



Evolutionary neighborhood discovery algorithm for agricultural routing planning in multiple fields

Amalia Utamima¹ · Torsten Reiners² · Amir H. Ansariipoor²

Accepted: 3 March 2022 / Published online: 2 May 2022
© The Authors 2022

Abstract

In recent years, operations research in agriculture has improved the harvested yield, reduced the cost and time required for field operations, and maintained economic and environmental sustainability. The heuristics method, named Evolutionary neighborhood discovery algorithm (ENDA), is applied to minimize the inter-field and intra-field distance of the routing planning of machines in multiple agricultural fields. The problem is an extended version of the Agricultural Routing Planning (ARP) that takes into consideration the different capacity of the machines and multiple agricultural fields. This research also describes the mathematical model to represent the proposed problem formulated as an integer program. The experimental results show that ENDA successfully solves ARP instances, giving the best results and the fastest running time compared to those obtained by Genetic Algorithms and Tabu Search. The results also show that ENDA can save an average of 11.72% of the distance traveled by the machines outside the working path (when making maneuvers, going to or from the entrances and going from and returning to the Depot).

Keywords Agriculture · Routing planning · Evolutionary neighborhood discovery algorithm

1 Introduction

Operations research in the domain of agriculture has become increasingly important since it can optimize the whole supply chain process and facilitate decision making (Zhang et al., 1990). Instances in this area include, among others, optimization of forest harvest scheduling (Neto et al., 2017; St. John and Tóth, 2013; Alonso-Ayuso et al., 2011), sugarcane harvest planning (Florentino et al., 2018; Sethanan and Neungmatcha, 2016), and crop cultivation (Aliano Filho et al., 2019). The domain also includes the management of farm and machinery as well as the routing of the machine(s) inside the field (Plà et al., 2014).

✉ Torsten Reiners
t.reiners@curtin.edu.au

Amalia Utamima
amalia@is.its.ac.id

¹ Faculty of Intelligent Electrical and Informatics Technology, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia

² Curtin Business School, Curtin University, Perth, Australia

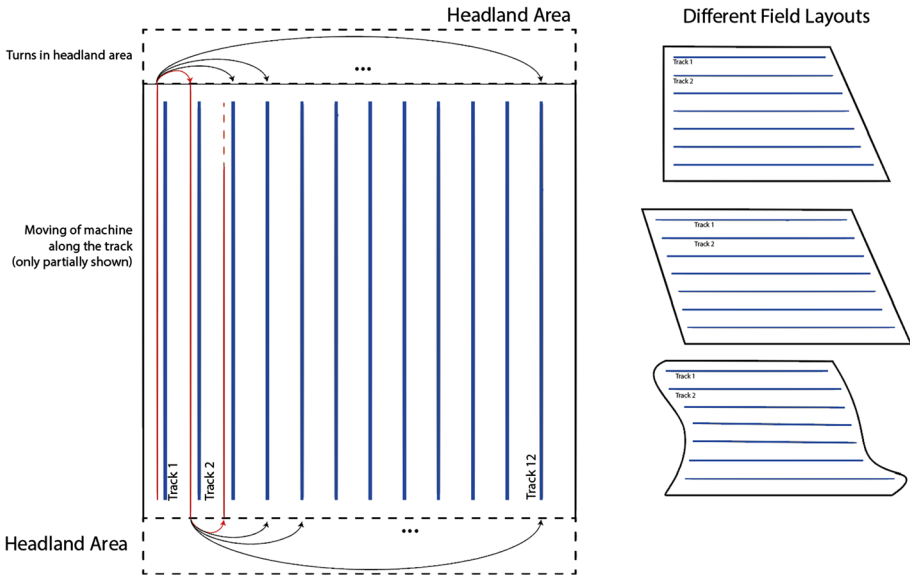


Fig. 1 Illustration of an agricultural field (Utamima et al., 2018)

Agricultural Routing Planning (ARP) in farm management is intended to design or schedule the movements of machines inside agricultural fields. A good design can minimize the distance of the machine’s tours, thereby leading to cost savings. Hence, it is essential to have an optimized plan for the routing of the machines being used for the agricultural tasks in the field (Jensen et al., 2015).

The focus of this research is on the routing optimization of machines inside multiple agricultural fields. Figure 1 illustrates an agricultural field while Fig. 2 shows the possible graph representation (right) of an ARP problem (left). The field has several established tracks with symmetrically-planted crops. These tracks can be traversed by both agricultural machine(s) and harvesters. The headland area to the North and South of the field is the non-working area (crop-free) where machines perform maneuvers to go to the next track. The farmer needs to determine which sequence of tracks will cover the shortest distance, thereby reducing the overall cost of harvesting.

To date, most research on ARP has considered only single-field configurations (Bochtis et al., 2015; Conesa-Muñoz et al., 2016; Seyyedhasani and Dvorak, 2018a). Most of the ARP research also utilized a single machine without a capacity constraint (Hameed et al., 2011; Edwards et al., 2017). Therefore, ARP with multiple fields and capacitated machines is still a potential research area. ARP belongs to the class of NP-complete problems; therefore, an exact optimization is too time-consuming and complex to be applied, and a heuristic algorithm is needed (Marinakakis et al., 2017). Hence, this research develops the evolutionary neighborhood discovery algorithm (ENDAs) to deal with ARP.

Most of the current research focuses on real-world scenarios and utilizes general algorithms rather than addressing unique ARP configurations and algorithm improvements. This study is a continuation of our recent work (Utamima et al., 2019) by extending the ARP problem. This study makes three significant contributions to the literature. First, the mathematical model for the extended ARP is formulated, taking into account multiple fields, barriers, and multiple machines with heterogeneous capacities. Second, seven problem sets for the extended ARP are generated and presented. Third, a hybrid heuristic and a set of

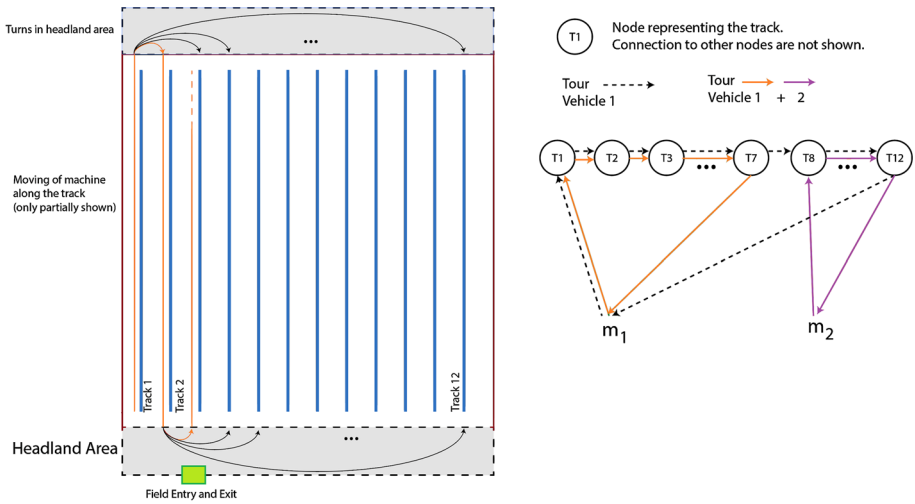


Fig. 2 Illustration of ARP problem (left) with possible graph configurations (right)

experiments are developed in order to compare the proposed heuristics with the Genetic Algorithm and Tabu Search from the previous literature.

This section is followed by a review of the literature pertaining to ARP and the algorithms that have been proposed to solve related problems (Sect. 2). The definition of the problem, the mathematical model, and the proposed algorithm, ENDA, are given in Sect. 3. Section 4 provides the experimental results, including descriptions of the problem set, the parameter settings, numerical results, and the analysis. Finally, Sect. 5 offers suggestions for future research in ARP, and concludes the paper.

2 Literature review

2.1 Agricultural routing planning

The ARP problem involves minimizing the distance traveled by machines when performing field operations inside an agricultural field (Utamima et al., 2018). This problem has been altered and extended in terms of the targets [e.g., improvement of time (Seyyedhasani and Dvorak, 2018b), minimization of the headland distance (Bochtis and Vougioukas, 2008)], specific field operations [e.g. herbicide application (Conesa-Muñoz et al., 2016), potato cultivation (Zhou et al., 2015), or orchard operation (Bochtis et al., 2015)], and limitations [e.g., restricted machine limit (Bakhtiari et al., 2013) and obstacles (Zhou et al., 2014)].

The optimization of the non-working distance, first introduced by (Bochtis and Vougioukas, 2008), is intended to minimize the distance in the headland area. The machine is not performing an agricultural operation while making the turning maneuvers in the headland area. Jensen et al. (2015) extend the problem by using capacity constraints. The model was adapted by Bochtis et al. (2015) for orchard operations, Conesa-Muñoz et al. (2016) for a weed-killing problem and by Seyyedhasani and Dvorak (2018a) for reducing the field work time. Barrientos et al. (2011) and Valente et al. (2013) also applied the same concept to aerial coverage planning.

Utamima et al. (2019) gathered several published ARP datasets from previous publications and presented a mathematical model to generalize the problem. All the datasets are single-field configuration. The collected datasets can be used as validation data for a new method when solving a new ARP problem. The work also conducts a comparative study of several metaheuristic algorithms in order to deal with the collected dataset.

Many of the recent ARP studies focus on a single field. Seyyedhasani and Dvorak (2017), Conesa-Muñoz et al. (2016), Zhou et al. (2014), Bakhtiari et al. (2013), Valente et al. (2013) and Hameed et al. (2011) consider both rectangular and non-convex fields. The rectangular field is the primary type that most researchers use in ARP. Whereas, the non-convex field is closer to actual cases and most farm situations (Zhou et al., 2014).

Currently, research on multiple crop fields is found in sugarcane harvesting (Kittilertpaisan and Pathumnakul, 2017; Sethanan and Neungmatcha, 2016). Sethanan and Neungmatcha (2016) focused on sugarcane field operations that minimized the traveling distance in multiple fields, while Kittilertpaisan and Pathumnakul (2017) focused on improving the planning of the cultivation of a new crop. However, neither study considered the tracks inside the fields and the maneuver of the machines. Hence, this study considers the tracks inside the fields and the machines' maneuvers to better address the harvesting problem.

2.2 Heuristic algorithms for agricultural routing planning

Previous studies on ARP focused mostly on real cases and solved the related problems with several different algorithms. The most frequently used algorithm is the Genetic Algorithm (GA). GA is an optimisation method based on the concept of genetics and natural selection. GA uses a population formed of individuals to evolve to a state that maximises fitness under the defined selection procedure (Haupt and Haupt, 2004; Karatas et al., 2021). In general, GA simulates a natural evolution process that generates individuals by selecting the best candidates from the current generation to be applied crossovers and mutations.

GA has been adapted for machine routing to decrease the total distance traveled in biomass transportation (Gracia et al., 2014). GA was also used for track sequence optimization in a field, and its solutions have proven to be more efficient compared to the conventional fieldwork patterns (Hameed et al., 2011). Recently, GA has been used in optimizing the scheduling of crop cultivation (Aliano Filho et al., 2019). GA has also been utilized to optimize the sugarcane harvesting plan in order to make the harvesting point as close as possible to the ideal (Florentino et al., 2018).

TS is an algorithm that traverses the solution space by moving iteratively from the current solution to a (best) solution in its neighborhood. TS improves the search by not re-evaluating points already visited in the search space. Such a point is considered 'tabu' to be rechecked (for several iterations). This method makes use of flexible memory structures with restrictions and aspiration criteria that exploit the search spaces (Glover and Laguna, 1997).

TS was applied to reduce the fieldwork time in agricultural operations (Seyyedhasani and Dvorak, 2018b). Seyyedhasani and Dvorak (2017, 2018a) utilized TS to improve field efficiency and the dynamic directing of several machines in farming tasks. The results show that TS can optimize the field work time more effectively than do the Clarke-Wright heuristics (Seyyedhasani and Dvorak, 2017). The TS approach was also applied to solve the location problem of a forest harvesting machine (Weintraub, 2007).

Sethanan and Neungmatcha (2016) used Particle Swarm Optimization (PSO) with a different structure for route planning in sugarcane fields, while Valente et al. (2013) employed Harmony Search to optimize coverage path planning in vineyard parcels. A hybrid Simu-

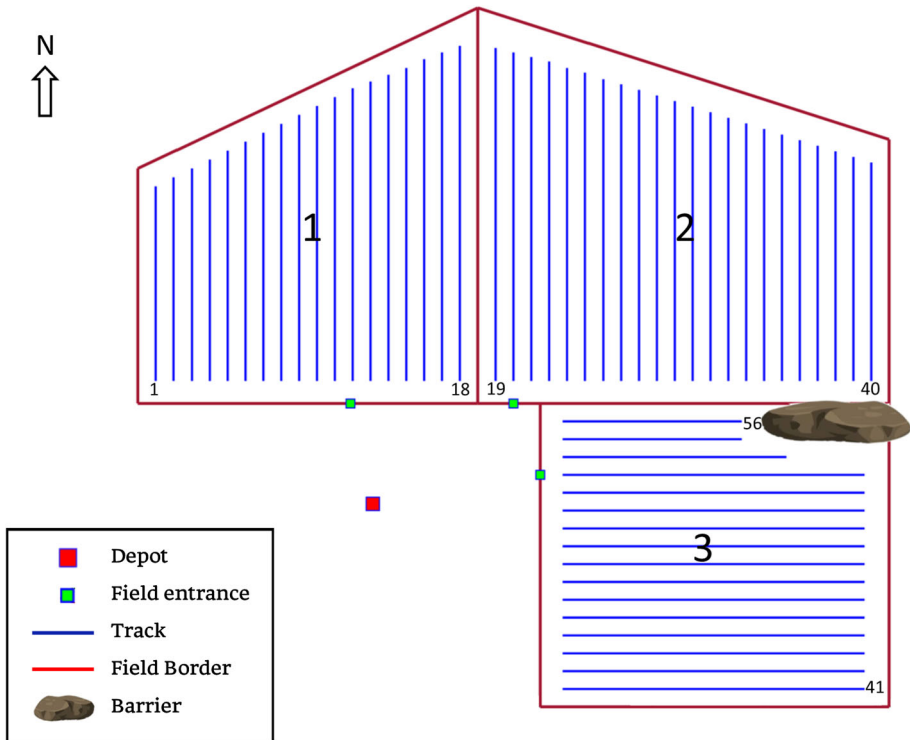


Fig. 3 Illustration of problem with three fields configurations

lated Annealing was used for route planning of autonomous vehicles in herbicide application (Conesa-Muñoz et al., 2016). Zhou et al. (2014) utilized Ant Colony Optimization (ACO) to create useful and efficient solutions for a field with obstacles as opposed to manual operations by farmers.

To conclude, there are several research gaps, as indicated earlier. First, there is very limited research on ARP involving multiple fields, and most of the ARP research utilized a single machine without capacity constraints. Hence, this study is the first to address several ARP constraints simultaneously. Second, the results of experiments conducted in this study show that the solutions given for several datasets in previous research can still be improved. Therefore, an improved, more efficient algorithm needs to be developed to enhance the quality and management of the solution.

3 Materials and methods

3.1 Problem description

Figure 3 illustrates the problem with three fields. The blue lines indicate the tracks while the red lines are the field border. Each track is differentiated by a number. Field 1 has 18 tracks (West to East, 1 to 18), Field 2 has 22 tracks (West to East, 19 to 40), and Field 3 has 16 tracks (South to North, 41 to 56). The barriers in the problem are the field borders and the rocks,

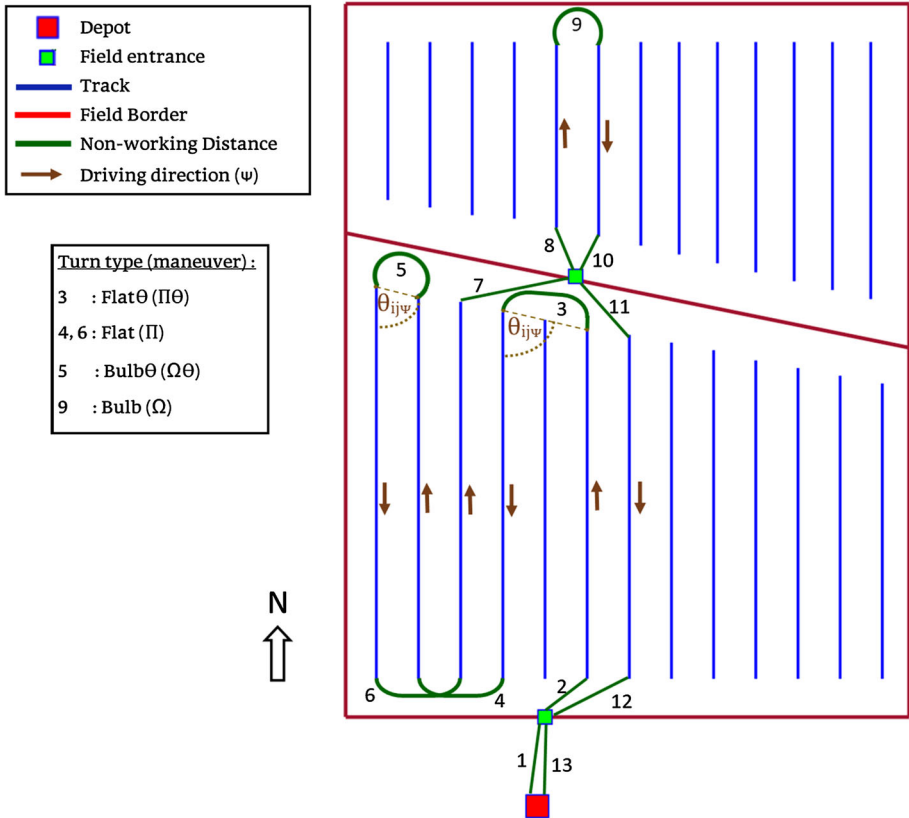


Fig. 4 Illustration of a machine’s tour and its maneuvers

which cannot be traversed by the machine. The proposed algorithm addresses the barriers by limiting the connection between different fields. Therefore, the machines can enter or exit a field or move from one field to another only from a specific location/entrance.

In Fig. 3, the machines can enter the field only through the field’s entrance (green square). The machine’s tour starts from and ends at the Depot (red square). The goal is to discover the minimum length of the non-working distance in all tours for a set of machines to harvest all tracks. The non-working distance in this study refers to the distance traveled by the machines outside the tracks when making maneuvers, going to or from the entrances and going from and returning to the Depot.

This research assumes that there are several available machines with different capacities to harvest several fields. The optimization will assign machines to the given routes, taking their capacities into consideration. If a machine reaches the maximum amount of crop it can hold, it needs to return to the Depot to unload the harvested crop. This study assumes a uniform amount of harvest per traveled distance. Hence, the machine capacity can be described as the maximum distance it can travel on tracks on one trip.

Figure 4 depicts a machine’s trip in two fields and its maneuvers. There are two adjacent fields in Fig. 4 where the field in the North can be entered only through the field in the South. The non-working distance is visualized with green lines. The numbers (near the green lines in Fig. 4) indicate the sequence of tracks that the machine uses. The non-working distances

can be differentiated by the maneuvers (3 to 6 and 9) and movements between Depot and entrances or the tracks to the entrances (1, 2, 7, 8, 10, 11, 12 and 13).

The machine starts from the Depot and traverses the south field’s entrance first (1) before harvests several tracks in the southern field (2–6). The machine then goes to the northern field’s entrance (7), harvests two tracks in the northern field (8–9), and returns to the entrance (10). Next, the machine harvests a track in the southern field (11) before going back to the Depot via the southern field’s entrance (12).

Figure 4 also depicts the four kinds of maneuvers that are considered in this study. The assumption in Fig. 4 is that the machine needs to skip at least one track to make a flat turn. Hence, if a machine’s next track is directly besides the current track, then the bulb or bulb⊖ turns will be made. In Fig. 4, the maneuvers are flat (4, 6), bulb (9), flat⊖ (3) and bulb⊖ (5). The flat⊖ and bulb⊖ turns are made when the current and next track are aligned with an acute angle θ ($\theta < 90^\circ$), while the flat and bulb turns are performed when the current and the next track are aligned with an acute angle θ ($\theta = 90^\circ$) and have the same Y-axis coordinate in the maneuver’s area. The conditions of these maneuvers are listed in Constraint 1 in Sect. 3.2 and adapted from Utamima et al. (2019) and Jin and Tang (2010).

3.2 Mathematical model

The mathematical model used to solve the extended ARP is formulated as an integer programming model. The group of tracks in the fields are assumed to be a set of vertices and edges in Graph G. Table 1 describes the parameters used in the mathematical model. The decision variables are listed in Table 2.

Equations 1–5, adapted from Jin and Tang (2010) and Utamima et al. (2019), are used to compute the distance the machine travels in order to make a turning maneuver in the headland area. The description of each notations is given in Table 1.

$$g_{ij\psi} = \begin{cases} \Pi(i, j, \theta_{ij\psi}), & \text{if } |i - j| \leq \frac{2r}{w} \wedge \theta_{ij\psi} = 90^\circ \\ \Omega(i, j, \theta_{ij\psi}), & \text{if } |i - j| > \frac{2r}{w} \wedge \theta_{ij\psi} = 90^\circ \\ \Pi\theta(i, j, \theta_{ij\psi}), & \text{if } |i - j| \geq \frac{2r}{w} \wedge \theta_{ij\psi} < 90^\circ \\ \Omega\theta(i, j, \theta_{ij\psi}), & \text{if } |i - j| > \frac{2r}{w} \wedge \theta_{ij\psi} < 90^\circ \end{cases} \tag{1}$$

$$\Pi(g_{ij\psi}) = |i - j|w + r(\pi - 2) \tag{2}$$

$$\Omega(g_{ij\psi}) = r \left(3\pi - 4 \sin^{-1} \left(\frac{2r + |i - j|w}{4r} \right) \right) \tag{3}$$

$$\Pi\theta(g_{ij\psi}) = |i - j|w(1 + \cot \theta_{ij\psi}) + r(\pi - 2) \tag{4}$$

$$\begin{aligned} \Omega\theta(g_{ij\psi}) &= \pi r + \frac{4r^2 - |i - j|w(4r + w \cot^2 \theta_{ij\psi} + w)}{4r - 2w} \\ &\times \sin^{-1} \frac{|i - j|w(4r \cot \theta_{ij\psi} - 2w \cot \theta_{ij\psi})}{4r^2 - |i - j|w(4r + w \cot^2 \theta_{ij\psi} + w)} \end{aligned} \tag{5}$$

The objective function is given in Eq. 6 which is to minimize the sum of all included edges in the solution, the distance of the tour between fields, the distance traveled by each machine to and from the Depot, and the distance between the entrance to the first and last track to be harvested. Meanwhile, Constraints 7–12 are the restrictions related to the tour between tracks inside a field. Constraint 7 and 8 specify that each route should start and end at the Depot (Vertex 0). Constraints 9 and 10 ensure that each node is visited only once.

Table 1 Parameters

Notations	Description
G	Graph $G = \{V, E\}$ representing the tracks in the fields
V	Set of vertices in the graph G
S	Subgraph of graph G , $VS \subseteq V$
E	Set of edges of the graph G
i, j	Vertex index ($i, j = (0), 1, \dots, n$), 0 is the Depot
F	Set of fields ($f = 1, \dots, F$)
a, b	Fields index ($a, b = 1, \dots, F$)
d_{ab}	The distance between two fields (from entrance field a to entrance of field b)
h_{ab}	A binary parameter, equal to 1 if there is a direct entrance from field a to field b ($a, b \in F$), otherwise 0
D_f	The distance from field f to Depot
T	Set of tracks ($t = 1, \dots, T$) in a field
δ_{ft}	The distance of track t to the entrance of field f
l_{ft}	The length of track t in field f
K	The set of machines available at the Depot, each referenced as k , where $k \in K$
r	Minimum turning radius of a machine
w	Effective operating width of a machine
ψ	The driving direction of the machine which is a binary variable that takes two values of 0 and 1
$g_{ij\psi}$	A maneuver's length between track i and j and driving direction ψ and angle $\theta_{ij\psi}$
$\theta_{ij\psi}$	The degree of tilt between track i and track j and driving direction ψ , note that $0 < \theta_{ij\psi} \leq 90$
$\Pi(i, j, \theta_{ij\psi})$	The flat type of maneuver turn executed by a machine in the headland area
$\Omega(i, j, \theta_{ij\psi})$	The bulb type of maneuver turn executed by a machine in the headland area
$\Pi\theta(i, j, \theta_{ij\psi})$	The flat type of maneuver turns with degree, executed by a machine in the headland area if the previous track and the next track are aligned with an angle θ
$\Omega\theta(i, j, \theta_{ij\psi})$	The bulb type of maneuver turns with degree, executed by a machine in the headland area if the previous track and the next track are aligned with an angle θ
Q_k	The capacity of a machine $k \in K$
P_{ft}	The volume of crops harvested in track t of field f

Table 2 Decision variables

Notations	Description
x_{fij}^k	Equal to 1 if machine moves from vertex i to vertex j in field f , otherwise 0 ($i, j \in E$)
y_{ab}	Equal to 1 if there is a tour from field a to field b , otherwise 0, ($a, b \in F$) After harvesting field a then the tour continues to b
z_{ft}	Equal to 1 if there is a tour from the entrance of field f to track t and vice versa, otherwise 0
s_f^k	Equal to 1 if machine k 's first visited track is on field f , otherwise 0
τ_f^k	Equal to 1 if machine k 's last visited track is on field f , otherwise 0

Constraint 11 ensures that when a machine enters a vertex, it will also leave that vertex. Constraint 12 is a sub-tour limitation that eliminates any disjoint sub-tours from a solution. Then, Constraint 13 ensures that the amount of harvested crop does not exceed the capacity of a machine. The distance traveled inside the track has the same dimension as the capacity of a machine. The assumption is that when a machine traverses a track, it completely harvests that track, so the maximum distance inside the track will depend on the maximum capacity of a machine. Finally, Constraints 14 and 15 ensure that each field is visited only once.

$$z = \text{Min} \left(\sum_{f \in F} \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} g_{ij\psi} x_{fij}^k + \sum_{a \in F} \sum_{b \in F} d_{ab} y_{ab} h_{ab} \right. \\ \left. + \sum_{f \in F} \sum_{k \in K} D_f (s_f^k + \tau_f^k) + \sum_{f \in F} \sum_{t \in T} z_{ft} \delta_{ft} \right) \tag{6}$$

subject to:

$$\sum_{j \in V} x_{f0,j}^k = 1, \quad \forall k \in K, \quad \forall f \in F \tag{7}$$

$$\sum_{i \in V} x_{fi,0}^k = 1, \quad \forall k \in K, \quad \forall f \in F \tag{8}$$

$$\sum_{k \in K} \sum_{i \in V} x_{fij}^k = 1, \quad i, j \neq 0, \quad \forall j \in V : i \neq j, \quad \forall f \in F \tag{9}$$

$$\sum_{k \in K} \sum_{j \in V} x_{fij}^k = 1, \quad i, j \neq 0, \quad \forall i \in V : i \neq j, \quad \forall f \in F \tag{10}$$

$$\sum_{k \in K} \sum_{i \in V} x_{fij}^k = \sum_{k \in K} \sum_{j \in V} x_{fji}^k, \quad i, j \neq 0, \quad \forall j \in V, \quad \forall f \in F \tag{11}$$

$$\sum_{i \in S} \sum_{j \in S} x_{fij}^k \leq \|S\| - 1, \quad \forall S \subseteq V, \quad \|S\| \geq 1, \quad \forall k \in K, \quad \forall f \in F \tag{12}$$

$$\sum_{f \in F} \sum_{i \in V} \sum_{j \in V} P_{ft} x_{fij}^k < Q_k, \quad \forall k \in K, \quad \forall t \in T \tag{13}$$

$$\sum_{a \in F} y_{ab} = 1, \quad \forall b \in F : a \neq b \tag{14}$$

$$\sum_{b \in F} y_{ab} = 1, \quad \forall a \in F : a \neq b \tag{15}$$

The total distance traveled for harvesting the crop fields is found with Eq. 16 which is the sum of the distance between the fields, the length of each track in each field, and the objective function.

$$\text{dist} = \sum_{f \in F} \sum_{t \in T} l_{ft} + z \tag{16}$$

3.3 Proposed algorithm

This study builds a new hybrid algorithm, named ENDA, to solve ARP. Algorithm 1 lists the ENDA pseudocode. ENDA has two main stages: the exploration stage that searches the solution among the individuals in a population, and the exploitation stage that searches

Fig. 5 Representation of individual

Order	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]
Tracks	7	4	1	6	2	8	3	5

for the best solution found so far. ENDA is a hybrid of three kinds of heuristics, namely distribution search, inter-neighborhood discovery, and intra-neighborhood discovery. The exploration stage consists of distribution search and inter-neighborhood discovery, while the exploitation stage contains the intra-neighborhood discovery procedure. The individuals are the candidate solution to the problem; the population refers to the group of individuals in the algorithm. The representation of an individual with 8 tracks is illustrated in Fig. 5.

Algorithm 1 The pseudocode of ENDA

```

1: procedure ENDA()
2:   Initialization_Read_Data()
3:   for g=1 to max_generation do
4:     OF ← Calculate_ObjectiveFunction()
5:     if in DistributionSearch_rate then
6:       Selected_Individuals ← Choose_BetterObjectives(OF)
7:       NewPopulation ← DistributionSearch(Selected_Individuals)
8:     else
9:       for i=1 to discovery_iteration do
10:        S ← RouletteWheel()
11:        [P] ← RandSelect_two_points()
12:        c ← Random(3)
13:        switch c do
14:          case 1:
15:            Flip(S,P)
16:          case 2:
17:            GuidedSwap(S,P)
18:          case 3:
19:            Inversion(S,P)
20:        end for
21:        NewPopulation ← Update_Population()
22:      end if
23:      OF ← Calculate_ObjectiveFunction(NewPopulation)
24:      currentBest ← Min(OF)
25:      if currentBest < globalBest then
26:        globalBest ← currentBest
27:      else
28:        ndSol ← globalBest
29:        while stuck ← false do
30:          ndSol ← Intra_NeighborhoodDiscovery()
31:          if ndSol > globalBest then
32:            stuck ← true
33:          end if
34:        end while
35:      end if
36:      if ndSol < globalBest then
37:        globalBest ← ndSol
38:      end if
39:      Elitism()
40:    end for
41: end procedure
    
```

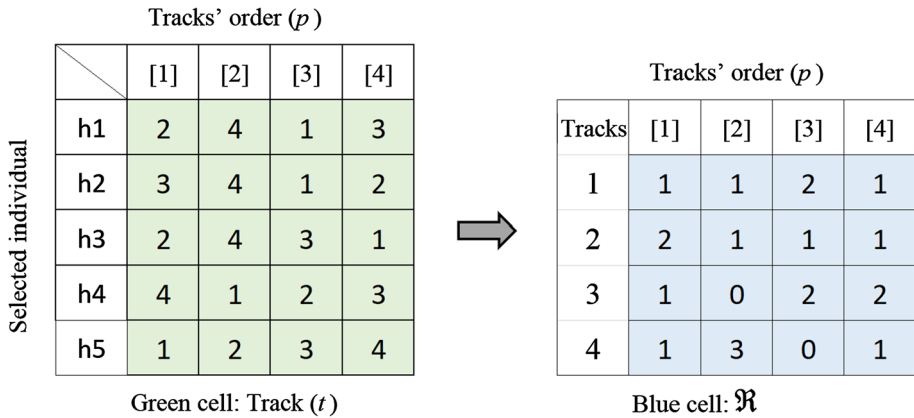


Fig. 6 Illustration of the construction of \mathfrak{R}

First, the parameters of the ENDA are initialized. The parameter settings are explained in more detail in Sect. 4.3. The details of the machines (number of machines, their various capacities, and turning radius) are also initialized at this point. Then, the algorithm reads the field data that contains the track coordinates, the distance between each entrance and the Depot, and the length of each track. The iterations start with the computation of the objective function of every individual. The exploration stage of ENDA will perform the distribution search according to either the rate or the neighborhood discovery.

The distribution search (Algorithm 1, Line 5–7) will build the new population according to the probability of the order of tracks for the selected individuals. The distribution search will run if the `DistributionSearch_rate` (the probability rate to execute the procedure) is met. The selected individuals are the group of individuals with a better objective function (H is set of selected individuals) that are labeled $h_1, h_2, h_3, \dots, h_n (h \in H)$. Equation 17 calculates the sum of every track being placed in an order. The graphical illustration of the calculation of \mathfrak{R} is shown in Fig. 6. In Eq. 17, Γ_{tp}^h is set to 1 if track t is visited at order p , otherwise, it is set to 0 ($t \in T, p \in P$; P is set of tracks' order). The size of T and P is equal to the number of tracks in the fields. The distribution search function will update each individual in the population by selecting the order of track proportionally according to \mathfrak{R} .

Next, the inter-neighborhood discovery procedure is listed in Line 9–20 of Algorithm 1. The procedure is adapted from Hansen et al. (2010) and Utamima et al. (2019). The procedure runs for `discovery_iteration` times. At the beginning of every iteration, the roulette wheel selection is performed, and an individual is selected (S). The roulette wheel selection expects that the higher the fitness of an individual implies the greater is the chance of its survival. The fitness function that is used in the Roulette Wheel selection is listed in Eq. 18 which is the inverse of the objective function (mentioned in Eq. 1). The chance of selection in this approach is proportional to the fitness of an individual.

$$\mathfrak{R}_{tp} = \sum_{h \in H} \sum_{t \in T} \sum_{p \in P} \Gamma_{tp}^h \tag{17}$$

$$fn = \frac{1}{z} \tag{18}$$

Table 3 Description of operators used in ENDA

No	Operators	Description
1	Flip	Flips the order of tracks between two points
2	Guided swap	Choose a track, then find its position in global best solution Then, place the selected track with the same position of global best by swapping with the other track in that position
3	Inversion	Choose two points, then moves the first point to the second point and slides forward the remaining points (starting from the first point + 1 until the second point)
4	Interchange	Interchange two elements randomly
5	Reverse	Reverse the tracks between two points

After selection, the algorithm chooses two points randomly from S and performs between flip, guided swap, or inversion operators. The details of the operators are listed in Table 3. Next, the new population is updated from the results of the inter-neighborhood discovery.

The objective function calculation evaluates the fitness of the new population and finds the current best solution in the population (Algorithm 1, Line 23). This solution is checked to determine whether it can replace the global best solution (the best solution found so far among generations). If the current best solution is not better than the global best solution (*globalBest*), then the second stage of ENDA is implemented (Algorithm 1, Line 28–33). In this stage, the intra-neighborhood discovery searches inside the *globalBest* to find a better solution. The procedure is adapted from the local search in Song et al. (2019). In this stage, the interchange and reverse operators are employed until no further improvement can be made. The operators are described in Table 3.

If the solution of the intra-neighborhood discovery (*ndSol*) is better than *globalBest*, then *ndSol* will become the *globalBest*. Next, the Elitism procedure is run to retain ten percent of the best individuals through generations in order to protect prospective solutions (Nouri and Ladhari, 2018). The recorded individuals are copied to the new population and are sent to the next generation.

3.4 Genetic algorithm and Tabu search

This study also re-coded GA and TS to solve the extended ARP problem set. GA and TS are chosen because both algorithms have been implemented recently in optimization in the agriculture domain (Florentino et al., 2018; Seyyedhasani and Dvorak, 2018a). The input and the fitness calculation are configured to meet the standard procedure of the algorithms. The number of generation and the size of population in GA and TS are set to be the same as those in ENDA.

Generally, GA starts from the generation of a new population and continues with the selection of chromosomes with a better fitness value. The higher the fit of a chromosome, the higher its chances to be selected. Crossover and mutation are applied to the chosen chromosomes with a given probability rate. The crossover process cuts two chromosomes at one point, and the halves are spliced together to create new chromosomes. Meanwhile, the mutation process changes the value of genes in a small group of chromosomes. After crossover and mutation, the fitness evaluation detects the best solution, which is recorded.

Table 4 The solution comparison of GAMS and ENDA

Problem	GAMS	GAMS's tour	ENDA	ENDA's tour
A	179.713	1: Field 1 [3, 1, 4, 2]; 2: Field 2 [6, 5]	179.713	1: Field 1 [3, 1, 4, 2]; 2: Field 2 [6, 5]
B	225.413	1: Field 1 [5, 3, 1, 2, 4, 6]; 2: Field 2 [9, 7, 10, 8]	225.413	1: Field 1 [5, 3, 1, 2, 4, 6]; 2: Field 2 [9, 7, 10, 8]

The whole process is repeated until a stopping criterion (i.e. maximum iteration) is met. The process of evolution in a population of chromosomes over multiple generations represents a search for a good and feasible solution.

Meanwhile, TS constructs several moves (candidate solution) in the neighborhood. The representation of the neighborhood structure in S is like the one provided in Table 5. A move in TS is using Swap techniques between two points (randomly chosen) in the neighborhood.

Next, TS picks a move and checks whether the move is tabu (the move is tabu if it is listed in the tabu list). If the move is not tabu, TS accepts the move. Otherwise, TS checks the aspiration criteria of the move. If a tabu move meets the aspiration criteria, the move will be accepted, otherwise, the move will be refused. If the new solution from an accepted move is better than the current solution, the solution will be updated. The decision process will be repeated for a number of iterations (Ji and Tang, 2004).

4 Experimental results and analysis

4.1 Validation of the mathematical model

In order to validate the mathematical model, it was implemented using General Algebraic Modeling System (GAMS) software. Two small problems are tested, and consist of multiple fields and multiple machines with different capacities. Each problem involves two fields with tracks measuring 4 meters in width, and two machines available with a turning radius of 3 meters.

Table 4 shows a comparison of the results of GAMS and ENDA (coded with MATLAB). Both GAMS and ENDA can obtain the same solution for the two problems as well as the same tours of machine 1 and machine 2.

The experiments show that the running time of ENDA is six times faster than that of GAMS. If the size of the problem is increased, the running time is expected to grow exponentially for the GAMS implementation. Therefore, the utilization of ENDA is required for both small and large problems because it can achieve a good solution with a faster running time.

Based on previous research by Bochtis and Sørensen (2009) and Utamima et al. (2018), ARP is similar to Vehicle Routing Problem (VRP) in the agricultural field. VRP itself is an NP-Complete problem (Toth and Vigo, 2002), hence ARP can also be considered as NP-complete. Therefore, finding exact solutions as done by GAMS is restricted to smaller instances as the running time is growing exponentially. Metaheuristics and evolutionary algorithms like ENDA are used on larger instances to achieve near-optimal solutions with short running times.

4.2 Problem sets description

The experiments conducted in this study are comprised of two steps. The first step involves the validation of the proposed algorithm by using the data in the literature. The datasets are gathered from published studies and by contacting the corresponding authors. Table 5 shows a description of the data. In Table 5, the first and second columns are the instance code and the total number of tracks in that instance, respectively. The track widths and lengths (meters), turning radii (meters), shapes, and machines used are listed in Column 3 to 6 in Table 5. The last column in Table 5 contains the references to the problem instances, and the details are listed immediately below the table.

In the second step, experiments are run using the extended ARP datasets. Table 6 gives the attributes of the datasets that are used in this study. There are seven datasets with multiple fields and different shapes. The problems are generated and vary in terms of the total number of tracks, their width (meters), and turning radius (meters). The number of fields ranges from two to five, with total tracks ranging from 30 to 112. The machines used (Table 7) have different capacities (ranging from 700 to 3000 m). As indicated earlier, this study assumes a uniform measure of harvest per traveled distance. Hence, the machine capacity is depicted with the maximum distance it can travel on one trip.

Figure 7 illustrates the seven problem instances: 2A(a), 2B(b), 3A(c), 3B(d), 4A(e), 4B(f), and 5A(g). The number in the middle of each field is the field number, and the small number near the track is the track number. These numbers are used to give a better understanding of the output of the program that is shown later in Table 12.

4.3 Parameters settings

The parameter set-up is acquired by directing a two-level factorial plan with four elements. Each factor joins high and low dimensions (Montgomery, 2013). The algorithm is run ten times to make up for the non-deterministic nature of the heuristics (Guan and Lin, 2016). Table 8 lists factors and the levels of each set-up. To adjust to the extent of the problem, the settings use n , which is the number of tracks in the fields. A larger problem requires a bigger

Table 5 Description of previous datasets

Instance code	Total tracks	Track width	Track length	Turning radius	Field shape	Machines	References
8rt	8	2.89	30	3.5	Rectangular	1	Bochtis and Vougioukas (2008)
12rta	12	2.5	40	3.5	Rectangular	1	Conesa-Muñoz et al. (2016)
12rtb	12	2.5	70	3.5	Rectangular	1	Conesa-Muñoz et al. (2016)
20rt	20	2.5	80	3.5	Rectangular	1	Conesa-Muñoz et al. (2016)
37nc	37	9	Vary	6	Non-convex	1	Conesa-Muñoz et al. (2016) and Hameed et al. (2011)
74nc	74	19	Vary	9.77	Non-convex	3 (Homogeneous)	Seyyedhasani and Dvorak (2018a)

Table 6 Description of the extended ARP problem set

No	Problem code	Number of fields	Total tracks	Tracks width	Turning radius	Shape
1	2A	2	30	8	7.5	Non-convex
2	2B	2	36	8	7.5	Non-convex
3	3A	3	56	7	5.5	Non-convex and rectangular
4	3B	3	62	6	4.8	Non-convex
5	4A	4	78	6	4.8	Non-convex and rectangular
6	4B	4	100	5	3.9	Non-convex and rectangular
7	5A	5	112	5	3.9	Non-convex

Table 7 Machines provided

Machines	Maximum distance when harvesting	Availability
1	3000	1
2	2500	2
3	2200	3
4	700	1

population and more generations. The value shown in bold indicates the better setting for ENDA. The selected settings for the number of generations and the size of the population are also applied to GA and TS.

4.4 Numerical results

The results of the conducted experiments are listed in Tables 9, 10, 11 and 12. Table 9 shows details of the first step of the experiments, while Table 10 lists the results of the second step of the tests. Table 11 presents the running time of all algorithms when solving the second step of the experiments. Table 12 lists the solution of ENDA in the second step of the tests that contains the order of tracks in the fields.

Table 9 shows details of the non-working distance of six published data. In Table 9, the first and second columns show the problem code and the associated references. The third column is the best-known solution to date obtained from the latest publication (Conesa-Muñoz et al., 2016; Seyyedhasani and Dvorak, 2018a), while the last column shows the results obtained by ENDA. The bold values in the third and fourth columns refer to the minimum value in that row. As shown by the results, ENDA can obtain the same solution for four instances (8rt, 12rta, 12rtb, 20rt) and achieve better solutions in two instances (37nc, 74nc). These results indicate that ENDA can achieve the expected results for ARP instances.

Table 10 presents the non-working distance of multiple field data with multiple machines of the extended ARP data. In Table 10, the first column lists the problem code, the second column specifies the number of tours required to complete all tracks in the fields, and the third, fourth, and fifth columns list the solutions obtained by ENDA, GA, and TS, respectively. The

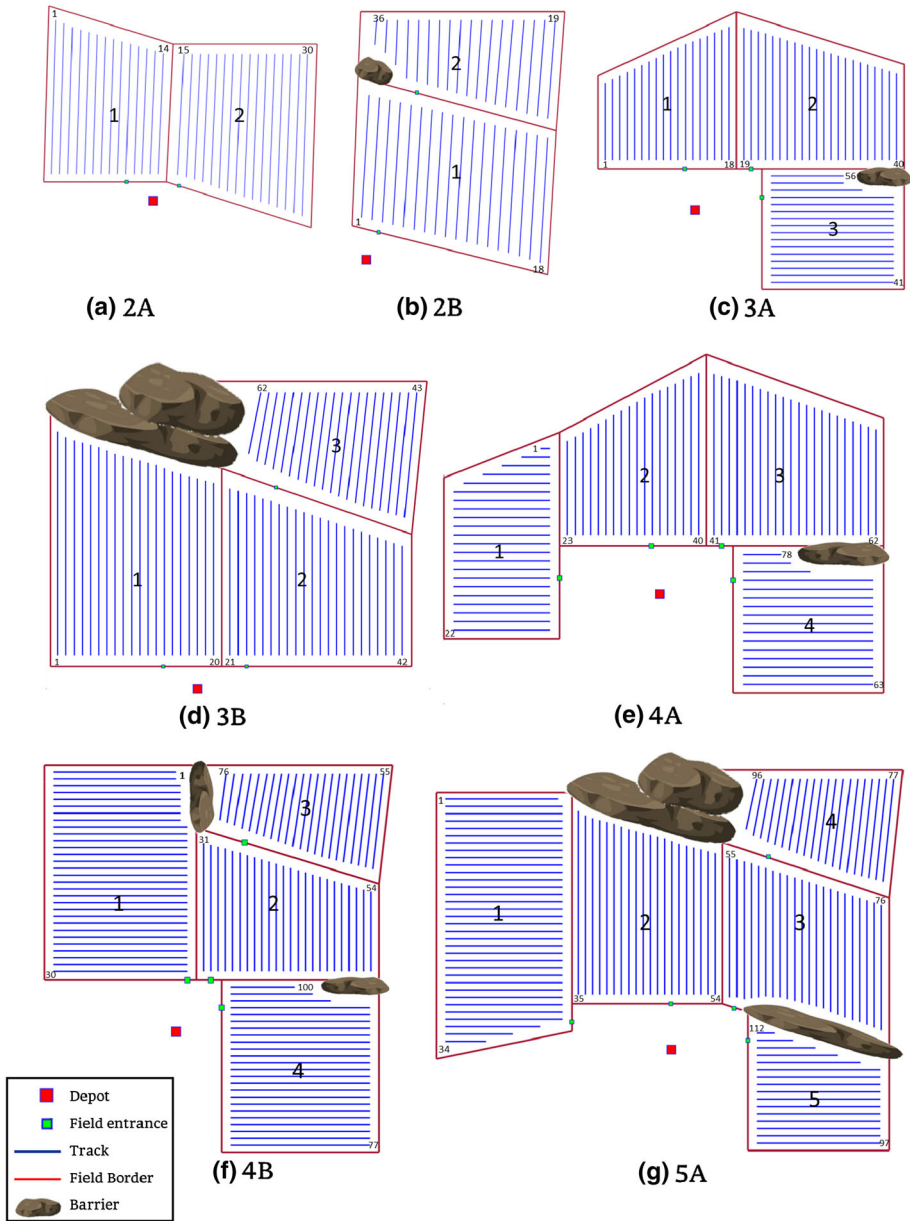


Fig. 7 Illustration of Problem 2A to 5A (a–g)

values in bold refers to the minimum value in that row. The last column in Table 10 is the average distance savings obtained by ENDA compared to those of GA and TS for every problem, while the last row in bold is the average of the distance savings for all problems. ENDA successfully achieves the shortest non-working distance for all problem sets compared to the performances of GA and TS. The GA solutions are better than TS in four instances

Table 8 Factor settings for ENDA

Factors	Levels	
	Low (n)	High (n)
Generations	40	50
Population size	4	5
Discovery iteration	2	2.5
DistributionSearch_rate	0.05	0.1

Table 9 The non-working distance of the published data (single field)

Problem	References*	Best-known solution	ENDA
8rt	Bochtis and Vougioukas (2008) and Conesa-Muñoz et al. (2016)	94.439	94.439
12rta	Bochtis and Vougioukas (2008) and Conesa-Muñoz et al. (2016)	146.027	146.027
12rtb	Bochtis and Vougioukas (2008) and Conesa-Muñoz et al. (2016)	145.602	145.602
20rt	Bochtis and Vougioukas (2008) and Conesa-Muñoz et al. (2016)	235.491	235.491
37nc	Conesa-Muñoz et al. (2016) and Hameed et al. (2011)	961.470	958.930
74nc	Seyyedhasani and Dvorak (2018a)	4416.300	3869.590

Table 10 The non-working distance of the extended ARP problem set (multiple fields)

Problem code	Number of tours	GA	TS	ENDA	Distance savings (%)
2A	2	1051.118	1029.329	955.869	8.10
2B	2	1101.292	1279.074	1044.038	11.79
3A	3	1121.185	1013.578	967.872	9.09
3B	3	2018.031	2333.044	1909.220	11.78
4A	4	1776.730	1645.868	1504.245	11.97
4B	6	2923.140	3063.433	2605.433	12.91
5A	6	4145.357	4782.591	3711.986	16.42
				Average:	11.72

Table 11 The running time of ENDA, GA, and TS for all problem set

Problem code	Number of tours	Average running time (s)		
		GA	TS	ENDA
2A	2	1.388	1.359	0.807
2B	2	6.842	9.171	4.565
3A	3	4.142	2.948	2.179
3B	3	12.264	16.986	7.974
4A	4	9.527	7.459	5.283
4B	6	25.608	23.814	11.942
5A	6	26.438	25.629	13.663

(2B, 3B, 4B, 5A), while the TS solutions for the remaining three instances (2A, 3A, 4A) are better than those obtained by GA. In terms of average distance savings, ENDA successfully saves 11.72% of the non-working distance compared to other algorithms.

Table 11 shows the running time of all the algorithms for each problem set. The running time of ENDA is the fastest compared to those of GA and TS. On the other hand, the runtimes

Table 12 The details of every tour in each problem set obtained by ENDA

Problem	Tours (max in-track distance)	Order of tracks in fields [field number (tracks's order)]
2A	1 (3000)	Field 1 [10, 8, 6, 4, 2, 1, 3, 5, 7, 9, 13, 14, 12, 11]; Field 2 [29, 27, 28, 30]
	2 (2500)	Field 2 [24, 26, 25, 23, 22, 20, 21, 19, 18, 16, 17, 15]
2B	1 (3000)	Field 1 [2, 1, 3, 6, 10, 11, 8]; Field 2 [33, 35, 36, 34, 32, 30, 28, 26, 24, 22, 21, 19, 20, 23, 25, 27, 29, 31]; Field 1 [4]
	2 (2500)	Field 1 [5, 7, 9, 12, 13, 14, 15, 16, 17, 18]
3A	1 (3000)	Field 1 [13, 15, 17, 18, 16, 14, 12, 11, 8, 7, 4, 3, 1, 2, 5, 6, 9, 10]; Field 2 [40, 39, 36, 35, 37, 38]
	2 (2500)	Field 2 [29, 31, 33, 34, 32, 30, 28, 27, 25, 23, 20, 19, 21, 22, 24, 26]; Field 3 [54, 56, 55, 53, 50, 52]
	3 (2500)	Field 3 [51, 49, 47, 45, 48, 46, 44, 42, 41, 43]
3B	1 (3000)	Field 1 [13, 11, 9, 7, 5, 3, 1, 2, 4, 6, 8, 10, 12, 14, 16, 15]
	2 (2500)	Field 1 [17, 18, 20, 19]; Field 2 [25]; Field 3 [52, 50, 51, 48, 49, 47, 45, 43, 44, 46, 42, 41, 39, 40]; Field 2 [23, 22, 21]
	3 (2500)	Field 2 [24]; Field 3 [57, 54, 53, 55, 56, 58, 60, 62, 61, 59]; Field 2 [26, 28, 27, 29, 30, 32, 31, 36, 35, 37, 38, 34, 33]
4A	1 (3000)	Field 1 [5, 3, 1, 2, 4, 6, 8, 10, 12, 14, 16, 18, 21, 19, 22, 20, 17, 15, 13, 11, 9, 7]; Field 2 [33, 34, 36, 38, 40, 39, 37, 35]
	2 (2500)	Field 2 [32, 31, 29, 30, 28, 27, 24, 23, 25, 26]; Field 3 [61, 62, 59, 60, 58, 57, 55, 56, 54, 53]
	3 (2500)	Field 3 [51, 52, 50, 49, 47, 48, 46, 45, 43, 44, 42, 41]; Field 4 [74, 72, 69, 66, 64, 67, 70, 77, 78, 76]
	4 (700)	Field 4 [75, 73, 71, 68, 65, 63]
4B	1 (3000)	Field 1 [1, 5, 7, 12, 15, 19, 22, 25, 27, 29, 30, 28, 26, 24, 21, 17, 13, 11, 9, 3]
	2 (2500)	Field 1 [2, 4, 6, 8, 10, 14, 16, 18, 20, 23]; Field 2 [37, 40, 42, 41, 38] Field 3 [70, 72]; Field 2 [32]
	3 (2500)	Field 2 [31, 33, 35]; Field 3 [74, 76, 75, 73, 71, 68, 66, 61, 59, 64, 57, 55, 56, 58, 60, 62, 63, 65, 67, 69]; Field 2 [34]
	4 (2200)	Field 2 [36, 44, 46, 48, 50, 52, 54, 53, 51, 49, 47, 45, 43, 39]; Field 4 [98, 100, 99, 97, 94, 96]
	5 (2200)	Field 4 [92, 90, 87, 85, 83, 81, 79, 77, 78, 80, 82, 84, 88, 95]
	6 (700)	Field 4 [93, 91, 89, 86]

Table 12 continued

Problem	Tours (max in-track distance)	Order of tracks in fields [field number (tracks's order)]
5A	1 (3000)	Field 1 [28, 21, 19, 17, 15, 13, 11, 9, 7, 5, 3, 1, 2, 4, 6, 8, 10, 12, 14, 16, 18, 30]
	2 (2500)	Field 1 [31, 34, 32, 33, 29, 27, 25, 23, 20, 22, 24, 26]; Field 2 [47, 45, 43, 44, 46, 48]
	3 (2500)	Field 2 [49, 51, 53, 54, 52, 50, 42, 41, 39, 37, 35, 36, 38, 40];
	4 (2200)	Field 3 [59]; Field 4 [92, 94, 93, 96, 95, 90, 89, 87, 85, 82, 80, 78, 77, 79, 81, 83, 84, 86, 88, 91]; Field 3[58]
	5 (2200)	Field 3 [62, 64, 66, 68, 70, 72, 74, 76, 75, 73, 71, 69, 67, 65, 60, 57]
	6 (2200)	Field 3 [55, 56, 61, 63]; Field 5 [112, 110, 108, 106, 104, 102, 100, 98, 97, 99, 101, 103, 105, 107, 109, 111]

of TS are faster than GA in five instances, while those for two problems (2B, 3B) are slower than GA.

Table 12 lists the details of the ENDA solution for each problem set. In Table 12, the first column lists the problem code, and the second column shows the tours and the maximum in-track distance of the machine. The last column is the order of tracks in the fields and lists the field number and the order of the tracks that correspond to Fig. 7. For example, problem 2A needs two machines (with maximum distance inside the tracks are 3000m and 2500m, respectively) to harvest the fields. The first machine goes to Field 1 and harvests 14 tracks in the order of [10, 8, 6, 4, 2, 1, 3, 5, 7, 9, 13, 14, 12, 11] and then goes to Field 2 and harvests four tracks [29, 27, 28, 30]. Next, the second machine goes to Field 2 and harvests 12 tracks in the order of [24, 26, 25, 23, 22, 20, 21, 19, 18, 16, 17, 15]. Note that every machine must start from the Depot and return to the Depot.

Figure 8 illustrates the maneuver of the machines in Problem 5A. In Fig. 8, the field number is in the center of each field, and the track numbers are indicated by the small number in the corner of each field. Each of the six tours of the machines is represented by a different colored line. In this problem, the turning radius (r) is 3.9 and the tracks' width (w) is 5. Thus, according to Constraint 1 in Sect. 3.2, the machine can perform a flat turn by skipping at least one track in order to go to the next track. As shown in Fig. 8, the predominant type of maneuver is the flat turn.

5 Conclusion

It is anticipated that this research will improve the planning of conventional machinery systems by producing better routing designs. The reduction of resources such as machines, fuel, and personnel can result in a competitive advantage and access to markets in a lower price sector. Subsequently, this contributes to addressing the issue of sustainability in agriculture. The scope of this research comprises ARP for a harvesting problem with the tracks inside several fields on a flat surface. This study also assumes that the machine capacity is the maximum distance covered on tracks during the trip.

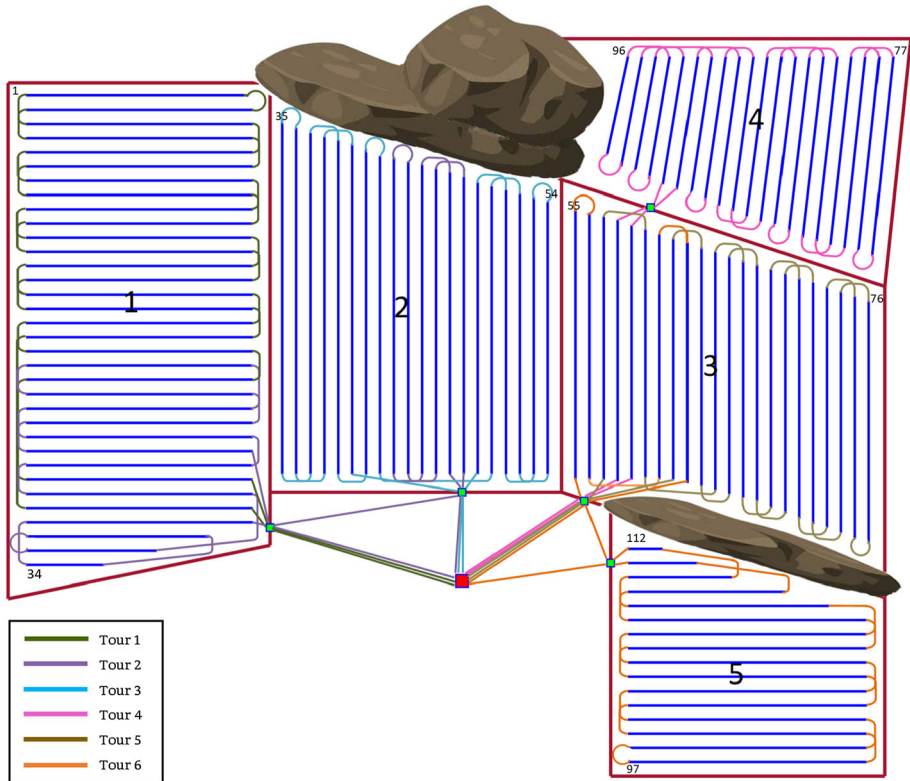


Fig. 8 Illustration of the tours and the maneuvers in five fields

This study presents the extended ARP by considering multiple constraints simultaneously in order to better generalize the harvesting problem. This research also presents seven problem sets for the extended ARP. Furthermore, it recommends the application of a new hybrid algorithm (ENDA) that outperforms GA and TS. The experimental results demonstrate that during the validation process with the previous ARP dataset, ENDA can achieve the same best solution in four instances, and produces a better solution for the other two instances. Moreover, ENDA successfully achieves the best objective function by saving an average of 11.72% of non-working distance compared to other algorithms in the seven problem sets of the extended ARP. ENDA also achieves the fastest running time for all the given problems.

The validation of the mathematical model with GAMS software is also provided in this research. Future research can focus on more experiments regarding the capacities of machines with different field layouts.

Acknowledgements The authors gratefully acknowledge financial support from the Institut Teknologi Sepuluh Nopember for this work, under project scheme of the Publication Writing and IPR Incentive Program (PPHKI).

Funding Open Access funding enabled and organized by CAUL and its Member Institutions.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the

article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Aliano Filho, A., de Oliveira, Florentino H., Pato, M. V., Poltroniere, S. C., & da Silva Costa, J. F. (2019). Exact and heuristic methods to solve a bi-objective problem of sustainable cultivation. *Annals of Operations Research*. <https://doi.org/10.1007/s10479-019-03468-9>.
- Alonso-Ayuso, A., Escudero, L. F., Guignard, M., Quinteros, M., Quinteros, M., & Weintraub, A. (2011). Forestry management under uncertainty. *Annals of Operations Research*, 190(1), 17–39. <https://doi.org/10.1007/s10479-009-0561-0>
- Bakhtiari, A., Navid, H., Mehri, J., Berruto, R., & Bochtis, D. D. (2013). Operations planning for agricultural harvesters using ant colony optimization. *Spanish Journal of Agricultural Research*, 11(3), 652–660. <https://doi.org/10.5424/sjar/2013113-3865>
- Barrientos, A., Colorado, J., Cerro, J. D., Martinez, A., Rossi, C., Sanz, D., & Valente, J. (2011). Aerial remote sensing in agriculture: A practical approach to area coverage and path planning for fleets of mini aerial robots. *Journal of Field Robotics*, 28(5), 667–689. <https://doi.org/10.1002/rob.20403>
- Bochtis, D., & Vougioukas, S. (2008). Minimising the non-working distance travelled by machines operating in a headland field pattern. *Biosystems Engineering*, 101(1), 1–12. <https://doi.org/10.1016/j.biosystemseng.2008.06.008>
- Bochtis, D., Griepentrog, H. W., Vougioukas, S., Busato, P., Berruto, R., & Zhou, K. (2015). Route planning for orchard operations. *Computers and Electronics in Agriculture*, 113, 51–60. <https://doi.org/10.1016/j.compag.2014.12.024>
- Bochtis, D. D., & Sørensen, C. G. (2009). The vehicle routing problem in field logistics part I. *Biosystems Engineering*, 104(4), 447–457. <https://doi.org/10.1016/j.biosystemseng.2009.09.003>
- Conesa-Muñoz, J., Bengochea-Guevara, J. M., Andujar, D., & Ribeiro, A. (2016). Route planning for agricultural tasks: A general approach for fleets of autonomous vehicles in site-specific herbicide applications. *Computers and Electronics in Agriculture*, 127, 204–220. <https://doi.org/10.1016/j.compag.2016.06.012>
- Edwards, G. T., Hinge, J., Skou-Nielsen, N., Villa-Henriksen, A., Sørensen, C. A. G., & Green, O. (2017). Route planning evaluation of a prototype optimised infield route planner for neutral material flow agricultural operations. *Biosystems Engineering*, 153, 149–157. <https://doi.org/10.1016/j.biosystemseng.2016.10.007>
- Florentino, H. O., Irawan, C., Aliano, A. F., Jones, D. F., Cantane, D. R., & Nervis, J. J. (2018). A multiple objective methodology for sugarcane harvest management with varying maturation periods. *Annals of Operations Research*, 267(1), 153–177. <https://doi.org/10.1007/s10479-017-2568-2>
- Gracia, C., Velázquez-Martí, B., & Estornell, J. (2014). An application of the vehicle routing problem to biomass transportation. *Biosystems Engineering*, 124, 40–52. <https://doi.org/10.1016/j.biosystemseng.2014.06.009>
- Guan, J., & Lin, G. (2016). Hybridizing variable neighborhood search with ant colony optimization for solving the single row facility layout problem. *European Journal of Operational Research*, 248(3), 899–909. <https://doi.org/10.1016/j.ejor.2015.08.014>
- Hameed, I. A., Bochtis, D. D., & Sorensen, C. G. (2011). Driving angle and track sequence optimization for operational path planning using genetic algorithms. *Applied Engineering in Agriculture*, 27(6), 1077–1086. <https://doi.org/10.13031/2013.40615>.
- Hansen, P., Mladenović, N., & Moreno Pérez, J. A. (2010). Variable neighbourhood search: Methods and applications. *Annals of Operations Research*, 175(1), 367–407. <https://doi.org/10.1007/s10479-009-0657-6>
- Haupt, R. L., & Haupt, S. E. (2004). *Practical genetic algorithms* (2nd ed.). Wiley-Interscience.
- Jensen, M. F., Bochtis, D., & Sørensen, C. G. (2015). Coverage planning for capacitated field operations, part II: Optimisation. *Biosystems Engineering*, 139, 149–164. <https://doi.org/10.1016/j.biosystemseng.2015.07.002>
- Ji, M., & Tang, H. (2004). Global optimizations and Tabu search based on memory. *Applied Mathematics and Computation*, 159(2), 449–457. <https://doi.org/10.1016/j.amc.2003.10.028>.
- Jin, J., & Tang, L. (2010). Optimal coverage path planning for arable farming on 2D surfaces. *Transactions of the ASABE*, 53(1), 283–295. <https://doi.org/10.13031/2013.29488>

- Karatas, M., Yakıcı, E., & Dasci, A. (2021). Solving a bi-objective unmanned aircraft system location-allocation problem. *Annals of Operations Research*. <https://doi.org/10.1007/s10479-020-03892-2>.
- Kittilertpaisan, K., & Pathumnakul, S. (2017). Integrating a multiple crop year routing design for sugarcane harvesters to plant a new crop. *Computers and Electronics in Agriculture*, 136, 58–70. <https://doi.org/10.1016/j.compag.2017.03.001>
- Marinakos, Y., Migdalas, A., & Sifaleras, A. (2017). A hybrid particle swarm optimization—Variable neighborhood search algorithm for constrained shortest path problems. *European Journal of Operational Research*, 261(3), 819–834. <https://doi.org/10.1016/j.ejor.2017.03.031>.
- Montgomery, D. (2013). *Design and analysis of experiments* (Vol. 8). Wiley.
- Neto, T., Constantino, M., Martins, I., & Pedroso, J. P. (2017). Forest harvest scheduling with clearcut and core area constraints. *Annals of Operations Research*, 258(2), 453–478. <https://doi.org/10.1007/s10479-016-2313-2>.
- Nouri, N., & Ladhari, T. (2018). Evolutionary multiobjective optimization for the multi-machine flow shop scheduling problem under blocking. *Annals of Operations Research*, 267(1–2), 413–430. <https://doi.org/10.1007/s10479-017-2465-8>.
- Plà, L. M., Sandars, D. L., & Higgins, A. J. (2014). A perspective on operational research prospects for agriculture. *Journal of the Operational Research Society*, 65(7), 1078–1089. <https://doi.org/10.1057/jors.2013.45>.
- Sethanan, K., & Neungmitcha, W. (2016). Multi-objective particle swarm optimization for mechanical harvester route planning of sugarcane field operations. *European Journal of Operational Research*, 252(3), 969–984. <https://doi.org/10.1016/j.ejor.2016.01.043>.
- Seyyedhasani, H., & Dvorak, J. S. (2017). Using the Vehicle Routing Problem to reduce field completion times with multiple machines. *Computers and Electronics in Agriculture*, 134, 142–150. <https://doi.org/10.1016/j.compag.2016.11.010>.
- Seyyedhasani, H., & Dvorak, J. S. (2018). Dynamic rerouting of a fleet of vehicles in agricultural operations through a dynamic multiple depot vehicle routing problem representation. *Biosystems Engineering*, 171, 63–77. <https://doi.org/10.1016/j.biosystemseng.2018.04.003>.
- Seyyedhasani, H., & Dvorak, J. S. (2018). Reducing field work time using fleet routing optimization. *Biosystems Engineering*, 169, 1–10. <https://doi.org/10.1016/j.biosystemseng.2018.01.006>.
- Song, X., Jones, D., Asgari, N., & Pigden, T. (2019). Multi-objective vehicle routing and loading with time window constraints: A real-life application. *Annals of Operations Research*. <https://doi.org/10.1007/s10479-019-03205-2>.
- St John, R., & Tóth, S. F. (2013). Spatially explicit forest harvest scheduling with difference equations. *Annals of Operations Research*, 232(1), 235–257. <https://doi.org/10.1007/s10479-012-1301-4>.
- Toth, P., & Vigo, D. (2002). *The vehicle routing problem*. Society for Industrial and Applied Mathematics.
- Utamima, A., Reiners, T., Ansariipoor, A., & Seyyedhasani, H. (2018). The agricultural routing planning in field logistics. In: *Contemporary approaches and strategies for applied logistics* (pp. 261–283). IGI Global. <https://doi.org/10.4018/978-1-5225-5273-4.ch010>
- Utamima, A., Reiners, T., & Ansariipoor, A. H. (2019). Optimisation of agricultural routing planning in field logistics with evolutionary hybrid neighbourhood search. *Biosystems Engineering*, 184, 166–180. <https://doi.org/10.1016/j.biosystemseng.2019.06.001>.
- Valente, J., Cerro, J. D., Barrientos, A., & Sanz, D. (2013). Aerial coverage optimization in precision agriculture management: A musical harmony inspired approach. *Computers and Electronics in Agriculture*, 99, 153–159. <https://doi.org/10.1016/j.compag.2013.09.008>.
- Weintraub, P. A. (2007). Integer programming in forestry. *Annals of Operations Research*, 149(1), 209–216. <https://doi.org/10.1007/s10479-006-0105-9>.
- Zhang, Z., Wang, C., & Zhao, Q. (1990). Application of operations research in agriculture decision making. *Annals of Operations Research*, 24(1), 299–307. <https://doi.org/10.1007/BF02216830>.
- Zhou, K., Leck Jensen, A., Sørensen, C. G., Busato, P., & Bothtis, D. D. (2014). Agricultural operations planning in fields with multiple obstacle areas. *Computers and Electronics in Agriculture*, 109, 12–22. <https://doi.org/10.1016/j.compag.2014.08.013>.
- Zhou, K., Jensen, A. L., Bothtis, D. D., & Sørensen, C. G. (2015). Quantifying the benefits of alternative fieldwork patterns in a potato cultivation system. *Computers and Electronics in Agriculture*, 119, 228–240. <https://doi.org/10.1016/j.compag.2015.10.012>.