RZESZOW UNIVERSITY OF TECHNOLOGY

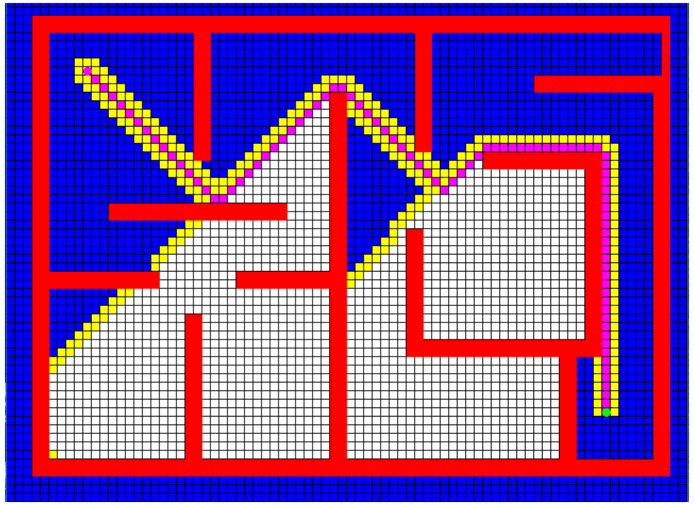
ROUTE DIFFICULTY AND ESTIMATED LENGTH ASSESSMENT USED FOR EMERGENCY DESTINATION EVALUATION

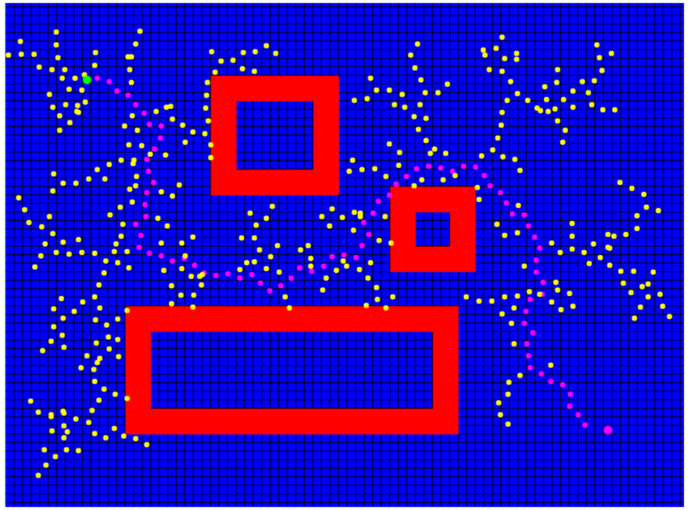
Piotr Grzybowski

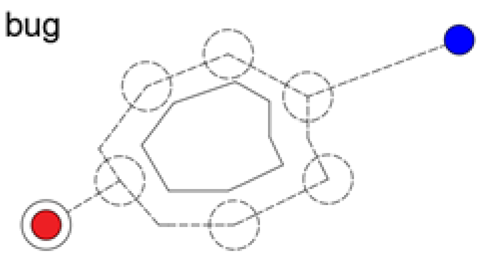
9th EASN Conference, Athens, 4th 2019

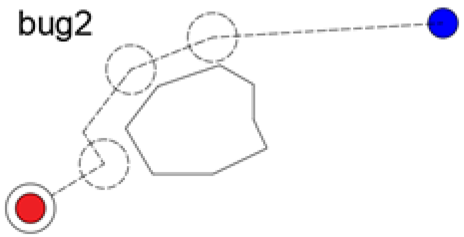
Introduction – multiple algorithms for path planning but which one to choose?

- Djikstra algorithm
- A*
- D*
- Bug
- Bug2
- RRT
- Dubbin’s path method
- Any- angle path planning









Motivation

- Do we need to consider all obstacles in calculation, which may not even be close to optimal path (e.g. in opposite direction)?
- Do we need to know every detail of obstacle, and consider it, if it will in most cases be bypassed with single turn?
- Is exact solution, which is given to late, better than rough estimation given in time?
- Should we bother to look for exact solutions for all possible scenarios when searching for emergency solution, considering it can be delivered to late, or should we concentrate on most promissing based on rough estimation, but do it on time?

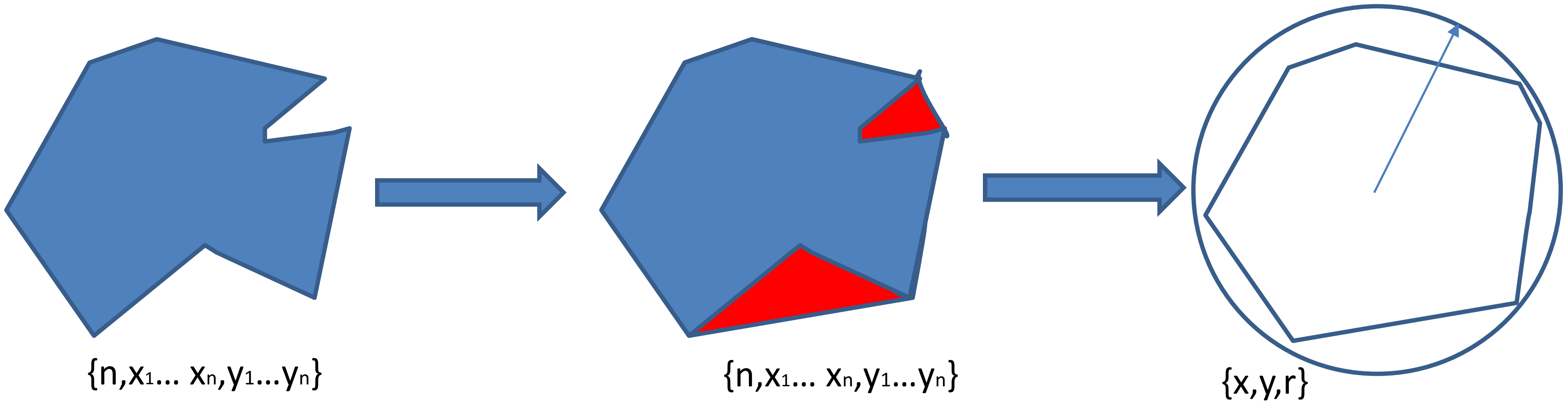
When searching for emergency solution more human questions are:

- Is route short?
- Is it safe to achieve?
- Will I get answer on time?

Obstacle descritpion

Problems to be dealt with:

- concave obstacles,
- obstacles with multiple vertexes,
- size of data to unequivocaly describe obstacle.



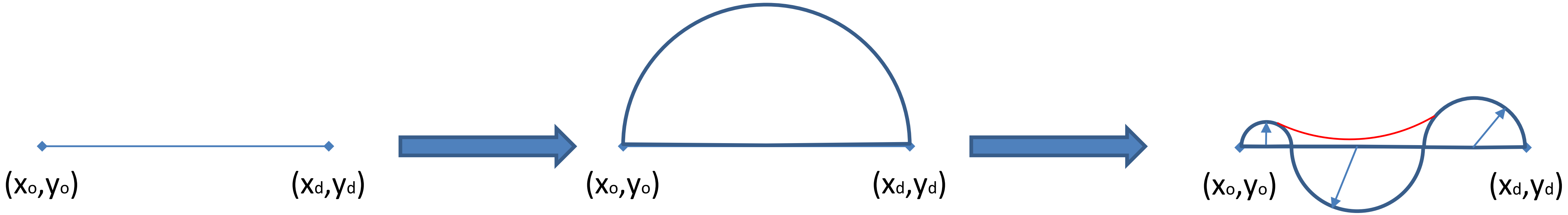
Route length boundaries

Defining boundaries of set:

- known minimal route length (lower boundary) $$L_{min} = \sqrt{(x_o - x_d)^2 + (y_o - y_d)^2}$$
- assumed maximal route length (upper boundary) $$L_{max} = \pi \cdot \frac{L_{min}}{2}$$

Assumption made for worst case scenario: single obstacle of diameter equal to route length

It is easy to prove that same maximal route length will be for case of multiple tangent obstacles with centers laying on minimal route.



Using squeeze theorem it can be proven that optimal route length is between boundaries .

Defining obstacles effecting route calculation

Search circle:

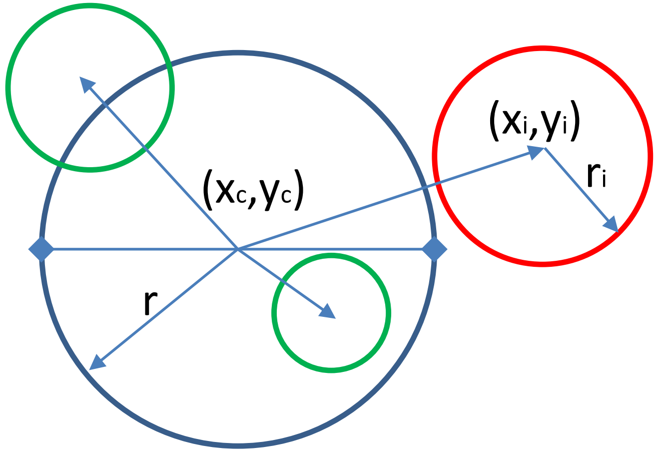
$$r = \frac{L_{min}}{2} \qquad x_c = \frac{x_o + x_d}{2} \qquad y_c = \frac{y_o + y_d}{2}$$

Let B be subset of all obstacles which can affect route

Search condition: $i \in B \leftrightarrow (r_i + r) \geq \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2}$

Which can be simplified for computing: $i \in B \leftrightarrow (r_i + r) \cdot (r_i + r) \geq (x_i - x_c) \cdot (x_i - x_c) + (y_i - y_c) \cdot (y_i - y_c)$

Number of obstacles affecting route n = card(B)



1st method of estimation – estimation by area ratio

Search area (circle area): $A = \pi r^2$

Total obstacles area: $A_{obst} = \sum_{i=1}^{n} \pi \cdot r_i^2$ for all obstacles in set B

Area ratio: $A_r = \frac{A_{obst}}{A}$

Estimated route length: $L_{est1} = L_{min} + A_r \cdot (L_{max} - L_{\mathrm{min}})$

Assumed number of possible routes to calculate: $R_{max} = 2^n$

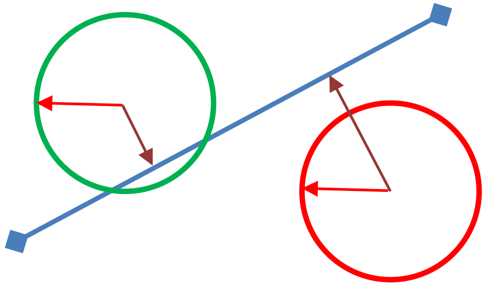
Assumed number of possible route segments: $R_{seg} = 2 \cdot R_{max} \cdot n$

This method is good for initial calculation, but as shown later not very accurate.

2st method of estimation – obstacles on shortest path

Procedure:

1. calculate coefficients for line passing through destination and origin
2. calculate distances between line and centers of obstacles d_j
3. create subset j $C \subseteq B$ from obstacles which satisfy criteria j $\in C \leftrightarrow r_j \geq d_j$
4. calculate number of turns on route: $\mathrm{m} = \mathrm{Card}(C)$
5. estimate route length: $L_{est2} = L_{min} + \sum_{j=1}^{m} (\pi - 2) \cdot r_j$
6. Assumed number of possible routes to calculate: $R_{max2} = 2^m$



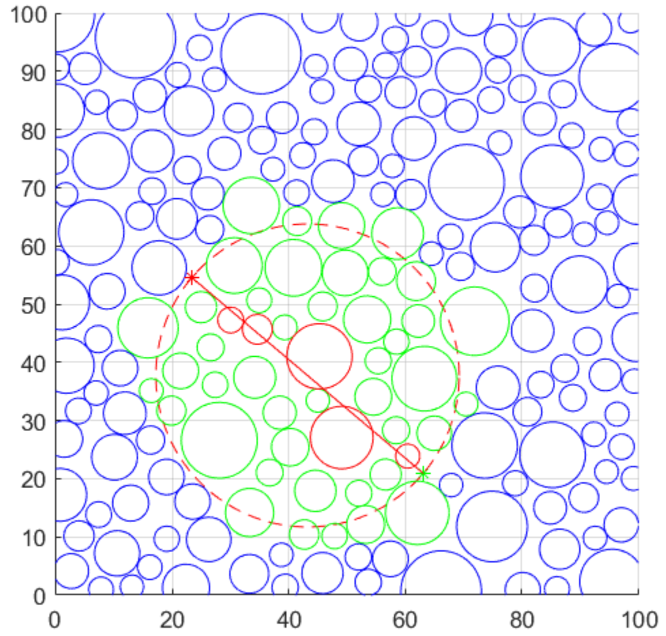
Assumed number of possible route segments: $R_{seg2} = 2 \cdot R_{max2} \cdot m$

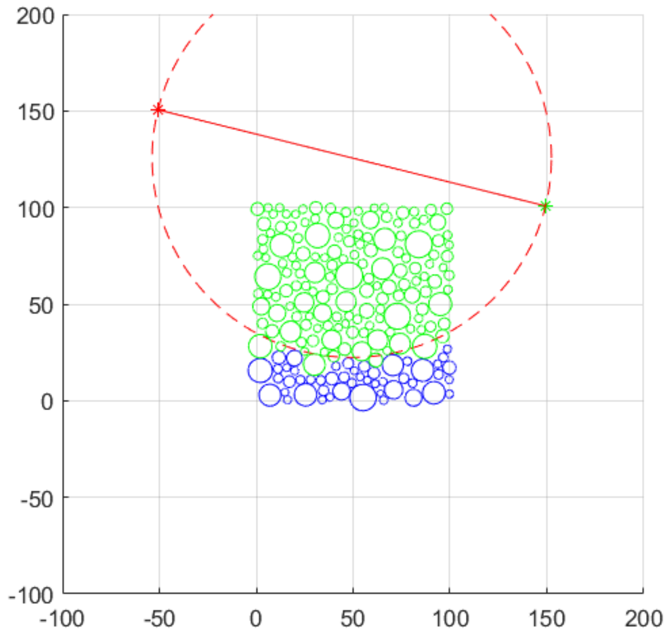
This method although more complex, as show later, is better for approximating route. In proposed method it has tendency to overestimate route length when compared to optimal solution (e.g. delivered with D* method).

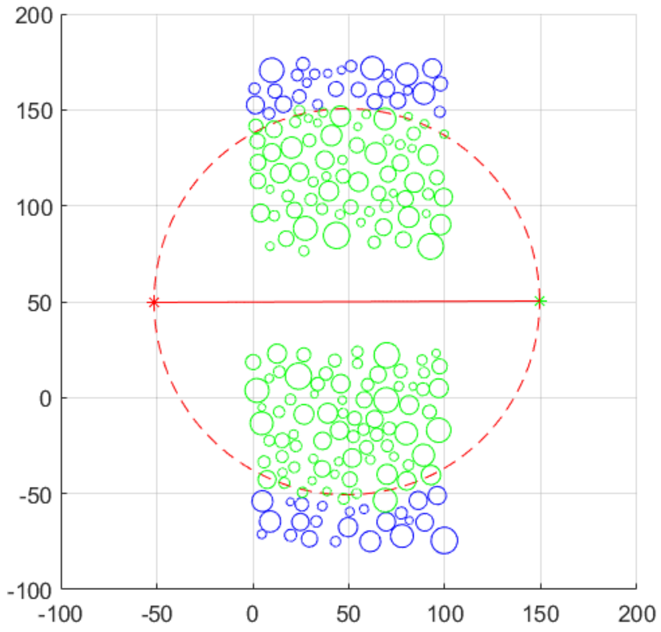
Verification

For verification software has been writen in MATLAB/Simulink environment.

Example calculations: 200 random obstacles – 3 scenarios (flying through, flying near, flying inbetween)







Verification

	x_o [-]	y_o [-]	x_d [-]	y_d [-]	Lmin [-]	Lmax [-]	n [-]	m [-]
Flying through	23.3645	54.5561	63.0841	20.9112	52.0541	81.7663	48	5
Flying near	-50.9346	150.5841	149.5327	100.8178	206.5522	324.4515	151	0
Flying inbetween	-51.6355	49.6495	149.5327	50.3505	201.1694	315.9962	149	0

	R_max [-]	R_seg [-]	A_r [-]	Lest1 [-]	Lest2 [-]	Max [%]	Est1 [%]	Est2 [%[	T [s]
Flying through	2.8147e+14	2.7022e+16	0.8897	78.4878	72.3846	57.0796	50.7814	39.0566	0.106611
Flying near	2.8545e+45	8.6206e+47	0.1529	224.5837	206.5522	57.0796	8.7298	0	0.281114
Flying inbetween	7.1362e+44	2.1266e+47	0.2208	226.5185	201.1694	57.0796	12.6009	0	0.155360

Conclusions

- proposed methods of route length estimation were defined and presented
- test cases for verification were created
- running test on implemented algorithms showed that:
  - first method lead to significant reduction of the set of obstacles
  - second method was more effective for route estimation, and calculation of minimal number of turns related to route difficulty
- time consumption (tested on regular PC) for delivering answer of estimation is satisfying, and can be used for assessing routes for multiple destinations in NRT regime
- presented method may be addopted to be used onboard of aircrafts in Free Route Airspace

RZESZOW UNIVERSITY OF TECHNOLOGY

Thank you for attention... Questions?