Simulated Annealing as an Optimization Algorithm in the Automatic Modular Design of Control Software for Robot Swarms*

Jonas Kuckling, Keneth Ubeda Arriaza, and Mauro Birattari

IRIDIA, Université libre de Bruxelles, Belgium
mbiro@ulb.ac.be

The main challenge of swarm robotics is the design of control software in such a way that a desired behavior emerges from the interactions of the swarm [1]. In the automatic off-line design, the design problem is transformed into an optimization problem [2]. The optimization algorithm combines modules into a target architecture creating instances of control software. The instances of control software are optimized with regard to a mission-dependent performance function. Francesca et al. developed two automatic modular design methods: AutoMoDe-`Vanilla` [3] and AutoMoDe-`Chocolate` [2]. `Chocolate` improves over `Vanilla` by using Iterated F-race instead of F-race as the optimization algorithm and is able to achieve significantly better results, indicating that the optimization algorithm has an important influence on the performance of the design method.

In this work we have implemented a modular design method (AutoMoDe-`IcePop`) based on the component based simulated annealing algorithm [4]. `IcePop` is based on `Chocolate` and only differs in the optimization algorithm that is used to generate the control software. It still uses the same set of six behaviors and six conditions as `Chocolate`, and assembles these modules into a finite-state machine with up to four states and four outgoing transitions per state. We have assessed the performance of `IcePop` on two missions: AAC and FORAGING. In AAC the swarm is tasked to aggregate on one black spot in the arena, having an ambient cue in form of a light source to give a general idea on the direction of the target. The mission FORAGING is an abstraction of the classical foraging task, but robots just need to pass over the areas instead of physically picking up items.

We have conducted experiments to investigate the influence of different parameters that could be adjusted in the design method. The first experiment investigates the influence of the budget on the performance of `IcePop`. The results show that with an increasing budget the solution quality improves, although it levels and shows diminishing returns. In a second experiment, we investigated different sample sizes to counter the effects of the stochastic assessment of performance. However neither the smaller sample size nor the larger sample size resulted in any significant changes in performance over the default sample size chosen for `IcePop`. A third experiment was conducted to investigate the effects

* JK and KUA contributed equally to this work and should be considered as co-first authors.

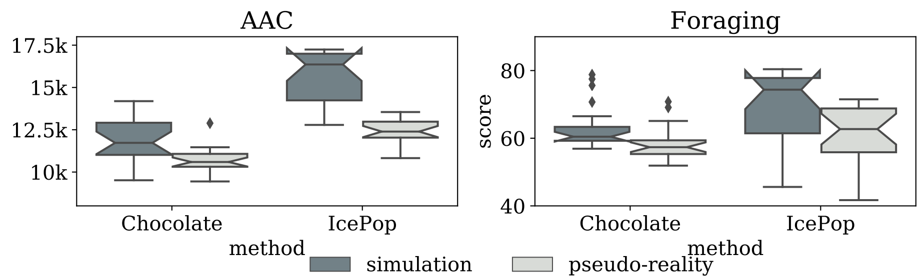

Fig. 1. Comparison for a budget of 100k simulations

of different restarting mechanisms for the simulated annealing algorithm. Here again no variant showed to be consistently better than the other considered variants. In a last experiment we compared the performance of `IcePop` to `Chocolate` over several budgets. Figure 1 shows the performance of both `Chocolate` and `IcePop` in the two missions for a budget of 100 000 simulations. In both missions `IcePop` performs significantly better than `Chocolate` in the design context. When assessed in pseudoreality [5], both methods perform similarly well as in the design context, indicating that the control software generated by `IcePop` possesses a good transferability.

The results show that simulated annealing is a viable optimization algorithm in the automatic modular design. Among the considered alternatives, no clear best was found. Future work will focus on investigating more variants from the large body of literature devoted to simulated annealing and to apply the algorithm to solve more complex missions.

Funding: FRS-FNRS (JK and MB); ERC (Demiurge: grant No 681872).

References

1. Brambilla, M., Ferrante, E., Birattari, M., Dorigo, M.: Swarm robotics: a review from the swarm engineering perspective. Swarm Intell. **7**(1), 1–41 (2013). https://doi.org/10.1007/s11721-012-0075-2
2. Francesca, G., Brambilla, M., Brutschy, A., Garattoni, L., Miletitch, R., Podevijn, G., Reina, A., Soleymani, T., Salvaro, M., Pinciroli, C., Birattari, M.: AutoMoDe-Chocolate: automatic design of control software for robot swarms. Swarm Intell. **9**(2/3), 125–152 (2015). https://doi.org/10.1007/s11721-015-0107-9
3. Francesca, G., Brambilla, M., Brutschy, A., Trianni, V., Birattari, M.: AutoMoDe: a novel approach to the automatic design of control software for robot swarms. Swarm Intell. **8**(2), 89–112 (2014). https://doi.org/10.1007/s11721-014-0092-4
4. Franzin, A., Stützle, T.: Revisiting simulated annealing: A component-based analysis. Computers & Operations Research **104**, 191 – 206 (2019). https://doi.org/https://doi.org/10.1016/j.cor.2018.12.015
5. Ligot, A., Birattari, M.: Simulation-only experiments to mimic the effects of the reality gap in the automatic design of robot swarms. Swarm Intelligence (1-24) (2019). https://doi.org/10.1007/s11721-019-00175-w