Les 7,8 et 9 octobre 2020
à l'Institut Supérieur D'Études Logistiques - ISEL
Université Le Havre Normandie
13èmes Rencontres Internationales de la Recherche en Logistique et en Supply Chain Management

Constructing Action Plans Based on Correlation Between Sequential Actions and Their Performance in Logistics Distribution Networks

Markus Rabe* [1], Majsa Ammouriova † [1]

[1] Department IT in Production and Logistics, TU- Dortmund University – Leonhard-Euler-Str. 5, 44227 Dortmund, Germany

Purpose of the communication: Management of logistics distribution networks is a challenging task for decision-makers. In order to assist them, logistics assistance systems have been developed, which are decision support systems for logistics. Logistics assistance systems can integrate a simheuristic approach that combines simulation and metaheuristics to optimize the performance of the logistics distribution network. Simulation models the complex system of the distribution network and metaheuristics optimizes this network. In the iterative optimization within the metaheuristic algorithm, the metaheuristic forms promising solutions and the simulation evaluates the solution based on the performance of the distribution network. The logistics assistance system recommends solutions in the form of action plans that consist of actions, e.g., increase the stock level of a stock keeping unit (SKU) or centralize it at a specific network node. This paper studies the correlation between sequential actions in the action plan and their impact on the performance of a logistics distribution network, e.g., cost and service level. It aims to utilize the correlation in the simheuristic approach in the logistics support system to improve the optimization method for the distribution network.
Research design, methodological approach: This research defines a correlation relation between sequential actions and their impact on the performance of the network. Then, a metaheuristic, evolutionary algorithm in the simheuristic approach is updated to utilize the correlation in the exploration of search space and action plan construction.

Results obtained: Analyzing sequential actions reveals that the impact of an action can depend on the previously applied actions on the distribution network. Utilizing this relation in an example showed a reduction in the number of iterations needed to find promising solutions.

Theoretical contributions: This research provides a relation between sequential actions and their impact on the performance of a logistics distribution network.

Managerial contributions: Theoretical contributions define action relations that can be used to improve the decision making in a logistics distribution network.
Limitations: The paper's scope is limited to distribution networks in material trading. The research focuses on actions in a logistics distribution network in material trading that consists

*Corresponding author: markus.rabe@tu-dortmund.de
†Speaker

of sites where items are stored as SKUs. The optimization solution is in the form of an action plan that specifies the actions to be applied to the distribution network as well as their order. The preliminary experiments are limited to examples and the method needs to be verified using a real-world case study.

Keywords: Distribution networks, Logistics assistance system, Domain specific information, Evolutionary algorithm

<u>CONSTRUCTING ACTION PLANS BASED ON CORRELATION BETWEEN SEQUENTIAL ACTIONS AND THEIR PERFORMANCE IN LOGISTICS DISTRIBUTION NETWORKS</u>

"work in progress"

<u>Purpose of the communication</u>: Management of logistics distribution networks is a challenging task for decision-makers. In order to assist them, logistics assistance systems have been developed, which are decision support systems for logistics. Logistics assistance systems can integrate a simheuristic approach that combines simulation and metaheuristics to optimize the performance of the logistics distribution network. Simulation models the complex system of the distribution network and metaheuristics optimizes this network. In the iterative optimization within the metaheuristic algorithm, the metaheuristic forms promising solutions and the simulation evaluates the solution based on the performance of the distribution network. The logistics assistance system recommends solutions in the form of action plans that consist of actions, e.g., increase the stock level of a stock keeping unit (SKU) or centralize it at a specific network node. This paper studies the correlation between sequential actions in the action plan and their impact on the performance of a logistics distribution network, e.g., cost and service level. It aims to utilize the correlation in the simheuristic approach in the logistics support system to improve the optimization method for the distribution network.

<u>Research design, methodological approach</u>: This research defines a correlation relation between sequential actions and their impact on the performance of the network. Then, a metaheuristic, evolutionary algorithm in the simheuristic approach is updated to utilize the correlation in the exploration of search space and action plan construction.

<u>Results obtained</u>: Analyzing sequential actions reveals that the impact of an action can depend on the previously applied actions on the distribution network. Utilizing this relation in an example showed a reduction in the number of iterations needed to find promising solutions.

<u>Theoretical contributions</u>: This research provides a relation between sequential actions and their impact on the performance of a logistics distribution network.

<u>Managerial contributions</u>: Theoretical contributions define action relations that can be used to improve the decision making in a logistics distribution network.

Limitations: The paper's scope is limited to distribution networks in material trading. The research focuses on actions in a logistics distribution network in material trading that consists of sites where items are stored as SKUs. The optimization solution is in the form of an action plan that specifies the actions to be applied to the distribution network as well as their order. The preliminary experiments are limited to examples and the method needs to be verified using a real-world case study.

Keywords: Distribution networks; logistics assistance system; domain-specific information; evolutionary algorithm.

1. INTRODUCTION

Decision making in logistics distribution networks is a challenging task. A logistics distribution network consists of sites, suppliers, and customers (Christopher, 2011). Sites are supplied with a variety of items by suppliers, and the items are stored as stock keeping units (SKUs). Customers receive the products according to their placed orders. The performance of a logistics distribution network is evaluated by a variety of performance measures that can be contradictory (Rushton, Croucher, & Baker, 2010), such as costs and service level.

In order to support decision-makers, decision support systems are used (Heilala et al., 2010). For this purpose, a model of the logistics distribution network is required (Riddalls, Bennett, & Tipi, 2000). Mathematical models or simulation models can be used. These models mimic a real logistics distribution network and can be used to evaluate the performance measures of the network after applying a variety of changes (Law, 2015). Logistics distribution networks are complex; thus, simulation tools, such as discrete event simulation, have been successfully used to model this kind of networks.

Optimization algorithms like metaheuristics have been developed to optimize the performance of systems (Talbi, 2009). Complex systems can be optimized using simheuristics (Juan, Faulin, Grasman, Rabe, & Figueira, 2015). Simheuristics combine simulation and metaheuristics, in which simulation is used to model complex systems (Juan & Rabe, 2013). As a result, decision support systems based on a simheuristic approach have been developed to support decision-makers to optimize complex systems. If the decision support system optimizes a logistics network, the system is called a logistics assistance system (LAS) (Liebler, Beissert, Motta, & Wagenitz, 2013). Rabe, Dross, Schmitt, Ammouriova, and Ipsen (2017) developed such a system, in which the optimization of the logistics networks is formulated as an NP-hard combinatorial optimization problem. The search space consists of elements presented as actions, e.g., increase the stock level of an SKU at a site. The size of the search space associated with a logistics distribution network is significantly ample and increases as the size of the logistics network increases. A large search space triggers a large number of iterations in the used optimization algorithms. As a result, there is – similar to comparable systems – a long response time because of the logistics network complexity. Some approaches have been developed to reduce the size of the search space of optimization problems (Ku & Arthanari, 2016; Rabe, Schmitt, & Ammouriova, 2018b).

This paper introduces an approach to guide the optimization algorithm in selecting a solution's elements without reducing the original search space. The approach aims in constructing more-promising solutions within a lower number of algorithm's iterations. The paper is organized as follows: Related work is summarized in Section 2. Section 3 defines an approach to improve the performance of the optimization algorithms. Section 4 presents the utilization of the approach in constructing solutions. Next, an example utilizing the approach is discussed in Section 5. The paper closes with an outlook.

2. OPTIMIZATION OF LOGISTICS DISTRIBUTION NETWORKS

Managing logistics distribution networks is a complex task (Christopher, 2011; Rushton et al., 2010). The complexity raises from the uncertainties, the interrelations between elements in the logistics distribution network, the contradiction between the performance measures of the networks, and the variety of decisions. The performance measures evaluate the logistics distribution network's performance and can be used to compare its performance to prior periods or studying the impact of changes applied to the network (Ghiani, Laporte, & Musmanno, 2013). Costs and service level are examples of these performance measures. The complex problem cannot be formulated using a mathematical equation and solved by exact algorithms.

2.1. Optimizing logistics distribution networks using logistics assistance systems

A simplified architecture of a LAS developed by Rabe et al. (2017) is shown in Figure 1. In this LAS, the optimization problem is the optimization of a logistics distribution network in material trading by minimizing costs and maximizing a service level. The LAS is based on a simheuristic approach that combines simulation and metaheuristics (Juan et al., 2015; Juan & Rabe, 2013). In simheuristics, the simulation is used to model a system and to evaluate the objective function value of an optimization problem. Metaheuristics are used to optimize the system.

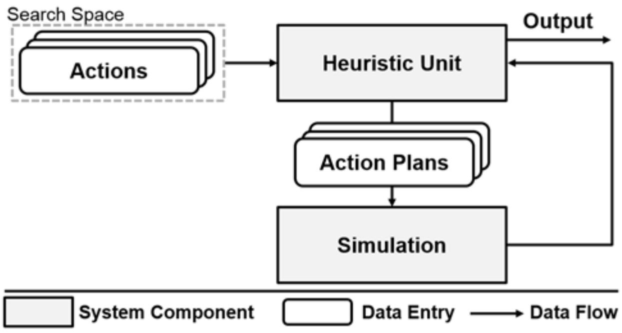


Figure 1: A simplified architecture of the developed LAS based on Rabe et al. (2017).

The logistics distribution network is modeled using data-driven discrete event simulation (Rabe et al., 2017). The structure of the network is stored in a database that includes the parameters of the network model. Changes to the network can be applied using SQL statements. These changes are associated with selected actions from the search space. An action presents an activity, such as “increase the stock level of SKU 1 in site A” or “centralize SKU 1 in site A”. The Heuristic Unit (HU) stores a library of metaheuristics algorithms. The metaheuristic algorithm selects actions and forms action plans. Actions are arranged in an action plan that also presents their order to be applied on the network. The impact of an action plan on the performance of the network is evaluated using simulation. For example, costs and service levels are considered performance measures in logistics distribution networks (Rushton et al., 2010). The performance measures are read from the database and are used by the metaheuristic algorithm to explore other actions and to form new action plans. The most-promising action plan is recommended after a termination criterion is met, such as reaching a maximum number of iterations in the HU or reaching stagnation.

One of the metaheuristics algorithms in the HU in the LAS is an evolutionary algorithm to construct solutions and explore the search space. The evolutionary algorithms solve problems with complex search spaces (Bonyadi, Michalewicz, Wagner, & Neumann, 2019). The development of evolutionary algorithms has been inspired by the natural evolution process (Talbi, 2009). The population in the algorithm consists of individuals that represent solutions of an optimization problem, which are in our case action plans in the LAS. The operators in the algorithms mimic those in the evolution process: selecting the fittest individuals, crossover, and mutation. The individuals are evaluated with respect to a fitness function that presents the objective function of the optimization problem. Forming subsequent generations is based on the selection of the fit individuals. New individuals are formed by crossover and mutation of the selected individuals. In the crossover, solution parts are exchanged between the selected individuals by defining one or more crossing points, such as one-point cross over or two-point crossover. In the mutation, a value is substituted by another value in an individual.

In the LAS, the actions are defined with the help of action types (Rabe et al., 2017). An action type is a generic description of the actions, e.g., “increase the stock level of an SKU in a site” without specifying the SKU and the site. Actions are derived from an action type by adding parameters to it, such as SKU 1 and site A. An action type’s definition includes the action type’s name, its description in natural language, required information for the execution and derivation of actions, and placeholders for the input parameters (Rabe et al., 2017). The information for the derivation and execution of actions from an action type is presented by SQL statements required to inquire the database and apply changes associated with the actions. Additionally, the action type’s definition includes domain-specific information that aims to improve the performance of optimization algorithms, such as success and type of changes (Rabe, Ammouriova, & Schmitt, 2018a). Additionally, a correlation between actions as domain-specific information is discussed in this paper.

2.2. Improving the performance of optimization algorithms

Researchers have studied the performance of optimization algorithms (Beiranvand, Hare, & Lucet, 2017; Talbi, 2009). The performance of optimization algorithms can be measured by their efficiency and the quality of found solutions (Beiranvand et al., 2017). The number of the algorithm’s iterations and the number of objective function evaluations express the algorithm’s efficiency. Additionally, researchers have studied approaches to increase the performance of the optimization algorithms (Bode, Reed, Reuschen, & Nowak, 2019; Karimi, Isazadeh, & Rahmani, 2017; Lokman, 2019). They aimed to reduce the number of iterations to find promising solutions. For example, an abstraction method to reduce the size of the search space has been used by Ku and Arthanari (2016) to reduce the number of algorithm’s iterations. Gomes and Saraiva (2016) studied elements to be included in a solution and construct good quality solutions. Sitek, Bzdyra, and Wikarek (2016) and Drugan (2018) reduced the size of the studied problem by simplifying it; they reduced the number of objective functions.

Amaran, Sahinidis, Sharda, & Bury (2016) stated that using knowledge about the optimization problem improves the performance of the optimization algorithms, especially when simulation is used. An example of such knowledge is domain-specific information. This information presents specific input about the optimization problem and its variables. Rabe et al. (2018a) used success and the type of actions (the type of changes) to improve the performance of an evolutionary algorithm in optimizing a logistics distribution network.

In this paper, the authors aim to introduce other domain-specific information. They define a correlation relation between actions. Correlation defines an existing relationship between variables (Sheskin, 2011). The authors illustrate a theoretical example to demonstrate their impact on the performance of the optimization algorithm.

3. DETERMINATION OF CORRELATION OF ACTIONS

A correlation relation refers to the relation between two variables. In this paper, the relation is related to the impact of two sequential actions on the considered performance measure of the logistics distribution network. This section defines the correlation relation between actions and action types in the LAS.

3.1. Definition of correlation between actions

The correlation relation focuses on two sequential actions, e.g., a_i and a_j, and defines their impact on the performance of a distribution network if applied sequentially. To define the relation, the impact of the sequential actions, $R([a_i, a_j])$, is compared with the expected impact of the single actions, $a_i$and a_j. The expected impact assumes the independence of the actions and is calculated as the summation of the impact of a_i and the impact of a_j, $R(a_i) + R(a_j)$. The relationship can be a positive $(+)$, a negative $(-)$, or a weak $(\sim)$ relation. The positive relation indicates that applying two sequential actions has a positive impact on the performance of the distribution network compared to the impact of the single considered actions, $R([a_i, a_j]) > R(a_i) + R(a_j)$. For example, applying a_1 followed by a_2, $[a_1, a_2]$, increased the costs reduction compared to the partial reductions after applying a_1 or a_2, only (Figure 2). The reduction in costs became 400 € compared with the expected reduction of 200 € when summing up the reductins for a_1 (50 €) and a_2 (150 €).

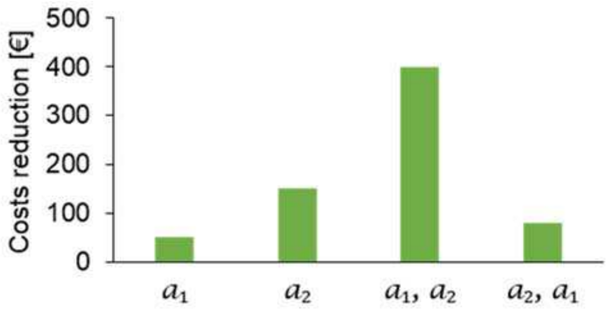


Figure 2: An impact of actions on the reduction of costs in a logistics distribution network.

On the other hand, the negative relation indicates a deterioration in the performance of the network. For example, applying a_1 after a_2 did not reduce the costs as the expectation from applying both a_1 and a_2 (Figure 2). The costs were reduced by 80 €, while the expectation was 200 €. Hence, the relation between a_2 followed by a_1, $[a_2, a_1]$, and the performance of the network is a negative relation. If the relation between the two considered sequential actions could not be classified neither as a positive nor as a negative relation, it is called a weak relation, in which no change in the performance happens. The relation between actions can be presented in a correlation matrix, in which the row presents the first applied action and the column presents the subsequently applied action, such as the correlation matrix in Table 2 in the Appendix.

The definition of the correlation relation between actions is assigned based on literature recommendations. For example, the stock level of an SKU in a site should be increased if the site became the centralized site (Rushton et al., 2010). Additionally, experiments can be conducted to study the relationship between actions, such as experiments on simulation models. An action is applied to the simulation model, and the performance of the model regarding the performance measures is recorded. Other actions are applied, and their impact on the performance measure is recorded. Then, sequential actions are applied, and their impact on the performance measures is compared to the performance measure of the single applied actions to determine the relation. In stochastic models, more than one comparison is needed to determine the correlation relation.

3.2. Defining the correlation between action types in the LAS

Since the actions are derived from action types in the LAS, the correlation relations are added to the definition of the action types. The correlation relation in the action types' definition is generalized and is associated with specific types of action pairs.

In order to determine the correlation relation between action types, the relation between actions derived from the two action types is studied. The studied actions affect the same entities in the logistics distribution network, e.g., SKU 1 in site A. For example, to study the relation between "centralize" and "increase the stock level" action types, the studied pairs of actions are: "centralize SKU 1 in site A" and "increase the stock level of SKU 1 in site A" as a pair of actions, and "centralize SKU 1 in site B" and "increase the stock level of SKU 1 in site B" as another pair of actions. When comparing the effect of the paired actions derived from these action types, a conclusion about the relationship is drawn. If most of the action pairs possess a positive relation, then the relationship of the action type pair is considered a positive relation.

Actions derived from the action type possess the defined attributes of their respective action types. For example, action type AT^1 has a positive relation with action type AT^2, and a negative relation with action type AT^3. Thus, a_1^1 derived from AT^1 is expected to have a positive relation with a_1^2 and a negative relation with a_1^3, as long as each pair of related actions affects the same entity in the logistics distribution network.

The definition of the relation between the two action types excludes the relation between actions affecting different entities, such as an action affecting SKU 2 in site A, and an action affecting SKU 2 in site C. These actions are considered independent, and accordingly $R([a_i, a_j]) = R(a_i) + R(a_j)$. Thus, the derived action pairs that affect different entities are considered weakly correlated. The relationship between these actions is not considered in the definition of the action type; the relations between these actions are mostly weak or difficult to predict because of the complex interactions between the elements in the network. Additionally, experiments for the determination of the relation between these action pairs can be time-consuming because of the enormous number of action pairs' combinations required to compare. For example, two action types are considered, and four actions are derived from each action type. Eight actions are derived and the number of action pairs to examine becomes 56 pairs

(8 × 7) considering the permutation of the actions in the action pair – this number of action pairs increases as the number of derived actions from action types increases.

As a result of the defined correlation between action types, a correlation matrix between the action types can be constructed (Table 1). The diagonal cells in the matrix present the duplication of an action. The duplication of an action can enhance the performance, deteriorate it, or have no effect. For example, duplication of "centralize an SKU in a site" does not change the performance of the logistics network: The costs of a logistics distribution network might be reduced when "centralize SKU 1 in site A" is applied, but duplicating the action does not cause further reductions. The expectation of the duplication becomes the doubeled reduction, and thus, the duplication of this action is a negative relation, as the effect of the sequential actions is only half of the effects' sum of the single actions.

	AT^1	AT^2	AT^3	AT^4
AT^1	−	+	+	~
AT^2	+	+	~	−
AT^3	+	~	−	−
AT^4	~	−	−	+

Table 1: An example of a correlation matrix between action types.

For an action derived from an action type, actions are classified into positively related actions (A^+), negatively related actions (A^-), and weakly related actions ($A^{\sim}$). Based on Table 1, action a_1^1 derived from AT^1 classifies actions as follows: a_1^3 that affects the same entity as a_1^1, and a_1^1 as A^+; a_1^2 as A^-; and all other actions as $A^{\sim}$.

4. UTILIZING CORRELATION IN SOLUTION CONSTRUCTION

After identifying the correlation between the action types, the LAS can classify actions into positively, negatively, or weakly correlated actions based on a selected action. Accordingly, the metaheuristic algorithm in the HU is adapted to utilize the classification of actions relations. In the algorithm's initial iteration, the algorithm constructs action plans by selecting actions and adding them to an action plan. Later, new action plans are formed based on the action plans constructed in the previous iterations.

4.1. Constructing action plans utilizing correlation relations

In the HU, the action plans are constructed in the initial iteration of the algorithm run. A solution is constructed by selecting an action at a time. To incorporate the correlation in the construction of action plans, several approaches can be adapted as follows: approach (1) that classifies actions based on the last added action to an action plan; approach (2) that updates the classification of actions after adding a new action to an action plan; approach (3) that classifies actions based on all actions in an action plan; approach (4) that updates the actions' selection probability based on the last added action to an action plan; and approach (5) that updates the actions' selection probability based on all actions in an action plan.

4.1.1. Approach (1): Classification of actions based on the last selected action

In correlation approach (1), actions are classified based on the last added action to an action plan. Once the first action in the action plan is selected, the selected action classifies actions as: $A^{+}_{a^{[1]}}$, $A^{\sim}_{a^{[1]}}$ and $A^{-}_{a^{[1]}}$. $A^{+}_{a^{[1]}}$ presents the positively related actions to $a^{[1]}$, where $a^{[1]}$ presents the first selected action in an action plan. Each class is given a probability to select from; higher selection probability is given to $A^{+}_{a^{[1]}}$. The lowest selection probability is assigned to $A^{-}_{a^{[1]}}$. After selecting the class, an action is selected randomly from it. The probability of selecting an action from the class is uniformly distributed. Figure 3a shows an example of the classification of actions based on $a^{[1]}$ that is presented as a_{10} in the example.

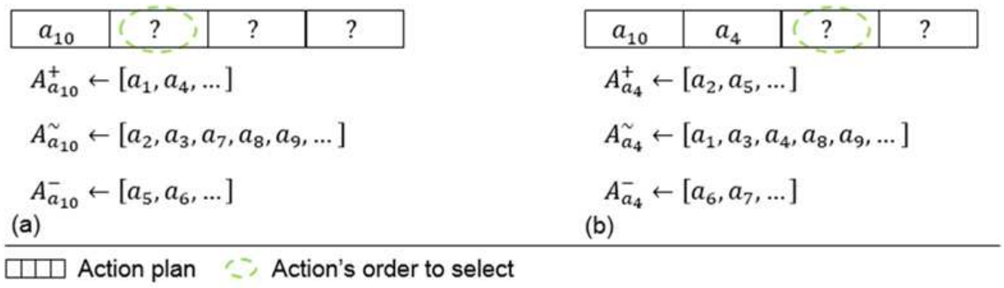


Figure 3: An example of selecting actions based on approach (1).

The selection of the third action in the action plan depends on the second selected action; the classification of the actions is based on $a^{[2]}$: $A^{+}_{a^{[2]}}$, $A^{\sim}_{a^{[2]}}$ and $A^{-}_{a^{[2]}}$. For example, the selection of the third action in Figure 3b is based on the classification of actions by a_4. The selection of the actions continues until the action plan is formed.

4.1.2. Approaches (2) and (3): Classification of actions based on selected actions in an action plan

These approaches differ from approach (1) described in 4.1.1 by using all the selected actions in the action plan to classify the actions into classes. At the beginning of the construction of an action plan, the previously selected actions classify the actions into the three relationship classes. These classes become the classes of the action plan (A_s^+, $A_s^\sim$ and A_s^-) and are updated with each selection of an action.

In correlation approach (2), once the second action is selected, the classes are updated according to the classification of actions related to the second selected action. If an action is classified as $A_{a[2]}^-$, it is moved from A_s^+ to $A_s^\sim$, or from $A_s^\sim$ to A_s^-. On the other side, if an action is classified as $A_{a[2]}^+$, it is moved from A_s^- to $A_s^\sim$, or from $A_s^\sim$ to A_s^+. For example, in Figure 4, actions are classified after selecting the first action, a_{10}, as A_s^+, $A_s^\sim$, and A_s^-. After selecting the second action, a_4, a_2 and a_5 are moved from $A_s^\sim$ to A_s^+ and from A_s^- to $A_s^\sim$, respectively, because they are classified as positively correlated actions by a_4. On the other side, a_7 is moved from $A_s^\sim$ to A_s^-. Thus, the actions are moved between the classes based on their positive or negative relations with the newly selected action.

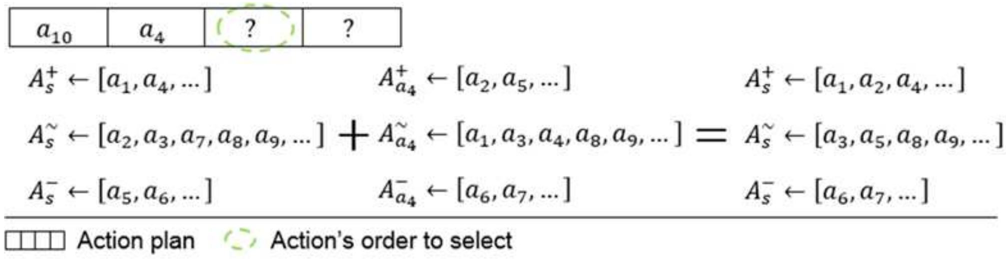


Figure 4: An example of selecting actions based on approach (2).

Correlation approach (3) is similar to the correlation approach (2), except that actions do not leave the negatively correlated actions class (A_s^-), and do not move to $A^\sim$; if an action is classified as $A_{a[i]}^-$, it is moved from A_s^+ or $A_s^\sim$to A_s^-. An action is moved from $A_s^\sim$ to A_s^+ if it is classified as $A_{a[i]}^+$; the action is not moved from A_s^- to $A_s^\sim$ if it is classified as $A_{a[i]}^+$. For example, in Figure 5, a_5 remains in A_s^- because it has a negative relationship with an action in the action plan, a_{10}.

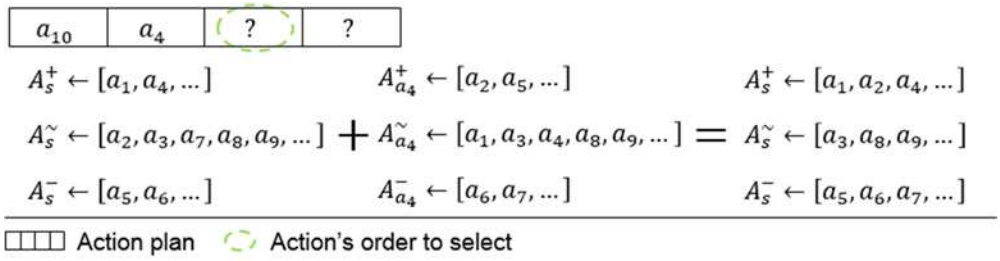


Figure 5: An example of selecting actions based on approach (3).

4.1.3. Approaches (4) and (5): Assigning selection probabilities to actions

In correlation approach (4), the selection probabilities are assigned to actions; instead of assigning them to the correlated actions' classes. Actions' selection probabilities are updated based on the selected action; the uniform selection probabilities of actions are updated based on $A^+_{a[i]}$ and $A^-_{a[i]}$, where i presents the last selected action. Actions in $A^+_{a[i]}$ get an increase in their selection probability, and actions in $A^-_{a[i]}$ get a decrease in their selection probability. In order to keep the summation of the probabilities of all actions equal to 1, the sum of the changes in the selection probabilities of actions in $A^+_{a[i]}$ should equal the sum of the changes in the selection probabilities of actions belonging to $A^-_{a[i]}$.Once an action $a^{[i]}$ is selected, the selection probability of the actions classified as $A^-_{a[i]}$ is halved. Then, $\sum_{a_k \in A^-} p_{a_k}$ presents the total decrease in the probabilities of negatively correlated actions, where $p_{a_k}$is the selection probability of action a_k. This sum is distributed over actions belonging to $A^+_{a[i]}$; the amount of increase of the selection probability of the actions depends on the number of actions that belong to $A^+_{a[i]}$. For example, actions are selected from 20 actions with a selection probability of 0.05. The first selected action classifies four actions as $A^+_{a[1]}$ and five actions as $A^-_{a[1]}$. The selection probability of actions in $A^-_{a[1]}$ is decreased to 0.025, and the total decrease in the selection probabilities becomes 0.125. Then, the selection probability of actions in $A^+_{a[1]}$ is increased to 0.08125 (= 0.05 + 0.125/4).

Once action $i + 1$ is selected, the selection probabilities are reset to the uniformly distributed probability, and then they are updated based on $A^+_{a[i+1]}$ and $A^-_{a[i+1]}$. As a result, an action's selection probability is one out of three values: actions belonging to $A^+_{a[i]}$ have the highest selection probability value, actions belonging to $A^{-}_{a^{[i]}}$ have the lowest selection probability value, and the remaining actions have an intermediate selection probability value.

In the correlation approach (5), the actions' selection probability is not reset to the uniformly distributed probability; thus, the selection probability of the actions varies and is not limited to three values as in approach (4).

4.2. Implementing correlation in an evolutionary algorithm

One of the realized metaheuristic algorithms in HU is an evolutionary algorithm. The initial algorithm's iteration starts by constructing solutions. A solution presents an action plan; actions are selected randomly to form the action plan. To utilize the correlation when constructing the action plans, the action selection is based on one of the correlation approaches described in Section 4.1.

The evaluated solutions form a population of the initial iteration. In the subsequent iterations, parent solutions are selected and crossover and mutation are used to form new solutions. Either type of the crossover can be used, such as 1-point crossover, 2-point crossover or uniform crossover (for crossover types see, e.g., Talbi 2009). Among the crossover types, 1-point crossover can keep the longest sequence of the correlated actions, mainly if correlation approaches (1) or (4) are used. The mutation is used to replace an action by another action. This replacement action can be selected based on the described correlation approaches. For example, if $a^{[3]}$ is to be replaced, the substitute action is selected based on $a^{[2]}$ in approaches (1) and (4), and based on $a^{[1]}$ and $a^{[2]}$ in approaches (2), (3), and (5).

5. EXPERIMENTAL RESULTS AND ANALYSIS

This section compares the correlation approaches defined in Section 4.1 to construct solutions. The described experiments use binary variables to illustrate the concept. First, Section 5.1 defines the correlation based on the correlation definition in Section 3, and then Section 5.2 compares the correlation approaches.

5.1. Experiment setup for the comparison between the constructions of solutions

Constructing action plans in the LAS is simplified to selecting binary variables and forming solutions. Two experiments are formulated: the correlation is defined between the variables, and the correlation is defined between action types (Sections 5.1.1 and 5.1.2, respectively). The experiments' results are analyzed in Section 5.2.

5.1.1. Experiment defining correlation relations between variables

The number of variables in the experiment is 20, and the length of a solution is selected as five. In each iteration, a solution is formed by selecting variables. Selected variables in a solution contribute to the value of an objective function (f) to be maximized. The objective function is presented in the Appendix. The relation between the variables can be studied as the discussed relation between actions in Section 3.1. A positive relation is defined if $f(x_i, x_j)$ is greater than $f(x_i) + f(x_j)$. The complete correlation matrix between the variables is shown in Table 2 in the Appendix. Each row in the correlation matrix presents the effect of a selected variable on the value of the objective function. For example, x_6 after x_1 decreases the value of the objective function, and x_3 after x_1 increases this value. The selection probability of positively, weakly, and negatively correlated actions was set to 0.7, 0.2, and 0.1, respectively.

The experiment compares a random selection of variables with the correlation approaches described in Section 4.1. The experiment includes two parts: (a) recording the maximum value of the objective function found in 5, 10, 20, and 30 iterations; and (b) recording the number of iterations required to find a solution having at least an objective function value of 25, 28, and 31. Each experiment setup is repeated for 100 times to perform statistical analysis afterwards.

5.1.2. Experiment defining correlation relations between action types

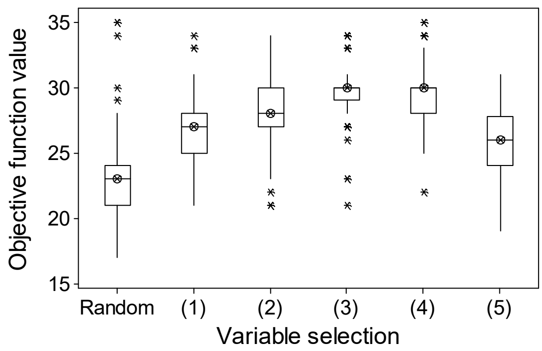
In this experiment, four action types are defined. Five variables belong to the same action type, e.g., x_1, x_2, x_3, x_4, and x_5 belong to AT^1. Variables x_1, x_6, x_{11}, and x_{16} affect the same entity, and the relation between the action types is defined based on the relations between the pairs of variables affecting the same entity. The correlation relation between variables is presented in

Table 2 in the Appendix, and the correlation matrix between the action types is summarized in Table 1. The relation between AT^1 followed by AT^2 was defined based on the relations between variables derived from them that affect the same entities: x_1 followed by x_6, x_2 followed by x_7, x_3 followed by x_8, x_4 followed by x_9, and x_5 followed by x_{10}. Four of these pairs have a positive relation; hence, the relation between AT^1 followed by AT^2 is a positive one.

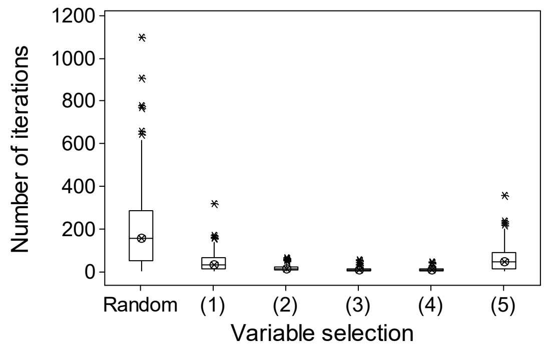
As in Section 5.1.1, the experiment involves selecting five variables to construct a solution, and the two parts of the experiment have been conducted.

5.2. Experiment results' analysis

Figure 6 shows a sample of the experiments' results described in Section 5.1.1. The sample shows the results of (a) recording the maximum objective function value in 20 iterations, and (b) recording the number of iterations to find a solution with at least 28 as a value of the objective function. It is seen that the correlation approaches can find better solutions within the specified number of iterations and can find the desired solution faster than the random selection of variables.



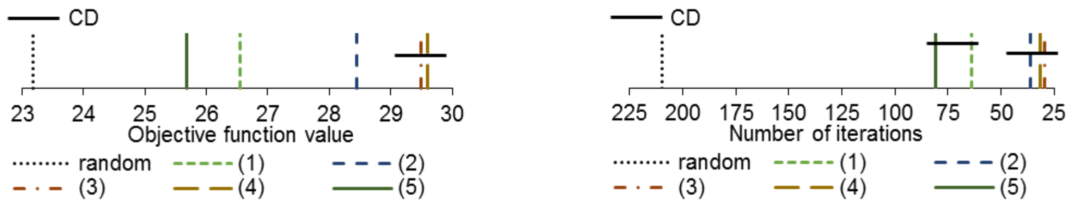
(a)



(b)

Figure 6: Box plots of the comparison between variables' selection in the experiment described in Section 5.1.1 regarding (a) maximum objective value found in 20 iterations for 100 runs, and (b) the number of iterations needed to find a solution having an objective value of at least 28 in 100 runs.

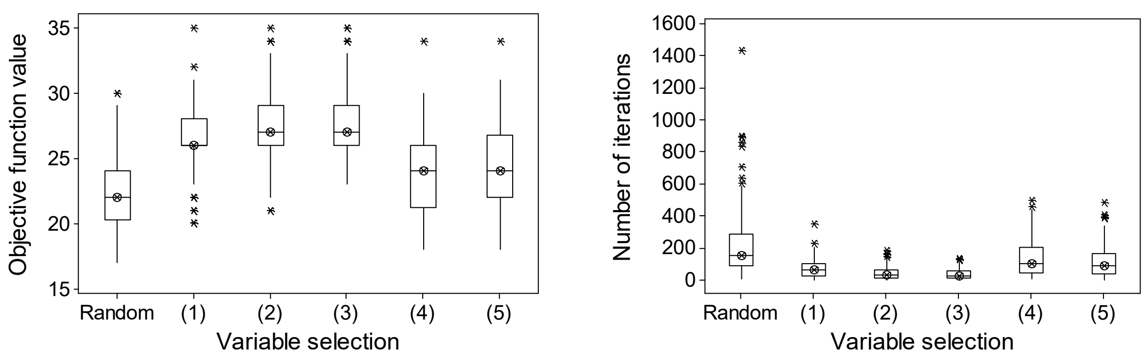
In order to define the significance of the selection method, analysis of variance (ANOVA) has been performed. The p-values of the test are 0.000 for both experiments' parts in Figure 6. Fisher's critical distance (CD) in Figure 7 indicates that the random selection of variables differs significantly from the correlation approaches. The correlation approaches (3) and (4) have the best performance and do not differ significantly in both experiment's parts (a) and (b). Based on the selection approaches' performance, the approaches can be ranked as correlation approach (3) and correlation approach (4) as the best approaches, followed by correlation approach (2), correlation approach (1), and correlation approach (5), and random selection as the worst performance in both experiments' parts.



(a) (b)

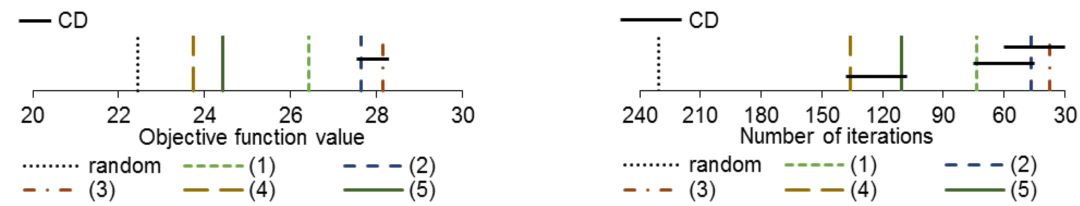
Figure 7: Comparison between variables' selection in the experiment described in Section 5.1.1 using Fisher's test regarding (a) maximum objective value found in 20 iterations for 100 runs, and (b) the number of iterations needed to find a solution having an objective value of at least 28 in 100 runs.

Figure 8 compares the experiment results considering the correlation relation definition for the action types (the experiment in Section 5.1.2). The ANOVA reveals that the methods differ significantly; the p-values are 0.000 for both experiments in Figure 8. Figure 9 indicates the significantly different selection approach in both experiment's parts. Based on the approaches' performance, the selection approaches are ranked as correlation approach (3) and correlation approach (2) as the best approaches, followed by correlation approach (1), correlation approach (5), correlation approach (4), and random selection. For the other experiment setups, the conclusions were similar to the one from the illustrated experiments' setup.



(a) (b)

Figure 8: Box plots of the comparison between variables' selection in the experiment described in Section 5.1.2 regarding (a) maximum objective value found in 20 iterations for 100 runs, and (b) the number of iterations needed to find a solution having an objective value of at least 28 in 100 runs.



(a) (b)

Figure 9: Comparison between variables' selection in the experiment described in Section 5.1.2 using Fisher's test regarding (a) maximum objective value found in 20 iterations for 100 runs, and (b) the number of iterations needed to find a solution having an objective value of at least 28 in 100 runs.

From both experiments, the correlation approach (3) had the best performance. This approach located good quality solutions within the lowest number of iterations in both experiments. It classifies variables based on their relation to the already selected variables in the solution and reduces the probability of selecting actions classified as A^- by keeping them in the A_S^-. In contrast, correlation approach (1) considers the last selected action only for the determination of the actions' classification. Actions in A_S^- are moved to $A_S^{\sim}$ in correlation approach (2) if a selected action classifies them as A^+. Both approaches (4) and (5) assign a distinguished selection probability, which can be computationally more demanding; thus, the recommended approach to utilize in the real-world case is the correlation approach (3).

The experiments' results revealed that the generalized correlation definition between action types differs significantly from the random selection. The definition of correlation between action types becomes more practical when more actions are considered in the search space; thus, utilizing correlation (3) based on the correlation relation defined between action types is recommended for real-world case studies.

6. CONCLUSION AND OUTLOOK

This work defines correlation as domain-specific information that aims to incorporate knowledge about the optimization problem in a logistics distribution network. The correlation presents a relation between two subsequent actions and their impact on the performance of the logistics distribution network. Five approaches to exploit correlation were discussed. The approaches were used to form solutions, and their performance was evaluated regarding the quality of formed solutions and the number of iterations to find a good solution. The defined correlation approaches found better solutions within fewer iterations compared to random selection. Additionally, the correlation was defined between action types, and the experiments' result showed that the generalization of the correlation definition found good solutions. Summarizing, correlation approaches constructed good solutions; thus, it is promising that optimization algorithms can benefit from utilizing them.

The proposed approach aims to guide the optimization algorithm to locate promising solutions within a lower number of evaluations while solving real-world problems; thus, the authors investigate the recommended approach, correlation approach (3), to optimize a logistics distribution network using the evolutionary algorithm and compare it to the results in this work in their further research.

Bibliography

Amaran, S., Sahinidis, N. V., Sharda, B., & Bury, S. J. (2016). Simulation Optimization: A Review of Algorithms and Applications. *Annals of Operations Research, 240*(1), 351–380.

Beiranvand, V., Hare, W., & Lucet, Y. (2017). Best Practices for Comparing Optimization Algorithms. *Optimization and Engineering, 18*(4), 815–848.

Bode, F., Reed, P., Reuschen, S., & Nowak, W. (2019). Search Space Representation and Reduction Methods to Enhance Multiobjective Water Supply Monitoring Design. *Water Resources Research, 55*(3), 2257–2278.

Bonyadi, M. R., Michalewicz, Z., Wagner, M., & Neumann, F. (2019). Evolutionary Computation for Multicomponent Problems: Opportunities and Future Directions. In S. Datta & J. P. Davim (Eds.), *Management and Industrial Engineering. Optimization in Industry: Present Practices and Future Scopes* (pp. 13–30). Cham, Switzerland: Springer International Publishing.

Christopher, M. (2011). *Logistics and Supply Chain Management* (4th ed.). Harlow, United Kingdom: Financial Times Prentice Hall.

Drugan, M. M. (2018). Scaling-up Many-objective Combinatorial Optimization with Cartesian Products of Scalarization Functions. *Journal of Heuristics, 24*(2), 135–172.

Ghiani, G., Laporte, G., & Musmanno, R. (2013). *Introduction to Logistics Systems Management* (2nd ed.). Chichester, United Kingdom: Wiley.

Gomes, P. V. & Saraiva, J. T. (2016). Evaluation of the Performance of Space Reduction Technique Using AC and DC Models in Transmission Expansion Problems. In *Proceedings of the 13th International Conference on the European Energy Market (EEM): 6-9 June 2016, Porto, Portugal* (pp. 1–5). Piscataway, New Jersey: IEEE.

Heilala, J., Montonen, J., Jarvinen, P., Kivikunnas, S., Maantila, M., Sillanpaa, J., & Jokinen, T. (2010). Developing Simulation-based Decision Support Systems for Customer-driven Manufacturing Operation Planning. In B. Johansson, S. Jain, J. Montoya-Torres, J. Hugan, & E. Yücesan (Eds.), *Proceedings of the 2010 Winter Simulation Conference* (pp. 3363–3375). Piscataway, New Jersey: IEEE.

Juan, A. A., Faulin, J., Grasman, S. E., Rabe, M., & Figueira, G. (2015). A Review of Simheuristics: Extending Metaheuristics to Deal with Stochastic Combinatorial Optimization Problems. *Operations Research Perspectives, 2*(1), 62–72.

Juan, A. A. & Rabe, M. (2013). Combining Simulation with Heuristics to Solve Stochastic Routing and Scheduling Problems. In W. Dangelmaier, C. Laroque, & A. Klaas (Eds.), *Simulation in Production and Logistics – Entscheidungsunterstützung von der Planung bis zur Steuerung* (pp. 641–649). Paderborn: HNI-Verlagsschriftenreihe.

Karimi, M. B., Isazadeh, A., & Rahmani, A. M. (2017). QoS-aware Service Composition in Cloud Computing Using Data Mining Techniques and Genetic Algorithm. *The Journal of Supercomputing, 73*(4), 1387–1415.

Ku, D. & Arthanari, T. S. (2016). On the Abstraction Method for the Container Relocation Problem. *Computers and Operations Research, 68,* 110–122.

Law, A. M. (2015). *Simulation Modeling and Analysis* (5th ed.). New York, New York: McGraw-Hill.

Liebler, K., Beissert, U., Motta, M., & Wagenitz, A. (2013). Introduction OTD-NET and LAS: Order-to-Delivery Network Simulation and Decision Support Systems in Complex Production and Logistics Networks. In R. Pasupathy, S.-H. Kim, A. Tolk, R. Hill, & M. E. Kuhl (Eds.), *Proceedings of the 2013 Winter Simulation Conference* (pp. 439–451). Piscataway, New Jersey: IEEE.

Lokman, B. (2019). Optimizing a Linear Function over the Nondominated Set of Multiobjective Integer Programs. *International Transactions in Operational Research, IN PRESS.*

Rabe, M., Dross, F., Schmitt, D., Ammouriova, M., & Ipsen, C. (2017). Decision Support for Logistics Networks in Materials Trading Using a Simheuristic Framework and Usergenerated Action Types. In S. Wenzel & T. Peter (Eds.), *Simulation in Production and Logistics 2017* (pp. 109–118). Kassel, Germany: Kassel University Press.

Rabe, M., Ammouriova, M., & Schmitt, D. (2018a). Utilizing Domain-specific Information for the Optimization of Logistics Networks. In M. Rabe, A. A. Juan, N. Mustafee, A. Skoogh, S. Jain, & B. Johansson (Eds.), *Proceeding of the 2018 Winter Simulation Conference* (pp. 2873–2884). Piscataway, New Jersey: IEEE.

Rabe, M., Schmitt, D., & Ammouriova, M. (2018b). Improving the Performance of a Logistics Assistance System for Materials Trading Networks by Grouping Similar Actions. In M. Rabe, A. A. Juan, N. Mustafee, A. Skoogh, S. Jain, & B. Johansson (Eds.), *Proceeding of the 2018 Winter Simulation Conference* (pp. 2861–2872). Piscataway, New Jersey: IEEE.

Riddalls, C. E., Bennett, S., & Tipi, N. S. (2000). Modelling the Dynamics of Supply Chains. *International Journal of Systems Science, 31*(8), 969–976.

Rushton, A., Croucher, P., & Baker, P. (2010). *The Handbook of Logistics and Distribution Management* (4th ed.). London: United Kingdom: Kogan Page.

Sheskin, D. (2011). *Handbook of Parametric and Nonparametric Statistical Procedures* (5th ed.). Boca Raton, Florida: CRC Press.

Sitek, P., Bzdyra, K., & Wikarek, J. (2016). A Hybrid Method for Modeling and Solving Supply Chain Optimization Problems with Soft and Logical Constraints. *Mathematical Problems in Engineering, 2016,* 1–16.

Talbi, E.-G. (2009). *Metaheuristics: From Design to Implementation*. Hoboken, New Jersey: John Wiley and Sons.

Appendix

The objective function value for the experiment used in Section 5, where x_i^2 presents the duplication of x_i:

$$
\begin{aligned}
f = 12 &- x_1 + x_2 - x_3 - 2x_4 + x_5 + 2x_6 - 2x_7 + x_8 + x_9 + x_{10} + x_{11} + 2x_{13} + x_{14} + x_{15} \\
&- x_{16} + x_{17} + x_{18} - x_{19} + 2x_{20} + x_2x_7 + 6x_3x_8 + 7x_5x_{10} + 5x_1x_{11} + 3x_2x_{12} \\
&+ 7x_3x_{13} - x_4x_{14} + 3x_5x_{15} + 2x_6x_{16} - 7x_7x_{17} - 8x_8x_{18} - 3x_9x_{19} - 3x_{10}x_{20} \\
&- 5x_{11}x_{16} - 4x_{12}x_{17} - x_{13}x_{18} - 6x_{14}x_{19} - 2x_{15}x_{20} - 3x_2x_8 + 2x_3x_{14} + x_4x_{17} \\
&- x_5x_{12} + x_6x_{18} + x_7x_{19} - x_8x_{20} + x_3x_9 + x_{10}x_{11} - x_7x_{11} + x_{12}x_{16} + x_1x_{13} - x_2x_{14} \\
&- 2x_4x_{15} - 3x_5x_{16} - x_9x_{17} + x_{10}x_{18} - 2x_{15}x_{19} - x_{13}x_{20} - 3x_6x_9 + x_1x_3 - x_9x_{14} \\
&- x_3x_{13} - 3x_1^2 - 4x_2^2 - 2x_3^2 - 5x_4^2 - x_5^2 + 2x_6^2 + x_7^2 - x_8^2 + 2x_9^2 + 4x_{10}^2 - 2x_{11}^2 \\
&- 3x_{12}^2 - x_{13}^2 - 2x_{14}^2 + 2x_{15}^2 + 4x_{16}^2 + 2x_{17}^2 + 3x_{18}^2 + 6x_{19}^2 + x_{20}^2
\end{aligned}
$$

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	x_{17}	x_{18}	x_{19}	x_{20}
x_1	−	~	+	~	~	~	~	~	~	~	+	~	+	~	~	~	~	~	~	~
x_2	~	−	~	~	~	~	+	−	~	~	~	+	~	−	~	~	~	~	~	~
x_3	+	~	−	~	~	~	~	+	+	~	~	~	+	+	~	~	~	~	~	~
x_4	~	~	~	−	~	~	~	~	+	~	~	~	~	−	−	~	+	~	~	~
x_5	~	~	~	~	−	~	~	~	~	+	~	−	~	~	+	−	~	~	~	~
x_6	~	~	~	~	~	~	~	~	−	~	~	~	~	~	~	+	~	+	~	~
x_7	~	+	~	~	~	~	+	~	~	~	−	~	~	~	~	~	−	~	+	~
x_8	~	−	+	~	~	~	~	−	~	~	~	~	~	~	~	~	~	−	~	−
x_9	~	~	+	+	~	−	~	~	+	~	~	~	~	−	~	~	−	~	−	~
x_{10}	~	~	~	~	+	~	~	~	~	+	+	~	~	~	~	~	~	+	~	−
x_{11}	+	~	~	~	~	~	−	~	~	+	−	~	~	~	~	−	~	~	~	~
x_{12}	~	+	~	~	−	~	~	~	~	~	~	−	~	~	~	+	−	~	~	~
x_{13}	+	~	+	~	~	~	~	~	~	~	~	~	−	~	~	~	~	−	~	−
x_{14}	~	−	+	+	~	~	~	~	−	~	~	~	~	−	~	~	~	~	−	~
x_{15}	~	~	~	−	+	~	~	~	~	~	~	~	~	~	+	~	~	~	−	−
x_{16}	~	~	~	~	−	+	~	~	~	~	−	+	~	~	~	+	~	~	~	~
x_{17}	~	~	~	+	~	~	−	~	−	~	~	−	~	~	~	~	+	~	~	~
x_{18}	~	~	~	~	~	+	~	−	~	+	~	~	−	~	~	~	~	+	~	~
x_{19}	~	~	~	~	~	~	+	~	−	~	~	~	~	−	−	~	~	~	+	~
x_{20}	~	~	~	~	~	~	~	−	~	−	~	~	−	~	−	~	~	~	~	−

Table 2: Correlation matrix for the variables in the experiments in Sections 5.1.1 and 5.1.2.