Novelty Search of Soft Robot Morphologies for Space Exploration

???

ABSTRACT

don't worry about this, we gonna write it at the end

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

General Terms

Theory

Keywords

soft robotics, novilty search, CPPN, HyperNEAT, VoxCAD

1. INTRODUCTION

Motivation, space, small bodies, passive actuation, story of Rosetta/Philae - stupid rigid probe without locomotion, we can do better!

2. BACKGROUND

- gaits at different gravity levels (Ariadna Space Gaits); fixed morphology, rigid body dynamics

2.1 Soft Robots

Soft robotics is a highly promising field of research dedicated to the science and engineering of soft materials in mobile machines. As the name suggests soft robots [23, 14] are made entirely of soft materials mimicking animals or animal-parts that consist only of soft tissue (elephant trunk, tongue, worm, octopus, etc.). Having no rigid parts in their design the degrees of freedom are infinite and the possible ways of motion can become extremely complex. In traditional robotics, joints and rigid parts predefine the space of possible movement and sometimes restrict the robot's locomotion strategy or *gait* to a specific set. In soft robotics, the absence of rigid parts can on the one hand make the design of the locomotion strategy exceptionally tortuous, on the other hand the gait alternatives are limitless.

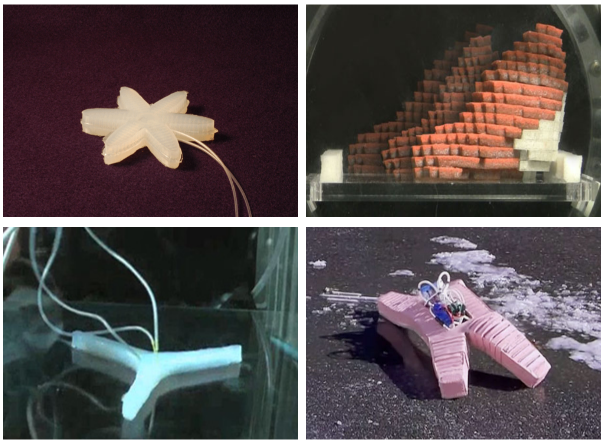

Figure 1: Soft robots can be actuated through air pressure tubes (top-left), pressure variations (top-right), or internal explosions (bottom-left). Autonomously actuated soft robot [22], it is able to withstand extreme temperatures and variant terrain types (Bottom-right).

The design and development of soft robotics is not an easy task, while the actuation of such soft structures is the most challenging task. Actuating soft materials can be done in many ways including pneumatic systems [7, 18], hydraulic, internal body explosions, passive actuation triggered by pressure or temperature variations and others [8, 17]. Figure 1 illustrates four different ways that soft robot bodies can be actuated. Regardless traditional ways of actuating soft material robots, three dimensional printing is now giving the freedom for multi-material structures to be created, which also explodes the number of possibilities for the design of a soft structure robot. Autonomously actuated soft robots [22] (see Fig. **??**) can also be designed having multiple advantages over rigid body robots such as resistance under extreme temperatures and the capability of locomotion on terrains of variant types.

Although soft robotics research field is in an early stage, it is growing fast. Some of the characteristics that make soft robots interesting to explore are the infinite number of degrees of freedom and the variety of materials (mostly elastic) that can be used, in the contrary to rigid robotics that are mostly made out of metals and plastic. Nevertheless, struc-

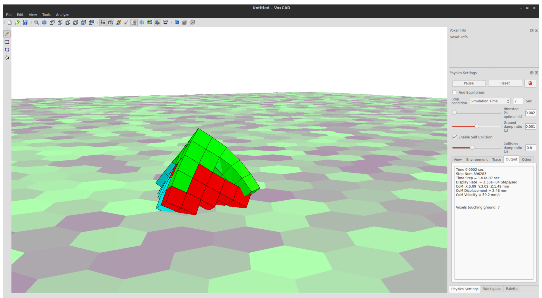

Figure 2: VoxCAD (Voxel CAD), a cross-platform open source voxel modeling and analyzing software.

ture design and control of soft robotics remain challenging mostly because of their soft bodies can only be represented in continuous state spaces, where only analytic methods can be proven successful.

2.2 Evolution of Soft Robots

Topological optimization techniques can be applied to soft robots [5] for producing functionality in the design. Evolution of soft material robots as it was shown in [3], can result in soft robots able to produce locomotion. The possibility of evolving these soft structures using an indirect encoding was of interest to be exploited by [1]. A powerful generative encoding, CPPNs [20], was used to generate soft voxel-formed three-dimensional structures, coupled with the use of NEAT algorithm which ensures the increasing complexity of the networks produced. The superiority of this kind of generative encoding was verified against direct encoding, showing how CPPNs can take advantage of their geometrical properties. Evaluation was done by a simple displacement measure while evolution tended to evolve different kinds of locomotion strategies and morphologies as the fitness function was penalized for different parameters. Furthermore, it has been shown that evolving morphologies (CPPNs) in lower resolutions and then applying the same networks for higher resolution structures can be beneficial, since the locomotion behaviors in lowers structures also apply in higher saving computational time. An earlier work [6], apart from the generative encoding of CPPNs, made use of *Gaussian Mixture* and *Discrete Cosine Transform* to produce amorphous soft body structures. The simultaneous evolution of soft robot morphology and control was also investigated by recent work [15]. Some aspects of soft robot evolution were verified in this work, namely muscle placement and muscle-firing patterns can be evolved given a fixed body shape and fixed material properties. Furthermore, material properties can be co-evolved alongside locomotion strategies. Finally, a developmental encoding was introduced, allowing more complex parts to be added to soft robotic structures during the evolution.

3. METHODOLOGY

3.1 VoxCAD simulator

Most work to simulate interactions and deformations within and between soft material bodies are mostly focused on the graphical part of the problem [2] sacrificing the accuracy of the simulation [21]. Three dimensional meshes [13] can represent these bodies including the dynamics of their materials. A recent work, *VoxCad* simulator [4], is focusing mostly on the physics side of the soft material interactions not at the expense of a low frame rate. VoxCad is a modeling and analyzing open-source software that can simulate soft material deformations and interactions. In Figure 2 the graphical user interphase of VoxCad software is presented during the simulation of the soft body robot in the simulator. VoxCad cannot model and simulate three dimensional meshes, yet a lattice is used to represent the 3D workspace where voxels (three dimensional pixels) can be assigned different materials. Materials themselves are passive and cannot actuate without external trigger. In this simulator this external force that can actuate the materials is the temperature of the environment. Furthermore, gravity acceleration of the environment can vary.

Materials

Within the VoxCad simulation software there is the option of defining and using a palette of materials. Materials can be *passive* or *active*. Passive materials do not react to temperature changes, while active materials expand and contract in respect to their thermal properties. Red and Green are the only actuated materials with non-zero and opposite thermal expansion coefficients, meaning that their phase in respect to the actuation from temperature changes is equal to half a circle. Green voxels contract the same time red expand and vice versa, mimicking living organisms' muscle tissue. The two additional materials represent soft non-actuated tissue that can be soft (soft tissue) or hard (bones). Cyan voxels are soft having five times smaller elastic modulus of their material than Blue which have 50 `MPa`.

3.2 Compositional Pattern-Producing Networks

Compositional pattern-producing networks [20] or CPPNs are artificial neural networks with an extended set of activation functions (see Fig. 3). This set of activation functions include repetitive, symmetrical, and linear functions as it is shown in the previous figure. CPPNs can generate phenotypes that can be interpreted as distributions of points in a multidimensional Cartesian space. The genotype (CPPN) can then be queried for each coordinate of the space and gives the phenotype representation of the genotype in multiple resolutions.

Therefore and for the purpose of this work, CPPNs are queried for every coordinate of the lattice space to form a soft robot morphology, the input nodes (neurons) of the CPPN are assigned to x,y,z normalized coordinates follow-

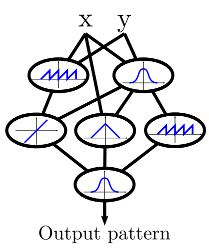

Figure 3: Compositional pattern-producing networks have identical network structure with artificial neural networks while they make use of a canonical set of activation functions.

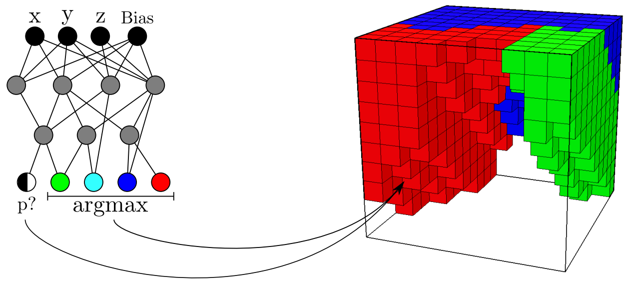

Figure 4: Each genotype (CPPN) is queried for every coordinate inside the lattice, its outputs determine the presence of a voxel and the type of its material.

ing [1], so that:

$$x, y, z \in [-1, 1]$$

A bias input node is also introduced in the genome CPPN representation, this will allow the network to produce arbitrary outputs different from the defaults when all other inputs values are set to zero. More inputs could be added to the CPPNs, for instance the distance from the center point of the Cartesian phenotype space (lattice) as described in [20] and used in [1], which naturally adds more bias towards symmetrical structures. However, the evolution of such aesthetic structures is not much of interest to exploit. CPPNs, as it is shown later in this thesis, can evolve symmetrical morphologies without this extra information input node(s). The proposed input nodes for the three dimensions of the Cartesian space provide the minimum bias to the network outputs. Figure 4 illustrates the topology of a random CPPN network with the input and output nodes previously described. The set of nodes and connections determine the *topology* of the network. The presence of a voxel in each coordinate of the lattice is determined by a single output of the CPPN, denoted with p while the selection of the material is determined by n-outputs. The node with the maximum value out of the n-outputs will determine which of the materials is going to be used in the specific voxel only in cases this is present.

3.3 Neuroevolution of Augmented Topologies

Neuroevolution of augmented topologies (NEAT) as it was first introduced by [19] is a neuroevolution method to evolve artificial neural networks. Originally, neuroevolution methods were developed to capture difficult sequential decision making, as well as to control problems. The sensory information is the input of these neural networks and decisions are the outputs. NEAT is yet another method for evolving ANNs where a few extra features are added, enables finding solutions in more demanding problems. NEAT starts the evolution process with a population of networks with simple topologies. Through the generations instead of just fixing the weights of the networks' connections, topologies are becoming more complex allowing nodes and links to be added. Meaning that during the evolution, more complicated networks will be produced, this *complexifying* technique leads to capturing more demanding solutions as it offers enough freedom to the evolution. Several aspects of this method worth mentioning where *speciation* is one of the most important. Speciation is the procedure that protects new *species* until they have enough time to evolve before comparing them with the rest of the population. The age of each species protects them for competing in equal terms with more optimized species, giving them in this way time to evolve further towards the objective function.

3.4 CPPN-NEAT

Compositional pattern-producing networks as described earlier are similar computational methods to ANNs in regards to their structure, so one can make use of the *complexifying* property to capture in this way more complex solutions. NEAT method can evolve CPPNs in the place of ANNs, since it only needs few modifications. The resulted method that evolves this generative type of genomes (CPPNs) is called CPPN-NEAT [20] and its only difference in respect to the original NEAT algorithm is the way new nodes are added to the network. The original NEAT algorithm evolves ANNs which are using sigmoid functions at every node, so every new node will carry this function. In the contrary, CPPNs use a variety of functions from a canonical set. CPPN-NEAT assigns a random function from this set to every newly added node. Previous work [1], showed that this method can indeed evolve the morphologies of the soft robots in the VoxCad simulation environment. *HyperNEAT*[1] is used for the implementation of the CPPN-NEAT algorithm.

3.5 Novelty search

Novelty search [9, 11, 10, 16] unlike traditional fitness based search is an alternative way of optimization towards an objective function without having knowledge of this objective. In simple words it is looking for a solution to a problem without knowing how close it is to solve it; fact that turns out to have a major impact to the increased performance of this method in several problem domains.

What novelty search seeks for is how interesting a new solution is in respect to all previously found ones. To define "interesting" we need to move our point of interest into behavior space which is a function of each phenotype, similar to the fitness function. Nevertheless, it fully or partially describes the behavior without directly implying the fitness function. As an example someone can think of a behavior could be defined as the recorded trajectory of a robot which tries to maximize its velocity.

As fitness is a function to measure the "goodness" of an individual, novelty measures how different an individual is from all previously found individuals. To define different a novelty metric measures the difference in the behavior space of the phenotype. Given the phenotype's behavior x a novelty measurement could be a function of x, $f(x)$ which computes how different (novel) is the specific behavior in respect to a set of other behaviors S in behavior space. As defined in [9, 11] *sparseness* can give a good measurement of how sparse is the area of a newly observed behavior. Given the behavior we can compute the sparseness by:

$$f(x) = \frac{1}{k} \sum_{i=1}^{k} dist(x, S_i) \tag{1}$$

where S is a sorted set of the closest behaviors. Sparsity measures the average distance from the k-closest behaviors.

[1]HyperNEAT `v4.0 C++` by J. Gauci code (url: `https://github.com/MisterTea/HyperNEAT`)

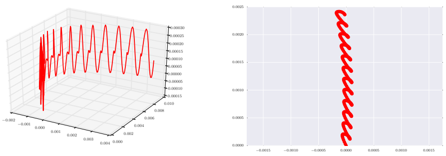

Figure 5: Observed behaviors of the soft robots used for the sparsity computation in novelty search. Three dimensional trajectories (left), and two dimensional trajectories (right).

One significant point here is that the behavior space in some domains can be limitless. However, a valid behavioral metric can be found excluding behaviors that are meaningless or do not comply with the natural limits of the problem. On the other hand, the search space in the genotype level can also be infinite especially in neuroevolution methods like NEAT where ANNs can grow during the evolution. A bounded space of understandable-valid behaviors is then the key idea of novelty search where increasingly complex behaviors present to the evolution as the complexity of the genotype grows along.

Behaviours in Novelty Search

Behavior can be defined as the way that a human/machine behaves towards or within an environment. Regarding the evolution of soft robots in the specific simulated environment, a behavior can be defined as the way soft robots behave in respect to their locomotion strategy. Every aspect of the soft robots movement that can be observed can be used to describe their behavior. Previous work [12] in a try to evolve walking three-dimensional virtual creatures used the evolved morphology of the creatures to describe their behavior. Although, in this work comparing the morphology of the evolved soft robots is similar to comparing the chromosome (CPPN) of each individual. Therefore, only the comparison of the observed behavior in phenotype level can lead the evolution towards more complex behaviors.

A straightforward function that determines the goodness of an individual is used in fitness based methods. It is expected that behaviors that contain information about the goodness (displacement) of individuals will be more successful than behaviors that include other aspects of the soft robots' behavior.

Behaviors that describe the morphology of the evolved robots have failed [12], since search is then forcing new types of morphologies without caring about the actual target of the evolution, which was the efficient locomotion. Using the same morphological novelty measure, we assure that there will not be exploration in the behavior aspect that affects the actual target of the evolution, which is to produce and evolve efficient locomotion strategies.

Figure 5 presents two of the behavior types used for the novelty metric computation. For all recorder behavior metrics a constant sampling rate ensures that all signals have the same length. The behaviors designed to describe the strategy and the efficiency of the evolved locomotion. They contain information that indirectly implies both the objectives of the evolution. *Trajectories* (2D and 3D), incorporate all the needed information such as speed, displacement, and locomotion strategy. To avert from same trajectories in all possible directions trajectories are normalized, meaning that their starting coordinates are always the start of the axes ($< 0, 0, 0 >$) and the point coordinates of the trajectory are rotated so their center of mass is normalized to a specific angle ($\theta = 90^o$). To measure the difference of two trajectories the Euclidean distances between coordinates at the same sampling time are measured, so that:

$$\text{First trajectory: } t_i = t_i^1, t_i^2, \ldots, t_i^N \tag{2}$$

$$\text{Second trajectory: } t_j = t_j^1, t_j^2, \ldots, t_j^N \tag{3}$$

$$\text{Difference: } t_i - t_j = \sum_{n=1}^{N} dist(t_i^n, t_j^n) \tag{4}$$

where n is the number of sampled coordinate points and $dist$ is the Euclidean distance. Apart from trajectory type behaviors, pace, kinetic energy, voxels touching the ground, and pressure were used to define other behavior types. The similarity or the difference of two of the same type behaviors can be determined by the equations provided while these measures of difference are used by the sparsity equation (see Eq. 1) to compute the sparseness of a given behavior in the behavior space. Individuals with novel observed behaviors (high sparseness value) are then stored in a list helping the evolution to avoid generating similar behaviors.

3.6 Fitness Elitism in Novelty Search

Elitism is the process of passing mutations or copies of the best individuals to the next generation. In this way best individuals are preserved and can be optimized later. The best individuals of each species generation are protected so they can contribute with their beneficial genes later in the evolution. Novelty search can include elitism in its selection process, and it does that by copying the most novel organisms of the current population of each species to the next. Since there is no point of changing this function, elitism can be used also to copy fit individuals within novelty search method. The way these two elitism functions can be combined together depends on the population size and the problem, while probabilistic methods can also be used. In the specific setting, both elitism function copy new individuals to the new generation with probability one. Moreover, evolution towards novelty does not get disturbed, at the same time fit individuals have the chance to be optimized further as long as they are the fittest within the species population.

3.7 Experimental setup

Each experiment consists of 10 runs under the same settings. As in [1] and for comparison purposes a population of 30 on each generation is used, the maximum number of generations in the evolution is set to 1000. Due to computationally expensive simulations, not all experiments are performed using a lattice resolution of 10^3, resolutions lower than 10^3 are used as well. More specifically, $5^3, 7^3, 10^3$ lattice resolutions are used. All settings regarding CPPN-NEAT algorithm are the same as in [1].

4. RESULTS

novelty (+ fitness) better than fitness examples of cool creatures taxonomy of the evolved creatures at different gravity levels (hoppers, 2-3-4 legs, crawler, tumbleweed)

This section presents the performance and the resulted evolved virtual soft robots by the methods described ear-

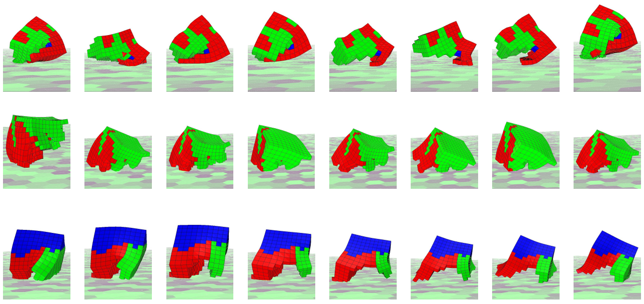

Figure 6: Champion (best overall) morphologies evolved in independent runs of fitness based search. Each row illustrates the locomotion strategy of the individuals created.

lier in this work. In addition, the performance and significant findings of the evolutionary methods used for the co-evolution of the morphology and the locomotion strategy of soft robots are discussed in details. Pure novelty search is compared in respect to the goodness measure used in the simulations (displacement of soft robots in body-lengths) to fitness based search. Elitism is used in a proposed methodology to incorporate fitness information in novelty search. Last, the performance of both methods are investigated for several levels of gravity. This will show that gravity conditions do not have any effect in favor of a specific search method. Furthermore, evolved locomotion strategies under different gravity conditions show how environmental conditions can affect the evolved morphologies and the strategies of soft robots.

In this section some of the evolved morphologies and their effective locomotion patterns evolved within fitness based and novelty search will be discussed. Apart from the performance that the two methods achieved, both of them were successful in evolving effective strategies for the locomotion of the evolved morphologies. Figure 6 shows four different gait types evolved by fitness based search. All of these morphologies are considered "good" in respect to their fitness value, meaning that they achieve to travel up to ~ 10

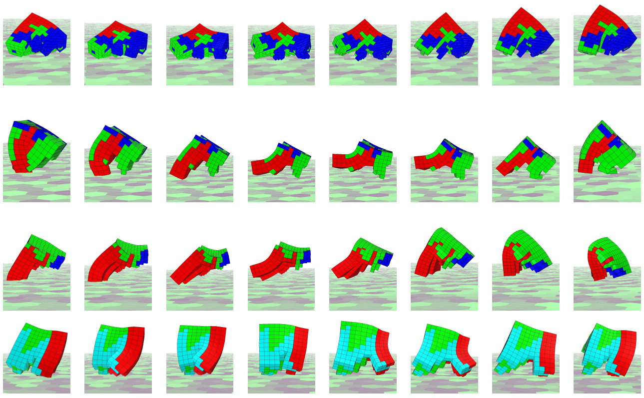

Figure 7: Champion morphologies evolved in independent runs of novelty search. Each row illustrates the locomotion strategy of the individuals created.

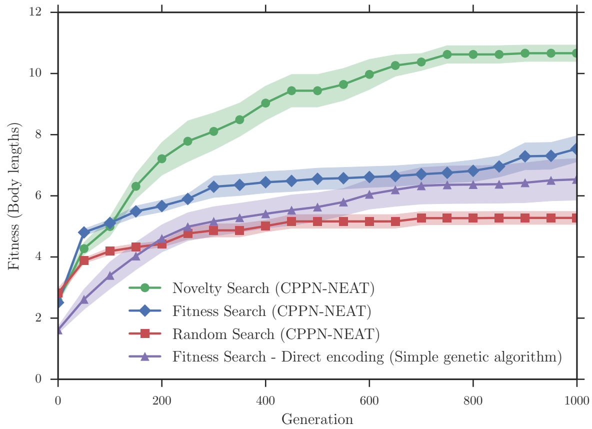

Figure 8: Comparison of simple genetic algorithm (direct encoding) against novelty-fitness-random search with generative encoding. Best fitness so far averaged over 10 runs.

body lengths during the simulation time (0.4 sec.). Considering that the lattice space used for this experiment was of size 10^3. The produced low-resolution soft robots cannot be compared with real-life organisms. However, the results are shown that even in such low dimensions life-like locomotion can be evolved.

4.1 Evolved Morphologies

Apart from the performance that the two methods achieved, both of them were successful in evolving effective strategies for the locomotion of the evolved morphologies. Figure 6 shows three different gait types evolved by fitness based search. All of these morphologies are considered "good" in respect to their fitness value, meaning that they achieve to travel up to ~ 10 body lengths during the simulation time (0.4 sec.). Considering that the lattice space used for this experiment was of size 10^3. The produced low-resolution soft robots cannot be compared with real-life organisms. However, the results are shown that even in such low dimensions life-like locomotion can be evolved. For the fitness based search, soft body morphologies can use their front leg(s) to pull themselves forward, evolve a four-leg locomotion where a nose and a tail are mostly used for stability, push and pull themselves forward, and gallop using both of their legs. Moving from fitness based search to novelty search locomotion strategies do not differ too much since the resolution does not allow the virtual soft robots to explore more locomotion techniques. However, novelty search proves its merits with regards to the morphologies evolved. More complicated structures are now evolved, this can be explained by the fact that novelty search pushes the evolution to investigate new kinds of behaviors resulting to more complex topologies for the networks (CPPNs) that represent the soft robot morphologies. Figure 7 presents four champion morphologies and their locomotion strategies. Two-legged galloping soft robots, animal-like locomotion based on four legs, and hopper soft robots are evolved.

4.2 Performance Comparison

To compare the performance achieved by novelty search

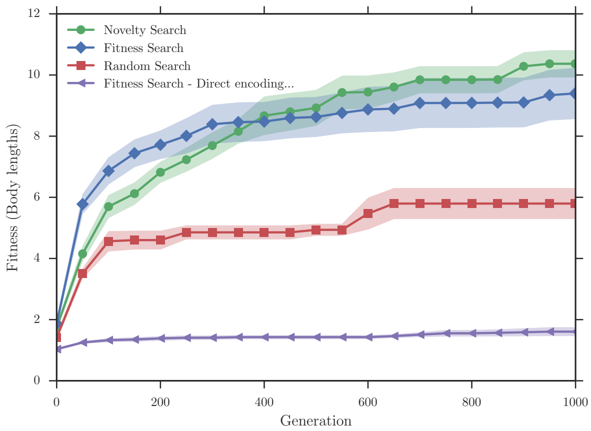

Figure 9: Comparison of simple genetic algorithm (direct encoding) against novelty-fitness-random search with generative encoding. Best fitness so far averaged over 10 runs.

method, its performance is set side by side with fitness based search, random search, and finally a simple genetic algorithm[2] with direct encoded genomes. The same experiment held under two different simulation settings (for resolutions 5^3 and 10^3). Notice, that the first three methods are referring to a generative encoding (CPPNs) evolved by CPPN-NEAT evolutionary algorithm and using selection in respect to novelty, fitness, and random selection. The last method uses a direct encoded genome driven by fitness within a simple genetic algorithm. Two dimensional trajectories, as described earlier, are used by novelty search in order to describe the novelty in the behavior space through sparsity equation. The objective function that describes the goodness of solutions is the displacement of the soft robot's center of mass from its initial position in body-lengths and it is used for all fitness based methods. Random selection in CPPN-NEAT achieved by choosing random selected individuals to breed on each generation.

Figure 8 presents the results for the low resolution soft robots (5^3). The average best displacement so far of the soft robots in body lengths is presented alongside the deviation error. Notice, the difference between novelty search and the other methods. Novelty search evolves structures that are superior than any other method does in these settings. Using the two-dimensional trajectories of the soft robots, novelty search visits optimal solutions that none of the other methods does. Local optima can prevent fitness based search to achieve the performance of novelty search. Encoding limitations in direct encoding cannot lead to optimal solutions for this settings. In the case of random search, having neither the information about their fitness, nor the driving force of novelty search that seeks for novel behaviors, it fails to evolve any decent locomotion. The only reason random search in CPPN-NEAT achieves to evolve displacement of ~ 5 body-lengths, is the powerful encoding used (CPPNs). The simple genetic algorithm approach which uses a direct encoded chromosome to represent the structure of the soft robots performs better than using random selection with an indirect encoding. Structural symmetry and regularity do not show all of their advantages in such low resolution settings.

Moving to higher resolution lattices, it is expected that generative encoding will prove its merits over the direct encoding scheme [1, 20]. More complicated morphologies can be produced (morphology space for 10^3 lattice resolution: 9.3×10^{698}). Furthermore, the space of behaviors, for instance two-dimensional trajectories, becomes larger since bigger and more detailed soft robots can achieve higher displacement and more complex gaits. As it has been shown before, these higher resolution morphologies can achieve life-like locomotion. Figure 9 illustrates the performance (i.e best displacement so far) of the four different methods in these higher resolution settings. Results reassure that novelty search achieves higher fitness ($>$ 1-bodylength) on average against fitness based search. Nevertheless, there is no tremendous difference as in the previous experiment. Both methods achieve to evolve the soft robot structure with the highest fitness found in all experiments ($\sim$ 14 Body lengths). Novelty search behaves more constant in evolving individuals with high fitness in all runs, on the other hand most of individual runs of fitness search are being trapped in low fitness local optima, trying to optimize specific individuals without trying to explore deeply the fitness landscape like novelty search successfully does. Random selection within CPPN-NEAT evolution produced low fitness morphologies for soft robots. The high difference between random selection evolution and novelty search proves that seeking novel behaviors in novelty search cannot be considered as a random search. The superiority of generative encoding (CPPN) over direct encoding can be evidently observed. Regular in shape morphologies can take advantage of their geometrical properties to locomote efficiently.

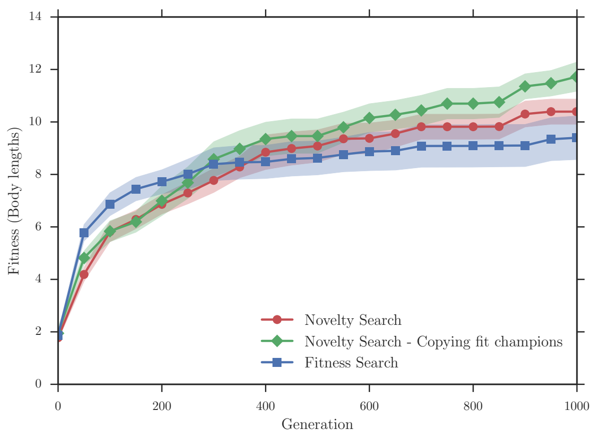

Figure 10: Best fitness so far, novelty search with and without copying *fit* champions (Fitness Elitism), and fitness search, averaged over 10 runs.

4.3 Fitness Elitism in Novelty Search

The reason that novelty search is considered such a revolutionary search method is because it finds solutions for deceptive problems, where the fitness landscape is not a straightforward function. What makes it so unique is the

[2]The GAlib C++ library [24] used for the implementation of this method. Source code used from [1].

fact that instead of looking for optimizing the solutions in respect to an objective function is looking for the novelty in the behavior space. On each generation of novelty search novel behaviors that are also fit in regards to the objective of the problem are discovered. Mutations of these solutions will yield in behaving similarly to their ancestors, resulting in similar behaviors. Thus, the novelty value of these individuals will be declined as similar behaviors will contribute in a denser area in the behavior space. Eventually these solutions will stop being selected, and evolution will not have the chance of carrying their valuable genes along. Mutations and other genetic operations can optimize these fit individuals more. These individuals (with high fitness value) can be seen as *stepping stones* [11] towards more optimized versions of them. Being blind to the objective function, novelty search will eventually stop producing new individuals out of them, which will lead to promising individuals being unable to survive through the evolution process.

Novelty search can include elitism in its selection process, and it does that by copying the most novel organisms of the current population of each species to the next. Since there is no point of changing this function, elitism can be used also to copy fit individuals within novelty search method. The way these two elitism functions can be combined together depends on the population size and the problem, while probabilistic methods can also be used. In the specific setting, both elitism function copy new individuals to the new generation with probability one. Moreover, evolution towards novelty does not get disturbed, at the same time fit individuals have the chance to be optimized further as long as they are the fittest within the species population. Figure 10 illustrates the gain in performance when fitness elitism is used in novelty search method compared with the pure novelty and fitness based search methods.

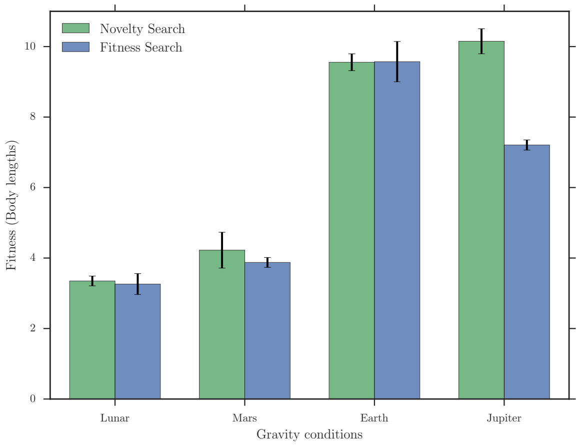

Figure 11: Novelty search performs better or equally good than fitness based search in all gravity conditions tested.

5. DISCUSSION

what's the use of this? getting inspiration for soft robotic probe / landers (tumbleweed) come back to asteroid scenario (passime motion), would it have saved Philae?

6. CONCLUSIONS

our setup better than results from "unshackling evoltion" methodology is suitable to design diverse gaits of soft robots at various gravity levels future work: ensamble of behaviors, very low gravity environments and rotational parameters of small body linked to actuation frequency

7. REFERENCES

[1] N. Cheney, R. MacCurdy, J. Clune, and H. Lipson. Unshackling evolution: evolving soft robots with multiple materials and a powerful generative encoding. In *Proceeding of the fifteenth annual conference on Genetic and evolutionary computation conference*, pages 167–174. ACM, 2013.

[2] P. Faloutsos, M. Van De Panne, and D. Terzopoulos. Dynamic free-form deformations for animation synthesis. *Visualization and Computer Graphics, IEEE Transactions on*, 3(3):201–214, 1997.

[3] J. Hiller and H. Lipson. Automatic design and manufacture of soft robots. *Robotics, IEEE Transactions on*, 28(2):457–466, 2012.

[4] J. Hiller and H. Lipson. Dynamic simulation of soft heterogeneous objects. *arXiv preprint arXiv:1212.2845*, 2012.

[5] J. D. Hiller and H. Lipson. Multi material topological optimization of structures and mechanisms. In *Proceedings of the 11th Annual conference on Genetic and evolutionary computation*, pages 1521–1528. ACM, 2009.

[6] J. D. Hiller and H. Lipson. Evolving amorphous robots. In *ALIFE*, pages 717–724, 2010.

[7] F. Ilievski, A. D. Mazzeo, R. F. Shepherd, X. Chen, and G. M. Whitesides. Soft robotics for chemists. *Angewandte Chemie*, 123(8):1930–1935, 2011.

[8] C. Laschi, M. Cianchetti, B. Mazzolai, L. Margheri, M. Follador, and P. Dario. Soft robot arm inspired by the octopus. *Advanced Robotics*, 26(7):709–727, 2012.

[9] J. Lehman and K. O. Stanley. Exploiting open-endedness to solve problems through the search for novelty. In *ALIFE*, pages 329–336, 2008.

[10] J. Lehman and K. O. Stanley. Revising the evolutionary computation abstraction: minimal criteria novelty search. In *Proceedings of the 12th annual conference on Genetic and evolutionary computation*, pages 103–110. ACM, 2010.

[11] J. Lehman and K. O. Stanley. Abandoning objectives: Evolution through the search for novelty alone. *Evolutionary computation*, 19(2):189–223, 2011.

[12] J. Lehman and K. O. Stanley. Evolving a diversity of virtual creatures through novelty search and local competition. In *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, pages 211–218. ACM, 2011.

[13] M. Müller, J. Dorsey, L. McMillan, R. Jagnow, and B. Cutler. Stable real-time deformations. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 49–54. ACM, 2002.

[14] R. Pfeifer, M. Lungarella, and F. Iida. The challenges ahead for bio-inspired'soft'robotics. *Communications of the ACM*, 55(11):76–87, 2012.

[15] J. Rieffel, D. Knox, S. Smith, and B. Trimmer. Growing and evolving soft robots. *Artificial life*, 20(1):143–162, 2014.

[16] S. Risi, S. D. Vanderbleek, C. E. Hughes, and K. O. Stanley. How novelty search escapes the deceptive trap of learning to learn. In *Proceedings of the 11th Annual conference on Genetic and evolutionary computation*, pages 153–160. ACM, 2009.

[17] S. Seok, C. D. Onal, R. Wood, D. Rus, and S. Kim. Peristaltic locomotion with antagonistic actuators in soft robotics. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 1228–1233. IEEE, 2010.

[18] R. F. Shepherd, F. Ilievski, W. Choi, S. A. Morin, A. A. Stokes, A. D. Mazzeo, X. Chen, M. Wang, and G. M. Whitesides. Multigait soft robot. *Proceedings of the National Academy of Sciences*, 108(51):20400–20403, 2011.

[19] K. Stanley and R. Miikkulainen. Evolving neural networks through augmenting topologies. *Evolutionary computation*, 10(2):99–127, 2002.

[20] K. O. Stanley. Compositional pattern producing networks: A novel abstraction of development. *Genetic programming and evolvable machines*, 8(2):131–162, 2007.

[21] M. Teschner, B. Heidelberger, M. Muller, and M. Gross. A versatile and robust model for geometrically complex deformable solids. In *Computer Graphics International, 2004. Proceedings*, pages 312–319. IEEE, 2004.

[22] M. T. Tolley, R. F. Shepherd, B. Mosadegh, K. C. Galloway, M. Wehner, M. Karpelson, R. J. Wood, and G. M. Whitesides. A resilient, untethered soft robot. *Soft Robotics*.

[23] D. Trivedi, C. D. Rahn, W. M. Kier, and I. D. Walker. Soft robotics: Biological inspiration, state of the art, and future research. *Applied Bionics and Biomechanics*, 5(3):99–117, 2008.

[24] M. Wall. Galib: A c++ library of genetic algorithm components. *Mechanical Engineering Department, Massachusetts Institute of Technology*, 87:54, 1996.