

Overview of a route-planning tool for capacitated field processes in arable farming

Santiago Focke Martinez¹, Thomas Wiemann^{1,2} and Joachim Hertzberg^{1,2}

Abstract: This paper presents an overview of a planning tool being developed for the generation of routes in the spatio-temporal domain for vehicles participating in capacitated arable farming operations. The planner supports processes where the field is divided into a surrounding headland and an inner-field (main) region. Two scenarios for output material flow operations are currently supported: 1) one machine equipped with a container works the field; and 2) a machine with no container works the field while being serviced on-the-run by one or more capacitated service units.

Keywords: Route Planning, Precision Farming, Smart Farming.

1 Introduction

Over the last years, the need to introduce new technologies in agricultural processes has increased considerably due to the integration of bigger and more sophisticated machinery and the consequent necessity to improve the efficiency of the processes, with concepts such as Precision Farming and Controlled Traffic Farming gaining traction in the field. Some effort has been put in the last years in the implementation of tools and methods to plan routes for the machines involved in arable farming processes [Mo20], including input and output material flow operations (e.g., spraying and fertilizing, and harvesting, respectively), involving both capacitated and non-capacitated scenarios. Route optimization can lead to reductions of energy consumption, labor costs, traffic intensity in the field, and soil compaction, among other benefits [Ed17, NZ20]. Additionally, if the machines' routes are planned in the spatio-temporal domain, it is possible to estimate the time needed to work on a field, as well as the time a machine is required to operate in a field, which can be used in multi-field campaign planning to improve the usage of resources [Wa20]. Moreover, the increasing development of agricultural robots [Fo20] requires generating plans that can be executed by autonomous systems.

This paper presents an overview of the current state of a tool designed to plan machines' routes in the spatio-temporal domain for arable farming. Two types of machines are considered: a primary machine (PM), which is the one performing the main work in the field (e.g., harvester); and service units (SU), which cooperate with the primary machine

¹ German Research Centre for Artificial Intelligence (DFKI), PBR, Berghoffstraße 11, 49090 Osnabrück, santiago.focke@dfki.de

² Osnabrück University, Berghoffstraße 11, 49090 Osnabrück, twiemann@uos.de, joachim.hertzberg@uos.de

in the process (e.g., transport vehicles). Currently, the planner supports two types of capacitated output-material flow operation types:

- CS1: the PM has a container and no SUs participate in the process. Once its capacity limits are reached, the PM must travel to an unloading facility and then return to continue working the field.
- CS2: the PM has no container, and a material overload operation is done on-the-run (i.e., without stopping) in cooperation with SUs, which are in charge of transporting the material to an unloading facility.

Exemplary results, limitations, future work, and conclusions are also included in the paper.

2 Methods

The main input parameters used by the tool for route planning are: field boundary (polygon), headland width (HW), reference linestring for inner-field tracks, distance between track points (resolution), list of participating machines with their parameters (incl. dimensions, working width, speeds, and capacity), field access points, unloading points, current states of the machines (location, load), out-of-field transit information (e.g. durations, distances), unloading durations, and material mass rate per area unit (either average for the field or a grid map). Other inputs are the speed function, used to compute the speed of the primary machine while working the field; and the cost function, used to compute the costs in the planning process and which can be designed to plan for different type of optimization goals. Introducing these functions as inputs for the planner allows a higher flexibility in the process. The output of the planner are the routes of all machines participating in the process, namely the PMs and the SUs (if applicable). The route points will contain specific information, including location, timestamp, and load information.

The first step in the planning process is the division of the field in two regions: headland and inner-field (IF). To obtain the boundary between the two regions (i.e. the IF boundary), the field boundary is offset inwards by the given distance HW (multiple of the PM's working width). Once the two regions are obtained, the tracks for both are generated. For the headland, the tracks are recursively generated by offsetting inwards the field boundary so that the distance between the tracks equals the working width and all the headland region is covered. In the case of the IF, the reference linestring is translated by a distance equal to the working width and extended until it intersects with the IF boundary. This procedure is done recursively over the IF area until the region is covered. Fig. 1(a) shows an example of the generated regions with their respective tracks, including the initial track order. The generated tracks are finally sampled using the given points' resolution. For the headland, the point order can be clockwise or counterclockwise.

Once the tracks are available, the so-called base route is generated, which represents the route of the PM which covers both regions without considering the capacity constraints or

unloading activities. The planner allows to plan for both regions of the field (as shown in this paper) or for only one region if desired. Whether the headland must be worked before the IF or vice versa depends on the nature of the process. Currently, only the scenario where the headland is worked before the IF region (e.g., harvesting) is supported. First, the headland tracks are connected to obtain the headland route segment, starting with the outmost track. Next, the first and last points of the first and last track of the IF are checked to obtain the point that can be connected to the last headland route-point with the shortest path. These connection paths are computed on the (already worked) headland. Based on the selected point, the first IF track in the sequence is defined, as well as the direction in which the track will be worked. Next, the sequence for the remaining IF tracks is selected. Several approaches have been presented over the last years to optimize the selection of the sequence in which these tracks should be traversed [Mo20]. In this paper, a simple sequencing approach is used, where the IF tracks are traversed in order from one side of the IF to the other. Finally, the headland route is connected to the first IF route point and the points of the IF tracks are added to the route following the track sequence. The connection between IF tracks is done in the headland region. Having the location of the route points, it is possible to compute the timestamp of each route point based on the distance between consecutive points, the machines speed limits, and the speed function. The timestamp of the first route point will be equal to the time required by the PM to reach it, based on the out-of-field information and the closest access point. Moreover, each route point will save the amount of material mass that has been worked until that point. To compute this mass, the material rate per area unit (MRPAU) grid map (if available) or the average MRPAU of the field is used. The area covered between two points is calculated using the distance between the points and the machine's working width, and the total material mass can be obtained from the average MRPAU on that covered area. This only applies for area being worked, hence no mass calculation is done for the points belonging to the headland-IF connection and the IF inter-track connections.

Next, a graph is constructed (Fig. 1(b)) based on the field geometries, the tracks, and the generated base route. The vertices of the graph correspond to each one of the track points, and are connected to other close vertices via edges. Vertices and corresponding edges are also added for field access points, unloading points, and current locations of the machines. Each vertex is assigned a timestamp corresponding to the time in which it is worked, based on the base route points. This allows to control that vertices are not visited by machines transiting on the field before they are worked by the PM. The vertices also hold information of the visits (visit windows) to avoid collision between machines. This graph is later used by the planner to search for the best path to reach a vertex corresponding to a goal location (e.g., an unloading point) from a vertex corresponding to a start location (e.g., a point inside the field) using A* search. The A* search is done in a constrained way to avoid undesired behaviors such as travelling over unworked areas and collision between machines. The search process not only delivers the resulting path points' locations, but also their timestamps, which were computed using the machine speed. The edge costs and heuristics used in the search will be defined by the given cost function. A cost function (CF) can be designed to optimize different aspects of the process. Two CFs are currently available in the planner: for machine's (non-working) time optimization (TOpt), and for

soil protection (SOpt). The TOpt aims to reduce the non-working time of a machine in the process, which translates into optimizing the non-working travels of the machines. The SOpt aims to reduce the damaging impact of machine traffic on the soil, taking into account the amount of mass that has travelled over the edges (dynamically updated during planning) and a cost grid map (if available) which depicts the state of the soil in the field.

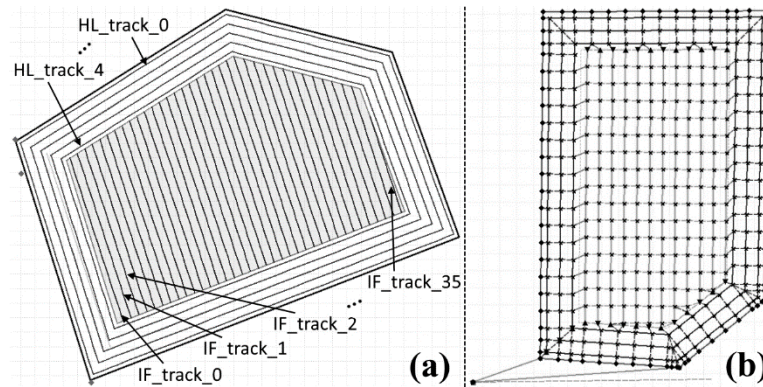


Fig. 1: (a) Field headland and inner-field regions with their respective tracks; (b) Graph example

The next step is to compute the working windows for the machines. The definition of these windows depends on the process. For CS1, a working window starts the moment in which the PM starts working a segment of the base route and finishes when it reaches its capacity limits and needs to travel to an unloading point. For CS2, the windows correspond to overloading windows, starting from the point in which the PM starts overloading the material to a SU and finishes when the SU reaches its capacity limit and must go unload, moment in which the PM waits until the next service unit arrives to continue overloading. If there are multiple SUs, the overloading windows will be assigned in an orderly fashion. If, for example, there are 5 overloading windows and two SUs, one SU will have windows $\{1,3,5\}$ assigned to it, and the other will have the remaining $\{2,4\}$. Different permutations are possible regarding the order in which the windows are assigned to the SUs. The planner generates plans for each permutation until all permutations are checked or until a planning duration limit is reached. The order in which the permutations are checked by the planner depends on the current states of the SUs, aiming to plan first for the permutations that are considered most likely to be the best. In both scenarios, it is possible to add the constraint that the switching between working windows in the IF is done only when a track has been completely worked, avoiding switches inside the IF region.

The overall planning for a permutation is divided in sets of sub-plans. In the case of CS1, the planning is done for the PM, whereas in CS2 the planning is done for the SUs. First, the PM starts following the original base route until the first working window is finished. In case of CS2, a SU follows the PM during this stage performing an overloading operation. Then, the planner searches for the best (sub) route to reach any of the unloading points using A* search in the generated graph, either for the PM (CS1) or the SU (CS2).

There, a route point is added corresponding to the unload duration, followed by a search for the best path from the unloading point to a vertex corresponding to the start point of the next window. This process is repeated until all windows have been worked, and so the route for the machine is generated. The route points' parameters (timestamp, load) are adjusted accordingly. Moreover, the graph vertex and edge parameters (incl. working timestamps, visit windows, and mass driven over the edges) are updated after each search. Lastly, the up-to-date graph is used to obtain the best paths for all machines to exit the field, which is added to the machines' routes. The resulting routes can be finally adjusted to the machines' kinematics by smoothing the sharp turns based on the turning radius of the corresponding machine. Fig. 2 shows two examples of routes generated for machines participating in the two scenarios. In both examples, planning was done following a soil-protection optimization approach which included a soil-state cost map, where areas with high compaction (white areas in the figures) should be avoided as much as possible when planning the routes from/to the unloading points. The left figure depicts the route of a capacitated harvester for CS1, whereas the right figure depicts the route of one of two SUs servicing a harvester for CS2.

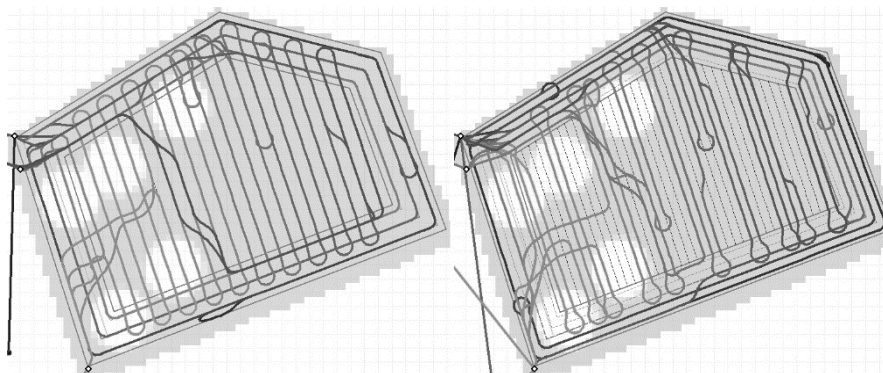


Fig. 2: Route for a harvester in CS1 (left) and route for transport vehicle in CS2 (right)

3 Limitations and future work

The current results are satisfactory; nevertheless, this tool has its limitations. One of them is that the planner does not handle obstacles, which could be solved by integrating a proper field segmentation into subfields and a corresponding subfield-sequencing planning. Another limitation is that only a headland surrounding the IF region is supported. However, some processes use headlands that are only present on two sides of the IF region where the IF tracks start/end, hence it is intended to expand the tool to support such type of headlands as well. Additionally, although the route generation following the presented graph approach gives some flexibility to control traffic over specific areas, it might generate paths that are impracticable or inadequate for vehicles whose turning radii are relatively high with respect to the working width of the process and/or the given track

point's resolution, making the integration of machine-specific kinematic constraints during route planning necessary. Finally, the planner optimizes the process by parts, which means that the resulting routes do not correspond to a global optimal plan.

It is intended to expand the planner to support more agricultural scenarios that could include capacitated input material processes such as fertilizing, as well as other output material processes where the interaction between the PM and the SUs differ from the presented scenarios (e.g., CS2 with a harvester equipped with a container). Dynamic planning is also a current development topic, where the routes are generated on partially worked fields, allowing online recalculation of the routes based on the current states of the machines and the field, and on data gathered during the actual process.

4 Conclusions

This paper presented an overview of a planning tool developed for route generation in capacitated arable farming processes. Two types of capacitated scenarios are currently supported. The planner can generate routes for primary machines and service units following an optimization objective defined by an input cost function (CF). Two CFs are currently offered which aim to minimize the process duration and to increase soil protection. Even though the route planner has limitations, the current results are satisfactory. Further work will be done to decrease the limitations and expand the list of farming scenarios supported by the planner.

The project SOILAssist is funded by the Federal Ministry of Education and Research (BMBF) within the framework of the BonaRes-initiative (grant no. 031B0684B). The DFKI Niedersachsen Lab (DFKI NI) is sponsored by the Ministry of Science and Culture of Lower Saxony and the VolkswagenStiftung.

References

- [Mo20] Moysiadis, V. et.al: Mobile Robotics in Agricultural Operations: A Narrative Review on Planning Aspects. *Applied Sciences* 10/20, p. 3453, 2020.
- [Ed17] Edwards, G. et.al: Route planning evaluation of a prototype optimised infield route planner for neutral material flow agricultural operations. *Biosystems Engineering* 153/17, p. 149-157, 2017.
- [NZ20] Nilsson, R.S.; Zhou, K. Decision Support Tool for Operational Planning of Field Operations. *Agronomy* 10/20, p. 229, 2020.
- [Wa20] de Wall, A. et.al: prospective.HARVEST–Optimizing Planning of Agricultural Harvest Logistic Chains, in 40.GIL-Jahrestagung, Dig. für Mensch,Umwelt und Tier, Bonn,2020
- [Fo20] Fountas, S. et.al: Agricultural Robotics for Field Operations. *Sensors* 20/20,p2672, 2020