# Underwater Glider Path Planning and Population Reduction in Differential Evolution

Eurocast 2015, Workshop on marine sensors and manipulators
Science Museum of Las Palmas de Gran Canaria
11th to 13th of February 2015
Canary Islands, Spain, EU

**Aleš Zamuda, José Daniel Hernández Sosa**

University of Maribor, Slovenia
University of Las Palmas de Gran Canaria, Spain

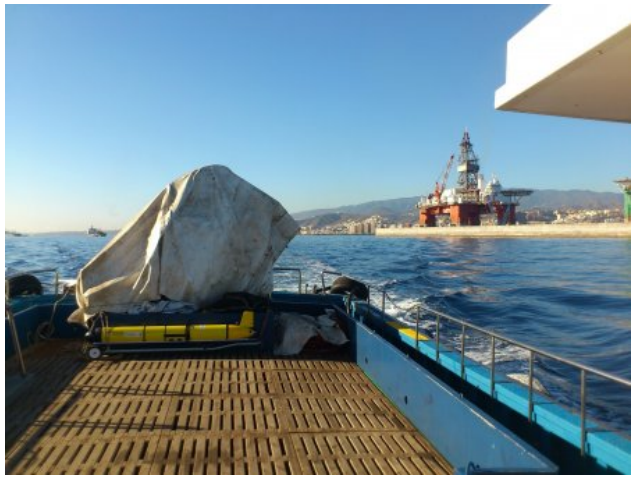
# Introduction

- 1) Underwater glider, mesoscale eddies
- 2) Path planning (ocean trajectory), trajectory test scenarios
- 3) Evolutionary optimization, differential evolution
  - improvement mechanism for trajectory optimization:
  - population size reduction in differential evolution.
- 4) Experiments, results

# Robotic unmanned sea glider Slocum G2

- High durability: 25 to 365 days,
- long range 600–1500 km (alk. batt.), 4000–6000 km ($Li^+$)
  - buoyancy-driven: horizontal 0.35m/s (0.68 knots),
  - 2 knots using propeller.
- Dive to depth 1000 meters, long range, modular,
- integrates sensors of physical and bio/chemical parameters[2]
  - temperature, salinity, dissolved oxygen, turbidity, chlorophyl and sea currents - possible rapid replacement of sensors.





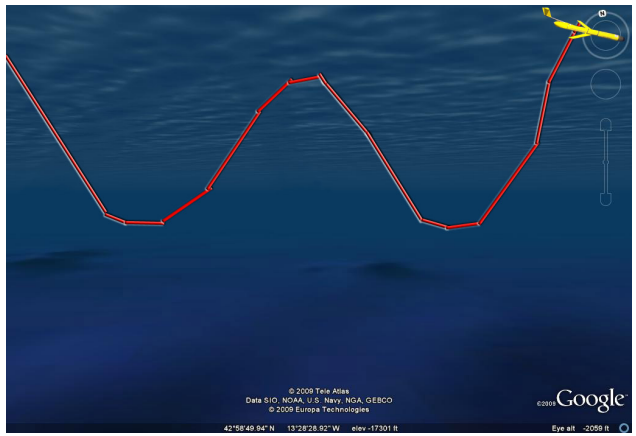
[1] lh6.googleusercontent.com/-Mq308aI1s2g/UHVf4k3uoiI/AAAAAAAACbw/LeiYHXMQRbs/s640/PA060013.JPG

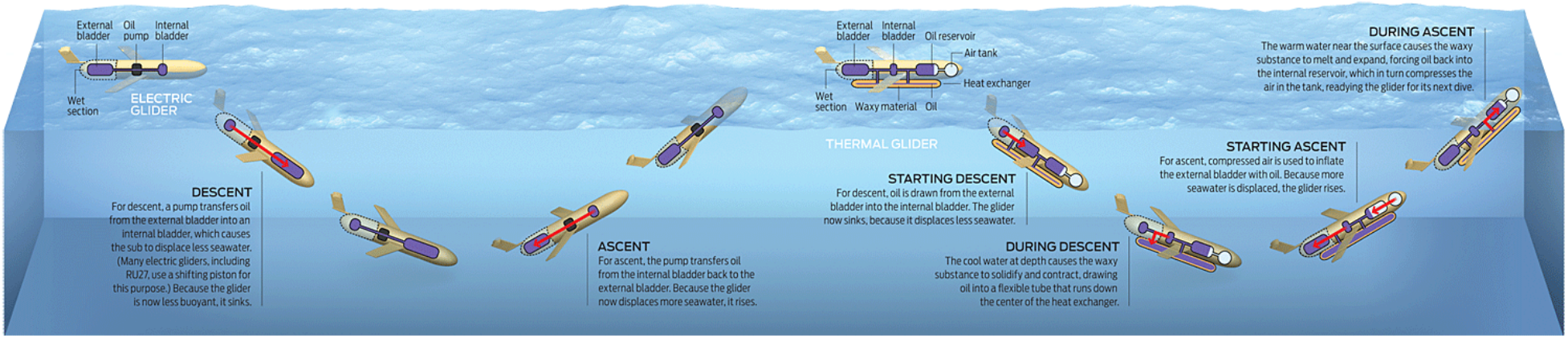
[2] http://www.webbresearch.com/pdf/Slocum_Glider_Data_Sheet.pdf

# The buoyancy drive and submarine probes usefulness

- Driving "yoyo" uses little energy, descent and rise (pump); also for maintaining direction little power is consumed.

+ Use: improving ocean models with real data,
+ the real data at the point of capture,
+ sampling flow of oil discharges,
+ monitoring cable lines, and
+ real-time monitoring of different sensor data.

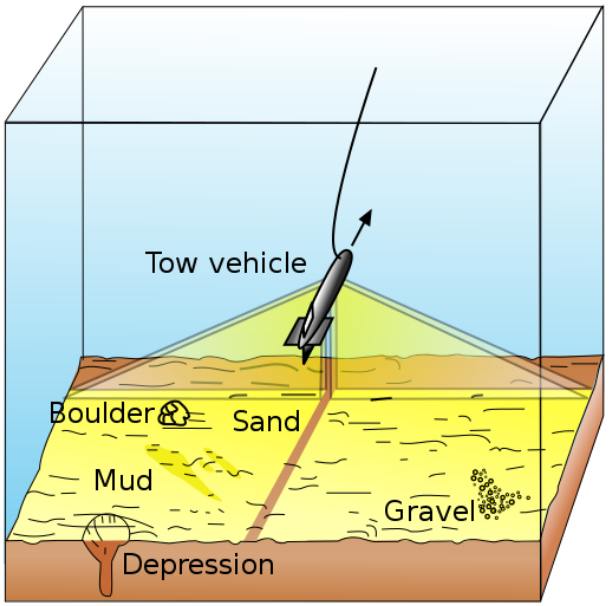


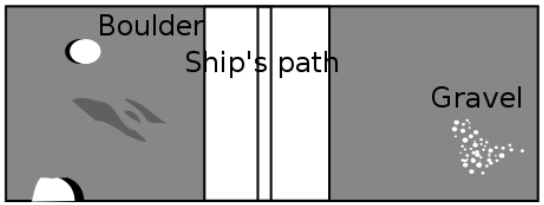



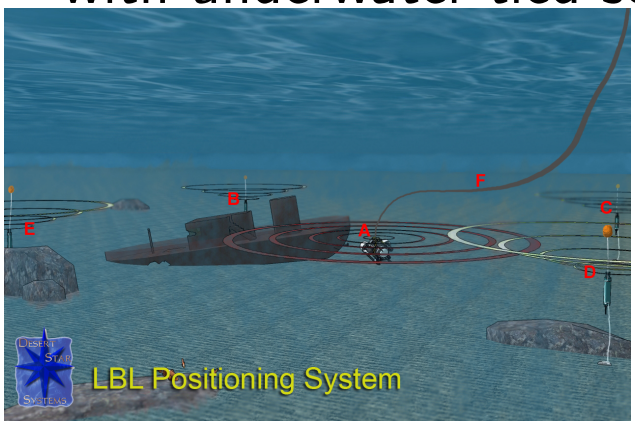

[1] http://www.i-cool.org/wp-content/uploads/2009/11/google-earth-glider-path.jpg
[2] http://spectrum.ieee.org/image/1523708

# Satellite navigation and autonomy

- Radio waves can not penetrate deep water, GPS signal is cut.
  - During a dive the AUV is autonomous,
  - AUV uses internal sensors for navigation,
    - compass, depth, sonar, relief sonar (mapping seabed[1]), gyroscope, accelerometer, magnetometer, thermistor, conductivity meter.
  - acoustic modem for wireless communication with underwater tied sensors[2].



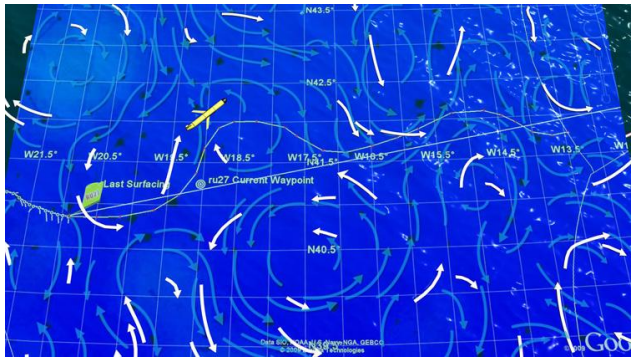



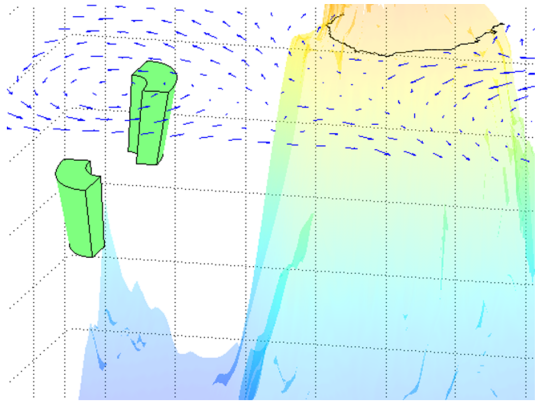




[1] http://upload.wikimedia.org/wikipedia/commons/5/5b/Side-scan_sonar.svg
[2] http://upload.wikimedia.org/wikipedia/commons/a/ad/LBL_Acoustic_Positioning_Aquamap_ROV.jpg
[3] http://www.ego-network.org/dokuwiki/lib/exe/fetch.php?media=img:glider3.gif

# Real-time data streams on the environment

- MyOcean IBI (`http://myodata.puertos.es/`),
- different satellite data about the sea (eg. currents),
- *Regional Ocean Modelling System*: refreshment each 4 hours,
- covers 19° W 5° / E 26° N 56° N, resolution 1/36°,
- furthermore: a surrogate currents model in 3D,
  - extrapolation from 2D surface data (3 days each 4 hours),
  - computed using 3D interpolation from neighboring points.



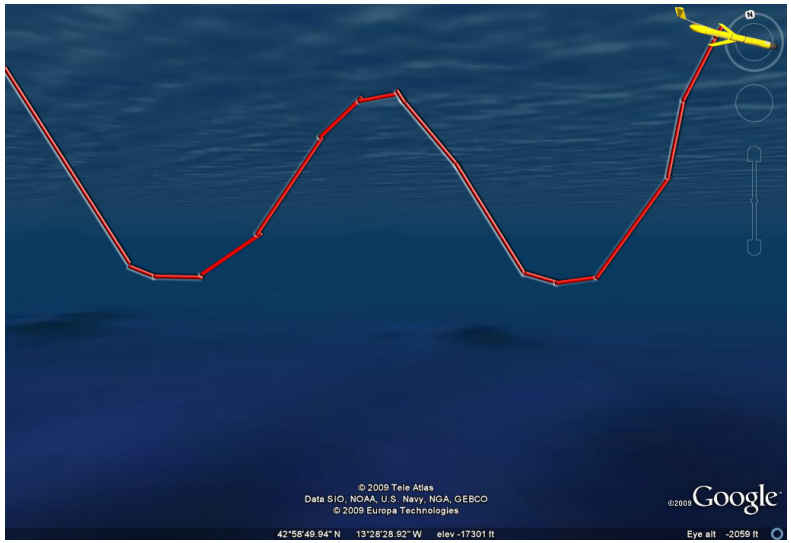
[1] `http://ocean.si.edu/sites/default/files/styles/colorbox_full/public/photos/glider_RU27_eddies_extra%20arrows.jpg?itok=pqbU1Vba`

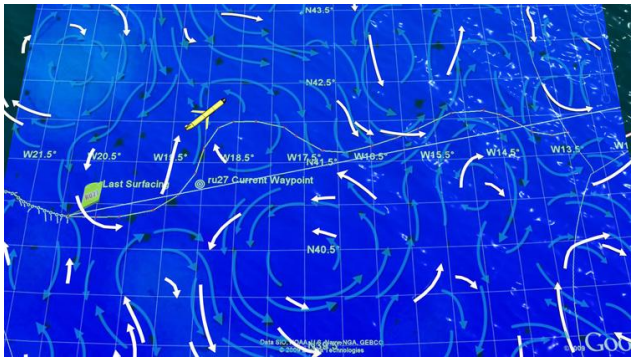
[2] `http://robotics.usc.edu/~ryan/Publications_files/GliderEddyPlan.pdf`

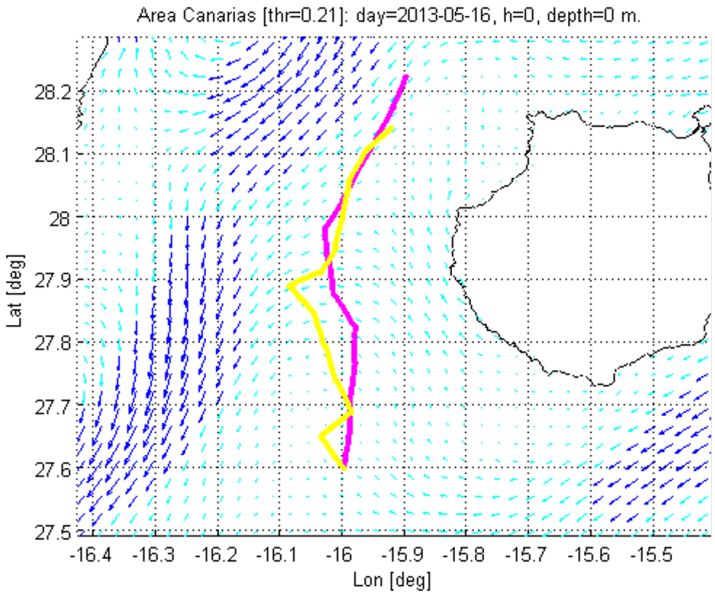
# The optimal trajectory task

fitness evaluation (UGPP simulation):

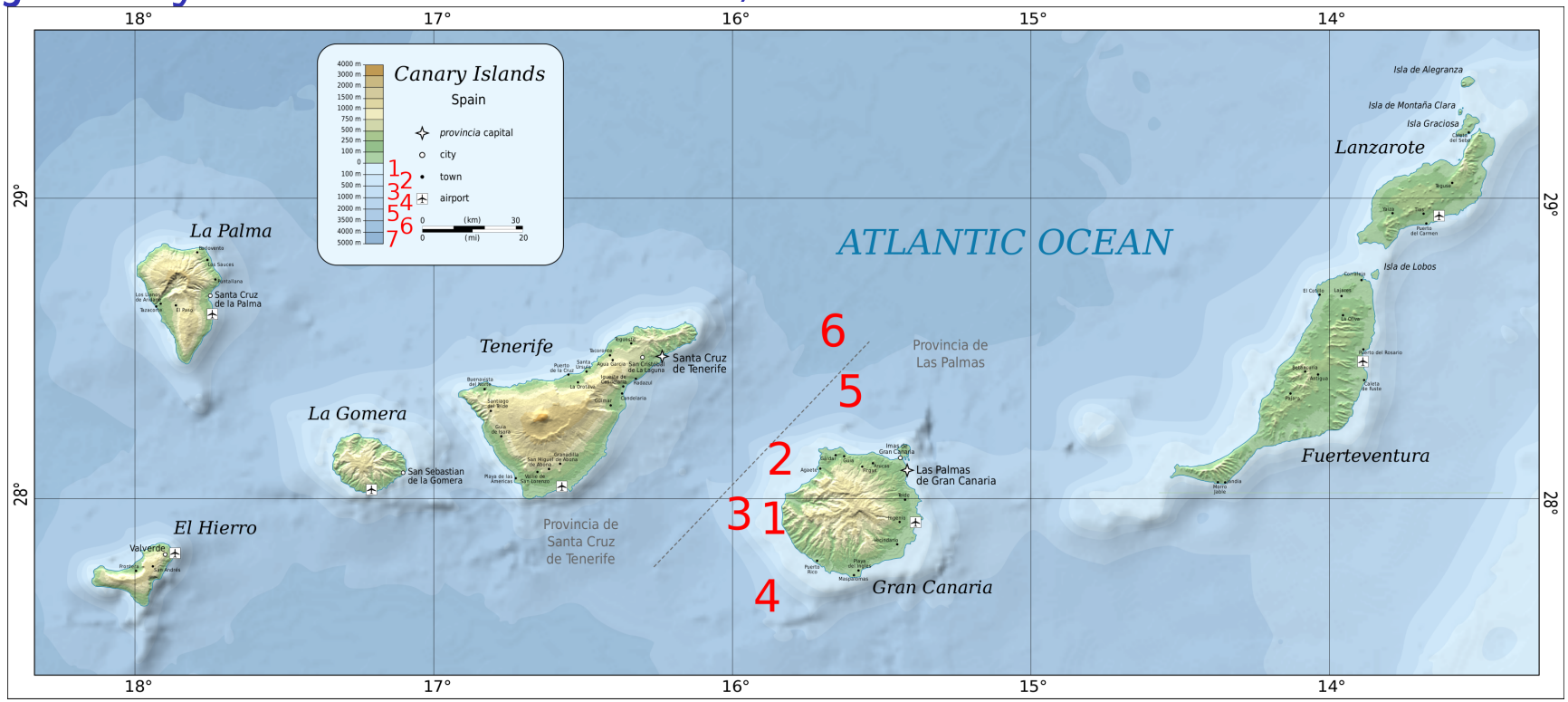
$b_0 = \text{bearing}(\mathbf{p}_0, \mathbf{p}_{\text{target}})$;
$b_1 = b_0 + u_{i,1,g+1}$;
$\mathbf{p}_1 = \text{simulate_stint}(\mathbf{p}_0, b_1, \text{map})$;
**for** $j = \{2, 3, ..., D\}$:
$b_j = b_{j-1} + u_{i,j,g}$;
$\mathbf{p}'_{j-1} = \text{simulate_drift}(\mathbf{p}_{j-1}, \text{map})$;
$\{\mathbf{p}_j\} = \{\mathbf{p}_{j-1}\} \cup \text{simulate_stint}(\mathbf{p}'_{j-1}, b_j, \text{map})$;
**end for**
$f(\mathbf{u}_{i,g}) = \| \mathbf{p}_D - \mathbf{p}_{\text{target}} \|$;



# Trajectory scenarios – *ocean, mesoscale eddies*





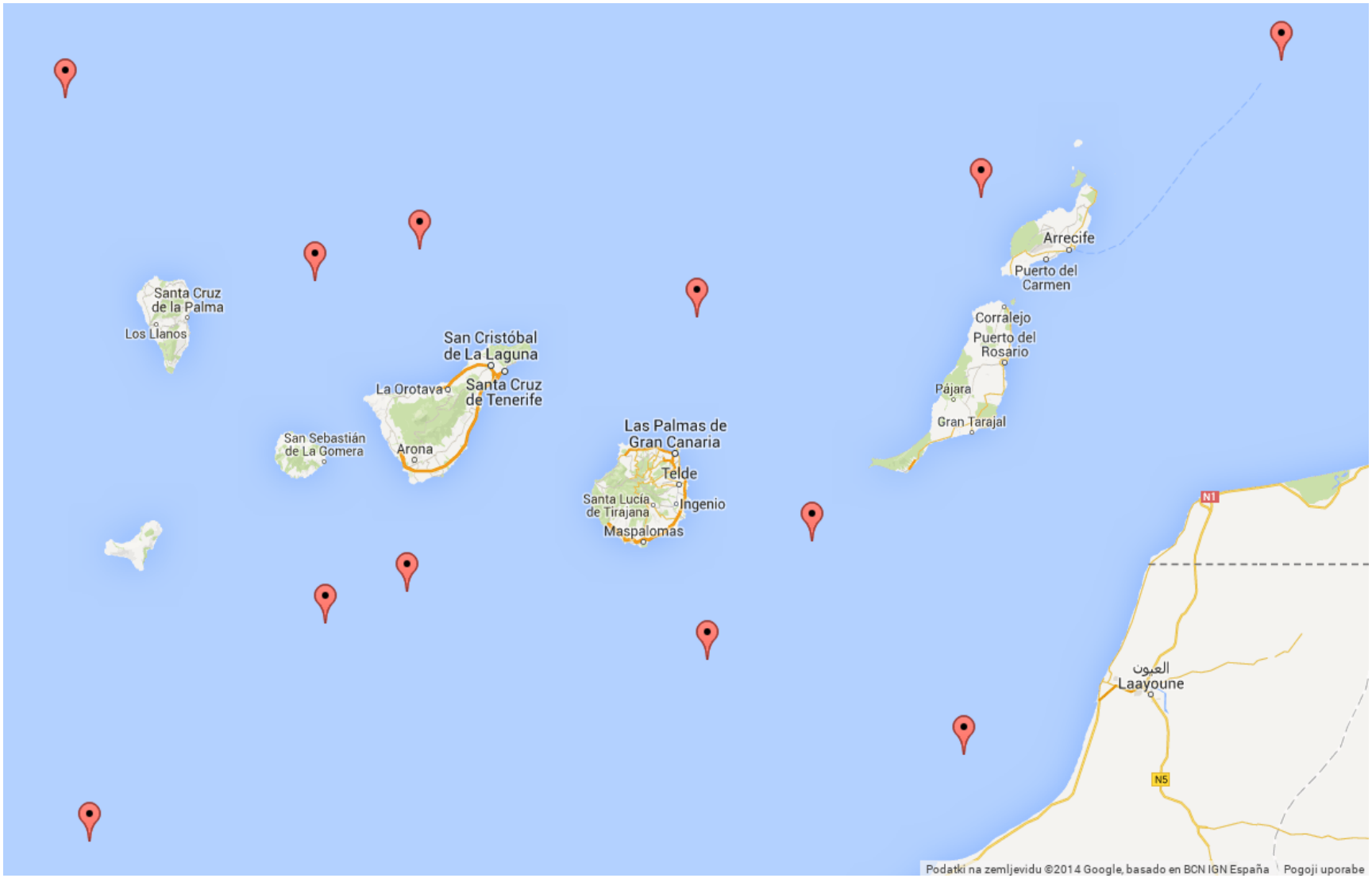




Images: Wikipedia, Google Earth

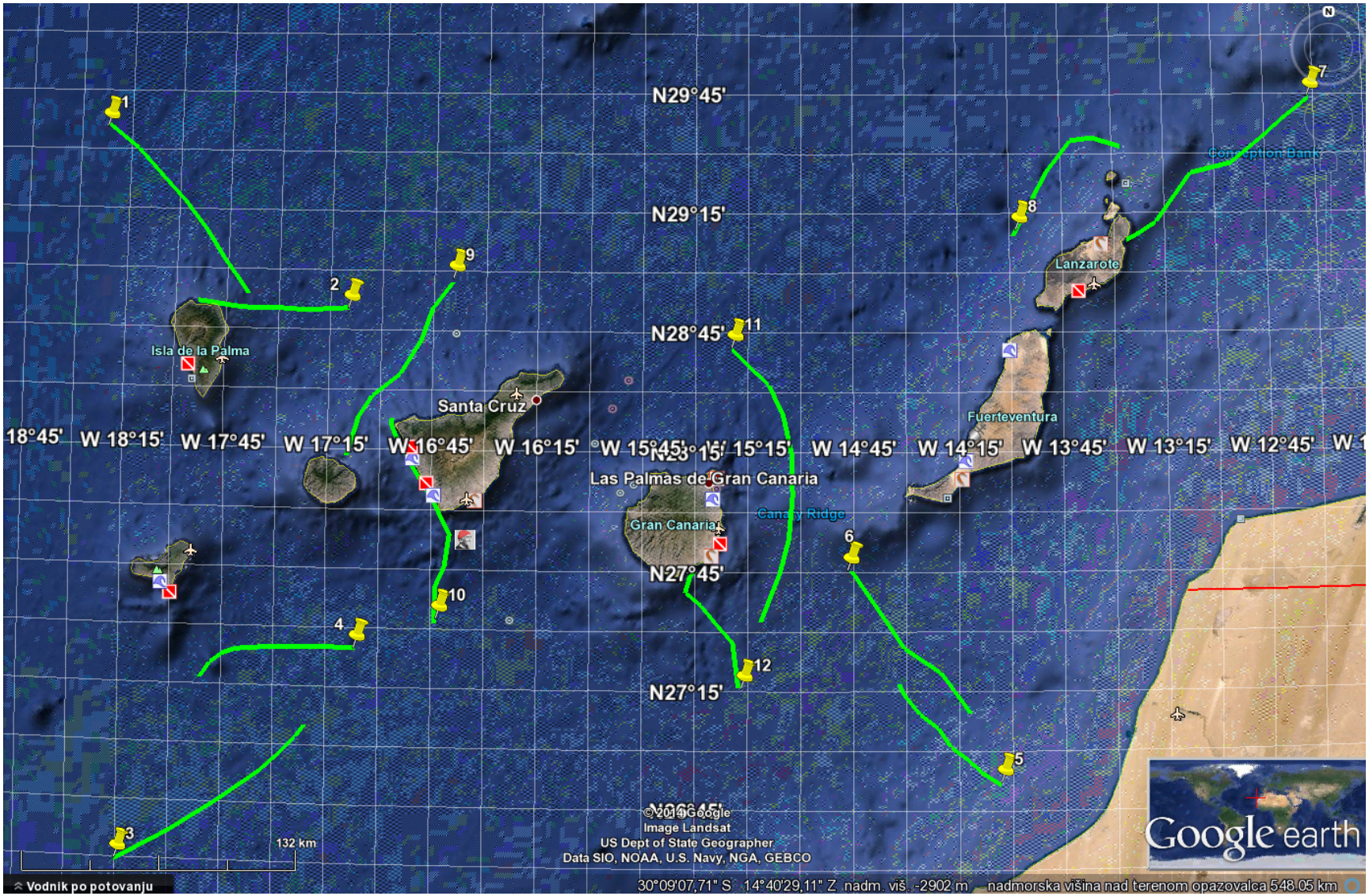
Aleš Zamuda, José Daniel Hernández Sosa

# Trajectory scenarios – *simulation points*



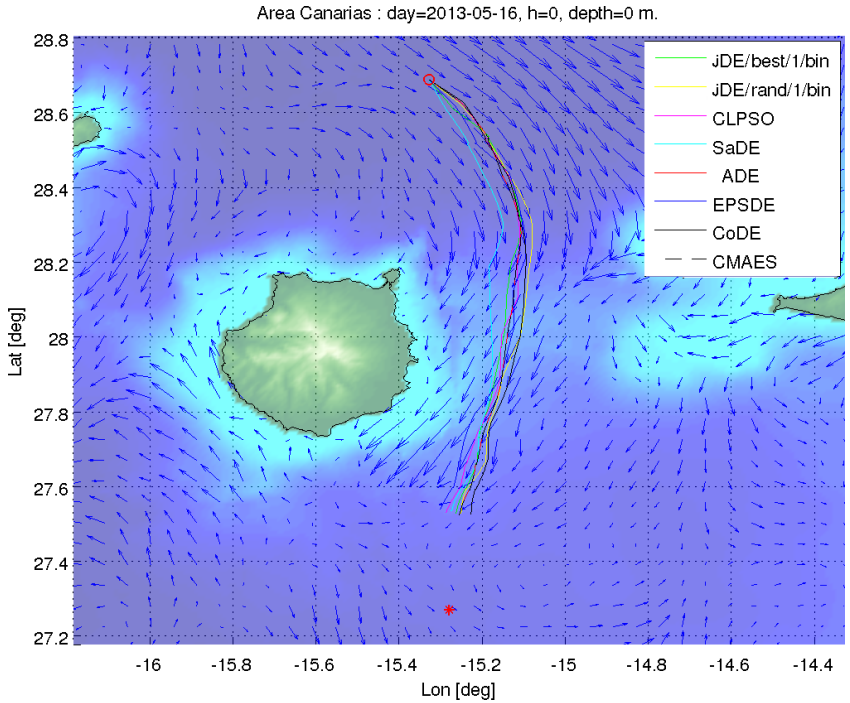

http://www.darrinward.com/lat-long/?id=379544

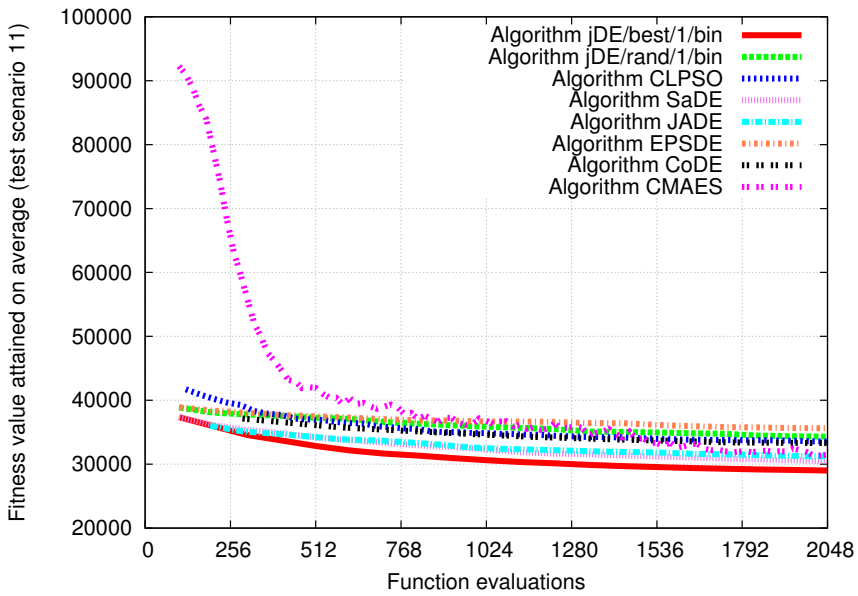
# Trajectory scenarios – *path planning optimization*




https://www.google.si/maps/@28.059806,-15.998355,650054m/data=!3m1!1e3

# Example trajectories optimization – scenario 11




Zamuda, J. D. Hernández Sosa. Differential Evolution and Underwater Glider Path Planning Applied to the Short-Term Opportunistic Sampling of Dynamic Mesoscale Ocean Structures. *Applied Soft Computing*, November 2014, vol. 24, pp. 95–108.

# Optimization algorithms

- Computing Machines + Intelligence = Artificial Intelligence
- Computational Intelligence
- Global Optimization

$$f'(x) = \frac{\Delta f(x)}{\Delta x}, \quad f^*(x) = f(x) + \Delta x f'(x).$$

- Mathematical Programming
- Evolutionary Computation
- **Evolutionary Algorithms** (EA)
  - population-based
  - mutation, crossover, selection

# Differential Evolution (DE) – small, versatile EA optimizer

- A floating point encoding EA for global optimization over continuous spaces,
  - through **generations**,
    **the evolution process** improves **population of vectors**,
  - iteratively by combining a parent individual and several other individuals of the same population.
- We choose the **strategy** *jDE/rand/1/bin*
  - **mutation**: $\mathbf{v}_{i,G+1} = \mathbf{x}_{r_1,G} + F \times (\mathbf{x}_{r_2,G} - \mathbf{x}_{r_3,G})$,
  - **crossover**:
    $$u_{i,j,G+1} = \begin{cases} v_{i,j,G+1} & \text{if } \mathrm{rand}(0,1) \leq CR \text{ or } j = j_{rand} \\ x_{i,j,G} & \text{otherwise} \end{cases},$$
  - **selection**: $\mathbf{x}_{i,G+1} = \begin{cases} \mathbf{u}_{i,G+1} & \text{if } f(\mathbf{u}_{i,G+1}) < f(\mathbf{x}_{i,G}) \\ \mathbf{x}_{i,G} & \text{otherwise} \end{cases},$
  - includes mechanism of $F$ and $CR$ control parameters self-adaptation.

# A DE algorithm extension: population size reduction

- Reducing population size by half (dynNP-DE),

$$G_p > \frac{N_{max_Feval}}{p_{max} NP_p},$$

  - when number of generations exceeds ratio between the number of function evaluations allowed and the population size.
  - Two parameters:
    - initial population size ($NP_{init}$) and
    - number of population reductions ($pmax$).
- Other extensions (unused here) of the original jDE algorithm:
  - multi-objective optimization,
  - SQP local search,
  - $\epsilon$-constraint handling,
  - three random strategy usage,
  - ageing of vectors, and
  - mutation rate $F$ sign changing,
  - multiple strategies.

# Some other instances of evolutionary algorithms

- CLPSO - a Particle Swarm Optimization algorithm,
- CMAES - an Evolutionary Strategy algorithm, Eigen-matrix,

- (jDE) - the Differential Evolution (DE) algorithm @ UM,
- SaDE - DE algorithm (NTU),
- JADE - DE (more greedy than basic jDE),
- EPSDE - DE (parameterization),
- CoDE - DE (parameterization).

- A comprehensive performance comparison published in:

  A. Zamuda, J. D. Hernández Sosa. Differential Evolution and Underwater Glider Path Planning Applied to the Short-Term Opportunistic Sampling of Dynamic Mesoscale Ocean Structures. *Applied Soft Computing*, November 2014, vol. 24, pp. 95–108.

# A new experimental setup and hypothesis

- Two new types of DE strategies (`DE/best` and `DE/rand`)
  - applied to underwater glider path planning (UGPP)
- The newly proposed DE instance algorithms
  - population size reduction on the `best` and `rand` DE strategies
  - assessed and compared on the 12 test scenarios.
- A Bonferroni-Dunns statistical hypothesis testing
  - to confirm outperformance of the favorized `DE/best` strategy
  - over the `DE/rand` strategy for the 12 UGGP scenarios utilized.

- The analysis suggests the approach can benefit from:
  - gradually reducing the population size and
  - also tuning the DE parameters.

# The code

**Algorithm :** Differential evolution for underwater glider path planning

**procedure** jDE_UGPP($\mathbf{x}^*$)
**Require:** $\mathbf{p}_0$ (current glider location), map (MyOcean IBI), $\mathbf{p}_{\text{target}}$ (mission target point), $MAX_FES$ (maximum number of FES allocated), $NP$ (DE population size), $F_l$, $F_u$, $\tau_1$, and $\tau_2$ (jDE constants)
**Ensure:** **x** – list of instructions on how to navigate glider
`uniform randomly generate DE initial population` $\mathbf{x}_{i,0}, \forall i \in \{1..NP\}$;
**for** DE generation loop $g$ (while FES <$MAX_FES$)**do**
**for** DE iteration loop $i$ (for all individuals $\mathbf{x}_{i,g}$ of a population) **do**
**DE population reduction;**
DE individual $\mathbf{x}_{i,g}$ computation (adaptation, mutation, crossover):

$$F_{i,g+1} = \begin{cases} F_l + rand_1 \times F_u & \text{if } rand_2 < \tau_1, \\ F_{i,g} & \text{otherwise} \end{cases};$$

$$CR_{i,g+1} = \begin{cases} rand_3 & \text{if } rand_4 < \tau_2, \\ CR_{i,g} & \text{otherwise} \end{cases};$$

$$\mathbf{v}_{i,g+1} = \mathbf{x}_{r_1,g} + F_{i,g+1}(\mathbf{x}_{r_2,g} - \mathbf{x}_{r_3,g});$$

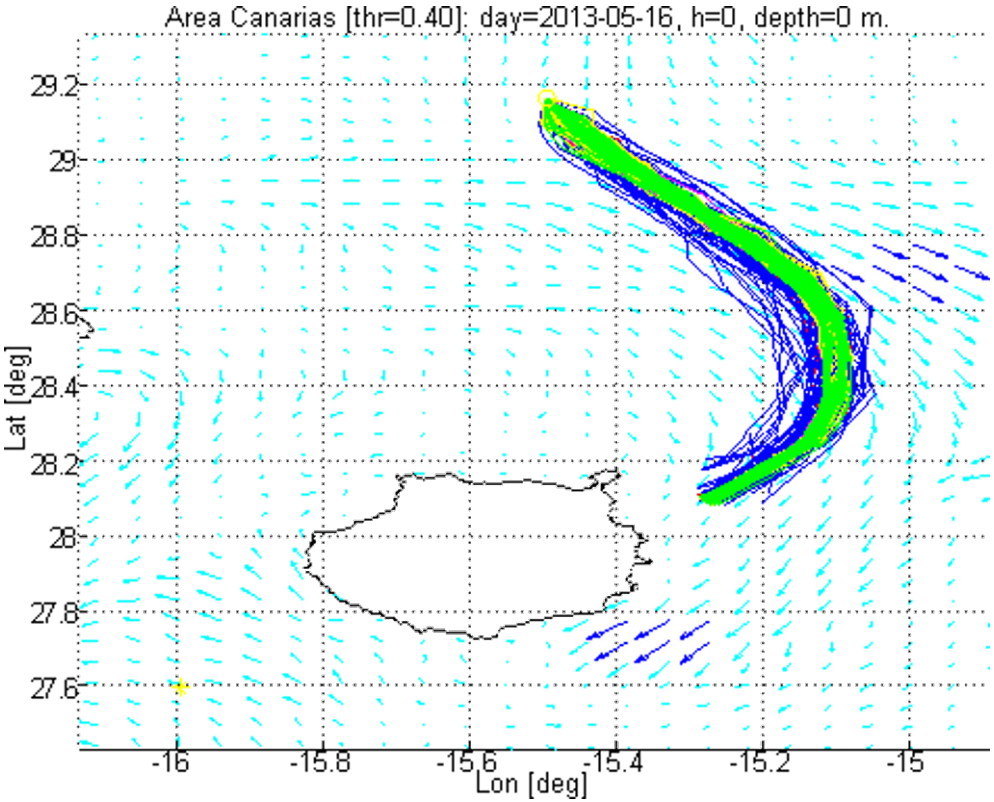
$$u_{i,j,g+1} = \begin{cases} v_{i,j,g+1} & \text{if} rand(0,1) \leq CR_{i,g+1} \text{or} j = j_{rand} \\ x_{i,j,g} & \text{otherwise} \end{cases}, \quad \forall \ j \in \{1, \ldots, D\};$$

DE fitness evaluation (UGPP simulation):
$b_0 = \text{bearing}(\mathbf{p}_0, \mathbf{p}_{\text{target}})$;
$b_1 = b_0 + u_{i,1,g+1}$;
$\mathbf{p}_1 = \text{simulate_stint}(\mathbf{p}_0, b_1, \text{map})$;
**for** $j = \{2, 3, \ldots, D\}$:
$b_j = b_{j-1} + u_{i,j,g}$;
$\mathbf{p}'_{j-1} = \text{simulate_drift}(\mathbf{p}_{j-1}, \text{map})$;
$\{\mathbf{p}_j\} = \{\mathbf{p}_{j-1}\} \cup \text{simulate_stint}(\mathbf{p}'_{j-1}, b_j, \text{map})$;
**end for**
$f(\mathbf{u}_{i,g}) = ||\mathbf{p}_D - \mathbf{p}_{\text{target}}||$;
DE selection:

$$\mathbf{x}_{i,g+1} = \begin{cases} \mathbf{u}_{i,g+1} & \text{if} f(\mathbf{u}_{i,g+1}) < f(\mathbf{x}_{i,g}) \\ \mathbf{x}_{i,g} & \text{otherwise} \end{cases};$$

**end for**
**end for**
**return** the best individual obtained among $\mathbf{x}_{i,G}$;

# Results – phenotype paths obtained using DE/best

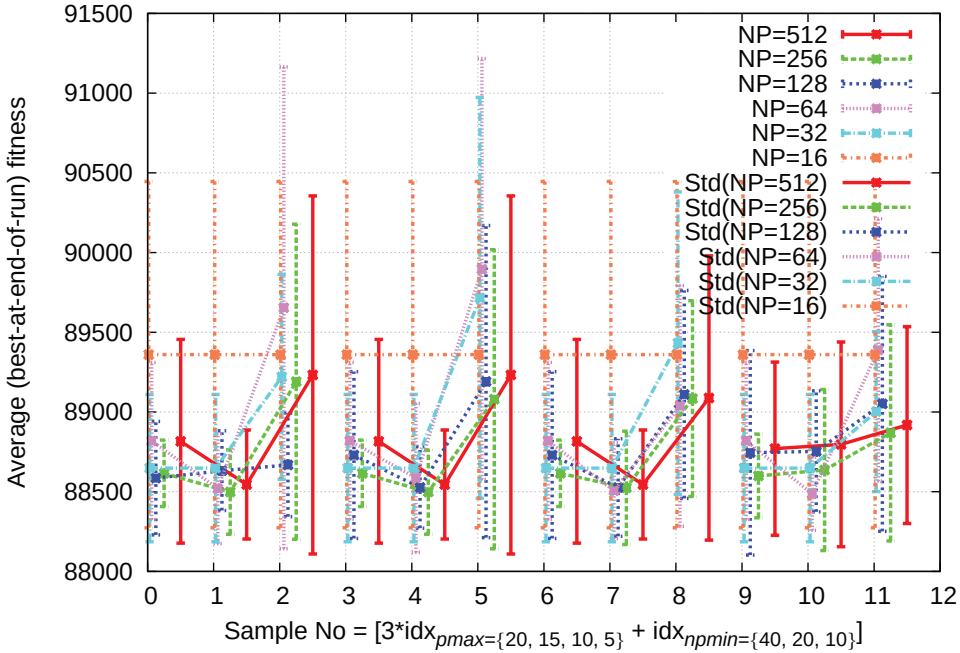



- Trajectory simulations:
  - 12 bearings
  - computed with best taking population size ($NP$) as the study factor.

- $NP$ values: 8 (blue), 32 (purple), 128 (yellow), 512 (green)

- 120 runs subsampled with step 10
  - one run drawn per $NP$, resulting in 12 trajectories shown.

# Results – phenotype paths obtained using DE/best




- Different population sizing settings
- aggregated on 10 independent runs,
- impact on mean final fitness value for different
  - minimal $NP$ ($NP_{\min}$) and
  - number of reductions ($pmax$).

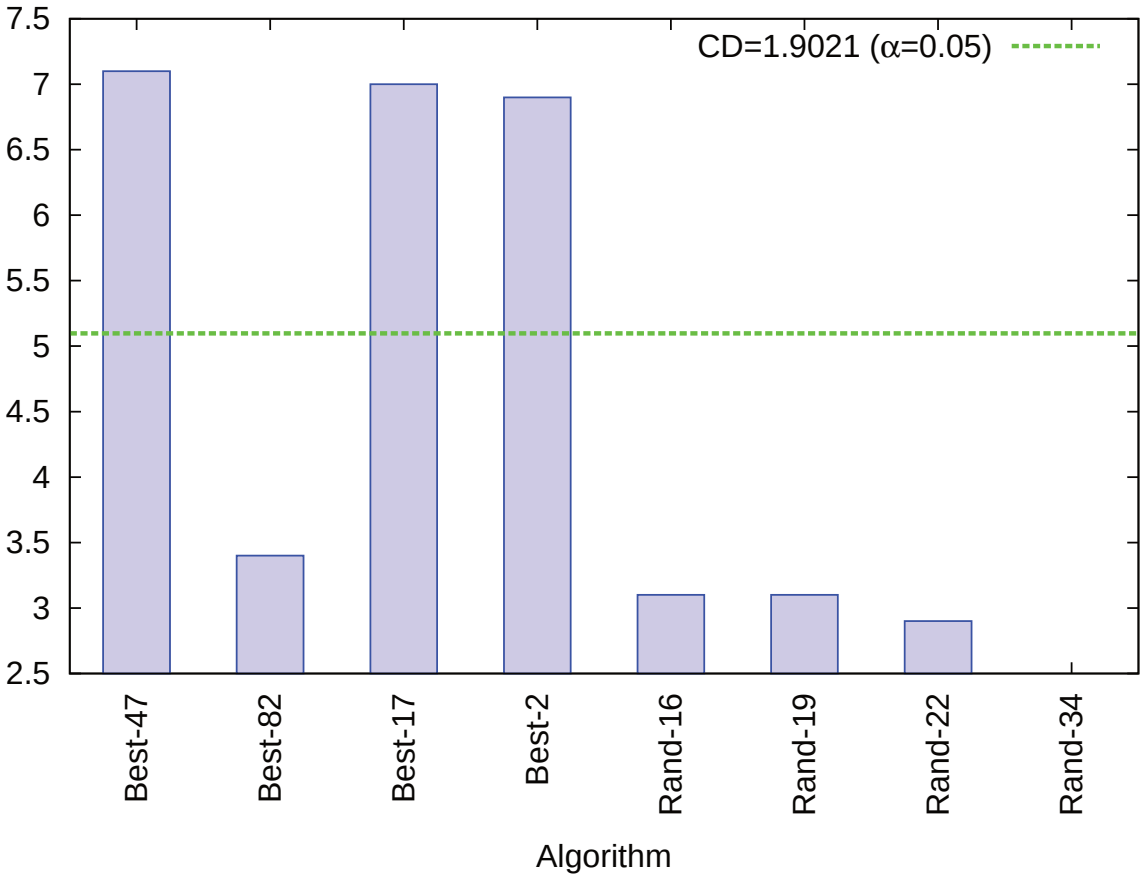
- Drawn are also standard deviation values of the means.

- Best results: DE/best with $NP$=64, $pmax$=5, $NP_{\min}$=20.

# Results – comparing selected optimization algorithms

Bonferroni-Dunn test on UGPP

- control algorithm: `DE/best` with $NP$=64, $pmax$=5, $NP_{\min}$=20,
- 51runs, 12scenarios, MAXFES 2048.




The control algorithm outperforms some `DE/best` and all `DE/rand` variants.

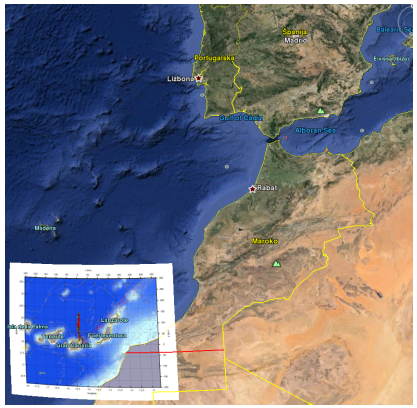
# Summary

- Underwater glider, satelite navigation and autonomy,
- real-time data streams about the environment,
- challenge of short-time mesoscale eddies on GC islands,
- the task of optimal trajectory, optimization algorithms, DE,
- example approach, real mission results, p201,ESTOC2013_3,
- differential evolution,
- mechanism of using population size reduction,
- improved results on test scenarios.


- This work was partially funded by the
  - Slovenian Research Agency, project P2-0041 and
  - Canary Island government and FEDER funds, project 2010/62.

# Thank you very much for listening!




Thanks to the organisers of the Eurocast 2015 conference.

# Questions and suggestions?