

Računska inteligenca in optimizacija

POSVET "Izzivi v svetu masivnih podatkov"
UM FERI, 3. 2. 2017 v predavalnici G2 Alfa

doc. dr. **Aleš Zamuda**

The
Austrian Federal Ministry of Science,
Research and Economy
In cooperation with the
Institute for the Danube Region and Central Europe
confers the

Danubius Young Scientist
Award 2016

To

Aleš Zamuda

Paul Böck

Dr. Erhard Basch
Deputy Director General and Chairman
of the Institute for the Danube Region
and Central Europe

Oliver

Dr. Reinhard Mitterhofer
Head of the Department of
Science, Research and Economy

Vienna, November 09, 2016





Winners of SPC 07 with the Jury members



SIIF Seoul 2012, IEEE R8SPC 2007, IEEE YP 2016

Pregled predavanja

1. Raziskovalna orodja
2. Računska inteligenca - primer diferencialne evolucije
3. Optimizacija navigacije avtonomnih globokomorskih sond
4. Optimizacija razporejanja energetske proizvodnje
5. Prostorska rekonstrukcija rastočih dreves in računanje z ekosistemi

Raziskovalna orodja: projekti

- ▶ COST CA15140: Improving Applicability of **Nature-Inspired Optimisation** by Joining Theory and Practice (ImAppNIO)
- ▶ ICT COST Action IC1406 High-Performance Modelling and Simulation for **Big Data** Applications (cHiPSet)
- ▶ Vabljena predavanja - Norveški finančni mehanizem, Erasmus+
- ▶ Projekti s študenti: Po kreativni poti, Omrežne in mobilne igre (igame.feri.um.si).

Raziskovalna orodja: literaturna mreža

- ▶ Odgovorni urednik (*Associate Editor*) pri reviji:
- ▶ **Swarm and Evolutionary Computation** (SWEVO, JCR&Scopus Q1).



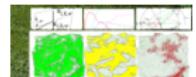
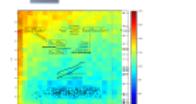
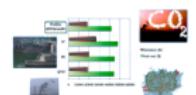
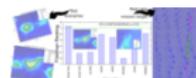
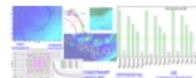
- ▶ Recenzent pri 30+ znanstvenih revijah (\approx Q1) in 50+ konferencah.

*"Nothing in the world
takes place without
optimization, and there
is no doubt that all
aspects of the world
that have a rational
basis can be explained
by optimization
methods"*

Leonhard Euler (1744)

Raziskovalna orodja: objave v literaturi (2014–2016)

- ▶ **A. Zamuda, J. D. Hernández Sosa, L. Adler. Constrained Differential Evolution Optimization for Underwater Glider Path Planning in Sub-mesoscale Eddy Sampling.** Applied Soft Computing, 2016, vol. 42, pp. 93-118. DOI 10.1016/j.asoc.2016.01.038.
- ▶ **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, vol. 24, November 2014, pp. 95-108. DOI 10.1016/j.asoc.2014.06.048.
- ▶ **A. Glotić, A. Zamuda. Short-term combined economic and emission hydrothermal optimization by surrogate differential evolution.** Applied Energy, 1 March 2015, vol. 141, pp. 42-56. DOI 10.1016/j.apenergy.2014.12.020.
- ▶ **A. Zamuda, J. Brest. Self-adaptive control parameters' randomization frequency and propagations in differential evolution.** Swarm and Evolutionary Computation, 2015, vol. 25C, pp. 72-99. DOI 10.1016/j.swevo.2015.10.007.
- ▶ **A. Zamuda, J. Brest. Vectorized Procedural Models for Animated Trees Reconstruction using Differential Evolution.** Information Sciences, vol. 278, pp. 1-21, 2014. DOI 10.1016/j.ins.2014.04.037.
- ▶ **A. Zamuda, J. Brest. Environmental Framework to Visualize Emergent Artificial Forest Ecosystems.** Information Sciences, vol. 220, pp. 522-540, 2013. DOI 10.1016/j.ins.2012.07.031.
- ▶ **H. Hamann, Y. Khaluf, J. Botev, M. Divband Soorati, E. Ferrante, O. Kosak, J.-M. Montanier, S. Mostaghim, R. Redpath, J. Timmis, F. Veenstra, M. Wahby and A. Zamuda. Hybrid Societies: Challenges and Perspectives in the Design of Collective Behavior in Self-organizing Systems.** Frontiers in Robotics and AI, 2016, vol. 3, no. 14. DOI 10.3389/frobt.2016.00014.

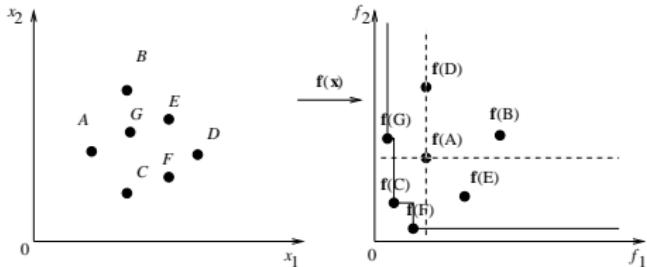


Računanje za uporabo v optimizaciji (velikih modelov)

- ▶ Čas: optimizacija porazdelitve stvari in protistvari.
- ▶ Združevanje organskih sistemov: življenje (optimizacija).
- ▶ Modeliranje optimalnosti in optimizacije (človeška orodja).
- ▶ Opis postopkov za doseg optimalnosti.
- ▶ Opis postopka optimizacije (Kepler).
- ▶ Pomik proti optimumu funkcije (Newton), gradientna metoda (Lagrange).

$$f'(x) = \frac{\Delta f(x)}{\Delta x},$$

$$f^*(x) = f(x) + \Delta x f'(x).$$



- ▶ Globalna optimizacija, matematično programiranje, digitalni računalniki.
- ▶ Prvi postopki: spuščanje po hribu, Nelder-Mead, nadzorovano naključno preiskovanje, simulirano ohlajanje, iskanje s tabuji.
- ▶ Numerična optimizacija: z omejitvami, večmodalna, večkriterijska, dinamična, s šumom, velikih dimenzij, zavajujoča, neločljiva.

Globalna optimizacija: diferencialna evolucija (DE)

- ▶ Populacijski realno kodiran iskalni algoritem za globalno optimizacijo v zveznih prostorih,
 - ▶ z **evolucijskim procesom** po **generacijah** izboljšuje **populacijo vektorjev**,
 - ▶ za vsak nov vektor v populaciji izvrši **evolucijske operatorje**,
- ▶ izbrana **strategija** *jDE/rand/1/bin*
 - ▶ **mutacija:** $v_{i,G+1} = x_{r_1,G} + F \times (x_{r_2,G} - x_{r_3,G})$,
 - ▶ **križanje:**

$$u_{i,j,G+1} = \begin{cases} v_{i,j,G+1} & \text{če } \text{rand}(0, 1) \leq CR \text{ ali } j = j_{rand} \\ x_{i,j,G} & \text{sicer} \end{cases} \text{ in}$$

$$\mathbf{x}_{i,G+1} = \begin{cases} \mathbf{u}_{i,G+1} & \text{če } f(\mathbf{u}_{i,G+1}) < f(\mathbf{x}_{i,G}) \\ \mathbf{x}_{i,G} & \text{sicer} \end{cases},$$

Algoritem DE

1: **postopek** osnovni algoritem DE/rand/1/bin (Storn, 1997)

Require: $f(\mathbf{x})$ – kriterijska funkcija; D , NP , G – krmilni parametri DE

Ensure: \mathbf{x}_{best} – optimirani parametri za dano kriterijsko funkcijo

2: Uniformno naključno inicializiciraj populacijo ($\mathbf{x}_{i,0}$, $i = 1..NP$);

3: **for** DE generacijska zanka g (dokler $g < G$) **do**

4: **for** DE iteracijska zanka i (za vse vektorje $\mathbf{x}_{i,g}$ v trenutni populaciji) **do**

5: DE izračun poskusnega vektorja $\mathbf{x}_{i,g}$ (mutacija, križanje):

$$6: \quad \mathbf{v}_{i,g+1} = \mathbf{x}_{r_1,g} + F \times (\mathbf{x}_{r_2,g} - \mathbf{x}_{r_3,g});$$

$$7: \quad \mathbf{u}_{i,j,g+1} = \begin{cases} \mathbf{v}_{i,j,g+1} & \text{če } rand(0, 1) \leq CR \text{ ali } j = j_{rand}; \\ \mathbf{x}_{i,j,g} & \text{sicer} \end{cases};$$

8: DE selekcija z oceno ustreznosti $f(\mathbf{u}_{i,G+1})$:

$$9: \quad \mathbf{x}_{i,g+1} = \begin{cases} \mathbf{u}_{i,g+1} & \text{če } f(\mathbf{u}_{i,g+1}) < f(\mathbf{x}_{i,g}); \\ \mathbf{x}_{i,g} & \text{sicer} \end{cases};$$

10: **end for**

11: **end for**

12: **vrni** najboljši najden posameznik (\mathbf{x}_{best});

Nekatere razširitve DE

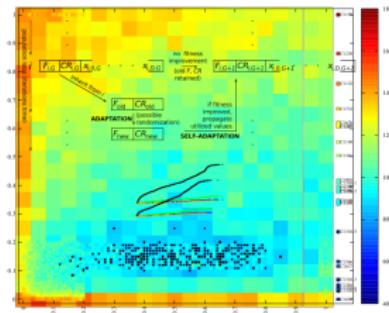
- ▶ Objavljeni algoritmi (CEC – svetovna prvenstva):
 - ▶ SA-DE (CEC 2005: SO) – poglavje v knjigi JCR,
 - ▶ MOjDE (CEC 2007: MO) – vs. DEMO 40/57 I_R , 39/57 I_H ,
 - ▶ DEMOwSA (CEC 2007: MO) – rang #3, 33 citatov,
 - ▶ DEwSAcc (CEC 2008: LSGO) – 37 citatov,
 - ▶ DEMOwSA-SQP (CEC 2009: CMO) – rang #2, 10 citatov,
 - ▶ DECMOSA-SQP (CEC 2009: CMO) – rang #3 na 2 funkcijah,
 - ▶ jDE_{NP,MM} (CEC 2011: RWIC) – objava SIDE 2012,
 - ▶ SPSRDEMMS (CEC 2013: RPSOO)
 - **SWEVO 2015 RAMONA / SNIP 5.220**
- ▶ Ovrednotenje na svetovnih tekmovanjih,
 - ▶ več krat po nekaj kriterijih najboljši.
- ▶ Ovrednotenje na različnih industrijskih izzivih
 - ▶ RWIC - CEC,
 - ▶ rekonstrukcija proceduralnih modelov dreves,
 - ▶ satelitska navigacija avtonomnih robotskih morskih sond,
 - ▶ procesiranje besedil, ...

Samoprilagajanje krmilnih parametrov

- ▶ Z bolj primernimi vrednostmi krmilnih parametrov ima proces iskanja boljšo konvergenco,
- ▶ zato se iskanje hitreje približuje boljšim rešitvam,
- ▶ tepreživijo z večjo verjetnostjo in ustvarijo več potomcev (razširijo še svoje krmilne parametre)
- ▶ Nova študija, cca. 10 milijonov zagonov SPSRDEMMS:
A. Zamuda, J. Brest. Self-adaptive control parameters' randomization frequency and propagations in differential evolution. Swarm and Evolutionary Computation, 2015, letn. 25C, str. 72-99.

DOI 10.1016/j.swevo.2015.10.007.

– **SWEVO 2015 RAMONA / SNIP 5.220**



Robotske morske sonde

- ▶ angl. *glider* - jadrilica; *underwater glider* - podmorska jadrilica?
- ▶ angl. *Autonomous Underwater Vehicle (AUV)*
≠
angl. *Unmanned Aerial Vehicle (UAV)*
- ▶ Primeri avtonomnih vozil in model Slocum:
oprema več mio EUR, ULPGC – prof. Hernandez Sosa

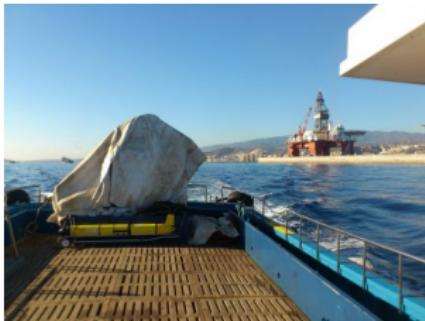


Viri:

http://upload.wikimedia.org/wikipedia/commons/e/ed/Black_Hornet_Nano_Helicopter_UAV.jpg
http://upload.wikimedia.org/wikipedia/commons/d/d2/MiniUmjetna_glider-Auvpicture_5.jpg

Robotska morska sonda Slocum G2

- ▶ Velika vzdržljivost: 25 do 365 dni,
- ▶ doseg 600 – 1500 km (alk. bat.), 4000 – 6000 km (Li⁺ bat.)
 - ▶ vzugonski pogon: horizontalno 0,35m/s (0,68 vozla),
 - ▶ 2 vozla s propelerjem.
- ▶ Potopi do globine 1000 metrov, dolg doseg, modularnost,
- ▶ integrira senzorje fizikalnih in biokemičnih parametrov²
 - ▶ temperature, zasoljenosti, raztopljenega kisika, kalnosti, klorofila in morskih tokov - možna hitra zamenjava senzorjev.

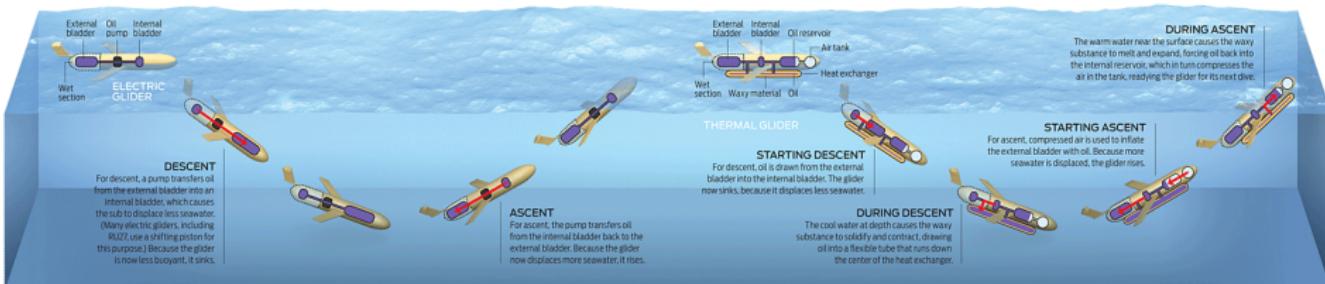
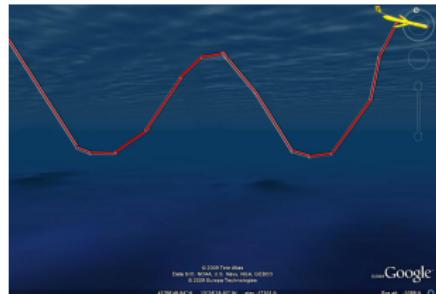


¹ lh6.googleusercontent.com/-Mq308aI1s2g/UHVf4k3uoI/AAAAAAAACbw/LeiYHXMQRbs/s640/PA060013.JPG

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

Vzgonski pogon in uporabnost podmorskih sond

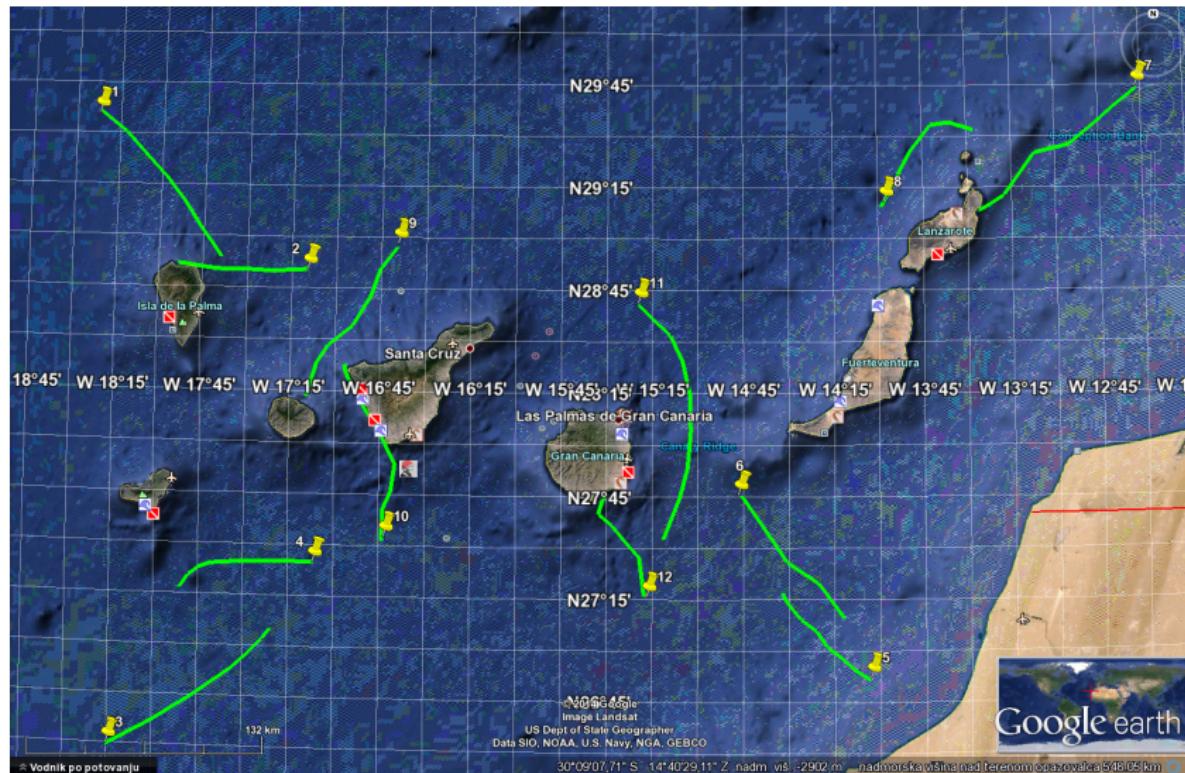
- ▶ Pogon "yo-yo" porabi malo energije, večinoma ob spustu in dvigu (črpalka), tudi za ohranjanje smeri porabi malo energije.
- + Uporaba: izboljševanje oceanskih modelov z realnimi podatki,
- + dejanski podatki na mestu zajema,
- + vzorčenje tokov naftnih izpustov,
- + inspekcija in detekcija,
- + spremljanje vodov in
- + realnočasovno spremljanje senzorskih podatkov.



¹ <http://www.i-cool.org/wp-content/uploads/2009/11/google-earth-glider-path.jpg>

² <http://spectrum.ieee.org/image/1523708>

Priprava – simulacijski scenariji



≈ Vodnik po potovanju

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

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 , τ_1 , and τ_2 (jDE constants)

Ensure: \mathbf{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 individual $\mathbf{x}_{i,g}$ computation (adaptation, mutation, crossover):

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

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

$$\mathbf{v}_{i,g+1} = \mathbf{x}_{r1,g} + F_{i,g+1}(\mathbf{x}_{r2,g} - \mathbf{x}_{r3,g});$$

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

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 $j = \{2, 3, \dots, D\}$:

$$b_j = b_{j-1} + u_{i,j,g+1};$$

$$\mathbf{p}'_{j-1} = \text{simulate_drift}(\mathbf{p}_{j-1}, \text{map});$$

$$\{\mathbf{p}_j\} = \{\mathbf{p}_{j-1}\} \cup \text{simulate_stint}(\mathbf{p}'_{j-1}, b_j, \text{map});$$

end for

$$f(\mathbf{u}_{i,g}) = ||\mathbf{p}_D - \mathbf{p}_{\text{target}}||;$$

DE selection:

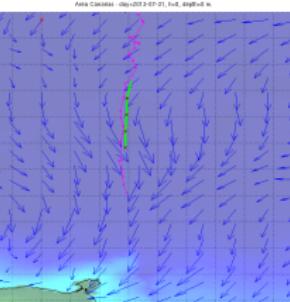
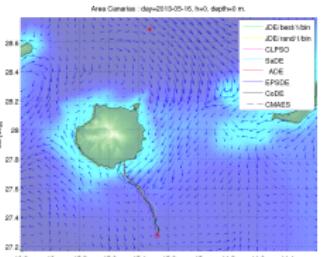
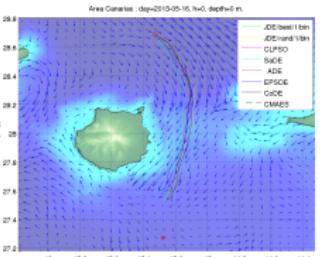
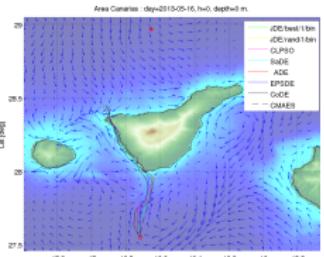
$$\mathbf{x}_{i,g+1} = \begin{cases} \mathbf{u}_{i,g+1} & \text{iff } f(\mathbf{u}_{i,g+1}) < f(\mathbf{x}_{i,g}) \\ \mathbf{x}_{i,g} & \text{otherwise} \end{cases};$$

end for

end for

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

Optimizacija trajektorije p201,ESTOC2013_3



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 $j = \{2, 3, \dots, D\}$:

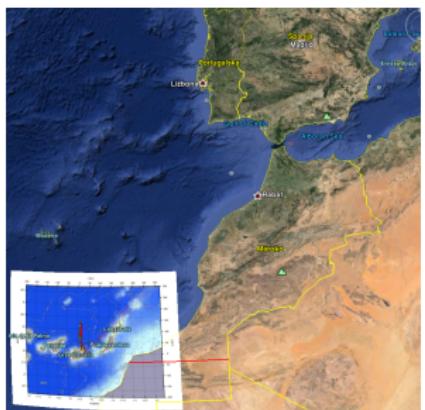
$$b_j = b_{j-1} + u_{i,j,g};$$

$$\mathbf{p}'_{j-1} = \text{simulate_drift}(\mathbf{p}_{j-1}, \text{map});$$

$$\{\mathbf{p}_j\} = \{\mathbf{p}_{j-1}\} \cup \text{simulate_stint}(\mathbf{p}'_{j-1}, b_j, \text{map});$$

end for

$$f(\mathbf{u}_{i,g}) = \| \mathbf{p}_D - \mathbf{p}_{\text{target}} \|;$$



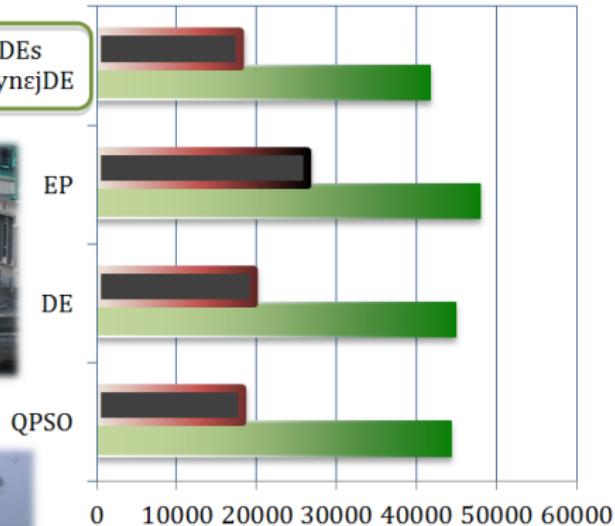
- + Big Data, MyOcean IBI, satelitska povezava, GPS lociranje

Trajektorija in zbrani podatki z Google Earth KML datoteko na COST EGO.

Razporejanje proizvodnje v energetiki z diferencialno evolucijo in omejitvami - uvod



PSADEs
@NPdynnejDE



■ Emission (lb)
■ Fuel cost (\$)



Modeli sistemov elektrarn (1/3)

- ▶ Nov pristop za optimizacijo razporejanja proizvodnje električne energije med **hidroelektrarnami** in **termoelektrarnami**.
- ▶ Pristop omogoča hitrejši izračun kot je to bilo možno do sedaj:
 - ▶ saj pristop predstavi **pred-izračunan nadomestni model**,
 - ▶ ta model se med optimizacijo celotnega optimiziranega modela hidro in termo enot ne spreminja (**predprocesiranje**) in
 - ▶ pristop hrani dobljene vrednosti parametrov nadomestnega modela v praktično uporabni natačnosti.

Modeli sistemov elektrarn (2/3)

- ▶ Rezultati tega problema razporejanja na uveljavljenih testih iz literature kažejo:
večjo izboljšavo na vseh scenarijih in vseh kriterijih
v primerjavi z do sedaj znanimi pristopi. (To velja tudi do današnjega dne.)
- ▶ Dva algoritma sta izdelana v tem pristopu:
 - ▶ prvi algoritem (NPdynējDE) naslavljaja posebno obravnavo omejitev in optimizira razporejanje za **termoelektrarne**;
 - ▶ drugi algoritem (PSADEs) **uporabi rezultate prvega** algoritma, da optimizira **skupno proizvodnjo** s hidro elektrarnami,
 - ▶ pri tem pa oba uporabita **praktično natančnost parametrov** za urenje obremenitev termoelektrarn.

Modeli sistemov elektrarn (3/3)

- ▶ Objava prispeva k **razvoju algoritmov** z namenom,
 - ▶ izboljšati učinkovitost **produkije električne energije** in
 - ▶ **zmanjšanja izpustov ter ogljičnega odtisa,**
 - ▶ s hkratno zadovoljtvijo **24-urnih sistemskih zahtev** v razporejanju povpraševane energije ter vseh operativnih zahtev.
- ▶ Literatura:

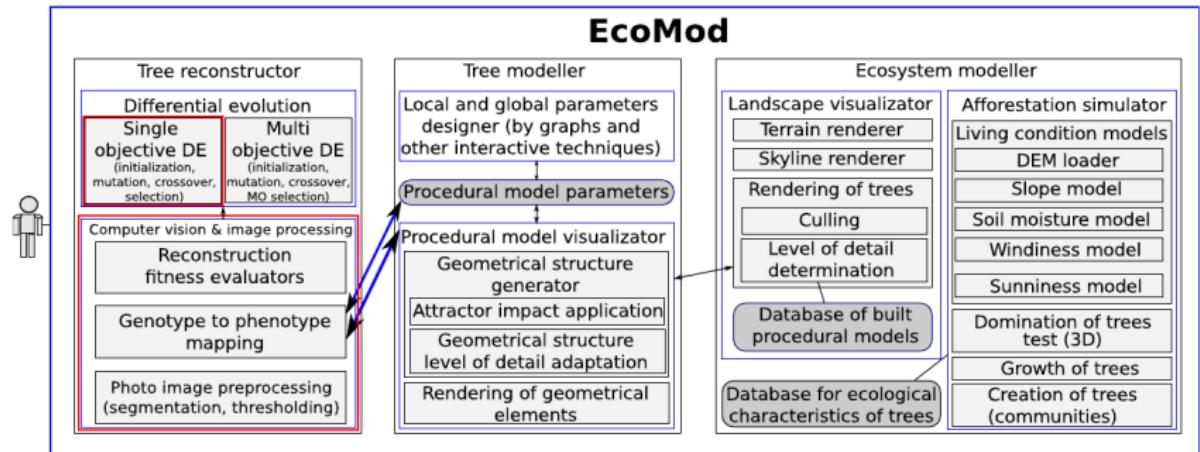
A. Glotić, A. Zamuda. *Short-term combined economic and emission hydrothermal optimization by surrogate differential evolution*. **Applied Energy**, 1 March 2015, letn. 141, str. 42-56. DOI 10.1016/j.apenergy.2014.12.020. **IF=5.613**

Rezultati in primerjave

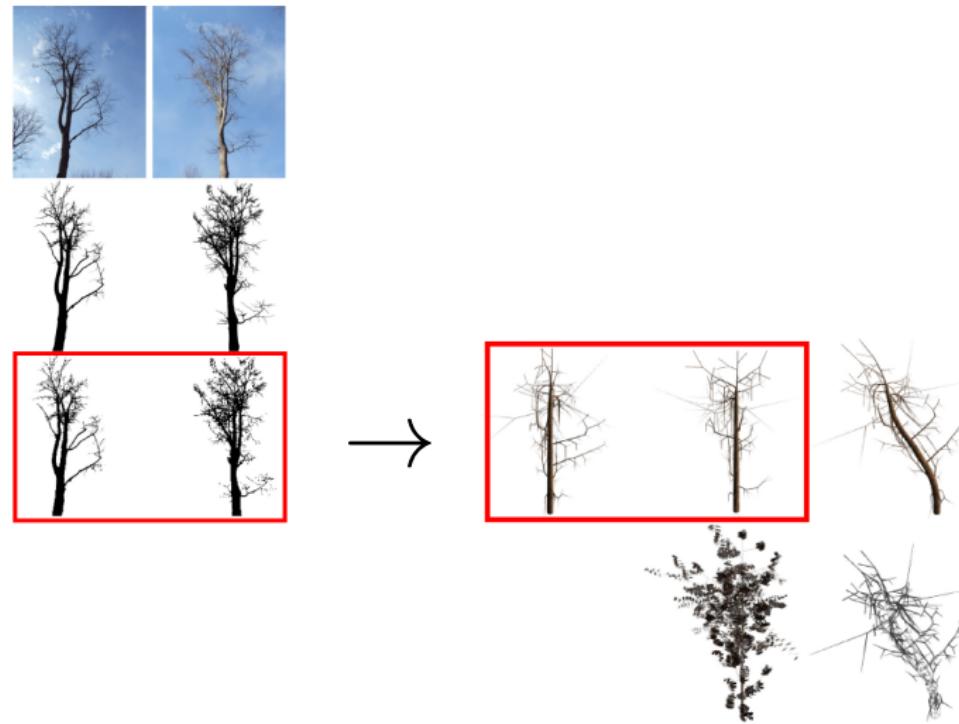
Primerjava rezultatov na različnih načinih razporejanja,
glede na ostale najboljše objave do sedaj

Methods	Case	Fuel cost (\$)	Emission (lb)	CPU time (s)
Proposed approach	ELS	40859.84	22767.78	26.24
	EES	49717.22	16495.62	26.25
	CEES	41697.23	17981.40	26.27
EP [17]	CEES	47906.00	26324.00	4582.00
	ELS	43500.00	21092.00	72.96
DE [32]	EES	51449.00	18257.00	72.74
	CEES	44914.00	19615.00	74.97
QPSO [27]	ELS	42359.00	31298.00	/
	EES	45271.00	17767.00	/
	CEES	44259.00	18229.00	/

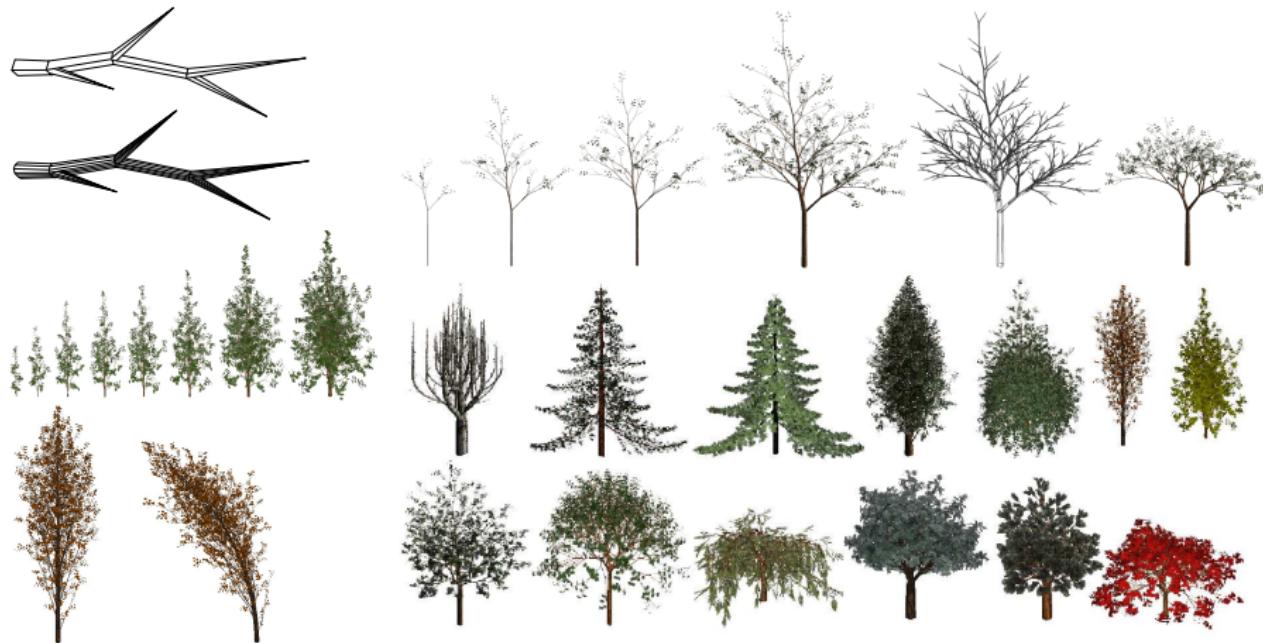
EcoMod: modeliranje ekosistemov z računsko inteligenco



Diferencialna evolucija in velikost genotipa proceduralnih modelov dreves



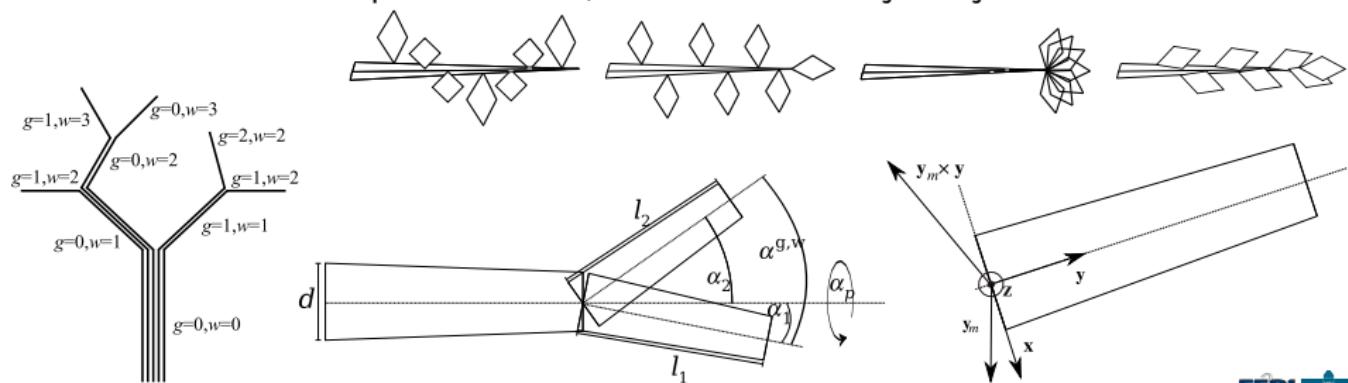
Predstavljivost dreves v paketu EcoMod



- ▶ Listavci ali iglavci z različnimi vejitvenimi strukturami,
- ▶ rast in zibanje v vetru: vsako vejo in list animiramo v realnem času.

Numerično kodirani proceduralni model drevesa

- ▶ 3D modele dreves učinkovito predstavimo s proceduralnimi modeli
 - ▶ naš programski sistem EcoMod uporablja numerično kodiran proceduralni model s fiksno dimenzijo,
 - ▶ posebej primerno za optimizacijo z jDE.
- ▶ Parametriziran proceduralni model zgradi 3D strukturo drevesa in vse njegove sestavne dele:
 - ▶ z rekurzivnim izvajanjem fiksnega postopka (procedure),
 - ▶ nad danim naborom numerično kodiranih vhodnih parametrov
 - ▶ npr. širina debla, relativna dolžina vej in vejitvena struktura.



1: postopek rekonstrukcija(\mathbf{z}^*)

Require: S_0 - največje število žil v deblu; drugi parametri jDE in EcoMod.

Ensure: rekonstruiran parametriziran proceduralni trodimenzionalni model drevesa.

2: Uniformno naključno generiraj začetno populacijo DE $\mathbf{x}_{i,0} \in [0, 1]$ za $i = 1..NP$;

3: for DE generacijska zanka g (dokler FEs < 10000) do

4: for DE iteracijska zanka i (za vse vektorje $\mathbf{x}_{i,g}$ v trenutni populaciji) do

5: DE izračun poskusnega vektorja $\mathbf{x}_{i,g}$ (samoprilagajanje, mutacija, križanje):

$$F_{i,g+1} = \begin{cases} F_i + rand_1 \times F_u & \text{če } rand_2 < \tau_1; \\ F_{i,g} & \text{sicer} \end{cases};$$

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

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

$$u_{i,j,g+1} = \begin{cases} v_{i,j,g+1} & \text{če } rand(0, 1) \leq CR_{i,g+1} \text{ ali } j = j_{rand}; \\ x_{i,j,g} & \text{sicer} \end{cases};$$

10: DE ocena ustreznosti (preslikava genotipa, upodabljanje in primerjava):

$$\mathbf{z}_1 = \mathbf{g}(\mathbf{u}_{i,g}, \beta_1), \mathbf{z}_2 = \mathbf{g}(\mathbf{u}_{i,g}, \beta_2) \quad \{\text{Postopek drevo, dvakrat za en genotip}\}$$

$$h(\mathbf{z}_1) = \sum_{x,y} m_1(z_{x,y}^1, z_{x,y}^*) + \sum_{x,y} m_1(z_{x,y}^*, z_{x,y}^1); \quad \{\text{Metrika podobnosti}\}$$

$$h(\mathbf{z}_2) = \sum_{x,y} m_1(z_{x,y}^2, z_{x,y}^*) + \sum_{x,y} m_1(z_{x,y}^*, z_{x,y}^2); \quad \{\text{Za 3D, rotacija } 90^\circ\}$$

$$f(\mathbf{u}_{i,g}) = f(\mathbf{g}(\mathbf{u}_{i,g}, \beta_1), \mathbf{g}(\mathbf{u}_{i,g}, \beta_2)) = h(\mathbf{z}_1) + h(\mathbf{z}_2); \quad \{\text{Ocena ustreznosti}\}$$

15: DE selekcija:

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

17: end for

18: end for

19: vrni najboljšega najdenega posameznika;

1: postopek drevo($g, w, S_0, L_0, l_0, \mathbf{M}_0, \mathbf{M}_{m;0}^{-1}, \mathbf{M}_{w;0}^{-1}$):

Require: g, w - Graveliusov in Weibullov indeks osnovne veje; S_0 - število žil v osnovni veji; L_0 - relativna dolžina osnovne veje; l_0 - dejanska dolžina osnovne veje; \mathbf{M}_0 - koordinatni sistem trenutne veje; $\mathbf{M}_{m;0}^{-1}$ - inverzna matrika zasukov za gravimorfizem v koordinatnem sistemu osnovne veje; $\mathbf{M}_{w;0}^{-1}$ - inverzna matrika zasukov za usmerjen veter v koordinatnem sistemu osnovne veje; globalni $k_d, k_c, l_{type}, k_s^{g,w}, M^{g,w}, m^{g,w}, k_l^{g,w}, \alpha_m^{g,w}, \alpha^{g,w}, t, k_f, w_s, w_g$.

Ensure: upodobljeno drevo

- 2: $d := k_d \sqrt{S_0}; \{izračun debeline osnovne veje\}$
- 3: upodobi osnovno vejo (\mathbf{M}_0, l_0, d);
- 4: **if** $S_0 = 1$ **then** upodobi liste(l_{type}); **return**; **end if**
- 8: $S_1 := \lceil 1 + k_s^{g,w} (S_0 - 2) \rceil; \{\text{število žil v glavni podveji}\} S_2 = S_0 - S_1; \{\text{število žil v stranski podveji}\}$
- 10: $r_1 := \max \left\{ \min \left\{ \sqrt{\frac{S_1}{S_0}}, M^{g,w} \right\}, m^{g,w} \right\}; r_2 := \max \left\{ \min \left\{ \sqrt{\frac{S_2}{S_0}}, M^{g,w} \right\}, m^{g,w} \right\}; \{\text{razmerja dolžin}\}$
- 12: $L_1 := r_1 L_0; L_2 := r_2 L_0; \{\text{relativna dolžina podvej}\}$
- 14: $l_1 := k_l^{g,w} L_1; l_2 := k_l^{g,w} L_2; \{\text{dejanska dolžina podvej}\}$
- 16: $\alpha_1 := k_c \sqrt{\frac{S_2}{S_0}} \alpha^{g,w}; \alpha_2 := \alpha^{g,w} - \alpha_1; \{\text{kota vejitev}\}$
- 18: $\alpha_x(t) := \sin(t + R_x) w_s (1 - k_f) l_0; \alpha_z(t) := \sin(t + R_z) w_s (1 - k_f) l_0; \{\text{animacija neusmerjenega vetra}\}$
- 20: $\alpha_w := \frac{S_0}{S} w_g; \{\text{animacija usmerjenega vetra}\}$
- 21: $\mathbf{M}_1 := \mathbf{R}_{w_0}(\alpha_w,) \mathbf{R}_z(\alpha_1 + \alpha_z(t)) \mathbf{R}_x(\alpha_x(t)) \mathbf{R}_y(\alpha_p) \mathbf{R}_{y \times y_m}(\alpha_m^{g,w}) \mathbf{T}_y(l_0) \mathbf{M}_0;$
- 22: $\mathbf{M}_2 := \mathbf{R}_{w_0}(\alpha_w,) \mathbf{R}_z(\alpha_2 + \alpha_z(t)) \mathbf{R}_x(\alpha_x(t)) \mathbf{R}_y(\alpha_p) \mathbf{R}_{y \times y_m}(\alpha_m^{g,w}) \mathbf{T}_y(l_0) \mathbf{M}_0;$
- 23: $\mathbf{M}_{m;1}^{-1} := \mathbf{R}_{y \times y_m}(-\alpha_m^{g,w}) \mathbf{R}_y(-\alpha_p) \mathbf{R}_x(-\alpha_x(t)) \mathbf{R}_z(-\alpha_1 - \alpha_z(t)) \mathbf{M}_{m;0}^{-1};$
- 24: $\mathbf{M}_{m;2}^{-1} := \mathbf{R}_{y \times y_m}(-\alpha_m^{g,w}) \mathbf{R}_y(-\alpha_p) \mathbf{R}_x(-\alpha_x(t)) \mathbf{R}_z(-\alpha_2 - \alpha_z(t)) \mathbf{M}_{m;0}^{-1};$
- 25: $\mathbf{M}_{w;1}^{-1} := \mathbf{R}_{y \times y_m}(-\alpha_m^{g,w}) \mathbf{R}_y(-\alpha_p) \mathbf{R}_x(-\alpha_x(t)) \mathbf{R}_z(-\alpha_1 - \alpha_z(t)) \mathbf{R}_{w_0}(-\alpha_w) \mathbf{M}_{w;0}^{-1};$
- 26: $\mathbf{M}_{w;2}^{-1} := \mathbf{R}_{y \times y_m}(-\alpha_m^{g,w}) \mathbf{R}_y(-\alpha_p) \mathbf{R}_x(-\alpha_x(t)) \mathbf{R}_z(-\alpha_2 - \alpha_z(t)) \mathbf{R}_{w_0}(-\alpha_w) \mathbf{M}_{w;0}^{-1};$
- 27: drevo ($g + 1, w + 1, S_2, L_2, l_2, \mathbf{M}_2, \mathbf{M}_{m;2}^{-1}, \mathbf{M}_{w;2}^{-1}$); **{razvijemo stransko vejo}**
- 28: drevo ($g, w + 1, S_1, L_1, l_1, \mathbf{M}_1, \mathbf{M}_{m;1}^{-1}, \mathbf{M}_{w;1}^{-1}$); **{razvijemo glavno vejo}**

Računanje z ekosistemi, algoritmi



1: **postopek** simulacija ekosistema

Require: v - seznam rastlinskih vrst;

r - seznam rastlin za rastlinske vrste;

f - seznam matrik za vrednosti faktorjev na površju;

Ensure: izvaja koračno simulacijo razvoja ekosistema

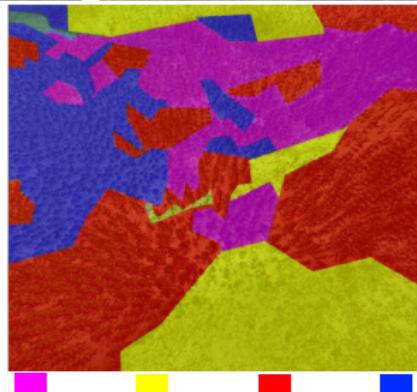
2: **loop**

3: dodaj nove rastline rastlinskim vrstam(v, r);

4: zrasti vse rastline (r, f);

5: odstrani odmrle rastline(r);

6: **end loop**



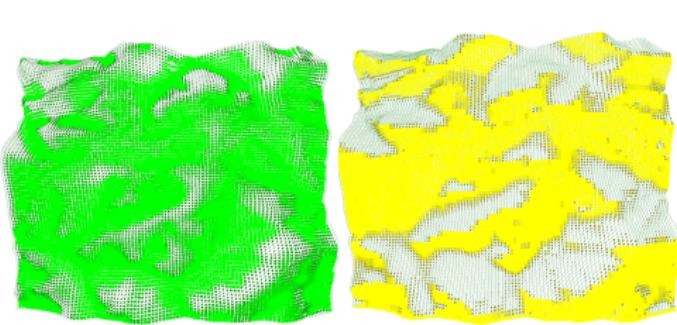
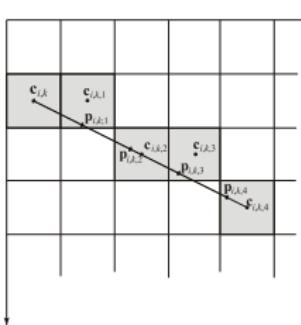
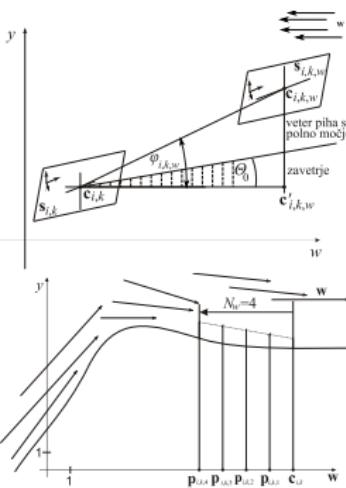
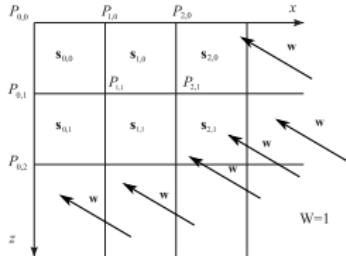
Bukve

Grmovje

Smreke

Javorji

Animacija ekosistemov: vlaga, vetrovnost



Zeleno: vlaga, rumeno: vetrovnost (jakosti).

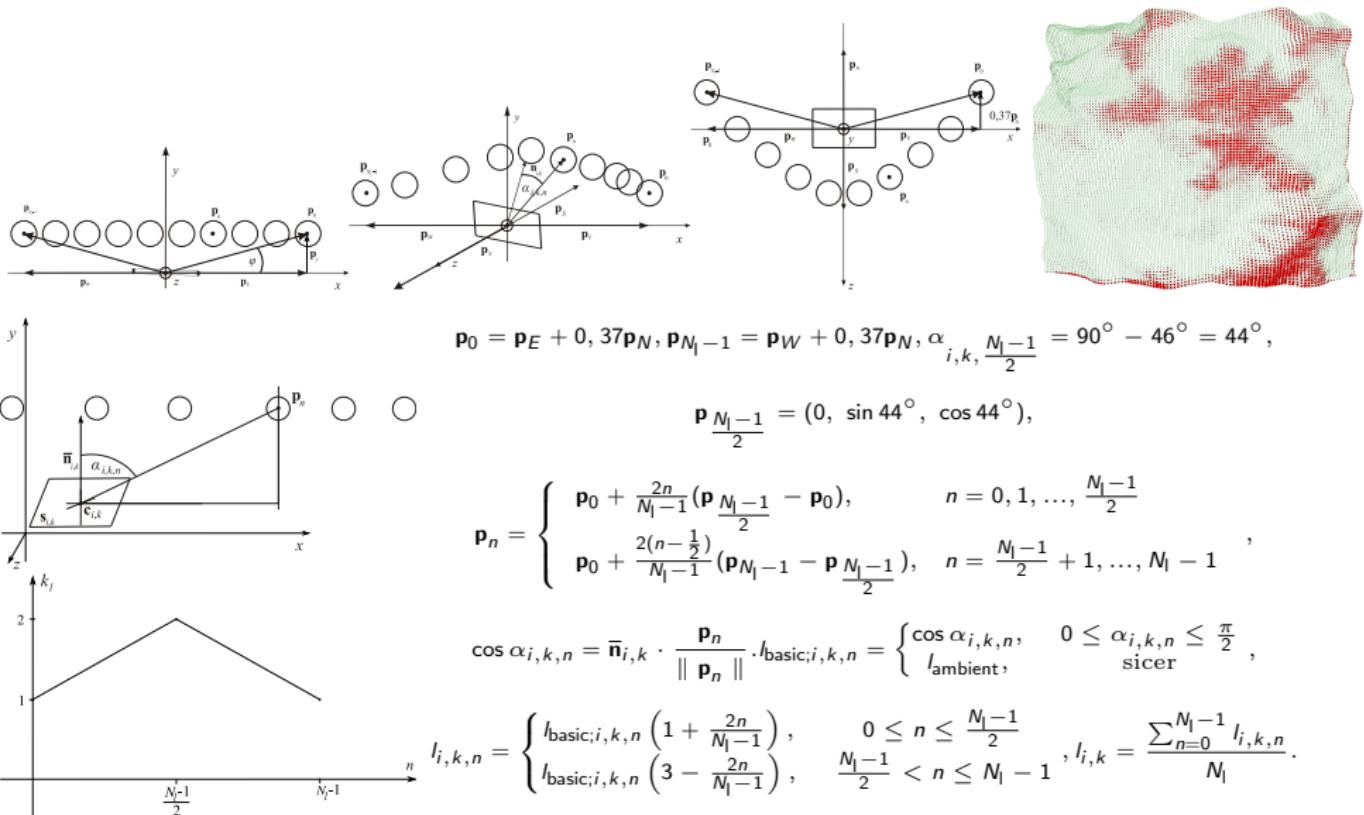
$$m_{i,k} := \begin{cases} m_{i,k} + m_w, & \bar{y}_{i,k} \leq \bar{y}_w \\ m_{i,k}, & \text{sicer} \end{cases}, \quad \Delta m_{i,k} = \begin{cases} 0, & n_{\text{neigh};i,k} = 0 \\ \frac{s_{i,k} k_m}{n_{\text{neigh};i,k}}, & n_{\text{neigh};i,k} = 1, 2, 3, 4 \end{cases},$$

$$m_{i\pm 1,k\pm 1} := m_{i\pm 1,k\pm 1} + \Delta m_{i,k}, \quad m_{i,k} := m_{i,k} - n_{\text{neigh};i,k} \Delta m_{i,k}.$$

$$\varphi_{i,k,w} = \angle (\mathbf{c}_{i,k,w}, \mathbf{c}_{i,k}, \mathbf{c}'_{i,k,w}), \quad \mathbf{c}_{i,k} = [c_{x;i,k} \quad c_{y;i,k} \quad c_{z;i,k}]^T,$$

$$\varphi_{i,k,w} = \arctan \frac{c_{y;i,k,w} - c_{y;i,k}}{\|\mathbf{c}'_{i,k,w} - \mathbf{c}_{i,k}\|}, \quad w_{i,k} = \begin{cases} \varphi_{i,k}, & \varphi_{i,k} \leq \Theta_0 \\ 1, & \text{sicer} \end{cases}.$$

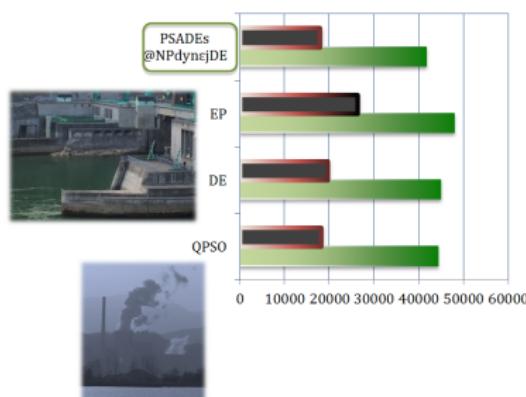
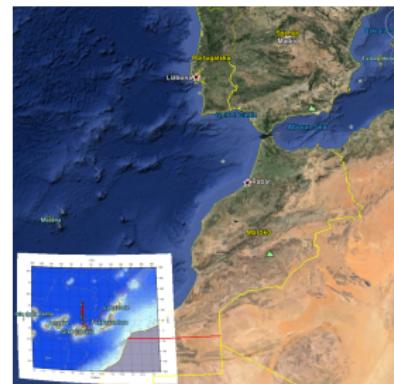
Animacija ekosistemov: osončenost



Povzetek

1. Raziskovalna orodja
2. Računska inteligenca - primer diferencialne evolucije
3. Optimizacija navigacije avtonomnih globokomorskih sond
4. Optimizacija razporejanja energetske proizvodnje
5. Prostorska rekonstrukcija rastočih dreves in računanje z ekosistemi

Vprašanja?



Emission (lb)
Fuel cost (\$)

