Crowdseeding: a novel approach for designing bioinspired machines

Mark D. Wagy and Josh C. Bongard

University of Vermont
Computer Science Department,
33 Colchester Ave., Burlington, VT, USA

Abstract. Crowdsourcing is a popular technique for distributing tasks to a group of anonymous workers over the web. Similarly, crowdseeding is any mechanism that extracts knowledge from the crowd, and then uses that knowledge to guide an automated process. Here we demonstrate a method that automatically distills features from a set of robot body plans designed by the crowd, and then uses those features to guide the automated design of robot body plans and controllers. This approach outperforms past work in which one feature was detected and distilled manually. This provides evidence that the crowd collectively possesses intuitions about the biomechanical advantages of certain body plans; we hypothesize that these intuitions derive from their experiences with biological organisms.

Keywords: Evolutionary Robotics, Biomimetics, Crowdsourcing, Data Mining

1 Introduction

Embodied cognition [17] is the view that the intelligent behavior of an animal or human is influenced not just by its nervous system but also by its body plan. Engineers that produce bio-inspired designs are (implicitly or explicitly) adhering to this view. Wings on an airplane strongly suggest the influence of the morphology of birds. Many robots that have been developed are either humanoid in form [19, 8, 10, 16] or resemble other animals, such as the canine *Bigdog* [18], the serpentine *OT-4* [3] or the chelonian *Aqua* [6]. Some biomimetic designs result from an explicit aim to exploit some desirable property of the behavior or feature of animals or of their environment. But in some cases the tendency to bias search toward specific design spaces could be considered an implicit tendency of collective human design behavior.

In [26], web participants collectively designed robot bodies. It was found that, among the successful designs, there was an overrepresentation of symmetric designs. This suggests that contributors have a strong proclivity for locomoting agents that resemble animals found in the physical world. Symmetry and single-component designs were the most explicit biases in robot bodies created by the crowd. However, there could be other, less obvious, traits and relations between them that could be exploited. In this study, we describe a novel methodology for extracting latent information from a group of participants in a crowdsourced experiment. Using the design of robot bodies and control as our domain, we use symbolic regression to discover implicit relations between properties of robot designs and an objective, in our case rapid forward locomotion. We then use these latent relationships to seed a new robot design process. This methodology represents a new mode of collaborative interaction between a crowd of human designers and a machine learning algorithm.

2 Related Work

There has been of late a great deal of interest in finding methods to utilize the collective intelligence of crowds to solve complex problems [21, 7, 9, 11]. The field of crowdsourcing has moved from being a convenient way to source simple, separable human intelligence tasks [14] that cannot yet be completed by machine intelligence [24] to being used to combine the efforts of individuals to solve larger problems that might not be amenable to reduction into a divide-and-conquer strategy [15, 29, 1, 23].

Use of human participants in evolutionary algorithms has been common for both selection of individuals in a population [5, 22] and in introducing variation in the evolutionary population [12]. But each of these examples involved direct user participation in the evolutionary search. Crowdsourcing in evolutionary robotics has been used to guide search for better robots and robot control [4, 2, 25]; but in these studies, the use of human intuition was used to actively guide search during the experiment rather than to distill out useful features in a crowdsourced study that were then incorporated into a separate search algorithm. In [26], features were extracted to seed the fitness objective of an evolutionary algorithm. But instead of using automated methods to find latent variables and their relations that contribute to performance – as is the case in this study – obvious characteristics of robot bodies favored by the crowd were distilled manually to seed the objective function.

3 Methods

We conducted an experiment in three stages. In the first stage of the experiment (Section 3.1), we deployed a web-based tool in which participants in a crowdsourced study designed robots collaboratively. In the second stage (Section 3.2), we used symbolic regression via genetic programming [13] to identify a novel relationship between the attributes of the crowdsourced robot designs and the distance that robots were able to move. In the third stage (Section 3.3), we used the relationship found through symbolic regression to augment a stochastic search process from a single-objective search problem to that of a multiobjective search.

3.1 First Stage: Crowdsourcing

We deployed a web-based tool that allowed participants to rapidly design robots using a simple grid-based drawing panel (Figure 1). We invited participants to design robots by recruiting them through the online forum *Reddit* (*www.reddit.com*). Participation was unpaid and voluntary. Participants were only given the instructions to design a robot that could "move farther". They were asked to connect dots to design a robot and click *GO*. They were told that their robot will learn new behaviors if they run the same robot multiple times.

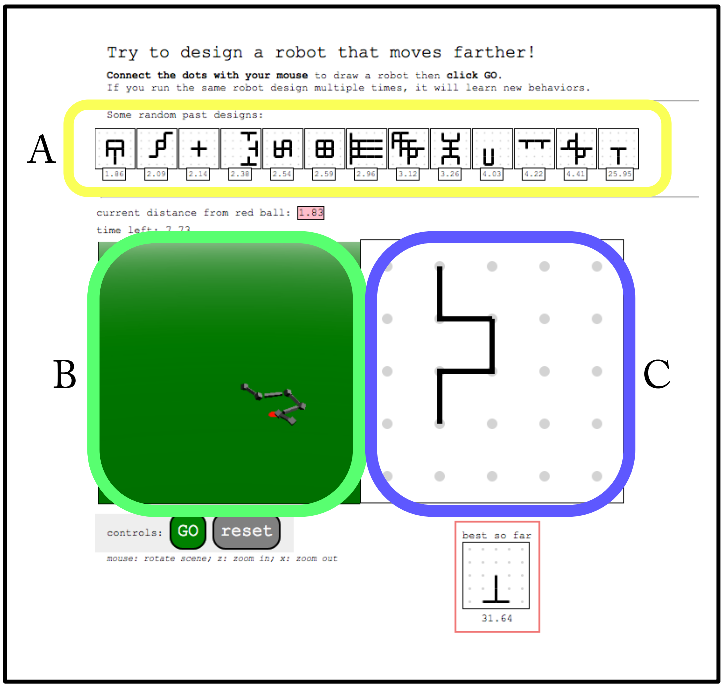

Fig. 1: Screenshot of web-based robot design tool. Users designed robot bodies by "connecting the dots" (Panel **C**). When they clicked "GO", they would see their robot move in a simulation (Panel **B**). They were shown a random sampling of 13 robots designed by others (Panel **A**).

Users connected dots in the design panel by clicking on a dot and dragging their mouse to another dot, which would form a line. Only lines between adjacent dots were allowed. When they clicked *GO*, each line was translated into a $0.1 \times 0.1 \times 0.1$ meter rigid segment in the simulation panel and each dot that was adjacent to a line was translated into a $0.2 \times 0.2 \times 0.2$ rigid cube. Cubes were connected to segments by a one-degree-of-freedom hinge joint. Robots were simulated in the three-dimensional physics simulation engine, Ammo.js[1], and were rendered using Web3D[2] in the participant's web browser.

Each of the robot's joints was assigned to move either in-phase ($0°$) or out-of-phase ($180°$) with other joints. When a particular robot design was run for the first time, it was assigned its own hill-climber algorithm on a central repository, which would determine whether each of its joints would be in-phase or out-of-phase. If the same or another user repeated that design for a run in the simulation, the joint configuration would either be repeated or a joint could be randomly mutated from in-phase to out-of-phase or vice-versa at a 0.1 mutation rate. Thus every time a participant clicked *GO*, it was contributing one run to the hillclimber for that particular robot morphology. All joints were actuated with displacement-controlled motors via a sinusoidal signal with a frequency of 1.5 Hz, and sweeping an angle of $[-45°, +45°]$. The axis of rotation was defined to be perpindicular to the normal of the ground plane and the normal of the faces of the cube and segment being connected. Each time a robot was simulated, it was allowed to run for 15 seconds of simulation time. The distance that the robot moved from its starting point was displayed in the browser above the simulation.

Participants were exposed to designs created by other participants at the top of their browser windows and were shown the best distance that that particular morphology was able to achieve. They were free to ignore these designs or use them as guidance in their own designs. Every time a user clicked *GO* or refreshed the web site, they would be exposed to another random sampling of 13 designs stored in the central repository of designs.

3.2 Second Stage: Mining Latent Features

The methodology of designing robots by connecting dots with lines was conducive to storing representations of these designs as a set of edges and nodes in a graph adjacency matrix. We stored the designs of all unique robots created by the crowd in the first stage of the experiment. Then for each of these robot design, we calculated network measures using its graph representation (for a list of network measures used, see Table 1a).

We then used these calculated network measures on the crowdsourced robot morphologies as explanatory variables in a symbolic regression model. The best distance that that morphology was able to move was the response variable used to train the model. We used the *Eureqa* [20] symbolic regression package to build the models. The set of functions allowed in the model are listed in Table 1b.

Eureqa is a multobjective search tool: it maintains non-dominated solutions along a Pareto front with respect to either minimum error or minimal solution complexity. We ran ten trials of symbolic regression until they reached 100% convergence and at least 80% maturity to find expressions that related distance to the set of explanatory network measures. We then selected the best solution of the ten trials with the minimum fit error value (and thus maximal complexity) for use in the third stage of the study.

[1] www.github.com/kripken/ammo.js
[2] www.khronos.org/webgl

Variable	Symbol
Maximal matching	M_{max}
Number of connected components	c
Maximum degree	D_{max}
Minimum degree	D_{min}
Number of limbs	L
Not a chordal graph	C
Symmetry	S
Number of segments	G
Average Degree	D_{ave}
Average degree connectivity	D_{con}
Average clustering	T_{ave}

(a) Variables

Function Name	Symbol
Addition	$+$
Subtraction	$-$
Multiplication	$\cdot$
Division	$/$
Logistic function	$\sigma()$
Indicator function	$I()$
Cosine	$cos()$
Sine	$sin()$
Tangent	$tan()$
Exponential	$exp()$
Natural Logarithm	$log()$
Power	x^y
Square root	$\sqrt{}$
Gaussian	$G()$
Less than or equal	$\leq$
Greater than or equal	$\geq$

(b) Functions

Table 1: (a) Variables used as explanatory variables and (b) functions allowed for use in symbolic regression expressions.

3.3 Third Stage: Seeding the Objective

In the third stage of our study, we used the best expression found using symbolic regression as an additional objective in a genetic algorithm to evolve new robots and their controllers. The genomes in this genetic algorithm consisted of bitstrings grouped into sets of three. Each of these groups corresponded to a potential lines position between adjacent dots in the same 5×5 grid that users in the crowdsourced portion of the study were given to design robots. Within each group of three bits, the center bit determined whether there was a segment present at that location. If a segment was present (a 1 in the center bit location) the bits to the left and right of the center bit determined whether the motor at the left/top or right/bottom hinge joint would move in-phase or out-of-phase with other motors depending on whether the line was vertical or horizontal (as illustrated in Figure 2).

We compared the best distances achieved by two different, *seeded* evolutionary algorithms: both had the primary objective of maximizing the distance that a robot was able to move in a physics simulation, but they were given additional objectives derived from the preferences of the crowdsourced portion of the study. The two treatments differed in this second, seeded, objective. In the first (control) treatment, the secondary objective was to maximize the ratio of symmetry to number of connected components that made up the robot body. This seeded objective was shown [26] to outperform an evolutionary algorithm with the single objective of maximizing distance that a robot could move. The second

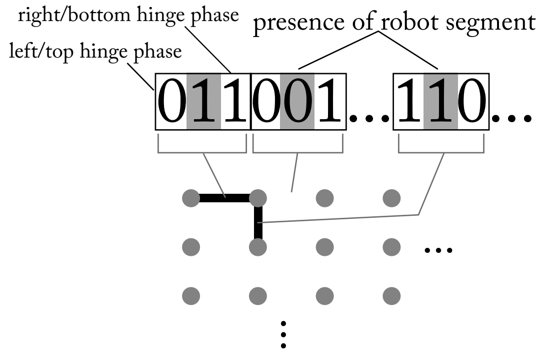

Fig. 2: Genotype to phenotype translation.

(experimental) treatment consisted of using the expression found by symbolic regression described in Section 3.2 to minimize the error between the expression and distance that a particular design with those parameters could achieve.

We performed 100 independent trials for each of the control and experimental treatments. Each evolutionary process ran for 100 generations with a population of 50 individuals. We used bit-flip mutation at a rate of 0.1 as well as uniform crossover at a rate of 0.1.

4 Results

In the first, crowdsourced, stage of the experiment, a total of 947 volunteers participated in robot design. They created 2292 unique designs. On average, 5.63 designs were created by each participant with a long-tailed distribution pattern of participation as is common in crowdsourced studies [27, 28]. Drawings of the best five designs can be seen in Figure 3.

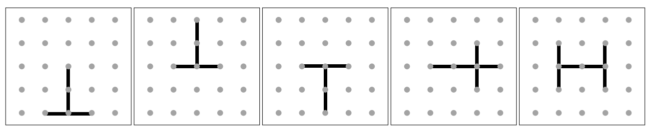

Fig. 3: Top five unique designs in stage one (left to right, top to bottom). Note that some designs that are considered unique are morphologically similar but at different grid coordinates.

Examples of the ten best symbolic regression expressions found in the second stage of the experiment can be seen in Table 2. The expression with the minimum error value (marked with *) was used in the obtaining distance values for the experimental treatment reported in Figure 4.

	Solution	Error
1*	$I(D_{max} \leq L) \cdot min(G(D_{conn}), S) + 0.177min(c + 6.800D_{conn}, D_{max}^{4.510-max(S,log(L))})$	0.771
2	$0.0.041 \cdot c \cdot S \cdot I(L \geq 0.044 \cdot c \cdot D_{max}) + min(c^{0.350}, D_{max})$	0.773
3	$min(D_{max} \cdot \sigma(M_{max}), 2.187) - 0.085 \cdot M_{max}$	0.804
4	$min(3.150, D_{max} + 0.170 \cdot M_{max}) - 0.144 \cdot M_{max}$	0.806
5	$\sqrt{D_{max}} - (0.003 \cdot N \cdot c^2)^S \cdot cos(D_{max})$	0.786
6	$min(3.054, D_{max} \cdot min(M_{max}, 1.325)) - 0.125 \cdot M_{max}$	0.807
7	$4.374 \cdot c \cdot min(0.033, D_{con}) + I(D_{max} \geq 3) + min(S, I(D_{max} \leq L))$	0.781
8	$0.053 \cdot c \cdot S \cdot I(L \geq 0.044 \cdot c \cdot D_{max}) + min(\log c + S, D_{max})$	0.773
9	$D_{con} + S \cdot I(L \geq D_{max}) + 0.196 \cdot min(3.471^{D_{max}}, c) - I(L \geq G)$	0.771
10	$2.819 \cdot \sigma(S) \cdot min(3.180, D_{max}) - 0.141 \cdot M_{max} \cdot min(M_{max}, D_{max}) - 2.960 \cdot I(3.887 - L \geq M_{max})$	0.779

Table 2: Lowest error solutions found in the ten symbolic regression trials. See Table 1 for variable definitions and the list of functions used in the expressions. Constants are rounded to the nearest thousandth for brevity. Expression 1 was used as the second objective in the experimental treatment.

The best distances achieved in the 100 independent runs of the control and experimental treatments in the third stage of the experiment are compared in Figure 4. The best five designs in this third stage of the study are illustrated in Figure 5.

5 Discussion

The introduction of an additional objective using a symbolic regression expression significantly outperformed the inclusion of manually-derived additional objectives to the fitness in the genetic algorithm as reported in [26] ($p < 0.0001$; independent two group t-test), which was itself shown to significantly outperform using the primary objective alone to guide search. We are comparing performance of these methods on the basis of the primary, distance, objective alone. Since adding additional objectives decreases selection pressure on the primary objective, this finding is surprising and encouraging for the method that we introduce here.

The expressions obtained using symbolic regression are quite complex. We took the expressions from the Pareto Front that had the lowest error of all solutions on the front, which necessarily means that these solutions would be the most complex expressions. Despite their complexity, we have seen empirically that they were able to outperform more simple multiobjective fitness values. This suggests that there is some valuable latent information in the symbolic regression expressions distilled from the crowd, despite the difficulty of parsing out exactly what the expressions entail at an intuitive level.

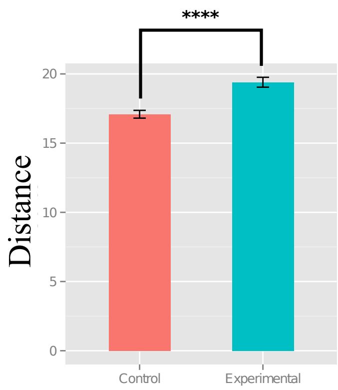

Fig. 4: Distance comparison between treatments. The control case (pink) shows the distance achieved by using the primary distance objective as well as a combined symmetry/number of components objective. The experimental case (blue) shows the distance achieved by using the primary distance objective as well as the expression obtained using symbolic regression. Error bars indicate the 95% confidence intervals around the mean distance value. The difference is significant at the $p < 0.0001$ level (independent two group t-test).

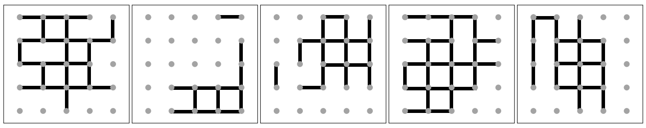

Fig. 5: Top five unique designs in stage three (left to right, top to bottom).

6 Conclusion and Future Work

In this work, we demonstrated a new process for automating the distillation of a crowd's design preferences into an additional objective used to seed an evolutionary algorithm. This objective was in addition to the primary objective of maximizing the distance that a robot was able to move from an initial fixed point in a simulation. We showed that this process significantly outperformed a manual method for extraction of information from crowdsourced studies, which itself was shown to outperform the use of the primary objective alone in the evolutionary algorithm.

In future work, we will investigate whether this technique can be incorporated into other domains. In an era of large amounts of data – much of it generated by humans – there is potential for mining features and expressions from that data that can then be used to improve training of machine learning algorithms, in the development of better design requirements for engineering projects or in seeding algorithms for creative algorithmic work.

Additionally, we will investigate methods to obtain intuitive information from the symbolic regression expressions themselves. In the present study, expressions were directly copied into a multiobjective search. However, there may be kernels of information in the expressions that could be distilled out to further to reduce complexity and drill down to the terms within the expression that are the basis for this method's ability to outperform manually-derived objectives.

Acknowledgments

This work was supported by the National Science Foundation (NSF) under project DGE-1144388. This work was also supported by the NSF under grant PECASE-0953837, and by the Defense Advanced Research Projects Agency (DARPA) under grants W911NF-11-1-0076 and FA8650-11-1-7155.

References

1. Vamshi Ambati, Stephan Vogel, and Jaime G Carbonell. Active learning and crowd-sourcing for machine translation. In *LREC*, volume 1, page 2. Citeseer, 2010.
2. Josh C Bongard and Gregory S Hornby. Combining fitness-based search and user modeling in evolutionary robotics. In *Proceeding of the fifteenth annual conference on Genetic and evolutionary computation conference*, pages 159–166. ACM, 2013.
3. Johann Borenstein, Malik Hansen, and Adam Borrell. The omnitread ot-4 serpentine robotdesign and performance. *Journal of Field Robotics*, 24(7):601–621, 2007.
4. Shane Celis, Gregory S Hornby, and Josh Bongard. Avoiding local optima with user demonstrations and low-level control. In *Evolutionary Computation (CEC), 2013 IEEE Congress on*, pages 3403–3410. IEEE, 2013.
5. Richard Dawkins. *The blind watchmaker: Why the evidence of evolution reveals a universe without design.* WW Norton & Company, 1996.
6. Gregory Dudek, Michael Jenkin, Chris Prahacs, Andrew Hogue, Junaed Sattar, Philippe Giguere, Andrew German, Hui Liu, Shane Saunderson, Arlene Ripsman, et al. A visually guided swimming robot. In *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 3604–3609. IEEE, 2005.
7. Huiji Gao, Geoffrey Barbier, Rebecca Goolsby, and Daniel Zeng. Harnessing the crowdsourcing power of social media for disaster relief. Technical report, DTIC Document, 2011.
8. Kazuo Hirai, Masato Hirose, Yuji Haikawa, and Toru Takenaka. The development of honda humanoid robot. In *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, volume 2, pages 1321–1326. IEEE, 1998.
9. Ece Kamar, Severin Hacker, and Eric Horvitz. Combining human and machine intelligence in large-scale crowdsourcing. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 467–474. International Foundation for Autonomous Agents and Multiagent Systems, 2012.
10. Kenji Kaneko, Kensuke Harada, Fumio Kanehiro, Gou Miyamori, and Kazuhiko Akachi. Humanoid robot hrp-3. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 2471–2478. IEEE, 2008.

11. Firas Khatib, Seth Cooper, Michael D Tyka, Kefan Xu, Ilya Makedon, Zoran Popović, David Baker, and Foldit Players. Algorithm discovery by protein folding game players. *Proceedings of the National Academy of Sciences*, 108(47):18949–18953, 2011.
12. Alex Kosorukoff. Human based genetic algorithm. In *Systems, Man, and Cybernetics, 2001 IEEE International Conference on*, volume 5, pages 3464–3469. IEEE, 2001.
13. John R Koza. *Genetic programming: on the programming of computers by means of natural selection*, volume 1. MIT press, 1992.
14. Edith Law and Luis von Ahn. Human computation. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 5(3):1–121, 2011.
15. Florian Laws, Christian Scheible, and Hinrich Schütze. Active learning with amazon mechanical turk. In *Proceedings of the conference on empirical methods in natural language processing*, pages 1546–1556. Association for Computational Linguistics, 2011.
16. Sebastian Lohmeier, Thomas Buschmann, and Heinz Ulbrich. Humanoid robot lola. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 775–780. IEEE, 2009.
17. Rolf Pfeifer and Josh Bongard. *How the body shapes the way we think: a new view of intelligence*. MIT press, 2006.
18. Marc Raibert, Kevin Blankespoor, Gabriel Nelson, Rob Playter, et al. Bigdog, the rough-terrain quadruped robot. In *Proceedings of the 17th World Congress*, volume 17, pages 10822–10825, 2008.
19. David Sanders and Yoshihiro Kusuda. Toyota's violin-playing robot. *Industrial Robot: An International Journal*, 35(6):504–506, 2008.
20. M Schmidt and H Lipson. Eureqa (version 0.98 beta)[software]. nutonian inc., cambridge, ma, 2014.
21. James Surowiecki, Mark P Silverman, et al. The wisdom of crowds. *American Journal of Physics*, 75(2):190–192, 2007.
22. Hideyuki Takagi. Interactive evolutionary computation: Fusion of the capabilities of ec optimization and human evaluation. *Proceedings of the IEEE*, 89(9):1275–1296, 2001.
23. Luis Von Ahn. Games with a purpose. *Computer*, 39(6):92–94, 2006.
24. Luis Von Ahn. Human computation. In *Design Automation Conference, 2009. DAC'09. 46th ACM/IEEE*, pages 418–419. IEEE, 2009.
25. Mark Wagy and Josh Bongard. Collective design of robot locomotion. In *ALIFE 14: The Fourteenth Conference on the Synthesis and Simulation of Living Systems*, volume 14, pages 138–145.
26. Mark D Wagy and Josh Bongard. Crowdseeding robot design. In *Proceedings of the 17th annual conference on Genetic and evolutionary computation*. ACM, forthcoming.
27. Dennis M Wilkinson. Strong regularities in online peer production. In *Proceedings of the 9th ACM conference on Electronic commerce*, pages 302–309. ACM, 2008.
28. Fang Wu, Dennis M Wilkinson, and Bernardo A Huberman. Feedback loops of attention in peer production. In *Computational Science and Engineering, 2009. CSE'09. International Conference on*, volume 4, pages 409–415. IEEE, 2009.
29. Yan Yan, Glenn M Fung, Rómer Rosales, and Jennifer G Dy. Active learning from crowds. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 1161–1168, 2011.