Physarum KISS: an Innovative Algorithm for the Preliminary Analysis of Multi-Gravity Assist Interplanetary Trajectories

L. Masi[a] , **M. Vasile**[a]

[a]Department of Mechanical & Aerospace Engineering
University of Strathclyde
75 Montrose Street, G1 1XJ, Glasgow, UK
luca.masi@strath.ac.uk, massimiliano.vasile@strath.ac.uk

Abstract

Multi-gravity assist (MGA) trajectories are used in space engineering for reducing the cost (propellant and time of flight) of interplanetary missions by using the gravitational field of celestial bodies. The probes Mariner 10 (mission to Mercury, launched by NASA on 1973), Voyager 1 (furthest human-made object, NASA, 1977), MESSENGER (mission to Mercury, NASA, 2004) and Cassini (mission to Saturn, NASA-ESA, 1997) are examples. Making use of gravity assist manoeuvres around planets, the velocity vector of the spacecraft relative to the Sun changes during a passage in proximity of the planet, while the velocity vector relative to the planet is rotated (no change in module). The problem of finding the sequence of planets (including resonances, i.e. multiple passages in proximity of the same planet) that guarantees the best transfer to a target space object is a fascinating problem in combinatorics and discrete optimisation. The computational complexity is NP-hard [1]. In this work, a simple and innovative bio-inspired multidirectional algorithm for preliminary analysis of multiple gravity assist planets' sequences is introduced and compared to a branch & cut algorithm.

Keywords: Physarum, Multi-Gravity Assist.

1 Introduction

In the MGA problems, algorithms using exact methods ([1], [2]) to analyse sequences of planets were proposed in the past: they are computationally expensive and engineering judgment is necessary before and during the optimisation process in order to discard not interesting solutions. [5] proposed a method for analysing low energy gravity-assist trajectories to Jupiter from Venus, Earth and Mars (number of fly-bys is fixed to 4) by dividing the problem in two stages: the first, called *path-finding*, uses a simplified model for trajectory evaluations based on gravity assist potential plot for each planet; after discarding unfeasible solutions, selected combinations of planets are analysed using the software STOUR. A similar method using Tisserand's graphs is proposed in [2]. In [7] a planning and scheduling approach is proposed; the solution is incrementally built with a modified ACO (ant colony optimisation) strategy. Recently [9] and [8] proposed to tackle the problem using a modified genetic algorithm (GA) where a "hidden" or "null" gene, representing the possibility of no fly-by, is introduced for dealing with sequences of variable length. In particular [9] proposed a solver divided in two parts: the *outer loop* and the *inner loop*. The outer loop generates the planet sequence through the GA that is passed to the inner loop for trajectory optimisation with a Monotonic Basin Hopping (MBH) algorithm.
In this work, a simple and innovative bio-inspired multidirectional algorithm for preliminary analysis of multiple gravity assist planets' sequences is presented. It is a variation of the algorithm developed in [3] for discrete decision making. As introduced in Sect. 2 the search strategy takes inspiration from the behaviour of Physarum Polycephalum, an amoeboid organism that in its plasmodium state extends and optimises a net of veins looking for food. Decision sequences are incrementally built from multiple directions: in a direct way, from Earth to a target planet, and in a backward way, from target planet to Earth. The

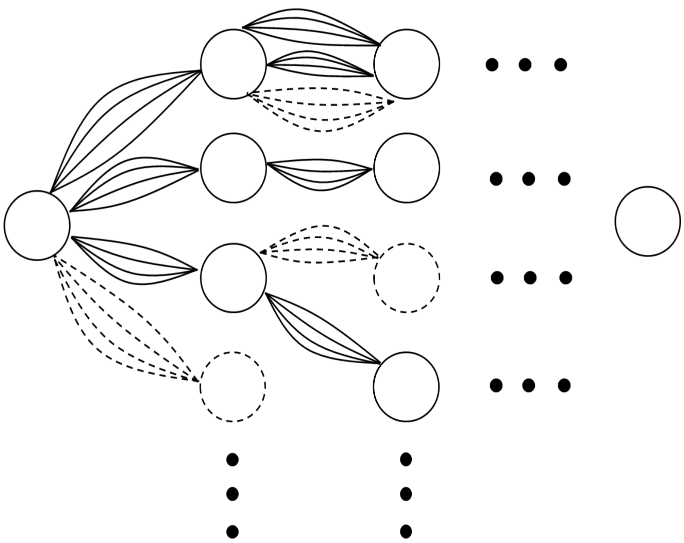

Figure 1: Graph modeling of MGA problems. Dotted lines are decisions not built yet.

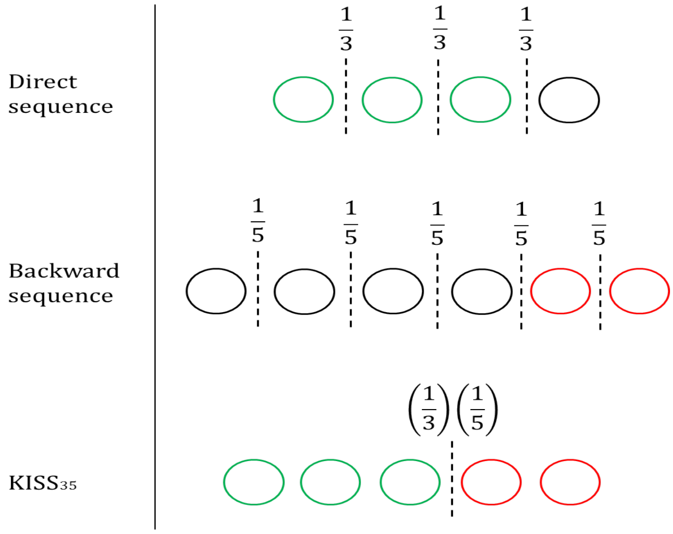

Figure 2: KISS communication strategy.

objective function to be minimised is the time of flight (ToF). The physical model developed for planet sequence evaluation, introduced in Sect. 3, is based on some simplified hypothesis. All planets are supposed to move in the same plane, with a circular orbit. Transfer feasibility is based on energy considerations only, using patched-conic approximation as in [1, 5]. No deep space manoeuvres are considered and phasing problem is neglected (each planet is assumed to be in the proper position for fly-by). The result is a set of optimal sequences that are feasible from an energy point of view, with phasing problem to be analysed at a later stage. The low computational effort needed to produce solutions, compared to exact methods as in [1] or branch and cut algorithms (Sect. 4), and their good quality is attractive and interesting for preliminary mission design, first guesses for more detailed analysis (*path-finding*) and *outer loop* sequence generation.

2 Biology and mathematical modeling

Physarum Polycephalum is a large, single-celled amoeboid organism that exhibits intelligent plant-like and animal-like characteristics. Its main vegetative state, the *plasmodium*, is formed of a network of veins (*pseudopodia*). The stream in these tubes is both a carrier of chemical and physical signals, and a supply network of nutrients throughout the organism [13]. *Physarum* searches for food by extending this net of veins, whose flux is incremented or decremented depending on the food position with reference to its centre. The longest is the path connecting the centre with the source of food, the smallest is the flux.
The multi-gravity assist problem can be modelled using a tree-like topology. Starting from the Earth, that represents the root node, each following planet for fly-by is a children. The graph can be grown incrementally by the algorithm with time, where each precedent child becomes the parent of the following children up until the target planet is reached. Virtual agents, following a defined heuristic , i.e. *Physarum*, build the graph looking for optimal sequences. Each arc connecting a parent and a children has an associated cost evaluated making use of the model in Sect. 3. Since the number of planets in the solar system is fixed, and 4 possible transfers are possible between two subsequent planets (see Sect. 3) , the tree can be mapped into a layer-graph where each arc has 4 possible costs associated. The topology is shown in Fig. 1. Due to physical model constraints, each arc has an identification number (ID) depending on the story (previous sequence of planets and fly-bys) of the walking virtual agent: each agent can build or walk an arc if arc's ID is consistent with agent's memory. This concept will be further discussed in Sect. 3. When the target planet is reached or the chosen sequence of planets is longer than a fixed value (i.e. 7),

Algorithm 1 Multidirectional incremental modified Physarum solver

```
initialize m, ρ, GF, N_agents, p_ram, r_ini
generate a random route from start to destination in DF and BF
for each generation do
  for each virtual agent in all directions (DF and BF) do
    if current node ≠ end node then
      if rand ≤ p_ram then
        create a new arc, building missing links and nodes
      else
        move on existing graph using Eq. (4).
      end if
    end if
  end for
  look for possible matchings with KISS
  contract and dilate veins using Eqs. (2), (3), (5)
  update fluxes and probabilities using Eqs. (1), (4)
end for
```

the sequence is terminated.

Physarum algorithmic for MGA problems Using the Hagen-Poiseuille law, the flux through the net of *Physarum* veins is [15, 11, 13, 14]:

$$Q_{ij} = \frac{\pi r_{ij}^4}{8\mu}\frac{\Delta p_{ij}}{L_{ij}} \tag{1}$$

where Q_{ij} is the flux between i and j, μ is the dynamic viscosity, r_{ij} the radius, L_{ij} the length and Δp_{ij} the pressure gradient. Diameter variations allow a change in the flux. Veins' dilation due to an increasing number of nutrients flowing can be modeled using a monotonic function of the flux:

$$\left.\frac{d}{dt}r_{ij}^{(n)}\right|_{dilation} = m\frac{r_{ij}^{(n)}}{L_{tot}^{(n)}} \tag{2}$$

where m is the linear dilation coefficient, $r_{ij}^{(n)}$ the radius of the veins traversed by virtual agent n and $L_{tot}^{(n)}$ the total length of the vein, i.e. the total cost of the complete decision (Earth to target planet) taken by agent n. Veins' contraction due to evaporative effect can be then assumed to be linear with radius:

$$\left.\frac{d}{dt}r_{ij}\right|_{contraction} = -\rho r_{ij} \tag{3}$$

where $\rho \in [0,1]$ is defined evaporation coefficient. The probability associated with each vein connecting i and j is then computed using a simple adjacency probability matrix based on fluxes:

$$P_{ij} = \begin{cases} \frac{Q_{ij}}{\sum_{j\in N_i} Q_{ij}} & if\ j \in N_i \\ 0 & if\ j \notin N_i \end{cases} \tag{4}$$

where N_i is the set of neighbour for i.

An additive term in the veins' dilation process, whose first main term is expressed in Eq. (2) was added in the algorithm. This dilation is:

$$\left.\frac{d}{dt}r_{ij_{best}}\right|_{elasticity} = GF r_{ij_{best}} \tag{5}$$

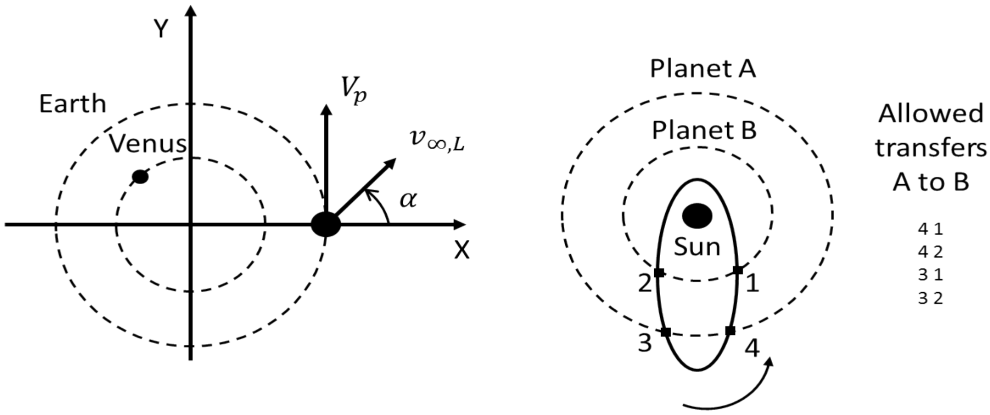

Figure 3: Initial conditions and different transfers on the same orbit.

where GF is the growth factor and $r_{ij_{best}}$ the veins' radii of the best chain of veins. In this work the initial value of the radius r_{ini} is set equals to 1 with the constraint that it can't be greater than 5, meaning that the vein dilation is blocked if $r \geq 5$ until $r < 5$ due to evaporative effect.
The incremental growth of the decision network is then based on a weighted roulette. Nutrients inside veins are interpreted as virtual agents that move in accord with adjacency probability matrix in Eq. (4). Once a node is selected (parent), there is an *a priori* probability p_{ram} of ramification towards new nodes (children) that are not yet connected with the actual parent node. If ramification is the choice, the virtual agents build an arc between the ramifying node and the closer (in terms of physical distance) missing planet. This process leads to Algorithm 1.

Introducing Multidirectionality and Matching: KISS Taking inspiration from the ideas proposed in [3] and [4], a multidirectional version of the algorithm was developed. Virtual agents are able to build arcs and nodes both in the classic direct way from Earth to the target planet (direct flux, DF) and in a backward direction from target planet to Earth (back flow, BF). Since the MGA model developed in Sect. 3 is not invertible, the local information used, i.e. L_{ij}, for arc evaluations is the distance between planets instead of the time of flight (ToF), while the global information, i.e. L_{tot}, is computed using the MGA model with the inverted backward sequence. The two superimposed graphs built using two counter-expanding *Physarums*, communicate using the KISS strategy: the best 10 BF and DF sequences are taken and each sequence DF communicate with each BF sequence. The communication occurs by randomly selecting a cut point (arc cut) for the two sequences and trying to connect the two graphs with a matching arc. This strategy is shown in Fig. 2, where the probability associated with cut at dotted lines (direct and backward sequence) and recombination ($KISS_{35}$ is an example) are reported.

3 MGA Model

The problem of reaching a target planet via multiple gravity assists with other planets, can be modeled using patched conic approximation. The problem is assumed to be bi-dimensional, i.e. the motion of the planets and the spacecraft occurs on a plane. All planets orbits are assumed to be circular. After launch from departure planet, a fly-by with a first planet at fixed altitude takes place, introducing a change in the velocity vector of the spacecraft w.r.t. the Sun. This change in velocity is the reflection of a change in the spacecraft energy due to the gravitational interaction between planet and spacecraft. A limited number of fly-bys can take place before the insertion in the target orbit. In the algorithm presented in this work, a maximum number of fly-bys can be selected, although the algorithm is able to build sequence of planets of

Table 1: Success rate: Physarum, Physarum KISS. Earth to Jupiter transfers: EJ7 ($v_{\infty,L} = 7km/s$, $\alpha = 1.5\pi$), EJ5 ($v_{\infty,L} = 5km/s$, $\alpha = 1.5\pi$).

	EJ5		EJ7	
Function evaluations:	*5000* ,	*30000*	*5000* ,	*30000*
Physarum	0.66	0.80	0.16	0.81
Physarum KISS	0.45	0.93	0.55	0.94

Table 2: Mean simulation time (non dimensional) at 30000 function evaluations. Non dimensional coefficient: Physarum mean simulation time at 30000 function evaluations. Earth to Jupiter transfers: EJ7 ($v_{\infty,L} = 7km/s$, $\alpha = 1.5\pi$), EJ5 ($v_{\infty,L} = 5km/s$, $\alpha = 1.5\pi$).

	EJ5	EJ7
Physarum	1	1
Physarum KISS	0.74	0.68

variable length. Resonances are taken into account, i.e. multiple passages in proximity of the same planet. Planets are assumed to be in the right position for fly-bys at every time, i.e. phasing is neglected. The result is a set of sequences that are feasible from an energy point of view but not from a phasing point of view, as in [1]. Having assigned a launch and direction velocity ($v_{\infty,L}$, α) from Earth (whose orbiting velocity is V_p) a set of transfers are possible (Fig. 3). In particular, referring to Fig. 3, if Planet A is the Earth and Planet B the following planet, and the launch happens in 4, then two possible transfer arcs are possible, i.e. 4-1 and 4-2. This is true for each transfer between two planets. In particular, since two intersections and two types of fly-bys are possible (accelerating or decelerating), then each doublet of planets leads to four possible transfers, as in Fig. 1.

4 Results and Conclusion

The testing procedure proposed in [16] was used to assess the performance of the proposed algorithms. In particular the parameter of performance chosen is the success rate that represents the number of time an algorithm is able to obtain the optimal solution over a defined number of runs, i.e. 200, with a defined tolerance, i.e. 10^{-5}. The success rate can be computed over the number of function evaluations. In this work each arc evaluated is considered a function evaluation, meaning that for example a sequence of 5 planets is made of 4 function evaluations.
The test cases here presented are the two Earth to Jupiter transfers EJ5 ($v_{\infty,L} = 5km/s$, $\alpha = 1.5\pi$) and EJ7 ($v_{\infty,L} = 7km/s$, $\alpha = 1.5\pi$). The limit altitude at fly-by epoch is $200km$ for Mercury, Venus, Earth, Mars and Pluto, 5 Jovian radii at Jupiter, 2 planetary radii at Saturn, and one planetary radius at Uranus and Neptune. Input parameters for the algorithms are: $m = 0.005$,$\rho = 0.0001$, $GF = 0.005$, $N_{agents} = 20$, $p_{ram} = 0.6$. The best solution found by the algorithm is the sequence of planets EVVEMJ (Earth, Venus, Venus, Earth, Mars, Jupiter) both for EJ7 and EJ5 with cost 1.48 and 1.38 years, and is in line with the results obtained with the exhaustive search presented in [1]. It should be noted that this low value in time of flight is due to the fact that phasing is neglected, meaning that the transfer is feasible from an energy point of view but not from a phasing point of view. This, as explained in [2], underestimates the time of flight required for most sequences. Table 1 shows that a multidirectional approach (Physarum KISS) introduces a considerable gain in the success rate (above 10% for both test cases at 30000 function

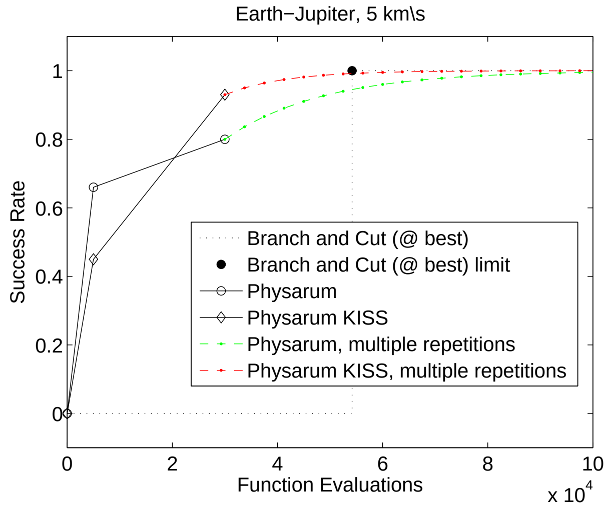

Figure 4: Success rate: Physarum, Physarum KISS and Branch & Cut. Earth to Jupiter transfer EJ5, $v_{\infty,L} = 5km/s$, $\alpha = 1.5\pi$.

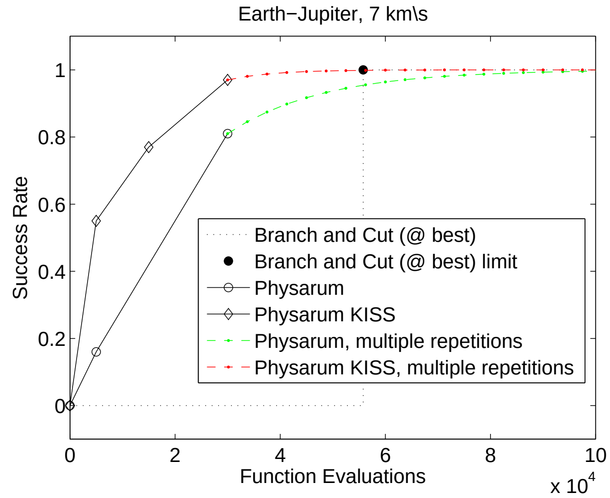

Figure 5: Success rate: Physarum, Physarum KISS and Branch & Cut. Earth to Jupiter transfer EJ7, $v_{\infty,L} = 7km/s$, $\alpha = 1.5\pi$.

evaluations and above 20% at 5000 function evaluations). Furthermore there is a considerable gain in computational time by using the Physarum KISS algorithm (around 30 time faster, as reported in Table 2) thanks to the matching approach that rapidly combines and discovers optimal sequences. It is of interest also a comparison among these two bio-inspired algorithms and branch & cut algorithms. Fig. 4 and Fig. 5 report this comparison in terms of the success rate vs. function evaluations. The branch & cut algorithm implemented starts building a tree of sequences from Earth and proceeds level by level, cutting branches when the value of the objective function for the calculated sequence so far is above a defined threshold (i.e. the best value found for a complete sequence during the branch & cut run). As in Fig. 4 and Fig. 5, the branch & cut algorithm shape behaviour is a step function and the best solution is reached at around 55000 function evaluations both for EJ5 and EJ7. Physarum algorithm is able to achieve 90% of success at around 40000 evaluations, while the addition of KISS strategy increase the success at 100% (note that these values are calculated considering the probability of success by repeating the algorithm multiple time, i.e. starting from experimental data at 30000 function evaluations, the probability of unsuccess is $1 - p_{30}^{n}$ where n is the number of repetions and p_{30} the experimental data at 30000 evaluations). These results demonstrates an advantage in using the proposed bio-inspired heuristic. Moreover a branch & cut strategy is not able to find certain sub-optimal sequences that are cut during the computation without being completed, and may be useful for preliminary analysis. This is not the case using Physarum heuristic.

5 Acknowledgments

This work has been partially supported by the CNES - Thales Alenia Space contract for the study on Global Optimisation of Interplanetary Trajectories.

References

[1] S Pessina, S Campagnola, M Vasile. Preliminary Analysis of Interplanetary Trajectories with Aero-gravity and Gravity Assist Manoeuvres. *54th International Astronautical Congress*, Bremen, Germany, 2003.

[2] N.J Strange, J.M Longuski. Graphical Method for Gravity-Assist Trajectory Design. *Journal of Spacecraft and Rockets*, 39(1):9–16, 2002.

[3] L Masi, M Vasile. A Multidirectional Modified Physarum Solver for Optimal Discrete Decision Making. *Proceedings of International Conference on Bio-Inspired Optimization Methods and their Applications, BIOMA 2012*, 1:207–218,2012.

[4] L Masi, M Vasile. Optimal Multi-Objective Discrete Decision Making Using a Multidirectional Modified Physarum Solver. *EVOLVE 2012 International Conference*, Mexico City, Mexico, 2012.

[5] A.E. Petropoulos, J.M Longuski, E.P. Bonfiglio. Trajectories to Jupiter via Gravity Assist from Venus, Earth and Mars. *Journal of Spacecraft and Rockets*, 37(6):776–783, 2000.

[6] M Vasile, P. De Pascale. On the Preliminary Design of Multiple Gravity-Assist Trajectories. *Journal of Spacecraft and Rockets*, 42(4): 794–805, 2006.

[7] M Ceriotti, M Vasile. Automated MGA Trajectory Planning with an ACO-inspired Algorithm. *Acta Astronautica*, 67(910): 1202-1217, 2010.

[8] A Gad, O Abdelkhalik. Hidden Genes Genetic Algorithm for Multi-Gravity-Assist Trajectory Optimization. *Journal of Spacecraft and Rockets*, 48(4): 629–641, 2011.

[9] J.A Englander, B.A Conway, T Williams. Automated Interplanetary Trajectories Planning. *AIAA/AAS Astrodynamics Specialist Conference*, Minneapolis, Minnesota, 2012.

[10] A Adamatzky, G.J Martinez, S.V Chapa-Vergara, R Asomoza-Palacio, C.R Stephens. Approximating Mexican Highways with Slime Mould. *Natural Computing*, 10(3): 1195–1214, 2011.

[11] D.S Hickey, L.A Noriega. Insights into Information Processing by the Single Cell Slime Mold Physarum Polycephalum. *UKACC Control Conference*, , Manchester, UK, 2008.

[12] T Nakagaki, H Yamada, A Toth. Maze-Solving by an Amoeboid Organism. *Nature*, 407(6803): 470, 2000.

[13] A Tero, K Yumiki, R Kobayashi, T Saigusa, T Nakagaki. Flow-Network Adaptation in Physarum Amoebae. *Theory in Biosciences*, 127(2): 89–94, 2008.

[14] A Tero, R Kobayashi, T Nakagaki. Physarum Solver: a Biologically Inspired Method of Road-Network Navigation. *Physica: A Statistical Mechanics and its Applications*, 363(1): 115–119, 2006.

[15] A Tero, K Yumiki, R Kobayashi, T Saigusa, T Nakagaki. Rules for Biologically Inspired Adaptive Network Design. *Science Signalling*, 327(5964): 439, 2010.

[16] M Vasile, E Minisci, M Locatelli. An Inflationary Differential Evolution Algorithm for Space Trajectory Optimization *IEEE Transaction on Evolutionary Computation*, 2(2): 267–281, 2011.