DOI: 10.2195/lj_Proc_wittwer_en_202112_01
URN: urn:nbn:de:0009-14-54486

Utilization-oriented Harvest Logistics: Vehicle Routing and Scheduling of Forage Harvesters, Transfer and Transport Vehicles with Synchronization Constraints

Auslastungsorientierte Erntelogistik: Routenplanung und Disposition von Feldhäckslern, Überladewagen und Transportfahrzeugen mit Synchronisationsnebenbedingungen

David Wittwer
Thorsten Schmidt

Chair of Material Handling
Institute of Material Handling and Industrial Engineering
Faculty of Mechanical Science and Engineering
TU Dresden

Agricultural contractors are logistics service providers for various field operations in agriculture. Restrictions imposed by nature, lack of personnel, high-priced agricultural machinery and small margins require contractors to make efficient use of their fleet of agricultural machinery. In order to achieve a short completion time, as is particularly necessary for harvesting, the dispatcher aims to minimize travel times between the operating locations and to increase the utilization of harvesting machines, e.g. forage harvesters and combine harvesters. This paper examines the harvesting process involving forage harvesters and support vehicles (transfer vehicles and transport vehicles). Here, if insufficient support vehicles are available, the forage harvester needs to pause operation on the field while waiting for a support vehicle that can receive the harvested biomass. Typically, teams of harvesters and support vehicles are assembled prior to the start of the harvesting process and remain in this constellation until all fields in the planning period have been harvested. In this study, we investigate the time savings when vehicles are not tied to their team and can thereby harvest each field in a new constellation of vehicles. This paper presents two mathematical models describing the harvesting process with forage harvesters, transfer vehicles and transporters with a focus on the utilization of the forage harvesters with fixed and variable team compositions, respectively. These models are solved via mixed-integer programming with the Gurobi solver.

[Keywords: harvest logistics, vehicle routing problem, synchronization, mixed-integer programming]

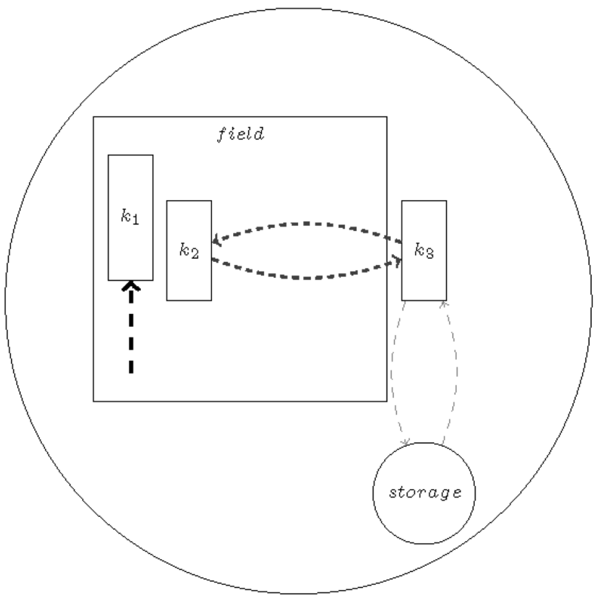
Landwirtschaftliche Lohnunternehmer sind Logistikdienstleister für verschiedene Feldarbeiten in der Landwirtschaft. Naturgegebene Restriktionen, Personalmangel, hochpreisige Landmaschinen und geringe Margen erfordern von den Lohnunternehmern eine effiziente Auslastung ihres Landmaschinenparks. Um kurze Einsatzzeiten zu erreichen, wie es insbesondere bei der Ernte notwendig ist, werden kurze Fahrzeiten zwischen den Einsatzorten und eine hohe Auslastung der Erntemaschinen, z.B. Feldhäcksler und Mähdreschern, angestrebt. In diesem Beitrag wird der Ernteprozess mit Feldhäckslern und Unterstützungsfahrzeugen (Überlade- und Transportfahrzeuge) untersucht. Dabei muss der Feldhäcksler, wenn nicht genügend Begleitfahrzeuge zur Verfügung stehen, die Ernte auf dem Feld unterbrechen und auf ein Begleitfahrzeug warten, das die geerntete Biomasse aufnehmen kann. Typischerweise werden Teams aus Erntemaschinen und Unterstützungsfahrzeugen vor Beginn des Erntevorgangs zusammengestellt und bleiben in dieser Konstellation, bis alle Felder im Planungszeitraum abgeerntet sind. Im Fokus dieser Studie steht die sich ergebende Zeitersparnis, wenn die Fahrzeuge nicht an ihr Team gebunden sind und somit jedes Feld in einer neuen Konstellation beernten können. In diesem Beitrag werden zwei mathematische Modelle vorgestellt, die den Ernteprozess mit Feldhäckslern, Übergabefahrzeugen und Transportern beschreiben, wobei der Schwerpunkt auf der Auslastung der Feldhäcksler bei fester bzw. variabler Teamzusammensetzung liegt. Diese Modelle werden mittels gemischt-ganzzahliger Programmierung mit dem Solver Gurobi gelöst.

[Schlüsselwörter: Erntelogistik, Vehicle Routing Problem, Synchronisierung, Gemischt-ganzzahlige Programmierung]

DOI: 10.2195/lj_Proc_wittwer_en_202112_01
URN: urn:nbn:de:0009-14-54486

1 INTRODUCTION

In this paper, we present a vehicle routing problem (VRP) with temporal and spatial synchronization of three different types of vehicles from the field of harvest logistics. The vehicle fleet is composed of forage harvesters and two types of support vehicles: transfer vehicles and transport vehicles. Figure 1 depicts a schematic illustration of the movement of the three vehicle types during harvest of a single field. While forage harvesters cover the field area, transfer vehicles travel between forage harvesters and the transport vehicles at the edge of the field for the transfer of biomass. Transport vehicles, in turn, travel between the field and the storage facility, which usually is a silo.



To fully utilize the forage harvester(s) in the field, certain numbers of transfer and transport vehicles are required depending on the field geometry and the distance between field and storage facility, respectively. Due to frequent staff shortages, high-priced machinery, and requirements regarding vehicle numbers that vary between fields, full utilization of forest harvesters is often not achieved. In this paper, instead of considering the exact movement of these three vehicle types on and next to the field, we assume that different numbers of transfer and transport vehicles result in different utilization rates of the forest harvester(s) and therefore different harvest durations (service times). These harvesting durations are field specific and can be determined in advance considering the distance between field and storage facility, the in-field track of harvesters and transfer vehicles, and vehicle specific characteristics such as working width of the forage harvester or loading capacity of the transfer and transport vehicles. Thus, the problem presented in this paper is the rough planning of the harvesting process. This reduction of complexity enables the simultaneous harvest planning of several fields as depicted in Figure 2. Here, one forest harvester returns to the depot after harvesting field 4, while the supporting vehicles move from field 4 to field 1 to support the other forage harvester. Field-specific requirements for the number of support vehicles on field 1 make the use of an additional forage harvester unnecessary, because it cannot reduce the processing time any further.

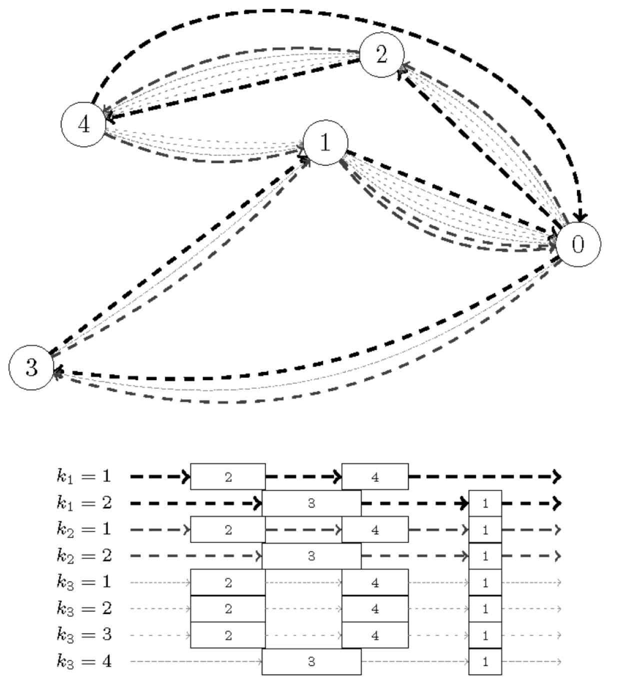


Figure 2. Graph and Gantt chart of an optimal solution with one depot (0), four fields, two forage harvesters (k_1), three transfer vehicles (k_2), and four transport vehicles (k_3)

In practice, harvest teams are usually assembled at the beginning of the planning horizon and remain in that composition until the end of the planning horizon. In this paper we will investigate the benefit regarding completion time (makespan) when vehicles are not tied to their harvest team and may leave their team when a field is completely harvested to join other vehicles to harvest the next field.

The harvesting process with three different types of vehicles involved, is one of two classic corn silage harvesting processes. Both processes differ mainly in the number of vehicle types used: Instead of three vehicle types as described above, the alternative harvesting process requires only harvesters and transporters, with transporters receiving the biomass from the harvester (instead of the transfer vehicle) in the field and transporting it directly to the storage facility. For a scheduling problem with only forage harvesters and transporters, [BoS10] and [ACC15] have each developed a two-stage heuristic for scheduling biomass transfers between two types of vehicles. [CCA17] have developed a heuristic for calculating harvester routes without considering other types of vehicles. [APC15] developed a

DOI: 10.2195/lj_Proc_wittwer_en_202112_01
URN: urn:nbn:de:0009-14-54486

simulation of logistic harvesting processes of forage harvesters and transport vehicles. [WiT21] developed a model for workload-based scheduling of two different types of vehicles and solved it with mixed integer programming (MIP). Harvest vehicle utilization was interpreted here in a similar way as in this paper. Therefore, we adopt the modeling approach from [WiT21] and extend it to include a third vehicle type. This results in a complex synchronized vehicle routing problem with support vehicle-dependent service times. For further details on synchronized VRP we refer to [Dre12]. For the classification of the presented problem we refer to [WiT21]. In contrast to former work, in particular the similar approach from [WiT21], we consider an additional vehicle type and enable multiple forage harvesters to harvest a field simultaneously. To the best of our knowledge, the logistic optimization with mathematical programming of the corn silage harvest with three vehicle types, i.e., forage harvesters, transfer vehicles, and transport vehicles, has not yet been studied. We make the following contributions to address this gap:

1. We modify and extend the model of a synchronized VRP with support vehicle and utilization dependent service times presented in [WiT21] by an additional vehicle type. Additionally, we present another very efficient MIP formulation of a similar problem, in which vehicles are assigned to teams they remain in during makespan.

2. We provide an analysis of the benefit of additional vehicles for each vehicle type. Increasing the number of vehicles in general reduces the makespan until saturation effects due to field characteristics and the ratio of vehicles to each other take effect.

3. We compare and evaluate two different harvest strategies regarding makespan and computation time. While enabling vehicles to flexibly move between fields is usually beneficial and never disadvantageous regarding makespan, solving such problems requires high computation capacities for large instances.

The remainder of this paper is structures as follows. In Section 2, we present two MIP models of the two different harvest strategies (with and without switching between teams). Section 3 contains computational experiments and discussions. Finally, we summarize the presented work and give an outlook on future work in Section 4.

2 PROBLEM FORMULATION

In this section, we present the assumptions made for the mathematical formulations, introduce the relevant notations for the models and finally present the models themselves. For the remainder of the paper, we call the model in which vehicles can move to the next field independently of each other the *independency* model. The model in which vehicles are permanently assigned to a group will be called the *group* model.

2.1 ASSUMPTIONS

We make the following assumptions regarding the harvest process with forage harvesters, transfer and transport vehicles:

- The fleets of forage harvesters, transfer and transport vehicles are all homogeneous.
- For an operation (harvest), at least one vehicle of each vehicle type is required.
- All vehicles have the same travel speed between fields.
- Vehicles can visit a field not more than once.
- An operation cannot start before all vehicles assigned to that operation have arrived. A vehicle cannot join or leave an ongoing operation.
- The number of primary vehicles and field-specific properties define the number of transfer and transport vehicles required for full utilization.
- As long as a vehicle type is the scarce resource out of all three types, additional vehicles of this type reduce the service time at a field linearly.

In both models, we minimize the makespan, defined as the return time to the depot of the last vehicle after all fields are fully harvested.

2.2 MATHEMATICAL PROGRAMMING FORMULATIONS

For the remainder of this paper, we will use the terms forage harvester and vehicle type 1, transfer vehicles and vehicle type 2, and transport vehicles and vehicle type 3 as synonyms.

2.2.1 SETS, PARAMETERS, DECISION VARIABLES

The problem is defined on a complete directed graph $G(N_0, A)$ with the set of nodes N_0 and the set of arcs A. The set of nodes representing the fields is defined as $N = N_0 \backslash \{0, n+1\}$ with 0 as start depot, with $n+1$ as the end depot and the number of fields n. To facilitate modelling we additionally introduce $N_0^+ = \{0\} \cup N$ and $N_0^- = N \cup \{n+1\}$. The set of vehicle types is defined as $R = \{1, 2, 3\}$. Forage harvesters are defined as vehicle type 1, transfer vehicles as vehicle type 2 and transport vehicles as vehicle type 3. The sets of each vehicle type are defined as $K_r \; \forall r \in R$. An additional set G is required for the *group* model defining the groups a vehicle can be assigned to. As all groups require at least one vehicle of each vehicle type the number of elements in G can be calculated as follows:

$$|G| = \min(|K_1|, |K_2|, |K_3|)$$

DOI: 10.2195/lj_Proc_wittwer_en_202112_01
URN: urn:nbn:de:0009-14-54486

Each field node $j \in N$ has a positive demand d_j corresponding to the harvest time of a field by a single forage harvester with maximum utilization. We denote the utilization in which a field j is harvested by k_1 forage harvesters, k_2 transfer vehicles and k_3 transport vehicles as $u^j_{k_1k_2k_3}$ and calculate it as follows:

$$u^j_{k_1k_2k_3} = \min\left(\frac{k_2}{k_2^j}, \frac{k_3}{k_3^j}, k_1\right)$$

With the field-specific required minimum number of transfer vehicles k_2^j and number of transport vehicles k_3^j. Note that the utilization can be greater than 1 if more than one forage harvester is involved. Each arc (i,j) in the arc set $A = \{(i,j) : i \in N_0^+, j \in N_0^-, i \neq j\}$ has a non-negative travel time τ_{ij}. The parameter M is a very large number at least as large as the unknown makespan. The value of this parameter is discussed further in the computational experiments in Section 3.

The decision variables to describe the two different mathematical programming formulations are given in Table 1.

Table 1. Decision variables used in the mathematical programming formulations of the independency (i) and group (g) model.

Name	Model	Definition
$t_j \in \mathbb{R}$	i, g	Start time of the service at node j
$z_r^g \in \mathbb{N}$	g	Amount of vehicles of type r in group g
$y^g_{k_1k_2k_3} \in \{0,1\}$	g	1, if k_1 vehicles of type 1, k_2 vehicles of type 2 and k_3 vehicles of type 3 are in group g, else: 0
$q^{jg}_{k_1k_2k_3} \in \{0,1\}$	g	1, if group g with k_1 vehicles of type 1, k_2 vehicles of type 2 and k_3 vehicles of type 3 visits node j, else: 0
$x^g_{ij} \in \{0,1\}$	g	1, if group g moves directly from node i to node j, else: 0
$p^j_{k_1k_2k_3} \in \{0,1\}$		1, if k_1 vehicles of type 1, k_2 vehicles of type 2 and k_3 vehicles of type 3 visit node j, else: 0
$w^r_{ij} \in \mathbb{N}$	i	Amount of vehicles of resource type r moving directly from node i to node j
$v^r_{ij} \in \{0,1\}$	i	1, if any vehicle of type r moves directly from node i to node j, else: 0

2.2.2 Mathematical Programming Formulation of the Group Model

The objective function (1) minimizing the makespan is identical for both models. Constraints and inequalities (2) - (12) define the *group* model, in which all vehicles are grouped once and remain in groups until all harvesting jobs are completed.

$$\min t_{n+1} \tag{1}$$

$$\sum_{k_1 \in K_1} \sum_{k_2 \in K_2} \sum_{k_3 \in K_3} y^g_{k_1k_2k_3} \cdot k_r = z_r^g \quad \forall g \in G, r \in R \tag{2}$$

$$\sum_{k_1 \in K_1} \sum_{k_2 \in K_2} \sum_{k_3 \in K_3} y^g_{k_1k_2k_3} = 1 \quad \forall g \in G \tag{3}$$

$$\sum_{g \in G} z_r^g = |K_r| \quad \forall r \in R \tag{4}$$

$$\sum_{g \in G} \sum_{j \in N} x^g_{0j} \leq |G| \tag{5}$$

$$\sum_{i \in N_0^+} \sum_{g \in G} x^g_{ij} = 1 \quad \forall i \in N \tag{6}$$

$$\sum_{h \in N_0^+} x^g_{hi} = \sum_{j \in N_0^-} x^g_{ij} \quad \forall i \in N, g \in G \tag{7}$$

$$t_j \geq t_i + \sum_{k_1 \in K_1} \sum_{k_2 \in K_2} \sum_{k_3 \in K_3} y^g_{k_1k_2k_3} \cdot \frac{d_i}{w^i_{k_1k_2k_3}} + \tau_{ij} \cdot x^g_{ij} + M \cdot \left(x^g_{ij} - 1\right) \quad \forall g \in G, i \in N_0^+, j \in N_0^- \tag{8}$$

$$y^g_{k_1k_2k_3} \geq q^{jg}_{k_1k_2k_3} \quad \forall g \in G, j \in N, k_1 \in K_1, k_2 \in K_2, k_3 \in K_3 \tag{9}$$

$$\sum_{i \in N_0^+} x^g_{ij} \leq \sum_{k_1 \in K_1} \sum_{k_2 \in K_2} \sum_{k_3 \in K_3} q^{jg}_{k_1k_2k_3} \quad \forall g \in G, j \in N \tag{10}$$

$$\sum_{g \in G} \sum_{k_1 \in K_1} \sum_{k_2 \in K_2} \sum_{k_3 \in K_3} q^{jg}_{k_1k_2k_3} = 1 \quad \forall j \in N \tag{11}$$

$$t_{n+1} \geq \sum_{i \in N_0^+} \sum_{j \in N_0^-} x^g_{ij} \cdot \tau_{ij} + \sum_{i \in N} \sum_{k_1 \in K_1} \sum_{k_2 \in K_2} \sum_{k_3 \in K_3} q^{ig}_{k_1k_2k_3} \cdot \frac{d_i}{u^i_{k_1k_2k_3}} \quad \forall g \in G \tag{12}$$

Constraints (2) - (4) control the vehicle composition of the groups. Constraints (2) connect binary variables $y^g_{k_1k_2k_3}$ with integer variables z_r^g to ensure the number of vehicles in a group of type r corresponds to the overall group composition defined by $y^g_{k_1k_2k_3}$. Constraints (3) guarantee that there is exactly one group composition per group. Constraints (4) enforce that the number of vehicles of all groups combined does not exceed the total number of vehicles available per vehicle type.

Constraints (5) - (8) define the routes of the vehicle groups. Constraint (5) ensures that not more groups leave the depot than there are available. Constraints (6) ensure that every field is visited by exactly on group. Constraints (7) are the flow conversation constraints that ensure that a group leaves the field it visits. Constraints (8) set the earliest start time of a group at all nodes. Here, $\frac{d_i}{w^i_{k_1k_2k_3}}$ corresponds to the service time at field i. Additionally, constraints (8) serve for subtour elimination.

Constraints (9) - (11) define variable $q^{jg}_{k_1k_2k_3}$ which is required for valid inequalities (12). Constraints (9) and (10) are an upper and lower bound for variables $q^{jg}_{k_1k_2k_3}$, respectively. Analogue to (3) Constraints (11) restrict the number of visits to a single group with a single group composition. As the linear relaxation of the mathematical program is rather weak, we added inequalities (12) by setting a lower bound for the start times (at the depot: arrival times) at the end depot of the groups. This results in shorter computation times.

2.2.3 MATHEMATICAL PROGRAMMING FORMULATION OF THE INDEPENDENCY MODEL

Besides objective function (1), the constraints and inequalities (13) - (20) form the mathematical programming formulation for the *independency* model, in which vehicles travel to the next field independently from each other.

$$\sum_{j\epsilon N_0^-} w^r_{0j} \le |K_r| \quad \forall r\epsilon R \tag{13}$$

$$\sum_{h\epsilon N_0^+} w^r_{hi} = \sum_{j\epsilon N_0^-} w^r_{ij} \quad \forall i\epsilon N, \forall r\epsilon R \tag{14}$$

$$w^r_{ij} \le v^r_{ij} \cdot |K_r| \quad \forall r\epsilon R, i\epsilon N_0^+, j\epsilon N_0^-, i \ne j \tag{15}$$

$$\begin{aligned} t_j \ge t_i + \sum_{k_1\in K_1}\sum_{k_2\in K_2}\sum_{k_3\in K_3} p^i_{k_1k_2k_3} \cdot \frac{d_i}{u^i_{k_1k_2k_3}} \\ + \tau_{ij} \cdot v^r_{ij} + M \cdot (v^r_{ij} - 1) \\ \forall r \in R, \forall i \in N_0^+, \forall j \in N_0^- \end{aligned} \tag{16}$$

$$\sum_{k_1\in K_1}\sum_{k_2\in K_2}\sum_{k_3\in K_3} p^i_{k_1k_2k_3} = 1 \quad \forall i \in N \tag{17}$$

$$\begin{aligned} \sum_{k_1\in K_1}\sum_{k_2\in K_2}\sum_{k_3\in K_3} p^j_{k_1k_2k_3} \cdot k_R = \sum_{i\epsilon N_0^+} w^r_{ij} \\ \forall j \in N, \forall r\epsilon R \end{aligned} \tag{18}$$

$$\begin{aligned} t_{n+1} \\ \ge \Bigg(\sum_{i\epsilon N_0^+}\sum_{j\epsilon N_0^-} v^r_{ij} \cdot \tau_{ij} \\ + \sum_{i\epsilon N}\sum_{k_1\epsilon K_1}\sum_{k_2\epsilon K_2}\sum_{k_3\epsilon K_3} y^i_{k_1k_2k_3} \\ \cdot \frac{d_i}{u^i_{k_1k_2k_3}} \Bigg) : |K_r| \quad \forall r\epsilon R \end{aligned} \tag{19}$$

$$w^r_{ij} \ge v^r_{ij} \quad \forall r\epsilon R, i\epsilon N_0^+, j\epsilon N_0^-, i \ne j \tag{20}$$

Constraints (13) - (16) define the routes of all vehicles involved. Constraints (13) ensure that not more vehicles of a vehicle type leave the depot than there are available in total. Constraints (14) are the flow conservation constraints that ensure that the same number of vehicles visit and leave a node. Constraints (15) guarantee that binary variables $v^r_{ij} = 1$ if a non-zero number of vehicles of type r travel on arc (i,j). Furthermore, they impose an upper bound to the number of vehicles allowed to travel on arc (i,j). Constraints (16) set the start times of all nodes with $\frac{d_i}{u^i_{k_1k_2k_3}}$ corresponding to the service time at field i. Additionally, constraints (16) serve for subtour elimination.

Constraints (17) and (18) define the configuration that a field is serviced with. Constraints (17) ensure that every field is serviced exactly once in exactly one configuration of vehicles. Constraints (18) connect binary variables $p^i_{k_1k_2k_3}$ with integer variables w^r_{ij} to ensure that the number of vehicles visiting a field corresponds to the configuration of vehicles a field is harvested with.

We introduce inequalities to strengthen the linear relaxation and improve computation times. Inequalities (19) set a lower bound on the makespan based on the time that vehicles spend traveling between nodes or serving customers. Inequalities (20) strengthen the relationship between binary and integer routing variables.

3 COMPUTATIONAL EXPERIMENTS

In this section, we describe the test setting and present the results of our experiments. The experiments include the analysis of the influence of additional vehicles and a comparison of two different planning strategies regarding makespan and computation time.

3.1 TEST INSTANCES AND SETTING

We generate instances with 4, 6 and 8 fields for the computational experiments. The fields and a depot are randomly distributed on a 100×100 *plane. Travel times correspond to the Euclidean distances between nodes. The number of transfer vehicles required to fully utilize a forage harvester ranges between* 0.5 *and* 1.5*. The number of transport vehicles to fully utilize a harvester ranges between* 2 *and* 4*. The demand of a field, i.e. the time required by a single harvester in full utilization (*$u^j_{k_1k_2k_3} = 1$*) to harvest a field ranges from 20 to 50. For each number of fields we vary the number of forage harvesters, transfer and transport vehicles in any combination of the values in the columns of Table 2:*

DOI: 10.2195/lj_Proc_wittwer_en_202112_01
URN: urn:nbn:de:0009-14-54486

Table 2. Number of forage harvesters (K_1), transfer vehicles (K_2) and transport vehicles (K_3) in the test instances

Vehicle types	Number of vehicles in instances	
K_1	2	3
K_2	1, 2, 3	2, 3, 4
K_3	4, 5, 6	6, 7, 8

Thus, the number of transfer vehicles can be smaller, equals or greater than the number of forage harvesters. Whereas the number of transport vehicles is at least twice the number of forage harvesters. For the remainder of this paper we abbreviate a configuration consisting of number of fields, number of forage harvesters, number of transfer vehicles and number of transport vehicles as $|N|\_|K_1|\_|K_2|\_|K_3|$. With five instances for each configuration the total number of instances considered is 270.

The mixed-integer linear program is implemented in Gurobi 9.1 via the Python 3.6 API. All instances are solved on an Intel(R) Xeon(R) CPU E5-2680 v3 with 2.50 GHz, 8 cores and 16 GB RAM. For each instance the maximum runtime is one hour. Each instances is solved three times to account for the randomness of the solver.

To strengthen the linear relaxation of the problems, the value of the parameter M should be chosen as small as possible. The minimum value is the optimal makespan, however, this is initially unknown. As the group model can be solved relatively fast we run the group model with a maximum runtime of 60 seconds to calculate M and pass the objective value as M and the best solution calculated as start solution to the independency model. Note that this is only possible, because the independency model is a relaxation of the group model. We include the additional runtime for the independency model in our results.

3.2 Experimental Results

In this section, we focus on aggregate data to provide general insights.

3.2.1 Influence of the Number of Vehicles on the Makespan

Figure 3 shows the average makespan obtained with the independency model for each configuration of fields, forage harvesters and *transfer vehicles*. Similarly, Figure 4 depicts the average makespan obtained for each configuration of fields, forage harvesters and *transport vehicles*. . In general, it can be deduced that a higher number of vehicles is beneficial in terms of makespan. This effect is greater when a scarce resource is increased, e.g., the reduction in makespan is slightly greater when the number of transport vehicles is increased from four to five than when it is increased from five to six. This behavior is similar for all vehicles, but is particularly prominent for forage harvesters and transfer vehicles. One reason for this may be that increasing the number of these vehicles makes a higher relative difference in the number of vehicles of this type. Since the number of primary vehicles determines the maximum number of support vehicles that can be reasonably deployed at a field, saturation effects can be observed here. For example, since some fields only require two transport vehicles to fully utilize a forage harvester, it does not make sense to have more vehicles in use there. For a larger number of fields, the observed effects are generally larger. On the one hand, this is due to the higher harvesting time associated with a higher number of fields. On the other hand, this effect is amplified by the fact that the ratio of field processing time to field change time increases, since the higher density of fields generally means that they are located closer together.

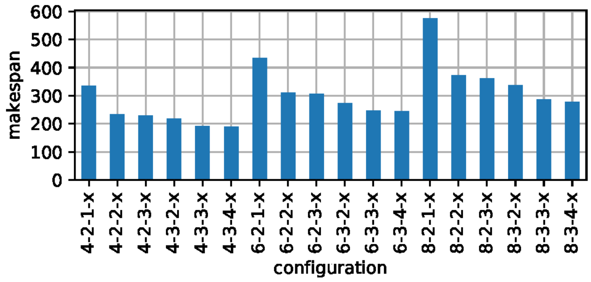


Figure 3. Average makespan of the independency model variant for each configuration of forage harvesters and transfer vehicles.

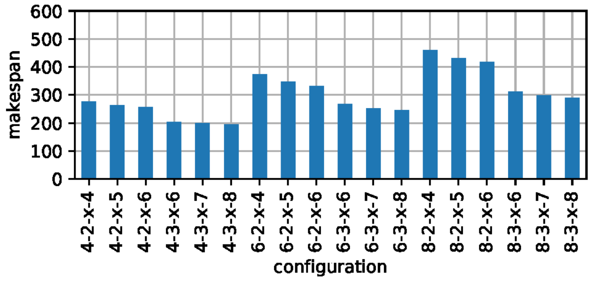


Figure 4. Average makespan of the independency model variant for each configuration of forage harvesters and transport vehicles.

3.2.2 Comparison of the Different Planning Strategies Regarding Makespan and Computation Time

Figure 5 presents the makespan reduction of the independency model relative to the group model. As expected, if there is only one transfer vehicle, there is no difference between the two models, since each type of vehicle must

be present at least once to harvest a field. Thus, all vehicles will harvest fields in the same compositions throughout the whole makespan. Due to rounding inaccuracies, there are minimal differences in makespan between the models for these instances.

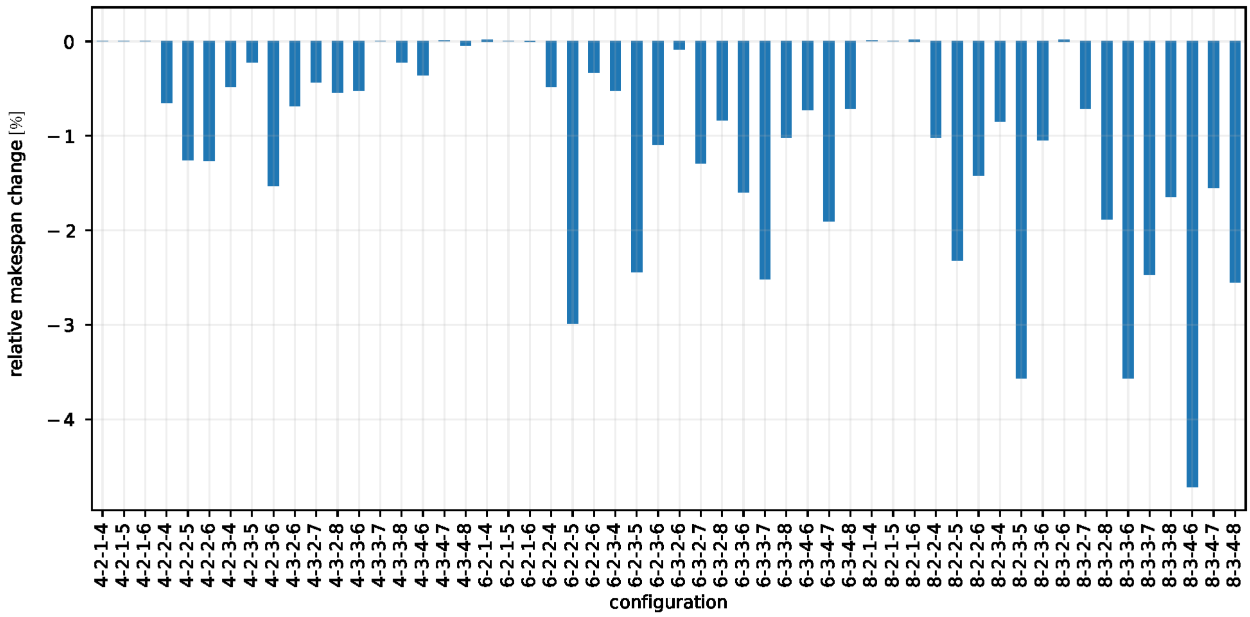


Figure 5. Average relative makespan change in percent of the independency variant compared to the group variant for all configuration of forage harvesters, transfer and transport vehicles.

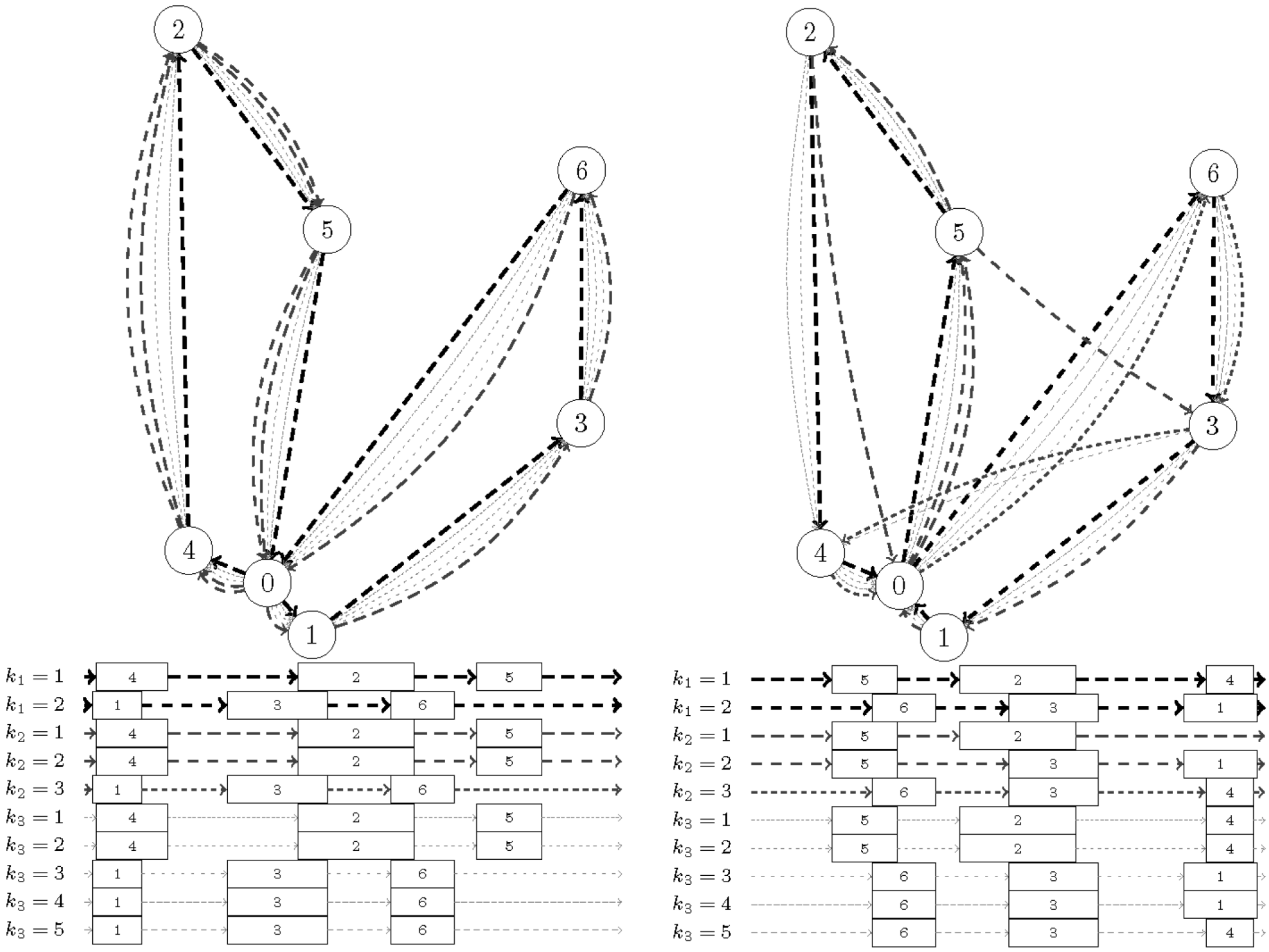


Figure 6. Example solutions of an instance with six fields, two forage harvesters (k_1), three transfer (k_2) and five transport vehicles (k_3) for the group model (left, makespan: 316.2) and the independency model (right, makespan: 302.7) variant

DOI: 10.2195/lj_Proc_wittwer_en_202112_01
URN: urn:nbn:de:0009-14-54486

An increased relative makespan reduction can be observed for instances with greater field numbers. This is probably due to the fact, that vehicles travel between more fields, which gives them more opportunities to form new constellations to harvest the next field. Another noteworthy observation is the fact that most of the greatest peaks occur for low or moderate numbers of transport vehicles (e.g. 6-2-2-5, 8-3-4-6) with a sharp decline, i.e. a lower makespan reduction, when adding another transport vehicle. This demonstrates that greater flexibility in harvest planning is particularly advantageous when vehicles are in short supply.

Figure 6 illustrates two exemplary solutions that show the advantage of flexible vehicle groups during harvest. The routes of the forage harvesters are inversed and additionally, one transfer vehicle ($k_2 = 2$) switches from field 5 to field 3 to shorten the harvest time of field 3 compared to the group model. Another switch occurs directly after field 3 is fully harvested: one transfer and one transport vehicle ($k_2 = 3$ and $k_3 = 5$) travel from field 3 to field 4, instead of going to field 1 together with the forage harvester on field 3. These recompositions of harvest teams shorten the total makespan by 4.4% compared to the model with immutable groups. The advantages in terms of makespan of the independency model are undeniable. However, the more complex planning situation also entails higher computation times. Besides the average makespan, Table 3 presents the average computation time as well as the average gap (relative difference between upper and lower bound) of the two different approaches. While instances with four and six fields are easily calculated within a few seconds, for eight fields several minutes are required when solving with the independency model. A gap greater than 0 indicates that some instances could not even be solved optimally within a calculation time of one hour. Runtime and gap are expected to grow exponentially, for greater instances, so the tradeoff between objective function value and runtime will become more important as instances sizes grow.

Table 3. Comparison of average makespan, runtime and gap between both modl variants

Model	$\|N\|$	**makespan**	**runtime [s]**	**gap [%]**
Group	4	234.0	0.16	0.0
	6	305.8	0.72	0.0
	8	372.4	27.55	0.0
Independency	4	233.0	0.39	0.0
	6	302.7	5.19	0.0
	8	369.9	266.56	0.3

4 Conclusion and Future Work

In this paper we present an extension to the synchronized vehicle routing problem presented in [WiT21] by a third vehicle type to model the harvest process with forage harvesters, transfer and transport vehicles. In this modelling approach the utilization rate of the harvesters is defined by the composition and number of support vehicles.

In this study, we introduce two MILP models: An extension to the model from [WiT21] and a new model formulation, which assigns vehicles to immutable groups. In computational experiments, we demonstrate the effect of additional vehicles on the makespan reduction. Increasing the number of a scarce vehicle type leads to relatively high reductions in makespan compared to vehicles that are already available in greater numbers. This is attributed to saturations effects due to a maximum number of support vehicles of a field in dependency of the forage harvester number. Finally, we show that it is possible to shorten the makespan of a harvest schedule significantly, when enabling to alter group compositions between fields. A disadvantage of this approach, however, is the additional computing time required. In addition, this approach entails an increased coordination effort in a practical application.

In this study, the vehicles were limited to changing between groups after finishing a field. A model, which enables vehicles to leave and join during an ongoing harvest process as a further generalization may be developed with the goal to decrease the makespan even further. Furthermore, some practical applications might require to solve greater instances within shorter computing time, e.g. for a longer planning horizon or smaller fields. This requires the development of adequate heuristics approaches or exact procedures.

Acknowledgment

The work was carried out as a part of the IGF research project 19509 BR “Landwirtschaftslogistik” (Logistikdienstleistungen in der Landwirtschaft - Aufträge sicherer und ressourcenoptimal planen) and supported by “Bundesvereinigung Logistik (BVL) e.V.”. The authors gratefully acknowledge the GWK support for funding this project by providing computing time through the Center for Information Services and HPC (ZIH) at TU Dresden.

Supported by:
Federal Ministry for Economic Affairs and Energy
on the basis of a decision by the German Bundestag

IGF
Industrielle Gemeinschaftsforschung

DOI: 10.2195/lj_Proc_wittwer_en_202112_01
URN: urn:nbn:de:0009-14-54486

LITERATURE

[ACC15] Amiama, Carlos, Noelia Cascudo, Luisa Carpente, und Ana Cerdeira-Pena. 2015. A Decision Tool for Maize Silage Harvest Operations. Biosystems Engineering 134: 94–104. https://doi.org/10.1016/j.biosystemseng.2015.04.004.

[APC15] Amiama, Carlos, José M. Pereira, Angel Castro, und Javier Bueno. 2015. *Modelling Corn Silage Harvest Logistics for a Cost Optimization Approach*. Computers and Electronics in Agriculture 118: 56–65. https://doi.org/10.1016/j.compag.2015.08.024

[BoS10] Bochtis, D.D., und C.G. Sørensen. 2010. *The Vehicle Routing Problem in Field Logistics: Part II*. Biosystems Engineer-ing 105 (2): 180–88. https://doi.org/10.1016/j.biosystemseng.2009.10.006..

[CCA17] Cerdeira-Pena, Ana, Luisa Carpente, und Carlos Amiama. 2017. Optimised Forage Harvester Routes as Solutions to a Traveling Salesman Problem with Clusters and Time Windows. Biosystems Engineering 164: 110–23. https://doi.org/10.1016/j.biosystemseng.2017.10.002

[Dre12] Drexl, Michael. 2012. Synchronization in Vehicle Routing—A Survey of VRPs with Multiple Synchronization Con-straints. Transportation Science 46 (3): 297–316. https://doi.org/10.1287/trsc.1110.0400.

[WiT21] Wittwer, David, und Felix Tamke. 2021. The synchronized vehicle routing and scheduling problemwith support ve-hicle dependent service times. [in review] https://tud.link/0136

David Wittwer, M.Sc., researcher at the Chair of Logistics Engineering, Institute of Material Handling and Industrial Engineering, Technische Universität Dresden.

David Wittwer received a masters degree in Economics and Engineering from Technische Universität Dresden. His research interests include application of mixed-integer-programming, constraint programming and heuristic algorithms for various variants and applications of scheduling and vehicle routing problems.

Address:
Professur für Technische Logistik, Technische Universität Dresden, 01062 Dresden, Germany,
Phone: +49 351 463 32539,
Fax: +49 463 35499,
E-Mail: david.wittwer@tu-dresden.de

Prof Dr.-Ing. Thorsten Schmidt, Head of the Chair of Logistics Engineering, Institute of Material handling and Industrial Engineering, Technische Universität Dresden.

Thorsten Schmidt is full professor at the TU Dresden and heads the Chair of Material Handling in the Mechanical Engineering faculty since 2008. He holds a diploma degree in mechanical engineering from the TU Dortmund and a Master degree in industrial engineering from the Georgia Institute of Technology. He received his Ph.D. from the TU Dortmund in 2001. His research areas are the design and optimization of facility logistics and production systems including a focus on the machinery and components involved. He currently works on energy efficient control strategies in material flow, fast approximation in early planning stages by means of standard design modules, online data analysis, formal verification of control logic, performance analysis of decentralized and selfcontrolled systems, lightweight structures in material handling and stress analysis on wire ropes and toothed belts, respectively.