/17521483.2021.1983173

[6] Reason, M. (2019). A prison audience: women prisoners, Shakespeare and spectatorship. *Cultural Trends*, *28*(2–3). https://doi.org/10.1080/09548963.2019.1617929

Appendix A. IMAGE ON 2 COLUMNS

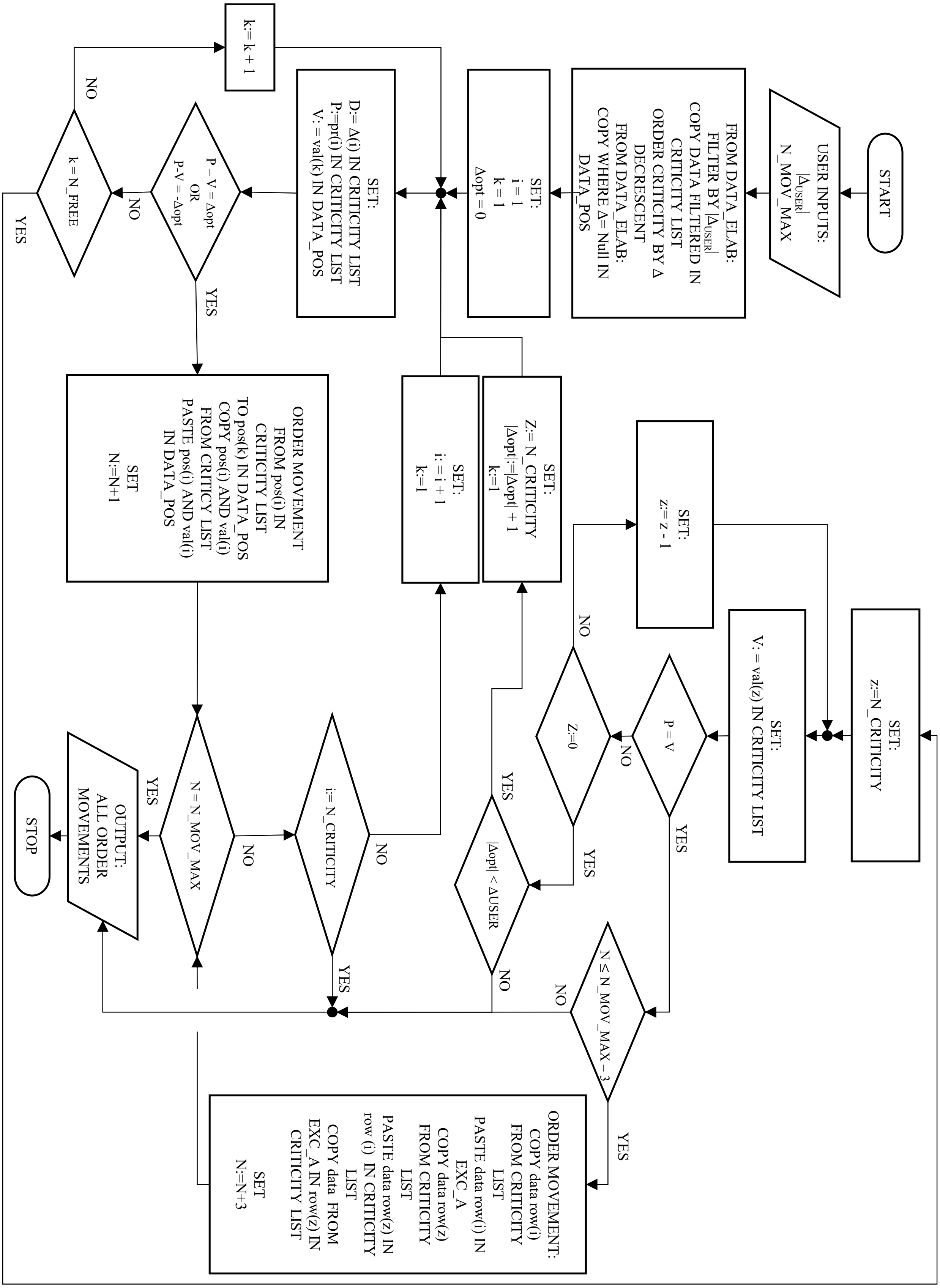


Figure 2 – Algorithm flowchart