

# Optimal route planning of agricultural field operations using ant colony optimization

A. A. Bakhtiari<sup>1</sup>, H. Navid<sup>1</sup>, J. Mehri<sup>2</sup>, D. D. Bochtis<sup>3</sup>

(1. Department of Agricultural Machinery Engineering, Faculty of Agriculture, University of Tabriz, Tabriz, Iran;

2. Department of Applied Mathematics, Faculty of Mathematics, University of Tabriz, Tabriz, Iran;

3. Department of Engineering, Faculty of Science and Technology, Aarhus University, Denmark)

**Abstract:** Farming operations efficiency is a crucial factor that determines the overall operational cost in agricultural production systems. Improved efficiency can be achieved by implementing advanced planning methods for the execution of field operations dealing, especially with the routing and area coverage optimisation aspects. Recently, a new type of field area coverage patterns, the *B-patterns*, has been introduced. *B-patterns* are the result of a combinatorial optimisation process that minimizes operational criteria such as, the operational time, non-working travelled distance, fuel consumption etc. In this paper an algorithmic approach for the generation of *B-patterns* based on ant colony optimisation is presented. Ant colony optimization metaheuristic was chosen for the solution of the graph optimisation problem inherent in the generation of *B-patterns*. Experimental results on two selected fields were presented for the demonstration of the effectiveness of the proposed approach. Based on the results, it was shown that it is feasible to use ant colony optimization for the generation of optimal routes for field area coverage while tests made on the resulting routes indicated that they can be followed by any farm machine equipped with auto-steering and navigation systems.

**Keywords:** route planning, B-patterns, area coverage, operation efficiency

**Citation:** Bakhtiari A. A., H. Navid, J. Mehri, and D. D. Bochtis. 2011. Optimal route planning of agricultural field operations using ant colony optimization. *Agric Eng Int: CIGR Journal*, 13(4).

## 1 Introduction

In intensive agriculture production systems, machinery investment is the second largest investment on most farms after real estate, while annual machinery cost is a significant part of a farm's total annual costs (Kay, Edwards, and Duffy, 2008). Nevertheless, mechanization has had a dramatic effect on production costs, efficiency levels, energy use, labor requirements, and product quality in agriculture all over the world.

Tractors and self-propelled farming machines moving on the fields are traditionally driven by a human driver. The human driver designs the operation strategy on the basis of type of task, type and specifications of the

machine, and, especially, on his own experience. This strategy directly affects field efficiency. Nevertheless, there are conditions where human planning can lead to low field efficiency, e.g. if the field shape is complex, or if there are obstacles within the field area, the selection of the in-field operating routes is too difficult. Therefore, it is necessary to use algorithmically planning methods based on advanced mathematical models to generate routes for automated field machinery.

The coverage path planning problem for agricultural field operations has been investigated by Sørensen *et al.* (2004). In their study, a method is presented for optimizing vehicle routes by defining the field nodes as a graph and formulating the routing problem as the Chinese Postman Problem and the Rural Postman Problem. Since these problems are NP-hard in nature, a heuristic approach was proposed for finding a solution. Bochtis

**Received date:** 2011-08-11 **Accepted date:** 2010-12-30

**Corresponding author's:** Bakhtiari, A. A. email: bakhtiari@tabrizu.ac.ir

et al. (2008) proposed a multi-travelling salesman problem for planning a fleet of combine harvesters operating in a field. Ryerson and Zhang (2007) have proposed genetic algorithms based approach to solve the coverage path planning problem. Although, this methodology did not result in completely optimized paths, the approach achieved 90% coverage of the field. Oksanen and Visala (2009) proposed an area-coverage planning algorithm for agricultural operations. The proposed algorithm included methods for the division of the coverage region into sub-regions, the selection of the sequence of those sub-regions, and the generation of a path that covers each sub-region taking into account the desired working direction. In Ali et al. (2009), the in-field logistics problem for combine harvesters was reformulated based on a modified vehicle routing problem (MVRP) and a modified minimum cost network flow problem (MCNFP).

Bochtis (2008) by introducing *B-patterns*, showed the potential of the implementation of computation optimisation approaches in cases of single or multiple machinery systems operating in one or multiple geographically dispersed fields. *B-patterns* are algorithmically-computed optimal fieldwork patterns based on the approach that field coverage is expressed as the traversal of a weighted graph, and the problem of finding optimal track traversal sequences is equivalent to finding the shortest tours in the graph. The weight of the graph arcs could be based on any relative optimisation criterion, such as total or non-working travelling distance, total or non-productive operational time, etc.

An implementation of *B-patterns* in conventional agricultural machines, supported by auto-steering systems, was presented by Bochtis and Vougioukas (2008). The experimental results showed that, by using *B-patterns* the total non-working distance can be reduced significantly by up to 50%. The same approach has been implemented for the mission planning of an autonomous tractor for area coverage operations such as grass mowing, seeding and spraying (Bochtis et al., 2009) and orchard operations as well (Bochtis et al., 2008).

Here, an approach based on ant colony optimisation

for the generation of *B-patterns* for optimal field coverage for the agricultural machinery operations is presented.

## 2 Ant colony optimisation algorithms overview

Ant Colony Optimization (ACO) is a population-based, general search technique for the solution of complex combinatorial problems. Ant Colony Optimization was first proposed by M. Dorigo (1992). ACO algorithms are inspired by the behaviour of real ants in the wild (Dorigo, Maniezzo, and Colomi, 1996), and more specifically, by the indirect communication between ants within the colony via the secretion of chemical pheromones (without central control). Within the Artificial Intelligence (AI) community, ant algorithms are considered under the category of swarm intelligence (Bonabeau, Dorigo, and Theraulaz, 1999). Swarm intelligence provides a basis with which it is possible to explore distributed problem solving without a centralized control. Further development led to a more general purpose optimization technique, known as Ant Colony Optimization (ACO), which was later formalized into a metaheuristic in (Dorigo and Di Caro, 1999; Dorigo, Di Caro, and Gambardella, 1999). The process above inspired the ACO metaheuristic. The main elements are *artificial ants* simple computational agents that individually and iteratively construct solutions for the problem, which has been modelled as a graph. Ants explore the graph visiting nodes connected by edges. A problem solution is an ordered sequence of nodes. Each artificial ant has an internal memory which is used to store the path followed by the ant. Although one ant is capable of building a solution, it is the behaviour of an ensemble of ants that exhibits the shortest path behaviour. Intermediate partial problem solutions are seen as *states*; at each iteration  $k$  of the algorithm each ant moves from state  $x_i$  to  $x_j$ , expanding the partial solution from node  $i$  adding node  $j$  (Rizzoli et al. 2007). In solution construction each ant is initially put on a randomly chosen start node and at each step iteratively adds unvisited nodes by choosing an action based on variable probability, random choice, and pheromone mediated.

The solution construction terminates once all graph nodes have been visited.

The first ant algorithm, named ‘‘Ant System’’ (AS), was developed by Dorigo et al. (1996). It is organized in two main stages: construction of a solution, and update of the pheromone trail. In AS each ant builds a solution. An ant is in a given state and it computes a set of feasible expansions from it. The ant selects the move to expand the state taking into account the following two values: the attractiveness,  $\eta_{ij}$ , of the move, as computed by some the heuristic information according to the information of the problem, and the pheromone trail level,  $\tau_{ij}$ , of the move that indicates how good the move was in the past. Given the attractiveness and the pheromone trail level, the probability of the  $k^{\text{th}}$  ant making the transition from node  $i$  to node  $j$  is given by:

$$p_{ij}^k = \begin{cases} \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum_{j \in N_i^k} [\tau_{ij}]^\alpha [\eta_{ij}]^\beta} & \text{if } j \in N_i^k \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where,  $\eta_{ij} = 1/d_{ij}$  is a heuristic value that is available *a priori*;  $\alpha$  and  $\beta$  are parameters which determine the relative influence of the pheromone trail and the heuristic information, and  $N_i^k$  is the feasible neighbourhood of ant  $k$  when being at city  $i$ , that is, the set of cities that ant  $k$  has not visited yet. Once complete solutions have been built, pheromone trails are updated. First, the pheromone is evaporated on all arcs and then all ants deposit pheromone on the arcs which are part of the solutions they have just computed. Evaporation of the pheromone trails is included to help ants ‘forget’ bad solutions that were learned early on in the algorithm run.

Following from the original AS, various approaches were made which gave rise to several improved ant algorithms which collectively form the domain of ACO algorithms. A first improvement on the initial AS, called the *elitist strategy* for Ant System (EAS), was introduced in Dorigo (1992) and Dorigo et al. (1991, 1996b). EAS idea is to provide strong additional reinforcement to the arcs belonging to the currently best graph tour found. The Max–Min Ant System (MMAS)

algorithm (Stutzle et al., 2000) differs from the AS in two principles, namely, only the best ant updates the pheromone trails, and the pheromone update function is bound. The rank-based Ant System (ASrank) (Bullnheimer, Hartl, and Strauss, 1996) incorporates the concept of ranking into the pheromone update procedure. Other approaches include the hyper-cube framework for ACO (Blum and Dorigo, 2004), which rescales pheromone values between 0 and 1, the beam-ACO (Blum, 2005), which hybridizes ACO with beam search and the Ant Colony System (ACS) (Dorigo and Gambardella, 1997; Gambardella and Dorigo, 1996) which differs mainly by its pheromone update function.

ACS local update is performed every time the edge from  $i$  to  $j$  is selected in the solution. The pheromone  $\tau_{ij}$  is then modified:

$$\tau_{ij} = (1 - \rho)\tau_{ij} + \rho\tau_0 \quad (2)$$

where,  $\rho \in (0, 1]$  is the pheromone evaporation rate;  $\tau_0$  is the initial pheromone value defined as  $\tau_0 = 1/(n \cdot C^m)$ , where  $n$  is the number of nodes included in the current solution and  $C^m$  the objective function produced by the execution of one ACS iteration without the pheromone component.

At each iteration of the basic Ant Colony method, each ant builds a solution of the problem step by step. Global update is carried out at the end of each iteration when each ant has built a complete solution. In ACS only one ant (the best-so-far ant) is allowed to add pheromone after each iteration and the only edges modified are those belonging to the best solution generated so far,  $T^{\text{bs}}$ . Thus the global update formula is:

$$\tau_{ij} = (1 - \rho)\tau_{ij} + \Delta\tau_{ij}^{\text{bs}} \quad \forall (i, j) \in T^{\text{bs}} \quad (3)$$

where,  $\Delta\tau_{ij}^{\text{bs}} = 1/C^{\text{bs}}$ , and  $C^{\text{bs}}$  is the length of a best-so-far tour.

ACS has been shown to be very efficient in solving routing problems. ACO algorithms can be used to solve both static and dynamic combinatorial optimization problems.

### 3 B-patterns approach overview

“B-patterns” is the result of a route planning method that provides optimal field area coverage planning. According to the B-patterns approach, as is depicted in Figure 1, each track is described by two points corresponding to two nodes in a graph. Abstractive, these nodes correspond to the “customers” in the vehicle routing problem (VRP) methodology. According to this representation, the set of nodes  $N$  is written as  $N = \{1, 2, 3, \dots, 2 \cdot |T|\}$ , where  $T = \{1, 2, 3, \dots\}$  denotes the set of the field-work tracks..

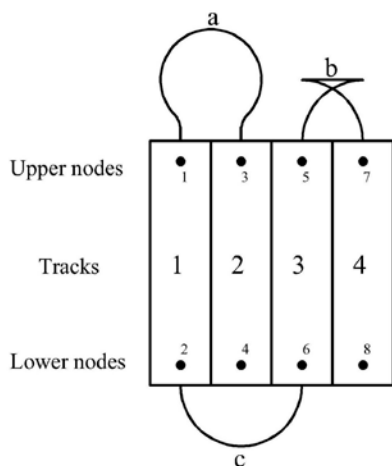


Figure 1 Representation of tracks, their nodes, and prevalent types of headland turnings (a)  $\Omega$ -turn, (b) T-turn, (c)  $\Pi$ -turn

For the solution of the corresponding VRP, let  $D$  be the  $2|T| \times 2|T|$  cost matrix with elements  $d_{ij}$ , that contains the traversal cost between each and every pair of nodes of set  $N$ , and let  $D_0$  be the row matrix of dimension  $1 \times 2|T|$ , where its element  $d_{0i}$  gives the cost for the machine's transition from the “depot” to the location that corresponds to node  $i$ .

$$d_{ij} = \begin{cases} 0 & \text{if } i = j \text{ or } i(j) = 2t_1 - 1, j(i) = 2t_2, t_1 = t_2 \\ M & \text{if } i(j) = 2t_1 - 1, j(i) = 2t_2, t_1 \neq t_2 \\ C_{ij} & \text{otherwise} \end{cases} \quad (4)$$

The  $C_{ij}$ , denotes the non-working distance travelled during headland turnings (e.g., the manoeuvre's length) from the end of track  $i$  to the beginning of track  $j$ . The turn types constitute common manoeuvres for any agricultural machine operating in a headland pattern including the loop turn ( $\Omega$ -turn), the double round corner (T - turn), and the reverse turn ( $\Pi$  - turn) (Figure 1).

According to Bochtis (2008) the minimum length for any turn type can be computed based on the kinematic equations of motion for a non-holonomic vehicle. For example, the following equations are the result for the case of the three most common headland manoeuvres performed by field machinery:

$$\Omega_{\min}(|i - j|) = r_{\min} \left[ 3\pi - 4 \sin^{-1} \left( \frac{2r_{\min} + |i - j|w}{4r_{\min}} \right) \right] \quad (5)$$

$$T_{\min}(|i - j|) = r_{\min} (2 + \pi) - |i - j|w \quad (6)$$

$$\Pi_{\min}(|i - j|) = |i - j|w + (\pi - 2)r_{\min} \quad (7)$$

where,  $r_{\min}$  is the minimum turning radius of the machine; and  $w$  is the operating width. The turning radius refers to the distance from the instantaneous centre of curvature (ICC) to the midpoint between the two rear wheels, while the steer-able wheels are at their maximum steering angle (Dudek and Jenkin, 2000). The  $\Omega$ -turn and  $\Pi$ -turn manoeuvre types result from the kinematic restrictions of the machine and are performed only when a T-turn cannot be executed.

By setting the intra-track transition cost equal to zero, the total cost for the graph traversal is equal to the non-working distance that is travelled by the machine during the headland manoeuvres.

A differentiation can be made between symmetric problems, where the distances between the nodes are independent of the direction of traversing the arcs, that is,  $d_{ij} = d_{ji}$  for every pair of nodes, and the asymmetric one, where at there is at least one pair of nodes  $i$  and  $j$  where  $d_{ij} \neq d_{ji}$  (Dorigo and Stutzle, 2004). Generally in agriculture field operations we confronted with symmetric problems.

In the VRP, the commodity refers to materials to be transported and delivered so the “capacity” of an agricultural machine determined to be equal to a measure (volume or weight) of the “commodity” that it is carries. In the B-patterns approach the “demand” of a node is designated as one half of the quantity of the “commodity” that has to be placed on the track area during an operation.

The demand's nature depends on the operation type and is identical to the nature of the capacity of the

machine that carries out the operation. For material input operations as well as for material output operations, the demand of each node and the machine's capacity are finite quantities. For material neutral operations, the demand of each track is zero and the machine's capacity is infinite. By setting Capacity to be infinite, we get an instance of the multiple travelling salesman problem (MTSP), and in the one - machine case the routing problem is degraded to the simple travelling salesman problem (TSP) (Toth and Vigo, 2002; Bochtis and Sørensen, 2009).

## 4 Result

A number of experiments were carried out in order to demonstrate the adoption of the ACO algorithm for the generation of the *B-patterns*. All the related computation processes were implemented on an Intel® Core™2 Due CPU with 4 GB RAM, using the version 7.8 (R2009a) of MATLAB software.. Two sets of experiments were performed corresponding to two fields referred to as “field A” and “field B”.

### 4.1 First set of experiments

In the first set of experiments, the operation of cultivating a small field (field A, approximately 45 m × 50 m) using a disk harrow with an operating width equal to 4.5 m was performed. The field was divided by the harrow working width into ten tracks ( $\|T_1\| = 10$ ) parallel to the field's longest edge. The minimum turning radius of the tractor used was measured as 5.4 m. The planned operation demanded that the machine starts and ends at the depot on the north-east side of the field.

When driver performed the cultivation following the continuous headland pattern where  $\sigma^{bl(1)} = \{1; 2; 3; 4; 5; 6; 7; 8; 9; 10\}$ , in which the  $\sigma^{bl(1)}$  indicate tracks sequence according to track numbers, the non-working distance was measured to be 209.7 m. The sequence started and ended at a depot and therefore the total non-working distance included the round trip to depot that was measured as 450.2 m.

In the second case, the driver operated according to the alternate overlapping (i.e., to avoid  $\Omega$ -turns) by using the most appropriate block length, which for the

particular field geometry and turning radius lead to the track sequence of the nine tracks per block:  $\sigma^{bl(9)} = \{1; 6; 2; 7; 3; 8; 4; 9; 5; 10\}$ , the non-working distance was measured as 240.3 m and the total non-working distance included also a round trip to depot was 480.8 m.

In the third pattern, the operator used seven tracks per block (resulting in a sequence  $\{1; 5; 2; 6; 3; 7; 4; 8; 9; 10\}$ ) at the end of the operation the driver was forced to perform two T-turns in order to complete the operation. In this operation the non-working distance was measured as 203.3 m and the total non-working distance with transport to depot was 443.8 m.

Finally, the optimal sequence of the tracks was computed by the optimization procedure and was found to be  $\sigma^{op} = \{2; 4; 6; 9; 7; 10; 8; 5; 3; 1\}$  (in Figure 2). The optimal sequence of nodes number was  $\{3, 4, 8, 7, 11, 12, 18, 17, 13, 14, 20, 19, 15, 16, 10, 9, 5, 6, 2, 1\}$ . The non-working distance for this sequence was measured as 171.9 m also the total non-working distance with transport to depot was 376.4 m.

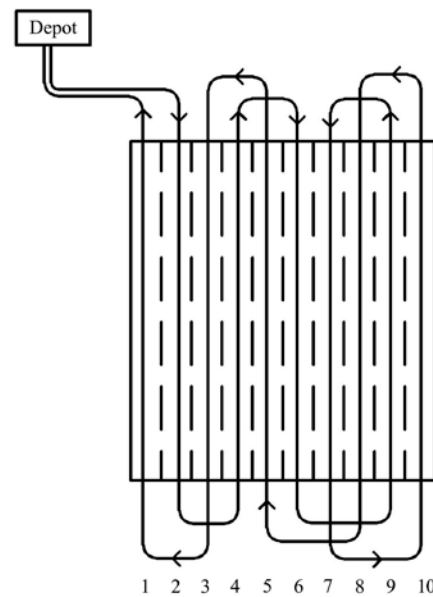


Figure 2 Optimal machine paths for covering field A

For finding the solution in the specific coverage problem the parameters of the ACO algorithm were set as:  $\alpha=1, \beta=2, \rho=0.01$  and the number of iteration ( $t$ ) =1000. Obviously, in more iteration the solution will be more reliable but on the other hand, computing time will be increased. Thus the number of iterations necessary to

find the optimal solution should be consistent with our study. As is mentioned above, the factors  $\eta_{ij} = 1/d_{ij}$  and  $\tau_0 = 1/(n \cdot C^m)$  has been represented. The number of ants to be used is equal to number of nodes in each problem. Furthermore for each problem instance 10 runs were performed to improve experiment accuracy.

The computational time of the whole process was measured and reported (Table 1). For example for this experiment the best solution iteration and its ant number were 206 and 12 respectively and total computational time was 49.3 s.

**Table 1 Final computational results of algorithm program for field A**

Final Outcome:											
Best Tour Selection =	3	4	8	7	11	12	18	17	13	14	20
	19	15	16	10	9	5	6	2	1	0	
Best Tour Length (m) =	376.4										
Best Tour Length (without depot travelling (m)) =	171.9										
Iteration Reference Number =	206										
Ant Reference Number =	12										
Total Computational Time (s) =	49.3										

Figure 3 illustrates how the ants' path searching for field A is biased toward the best solutions. As previously mentioned, it shows in 1000 iterations of algorithm, the best solution is reached in iteration 206 and in continuation no ant can find the better solution.

The non-working distance of each of the four

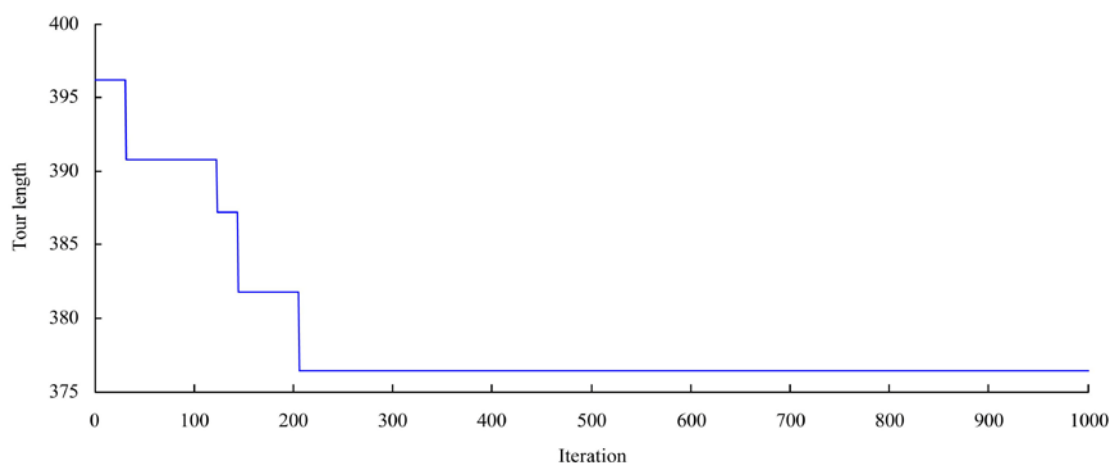


Figure 3 Convergence of solution (Field A)

**Table 2 Measured non-working distances for different patterns and savings by adopting the optimum pattern.**

Field pattern	Field A	Field B
---------------	---------	---------

operations is given in Table 2. The “saving” rows show the presentence of savings for the non-working distance achieved by adopting the *B-patterns* instead of traditional patterns. Also for determination of field efficiency on different path sequence, the distance based field efficiency (Bochtis et al., 2009b) that is given by the relation between the effective travelling distance (the distance that the machine travels while operating) and the total distance travelled by the machine is computed by the following equation:

$$E_{f(distan\ cebased)} = \frac{d_{ef}}{d_{ef} + d_{nef}} \quad (8)$$

where,  $d_{ef}$  is the effective travelling distance and  $d_{nef}$  is the non-effective distance.

	$\sigma^{op}$	$\sigma^{bl(1)}$	$\sigma^{bl(9)}$	$\sigma^{bl(7)}$	$\sigma^{op}$	$\sigma^{bl(17)}$
In-field distance/m	171.9	209.7	240.3	203.3	352	862.9
Savings/%	-	18	28.5	15.4	-	59.2
Field efficiency (distance based)/%	74.4	70.4	67.5	71.1	86.2	71.9
Total Non-working distance/m	376.4	450.2	480.8	443.8	556.5	1168.4
Savings/%	-	16.4	21.7	15.2	-	52.4
Field efficiency (distance based)/%	57	52.6	51	53	79.9	65.4

**4.2 Second set of experiments**

In the second set of experiments, a pentagon field (field B) was covered. The fields' dimension is shown in Figure 4, together with each of the two nodes of tracks.

The working direction was parallel to the 130 m field edge, and consequently, the number of required tracks was  $||T_2|| = 18$ . The initial and final machine locations both correspond at the depot.

According to the optimal planning, the sequence of the tracks was:  $\sigma^{op} = \{1, 3, 5, 7, 9, 11, 12, 14, 16, 18, 17, 15, 13, 10, 8, 6, 4, 2\}$ . The non-working distance was measured as 352 m and the total non-working distance including the round trip to depot was 556.5 m. Other details of solution are presented in Table 3.

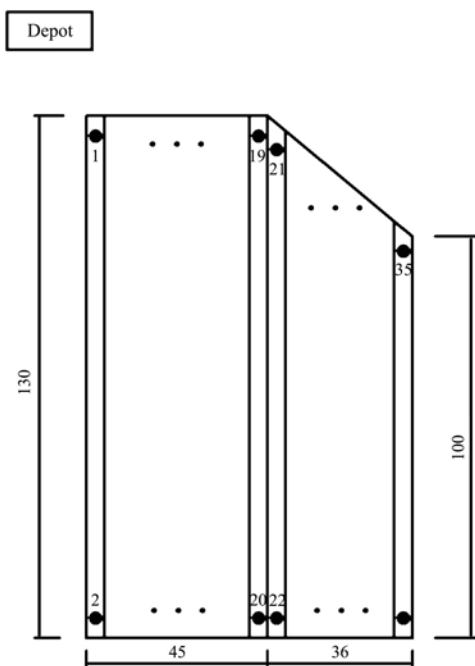


Figure 4 Representation of field B

Table 3 Final computational results of algorithm program for field B

1.1.1 Final Outcome :

1.1.2 Best Tour Selection = 1 2 6 5 9 10 14 13 17 18  
 22 21 23 24 28 27 31 32 36 35 33 34 30  
 29 25 26 20 19 15 16 12 11 7 8 4 3  
 0

1.1.3 Best Tour Length = 556.5

1.1.4 Best Tour Length(without depot travelling) = 352

1.1.5 Iteration Reference Number = 1285

1.1.6 Ant Reference Number = 18

Total Computational Time = 240.7555

All the parameters of the ACO algorithm were set as in the first experiment except the number of iterations which here has been increased to  $(t) = 1500$ . Because of the increasing in tracks and nodes the iteration had to be increased too. As can be seen in Figure 5, the optimum solution was reached in iteration  $(t) = 1285$ , and so, if the algorithm was applied on 1000 iteration a sub-optimal solution, most likely, would be obtained.

When the machine operator was asked to carry out the operation according to his experience, the track sequence he followed for covering field B was:  $\sigma^{bl(17)} = \{1; 10; 2; 11; 3; 12; 4; 13; 5; 14; 6; 15; 7; 16; 8; 17; 9; 18\}$ . The non-working distance for this sequence was measured as 862.9 m, and the total non-working distance including travelling to depot was 1,168.4 m. Non-working distance, savings, and field efficiency of optimal patterns instead of planning based on the traditional pattern, are presented in Table 2.

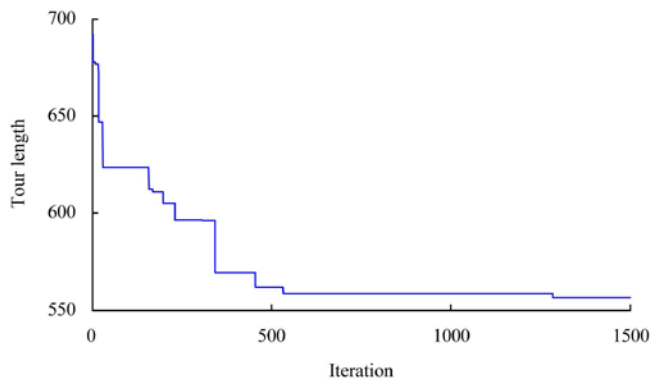


Figure 5 Convergence of solution (Field B)

## 5 Discussion

Regarding the comparison of total computational time, as it was expected, the computational time of algorithm was increased when the number of nodes was increased. Although the number of iterations was raised in the case of field B, the increasing in computational time wasn't harmonized with the increasing in iterations. This means that the relationship between the computation time and the number of nodes is not linear. Nonetheless, insignificant algorithm implementation time is considerable. As it can be seen in Table 2, the increase in the complexity of the shape of the field results in increased savings in the non-working distance.

By examining the examples solutions it can be seen that optimal path sequences (*B-patterns*) start and end at the nearest to the depot tracks. So, the machine doesn't travel across the field headland without operating.

The proposed path algorithm can accommodate any type of turn, as long as the turn's length can be computed in advance and incorporated to cost  $c_{ij}$ . If a library of available turns exists, the algorithm will automatically pick the best one for each turn. As the optimal path sequence examples showed, there is no any  $\Omega$ -turn included in the optimal pattern. This is because of the large cost, in terms of travelling distance required to execute  $\Omega$ -turns. Bochtis and Vougioukas (2008) emphasized this point and mentioned two advantages of

this matter. The first is reduced soil disturbance because smooth  $\Pi$ -turns result in smaller lateral forces during manoeuvring and the second benefit is a reduction in the headland area itself.

Results showed significant savings on non-working distance travelled and indisputable the total operation time, in comparison with the typical unplanned method that is used by the farmers. According to the experimental results machinery efficiency could be improved significantly by computing optimal fieldwork patterns (*B-patterns*) for the agricultural machines which minimize turning time as well as transport distance. Especially for larger machines, it is important to maintain a high efficiency as the non-productive time elements represent a greater proportional loss in potential machine production (Søgaard and Sørensen, 2004).

The dynamic nature of the debated problems demands optimization in almost real time. This necessity prohibits the use of exact algorithms, because of the large computational cost; therefore the implementation of heuristic and metaheuristic algorithms is more suitable. The proposed method based on ACO can easily handle problem instances with many tracks and different field shapes.

## 6 Conclusions

It was shown to be feasible to implement an ant colony optimization (ACO) algorithm to create the optimal field area coverage patterns, *B-patterns*. Experimental examples included small-sized problems. Nevertheless, more complex situations could be considered following the same approach as well. The ACO algorithm can successfully generate routes that can be followed by farm machinery equipped with automation systems such as auto-steering navigation systems.



## References

- Ali, O., B. Verlinden, and D. Van Oudheusden. 2009. Infield logistics planning for crop-harvesting operations. *Engineering Optimization*, 41(2): 183-197.
- Basnet, C. B., L. R. Foulds, J. M. Wilson. 2006. Scheduling contractors' farm-to-farm crop harvesting operations. *International Transactions in Operational Research*, 13(1): 1-15.
- Blum, C. 2005. Beam-ACO—Hybridizing ant colony optimization with beam search: an application to open shop scheduling. *Computers and Operations Research*, 32(6): 1565-1591.
- Blum, C., and M. Dorigo. 2004. The hyper-cube framework for ant colony optimization. *IEEE Transactions on Systems, Man and Cybernetics—Part B: Cybernetics*, 34(2): 1161-1172.
- Bochtis, D. 2008. Planning and Control of a Fleet of Agricultural Machines for Optimal Management of Field Operations. Ph.D. Thesis. ATh, Faculty of Agriculture, Department of Agricultural Engineering, Greece.
- Bochtis, D. D., C. G. Sørensen, R. N. Jørgensen, and O. Green. 2009b. Modelling of material handling operations using controlled traffic. *Biosystems Engineering*, 103(4): 397-408.
- Bochtis, D. D., and S. G. Vougioukas. 2008. Minimising the non-working distance travelled by machines operating in a headland field pattern. *Biosystems Engineering*, 101(1): 1-12.
- Bochtis, D. D., S. G. Vougioukas, H. W. Griepentrog; and N. A. Andersen. 2008. Effects of capacity constraints on the motion pattern of an autonomous orchard Sprayer. In *Proc. Agricultural and Biosystems Engineering for a Sustainable World, EuAgEng, OP-375*, 8pp, Crete, Greece.
- Bochtis, D. D., and C. G. Sørensen. 2009. The Vehicle Routing Problem in Field Logistics Part I. *Biosystems Engineering*, 104(4): 447-457
- Bochtis, D., S. Vougioukas, C. Tsatsarelis, and Y. Ampatzidis. 2007. Optimal dynamic motion sequence generation for multiple harvesters. *Agricultural Engineering International: The CIGR Ejournal*, XI (Manuscript ATOE 07-001).
- Bochtis, D. D., S. G. Vougioukas, and H. W. Griepentrog. 2009. A Mission Planner for an Autonomous Tractor. *Transactions of ASABE*, 52(5): 1429-1440.
- Bonabeau, B., M. Dorigo, and G. Theraulaz. 1999. *Swarm intelligence: From natural to artificial systems*. New York, NY: Oxford University Press.
- Bullnheimer, B., R. F. Hartl, and C. Strauss. 1996. A new rank-based version of the Ant System: A computational study. *Central European Journal for Operations Research and Economics*, 7(1): 25-38.
- Cordeau, J. F., M. Gendreau, G. Laporte, J. Y. Potivn, F. Semet. 2002. A guide to vehicles routing heuristics. *Journal of Operational Research Society*, 53(5): 512-522.
- Dantzig, G. B., and J. H. Ramser. 1959. The truck dispatching problem, *Management Science*, 6(1): 80-91.
- Dorigo, M. 1992. Optimization, learning and natural algorithms. PhD thesis, Dipartimento di Elettronica, Politecnico di Milano, Italy.
- Dorigo, M., and G. Di Caro. 1999. The Ant Colony optimization metaheuristic. *New Ideas in Optimization* (pp. 11-32).
- Dorigo, M., and L. M. Gambardella. 1996b. A study of some properties of Ant-Q. In *PPSN IV: Proceedings of the 4th international conference on parallel problem solving from nature* (pp. 656-665). Springer-Verlag.
- Dorigo, M., and L. M. Gambardella. 1997. Ant Colony System: A cooperating learning approach to the travelling salesman problem. *IEEE Transactions on Evolutionary Computation*, 1(1), 53-66.
- Dorigo, M., and T. Stutzle. 2004. *Ant colony optimization*. A Bradford Book, the MIT Press, Cambridge, Massachusetts, London, England.
- Dorigo, M., G. Di Caro, and M. Gambardella. 1999. Ant algorithms for discrete optimization. *Artificial Life*, 5(2): 137-172.
- Dorigo, M., V. Maniezzo, and A. Colomi. 1991. Positive feedback as a search strategy. Technical report 91-016, Dipartimento di Elettronica, Politecnico di Milano, Italy.
- Dorigo, M., V. Maniezzo, and A. Colomi. 1996. Ant System: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics – Part B*, 26(1): 29-41.
- Dudek, G., M. Jenkin. 2000. *Computational Principles of Mobile Robotics*. Cambridge University Press.
- Gambardella, L. M., and M. Dorigo. 1996. Solving symmetric and asymmetric TSPs by ant colonies. *Proceedings of the 1996 IEEE international conference on evolutionary computation (ICEC'96)* (pp. 622-627). Piscataway, NJ: IEEE Press.
- Kay, R. D., W. M. Edwards, and P. A. Duffy. 2008. *Farm management*. Published by McGraw-Hill. Sixth edition.

- Kirkpatrick, S., D. Gellat, M. P. Vecchi. 1983. Optimizations by simulated annealing. *Science*, 220(No 4598), 671–680.
- Oksanen, T., and A. Visala. 2009. Coverage path planning algorithms for agricultural field machines. *Journal of Field Robotics*, 26(8): 651–668.
- Oppen, J., A. Løkketangen. 2008. A tabu search approach for the livestock collection problem. *Computers and Operation Research*, 35(10): 3213–3229.
- Osvald, A., and L. Z. Strim. 2008. A vehicle routing algorithm for the distribution of fresh vegetables and similar perishable food. *Journal of Food Engineering*, 85(2): 285–295.
- Reimann, M., K. Doerner, R. F. Hartl. 2004. D-ants: savings based ants divide and conquer the vehicle routing problem. *Computers and Operations Research*, 31(4): 563–591.
- Rizzoli, A. E., R. Montemanni, E. Lucibello, and L. M. Gambardella. 2007. Ant colony optimization for real-world vehicle routing problems. *Swarm Intelligence*, 1(2): 135–151.
- Ryerson, A. E. F., and Q. Zhang. 2007. Vehicle path planning for complete field coverage using genetic algorithms. *Agricultural Engineering International: The CIGR Ejournal*, IX (Manuscript PM ATOE 07).
- Sigurd, M., D. Pisinger, M. Sig. 2004. Scheduling transportation of live animals to avoid the spread of diseases. *Transportation Science*, 38, 197–209.
- Søgaard, H. T., C. G. Sørensen. 2004. A model for optimal selection of machinery sizes within the farm machinery system. *Biosystems Engineering*, 89(1): 13–28.
- Sørensen, C. G., and D. D. Bochtis. 2010. Conceptual model of fleet management in agriculture. *Biosystems Engineering*, 105(1): 41–50.
- Sørensen, C. G., T. Bak, and R. N. Jørgensen. 2004. Mission planner for agricultural robotics. *Proceedings of AgEng, Leuven, Belgium, 12–16 September 2004*, 894–895.
- Stutzle, T., and H. H. Hoos. 2000. Max–min ant system. *Future Generation Computer Systems*, 16(8): 889–914.
- Toth, P., and D. Vigo. 2001. An overview of vehicle routing problems. In P. Toth & D. Vigo (Eds.). *The vehicle routing problem* (pp. 1–26). Philadelphia: SIAM.
- Toth, P., and D. Vigo. 2002. *The Vehicle Routing Problem*. SIAM Monographs on Discrete Mathematics and Applications, Vol. 9. Philadelphia, Pa.: Society for Industrial and Applied Mathematics.