TRACC:
Taxi Routes for Aircraft: Creation and Controlling

Deutsches Zentrum
für Luft- und Raumfahrt e.V.
in der Helmholtz-Gemeinschaft

flexiGuide Context

- Task: Applying the principles of 4D trajectories to the ground:
  - Construction of conflict-free taxi-routes which are optimized in relation to the parameters time, speed, length of route etc..
  - Creation of appropriate taxi and speed commands for controllers and pilots for an easy guidance of an aircraft on the optimized and conflict-free route.
  - Solving of tactical and pre-tactical conflicts.

- Connections to other elements of flexiGuide:
  - flexiGuide-Controller-HMI: Presentation of the actual traffic and the necessary advisories to controllers and pilots.
  - DMAN-PVM / AMAN: Actual Information about estimated and target times for the arriving and departing traffic.
  - NARSIM: Steering, visualization and supervision of the airport traffic and transmission of actual positions data of all taxiing aircraft.

Procedures today

- Many hubs and almost all smaller airport conduct taxiing without computer assistance.
- Many airports use displays for visualization and monitoring.
- The focus of actual surveillance tools lie mainly on visualization of traffic, management of restricted areas of the airports and the forecast of runway conflicts instead of solving conflicts for all types of airport traffic or the creation of 4D-Ground-trajectories.
- For an increase of safety and the simplification of conflict awareness and avoidance some airports use pre-definied standard taxi routes (e.g. Standard Short Cuts in Frankfurt).
- Speed advisories are somewhat “fuzzy”.

Fundamental Principles

1. **Definition of a taxi route:**

   Each route consists of a push-back / roll-through and the taxi route to the runway holding point, respectively the touch down and the route to the parking position.

2. **Optimization object:**

   The aircraft are optimized in a sequence created by “First Come, First Served“ or depending on the controller preference. For each aircraft the optimization takes all other already planned aircraft into account but is carried out sequential.

3. **Principle of “user pays”:**

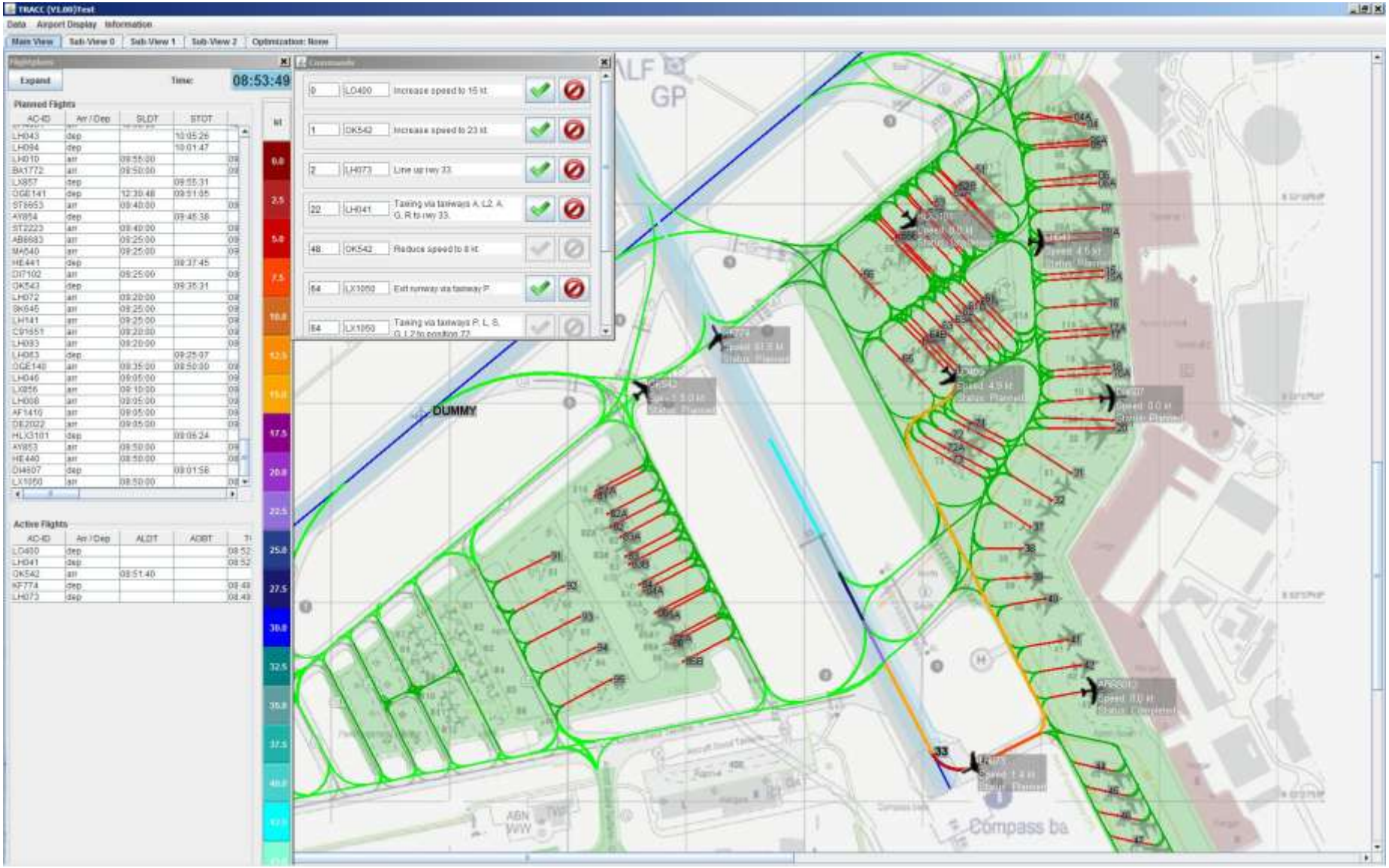
   If an aircraft deviates from the advised route only this aircraft is optimized again.

4. **Principle of highest similarity / reliability:**

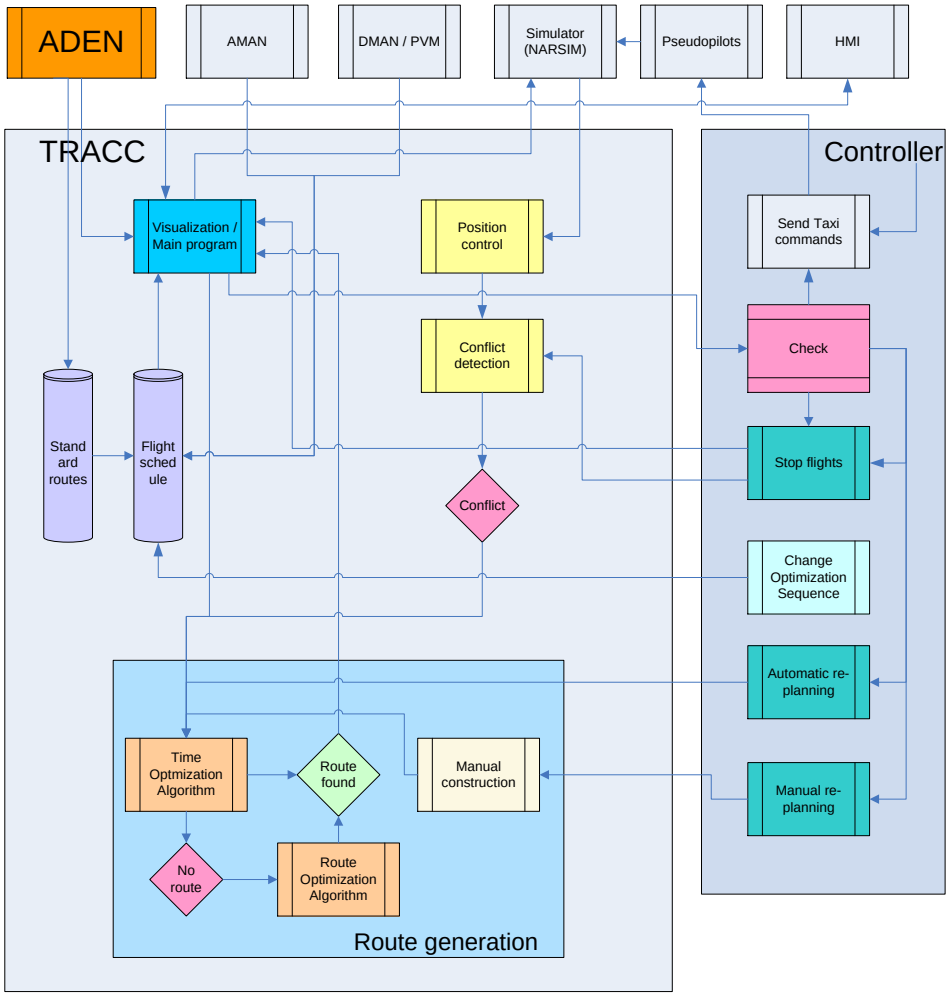
   The newly created taxi route should differ as little as possible in relation to run and speed from the normal standard routes used today.

5. **Principle of lowest workload:**

   Changes for a confirmed route should arise as seldom as possible or should not be visible to the controller (speed advisories and holdings). Only the taxi route is shown to the controller in the beginning.


TRACC (V1.00)Test
Data Airport Display Information
Main View Sub-View 0 Sub-View 1 Sub-View 2 Optimization: None
Expand
Time: 08:53:49
Planned Flights
Active Flights
Increase speed to 15 kt.
Increase speed to 23 kt.
Line up rwy 33.
Taxiing via taxiways A, L2, A, G, R to rwy 33.
Reduce speed to 8 kt.
Exit runway via taxiway P.
DUMMY

Sequence of Operation diagramm



Controller Command Panel

- Shows the translated controller taxi commands.
- For each aircraft only the next command can be processed.
- Each command is visible 90 seconds before the time of application.
- Controllers have three possibilities:
  - Accept: the commands is handed to the pilot / simulation.
  - Reject: TRACC removes the command, checks for conflicts and tries to create an new route in time if necessary without stopping the aircraft.
  - Ignore: TRACC removes the command, checks for conflicts and stops the aircraft for the time needed to create a new route if necessary.

Optimization Sequence

- Start sequence: „First-Come, First- Served“.
- The sequence can be changed by dragging of a flight to another row (planned flights only).
- If the sequence was changed by hand, all flights behind the moved flights have to be optimized either.
- If a flight becomes active, it is moved from the “Planned Flights”-Table to the “Active Flights”-Table.
- In case of deviations from the advised route which result in the necessity of a new optimization the affected aircraft is moved back to the optimization list. The position in the list depends on the time when the new start-waypoint is reached.

Types of Conflicts

There are three different types of conflicts which are tested consecutively:

- Conflicts between pushing and taxiing aircraft:

  Before push-back a check for conflicts with all already moving aircraft is carried out.

- Conflicts between arriving / departing aircraft and taxiing aircraft in case of crossing an active runway:

  Comparison of runway occupancy times with runway crossing times.

- Conflicts between a minimum of two taxiing aircraft:

  Calculating the minimum distance between all route parts of the affected aircraft.

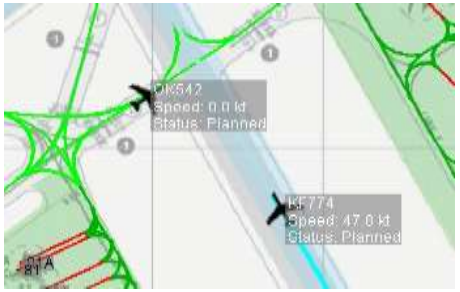
Conflict resolution

Conflicts between pushing and taxiing aircraft:

A waiting time is added to the push-back until no more conflicts occur. The same method is used as for the conflict detection between taxiing aircraft.

Conflicts between arriving / departing aircraft and taxiing aircraft in case of crossing an active runway:

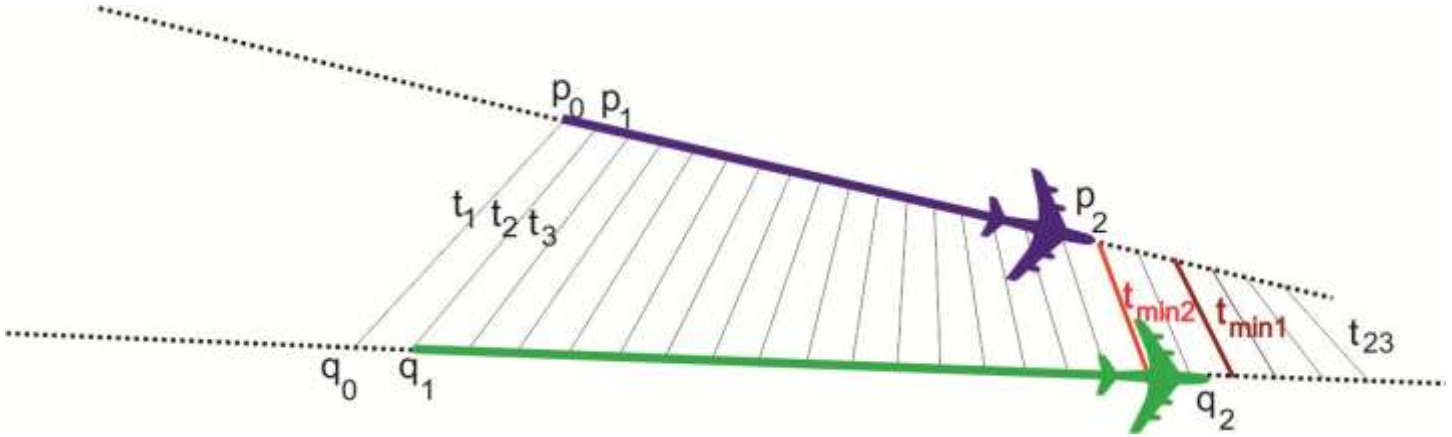
A holding is added in front of the runway.



Conflicts between a minimum of two taxiing aircraft :

A new conflict-free route is developed for the actual aircraft using evolutionary algorithms.

Conflict detection



Position p on the link W:

$$p = w_i + \lambda_i \cdot (v_i + \Delta v_i \cdot t) \cdot t \cdot (w_{(i+1)} - w_i)$$

Minimum distance between two links U and W (Calculation of extreme values):

$$dis^2 = \sum_{j=1}^{2} (w_{i,j} - u_{k,j} + (v_1 t + \Delta v_1 t^2)\lambda_1 (w_{i+1,j} - w_{i,j}) - (v_2 t + \Delta v_2 t^2)\lambda_2 (u_{k+1,j} - u_{k,j}))^2$$

„Safe-Node“-Concept

Important Task: Creation of new taxi routes in case of plan deviation for the ongoing traffic.

- **Two important questions before starting the conflict resolution process:**
  - **How much time can be used for the optimization before the conflict occurs.**
  - **Which waypoint of the route should be used as starting point for the optimzation process / creation of a new route start.**
- Determination of the safe part of the taxi route for the prevention of an emergency stop.
- Calculation of the so-called “Safe-Node” before the conflict occurs under consideration of the necessary minimum distance between the aircraft.
- Determination of the node which will be reached within the next 30 seconds (time used for optimization).
- Determination of a new start node for the following optimization process under consideration of safe node and necessary time.

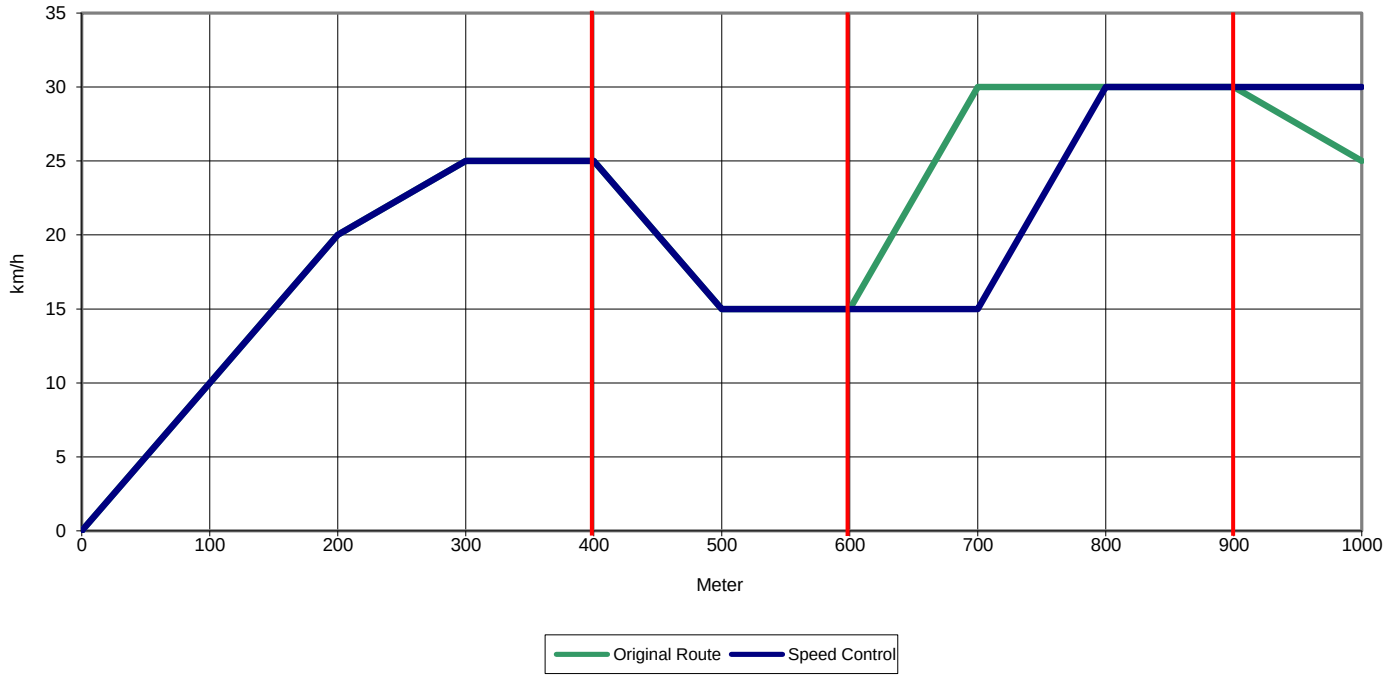
Optimization Concepts

- Implementation of two different optimization Algorithms:
- TOA (Time Optimization Algorithm):
  - Usage of a pre-defined standard taxi route and adaption of taxi speeds and holding times for the creation of a new conflict free route.
  - The probability for inserting a holding time depends on the difficulties to create a conflict-free route (Safe-Node).
- ROA (Route Optimization Algorithm):
  - Modification a the complete taxi route referring to the run of the route, speed profile and holding times.
  - Holding times are created with a predefined probability from the start.

For both algorithm a special operator flats the speed profiles by removing speed changes within a defined distance.

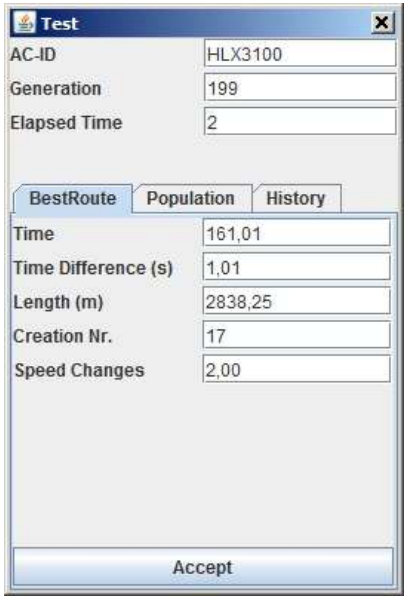
Speed correction for a minimum length of 300 meters

Distance for the speed correction marked by red lines.

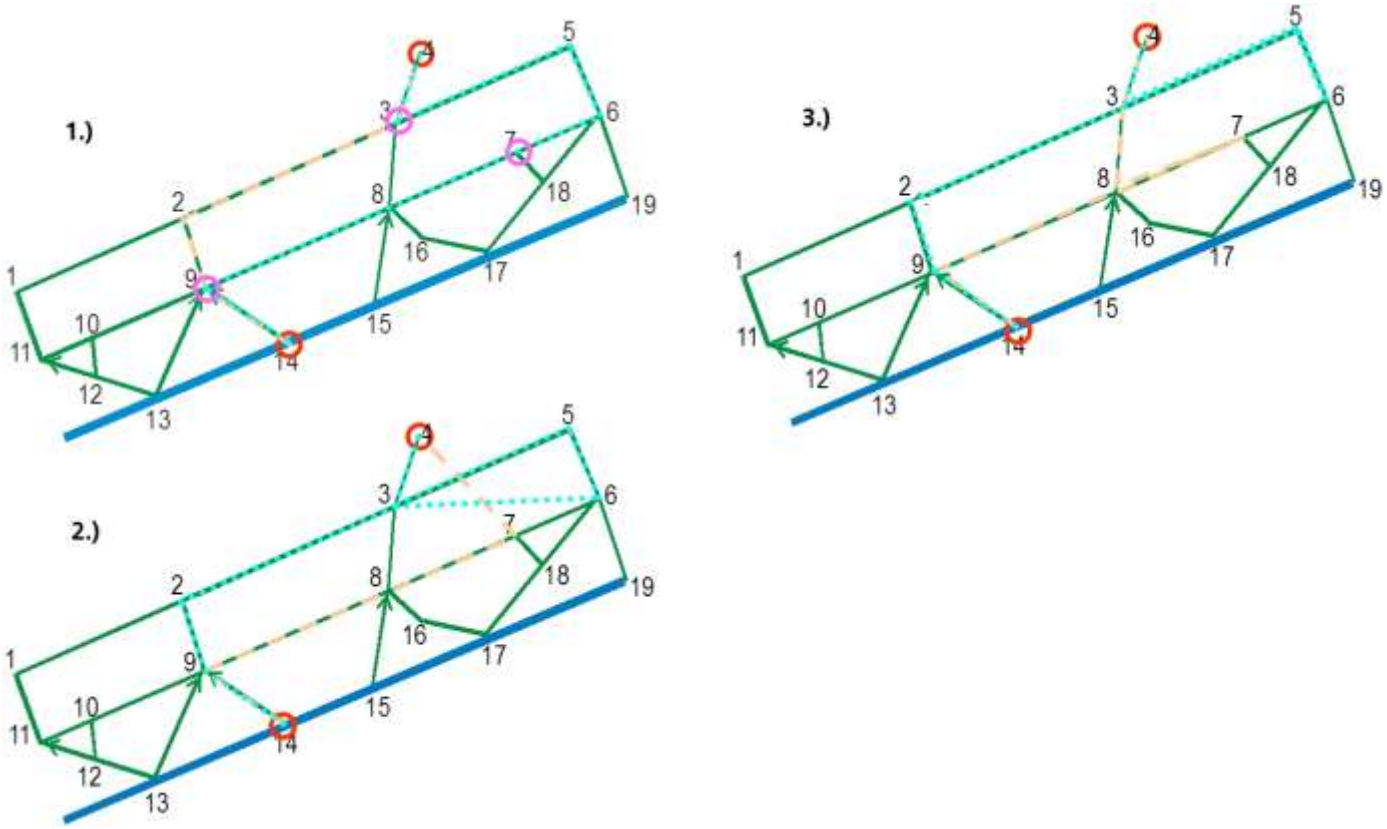


Evolutionary Algorithms

- Imitation of the evolutionary principles of the nature: Survival of the fittest!
- Problem solutions are coded in form of „chromosoms“: Sequence of waypoints with additional infos (e.g. speed, holding time) as genes.
- Population of problem solutions, which are mutated and mixed for the creation of new solutions.
- Usage of crossover, mutation and several problem dependant additional operators.
- Selection of elements of the actual population for the next population with the help of an evaluation function, which can be adapted to several different criterieas (e.g. free of conflicts, punctionality, number of speed changes).
- Stop of the optimization process by hand or when fulfilling a special stop criterion.



Example for the Crossover-Operator



Thank you for your interest!

?

TRACC-Functionality

- Creation of conflict-free 4D-taxi routes using evolutionary algorithms.
- Direct usage of NARSIM ground data for the airport display of the user interface and for the construction of the taxi route.
- Overview about all flight information within two different expandable tables for already moving and announced flights.
- Creation of the necessary controller taxi commands from the technical route (list of waypoints).
- Visualization of taxi routes as colored lines with the color depending on the average speed.
- Three Sub-Views for a better overview for special parts of the airports.
- Visualization of the optimization process inside a special panel.
- Reacts to acceptance or rejection of commands by controllers.
- Controllers can change the optimization sequence by moving aircraft in the information table.

Examples for Controller Commands

Speed Change:

Accept:

Command is carried out.

Reject:

No speed change is carried out. If the resulting route has conflicts, a new route is created (TOA).

Ignored:

Like Reject.

Taxi route:

Accept:

Command is carried out.

Reject:

If there is not enough time left for the creation of a new route the aircraft has to wait at the next possible waypoint for the time of the route creation process. Otherwise the optimization process is carried out without a stop. For the optimization the ROA is used.

Ignored:

The aircraft has to wait at the next possible waypoint and a new route is created using ROA.