System Identification of Self-Organizing Robotic Swarms

Nikolaus Correll and Alcherio Martinoli

Swarm-Intelligent Systems Group, École Polytechnique Fédérale Lausanne
`firstname.lastname@epfl.ch`

We discuss system identification of self-organizing, swarm robotic systems using a "gray-box" approach, based on probabilistic macroscopic models. Using a well known case study concerned with the autonomous inspection of a regular structure by a swarm of miniature robots, we show how to achieve highly accurate predictive models by combining previously developed probabilistic modeling and calibration methods, with parameter optimization based on experimental data (80 experiments involving 5-20 real robots).
Key properties of the optimization process are outlined with the help of a simple scenario and a model that can be solved analytically. Concepts are then validated numerically for the more complex, non-linear inspection scenario.

1 Introduction

Self-organization emerges from the interplay of four ingredients. Positive feedback (e.g., amplification) and negative feedback (e.g., saturation, resource exhaustion), randomness, and multiple interactions between individuals [2]. Although self-organization might achieve less efficient coordination than other distributed control schemes, it can provide extremely high levels of robustness and can be applied to miniature robotic platforms such as those mentioned in this paper.

For designing and formally analyzing self-organized robotic systems, appropriate models are necessary. Modeling allows us to focus on key design parameters, and costly, time-consuming experiments can be reduced to a minimum. As randomness and fully distributed control are at the core of self-organized swarm coordination, we make use of probabilistic macroscopic models to statistically capture the average dynamics and performances of a self-organized robotic swarm.

System modeling and identification [6, 8] is the process of deriving a mathematical model for a metric of interest from observed data. For swarm-robotic systems, probabilistic models have been successfully applied to several case

studies (see for instance [9] or [7, 11]), and lead to good quantitative agreement between reality and the prediction of the model. In this contribution, we take a step forwards and show how the identification process can be complemented by optimization procedures for improving the quantitative agreement between model prediction and experimental data. Such improved models can then serve as a baseline for exploring other aspects of the system without performing additional experiments.

As a case study we consider a homogeneous swarm engaged in the inspection of a regular structure [5], where modeling assumptions (homogeneous distribution of robots and objects, see [9] for more details) are only partially fulfilled. In this case, optimization of model parameters allows us not only to achieve accurate prediction in the model, but also yields valuable insight into improving the structure of the model. Although results are validated on a particular case study, the proposed method is generally applicable for system identification of self-organized robotic systems whose dynamics can be captured by probabilistic models.

2 Probabilistic Modeling of Swarm-Robotic Systems

As more extensively detailed in [9] we abstract the robots' behavior as an arbitrary Probabilistic Finite State Machine (PFSM), whose states are chosen according to the metric of interest. Interactions among the robots or with the environment are represented by state transitions and abstracted to encountering probabilities, whereas the time spent in a certain state is captured by the average interaction time. We hereby assume that the robots and objects are uniformly distributed in the environment, that the system is markovian (i.e. the system's state depends only on its previous state), and that the encountering probabilities scale linearly with the number of objects (i.e. the chance to encounter an object is ten times as high when there are ten objects than when there is only one), which is reasonable when the ratio of free-space to space occupied by robots and objects is large.

Calibration of Model Parameters: Following [4, 9], we calculate the *geometric probability* of encountering an object from the ratio of the object's detection area (the area in that it can be detected by another robot), and the total area of the arena. The (unit less) geometric probability can then be converted into the object's *encountering probability* per time-step, using a simple heuristic based on the area that a robot sweeps with its sensors in this period (based on the characteristics of its sensors and its speed).

The interaction time—if not directly specified within the robot's controller—cannot be calibrated as easily, but needs to be measured using systematic experiments with two robots, or one robot and one object [5].

We validated this approach systematically for simple scenarios in [4], and it showed quantitatively correct predictions for various swarm-robotic case

studies [9]. Nevertheless, the calibration methodology reaches its limitations in overcrowded scenarios, and for nonuniform spatial distributions of robots (as in [5], for instance).

3 Identification of a Linear Swarm-Robotic System

For introducing the concepts applied in this paper more formally, we consider a simple case study first. A swarm of robots is moving on a bounded arena, performing collision avoidance with other robots and the boundary walls using a reactive controller [1]. We are now interested in finding a model for the average number of robots in the search and collision avoidance states from experimental data. Finding a model involves three basic steps [8]:

1. Performing a set of experiments measuring the metric of interest for different parameter choices;
2. Deriving a candidate model;
3. Finding optimal parameters that minimize the mismatch between candidate model and experimental data.

Note, that we consider the system in discrete time, as the observed data for system identification is collected by sampling.

An Identification Experiment: For simplicity, we consider a system without time varying inputs, where an experiment is characterized over a time interval $0 \leq k \leq n$ by its state vector $N(k)$ and parameters that are set by the experimenter (e.g., the number of robots N_0). The state variables are measurements of an arbitrary metric of interest, for instance, the average number of robots searching at time k, $N_s(k)$.

A Candidate Model: Following the methodology outlined in section 2, we model the system by a PFSM with three states: search, avoidance of walls, and avoidance of robots. Our approach involves the following assumptions. Every time step, one of N_0 robots can encounter another robot with probability p_r (and any other robot with probability $p_R = p_r(N_0 - 1)$), and a wall with probability p_w. We further assume that a collision can be sufficiently characterized by its mean duration (T_r and T_w). We can then write the following set of difference equations:

$$\underbrace{\begin{pmatrix} N_{ar}(k+1) \\ N_{aw}(k+1) \\ N_s(k+1) \end{pmatrix}}_{N(k)} = \underbrace{\begin{pmatrix} 1 - \frac{1}{T_r} & 0 & p_R \\ 0 & 1 - \frac{1}{T_w} & p_w \\ \frac{1}{T_r} & \frac{1}{T_w} & 1 - p_r - p_w \end{pmatrix}}_{\theta} \begin{pmatrix} N_{ar}(k) \\ N_{aw}(k) \\ N_s(k) \end{pmatrix}, \qquad (1)$$

and the initial conditions

$$\begin{pmatrix} N_{ar}(0) & N_{aw}(0) & N_s(0) \end{pmatrix}^T = \begin{pmatrix} 0 & 0 & N_0 \end{pmatrix}^T \qquad (2)$$

with $N_s(k)$ being the number of robots searching at time k, $N_{ar}(k)$ the number of robots avoiding a robot, $N_{aw}(k)$ the number of robots avoiding a wall, and

N_0 the total number of robots. We can interpret the first row of (1) as follows. The number of robots avoiding another robot is decreased by those that return from a collision ($\frac{1}{T_r}N_{ar}(k)$), and increased by searching robots colliding with another robot ($p_R N_s(k)$). The other rows of (1) can be interpreted in a similar fashion. Equation (1) can be reformulated as

$$\hat{N}(k+1)^T = N(k)^T \theta^T, \tag{3}$$

where $\hat{N}(k+1)$ is the *estimate* based on the measurements of the real system $N(k)$ and the parameters θ.

Analytical Optimization: Provided the state vector measurements $N(k) = (N_{ar}(k) \quad N_{aw}(k) \quad N_s(k))^T$ in the interval $0 \leq k \leq n$, we can now calculate the prediction error of the model estimate $\hat{N}(k)$. Optimal parameters (θ) can then be found using the least-squares method that can be formulated as

$$\min_\theta \frac{1}{n}\sum_{k=1}^{n}(N(k)^T - \hat{N}(k)^T)^2 = \min_\theta \frac{1}{n}\sum_{k=1}^{n}(N(k)^T - N(k-1)^T\theta^T)^2 \tag{4}$$

We denote the matrix θ that minimizes (4) by $\hat{\theta}_n$:

$$\hat{\theta}_n = \arg\min_\theta \frac{1}{n}\sum_{k=1}^{N}(N(k)^T - N(k-1)^T\theta^T)^2 \tag{5}$$

Since (4) is quadratic in θ, we can find the minimum value easily by setting the derivative of (4) to zero:

$$0 = \frac{2}{n}\sum_{k=1}^{n}N(k-1)(N(k)^T - N(k-1)^T\hat{\theta}_n^T), \tag{6}$$

yielding

$$\hat{\theta}_n = \left[\sum_{k=1}^{n}N(k-1)N(k-1)^T\right]^{-1}\sum_{k=1}^{n}N(k-1)N(k)^T, \tag{7}$$

which is straightforward to compute given the availability of the measured state variables $N(k)$.

Initial Parameter Estimation: In the above experiment, measurements for all state variables (N_{ar}, N_{aw}, and N_s) are available. Imagine now that it is not possible to measure N_{ar} and N_{aw} independently from each other (this is reasonable for collisions with robots close to the wall for instance). Then, p_R and p_w cannot be estimated (1), but only $1 - p_R - p_w$ (using numerical methods). As a work-around, additional experimental data need to be gathered by varying other parameters, for instance changing the number of robots. Such a procedure leads to an identification problem with a smaller number of degrees of freedom, but it might not be feasible to conduct it for every single parameter; in particular for systems where the ratio of parameters to the number of observed state variables is high. Then, an initial estimate using the calibration heuristic from [9] is extremely helpful, as shown below.

4 Identification of a Non-Linear Swarm-Robotic System

We consider a case study introduced in [4, 5]. A swarm of miniature robots (*Alice* [3]) is concerned with the inspection of a simplified jet turbine (Figure 1, *right*). Robots are randomly searching through a bounded arena, and inspect objects in the environment (turbine blades) by circumnavigating them. The robots's controllers are endowed with a timeout (T_{max}), that indicates when a blade can be left. After the timeout is expired, the robot leaves the blade at the next tip it encounters (compare the Finite State Machine (FSM) of the individual robot in Figure 1, *middle*).

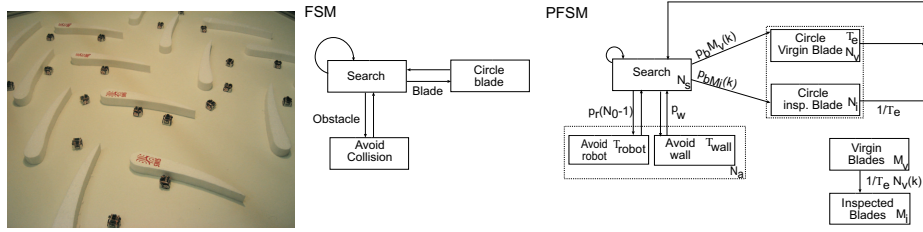We have selected this case study for the following reasons: in [5], our mod-

Fig. 1. *Left*: Experimental setup involving 16 blades and up to 20 miniature robots. *Middle*: Finite State Machine describing individual robot's behavior. *Right*: Probabilistic Finite State Machine describing the state of an individual robot and the environment (number of blades inspected).

eling approach showed certain discrepancies when compared to the dynamics of the real system. We identified two problems. First, the regular structure of the environment imposing drift on the robotic swarm, which in turn favors the inspection of some parts of the arena due to nonuniform distribution of the robots. Second, noisy perception of environmental features led to artifacts such as robots getting stuck at the outer walls or at the tips of individual blades. Using system identification, we show how to get further insight into the system, that the model is indeed able to reproduce the observed dynamics, and that it has the potential to serve as a predictor for the system.

Identification Experiment: The simplified turbine environment was implemented in a rectangular arena of $1.10m \times 1m$ [5]. The robots are endowed with a PIC micro controller (368 bytes RAM, 8Kb FLASH), have a length of 22mm, and a maximal speed of 4cm/s. Four IR modules can serve as crude proximity sensors (up to 3cm) and local communication devices (up to 6 cm). We performed four sets of 20 runs each (80 experiments altogether) for $N_0 = 5, 10, 16$ and 20 robots in an environment with $M_0 = 16$ blades. The inspection progress (the number of inspected blades) is monitored (sampling rate 1Hz) using an overhead camera and the tracking software *SwisTrack* [1], that records the robots's trajectories at a frame rate of 30Hz.

---

[1] http://swistrack.sourceforge.net

Candidate Model: The model for the inspection case study is extensively detailed in [4, 5], and will only briefly be summarized here. Following our modeling methodology [9], we sketch a PFSM of the system dynamics according to our metric of interest (the number of blades inspected), based on an individual robot's FSM (compare Figure 1, middle and right). Initially, a robot is in search mode. At every time-step, a robot may encounter the outer wall, an other robot, or a blade with probabilities p_w, p_R, and p_b, respectively, causing it to enter an avoidance or inspection state. Every state is associated with an interaction time (T_w, T_r, and T_b); the average time the robot spends in this state. In the probabilistic model, this is equivalent to leaving the state with probability $\frac{1}{T_w}$, $\frac{1}{T_r}$, and $\frac{1}{T_b}$.
In our PFSM we also keep track of whether the robot is inspecting a "virgin" blade or a previously inspected blade. With N_0 being the number of robots, and M_0 the number of blades, we can hence derive the following difference equations for the robotic states:

$$\hat{N}_{ar}(k+1) = \hat{N}_{ar}(k) + p_r\hat{N}_s(k)(N_0 - 1) - \frac{1}{T_r}\hat{N}_{ar}(k) \tag{8}$$

$$\hat{N}_{aw}(k+1) = \hat{N}_{aw}(k) + p_w\hat{N}_s(k) - \frac{1}{T_w}\hat{N}_{aw}(k) \tag{9}$$

$$\hat{N}_v(k+1) = \hat{N}_v(k) + p_b\hat{M}_v(k)\hat{N}_s(k) - \frac{1}{T_b}\hat{N}_v(k) \tag{10}$$

$$\hat{N}_i(k+1) = \hat{N}_i(k) + p_b\hat{M}_i(k)\hat{N}_s(k) - \frac{1}{T_b}\hat{N}_i(k) \tag{11}$$

$$\hat{N}_s(k+1) = N_0 - \hat{N}_{ar}(k+1) - \hat{N}_{aw}(k+1) - \hat{N}_v(k+1) - \hat{N}_i(k+1) \tag{12}$$

and for the environmental states:

$$\hat{M}_v(k+1) = \hat{M}_v(k) - \frac{1}{T_b}\hat{N}_v(k) \tag{13}$$

$$\hat{M}_i(k+1) = M_0 - \hat{M}_v(k+1) \tag{14}$$

with initial conditions $N_s(0) = N_0$ and $M_v(0) = M_0$, and all other states zero. Note, that the system described above is non-linear and thus cannot be written in matrix notation as the simple model in section 3.

Initial Calibration of Model Parameters: During the experiments we recorded only the inspection progress itself ($M_i(k)$), whereas the number of unknown parameters is 6 (encountering probabilities and interaction times for blades, robots, and walls). Although the problem of an under-determined system as outlined in section 3 is mitigated by performing experiments with different numbers of robots, there is still an infinite number of possible combinations for encountering probabilities and interaction times. For example, inspection of one blade could be very fast, whereas collisions take very long, or the other way around; both leading to the same measured inspection time. As this is acceptable so long as the system dynamics are reproduced faithfully, the resulting models might perform poorly when used as predictors for other parameters or modified robot controllers.

In order to improve the potential predictive quality of the model after identification, we provide the identification process with an initial guess using the calibration heuristic from [9], which we have good reason to believe [4] yields values that come close to the parameters of the "real" system. Initial guesses for the parameters of the experiment are picked up from [5], and are summarized in Table 1.

Parameter Optimization: We are interested in finding the optimal parameter set θ_N that minimizes the difference between the model's prediction $\hat{M}_v(k, \theta, N_0)$ and measurements of the real system $M_v(k, N_0)$. At the same time, we want to minimize the difference between the optimal parameters and the initial guess $\theta_0 = (p_r \quad p_w \quad p_b \quad T_r \quad T_w \quad T_b)$ (compare Table 1). We thus formulate the following optimization problem

$$\theta_n = \arg\min_{\theta} \left[\left(\sum_{N_0 \epsilon R} \sum_{k=0}^{n} (\hat{M}_v(k, \theta, N_0) - M_v(k, N_0))^2 \right) + (\theta - \theta_0)^2 \right] \qquad (15)$$

With $R = \{5, 10, 16, 20\}$ the number of robots in an experiment (N_0), and n the duration of the longest experiment in the set. Obtaining an analytical solution to (15) is unfeasible. Nevertheless, as it is quadratic in θ, we can attempt to solve it using a convex optimization algorithm (for instance *fmincon* in Matlab^{TM}). Here, the optimization routine integrates (14) numerically until $\hat{M}_v(k) = 0.01$ ($\hat{M}_v(k)$ converges to zero asymptotically) to find an optimal solution for θ.

Table 1. Initial guesses for model parameters (heuristic calibration/measurements of the real system), time discretization of the model T=1s

	Wall	Blade	Robot
Encountering probability	$p_w = 0.045$	$p_b = 0.0106$	$p_r = 0.0015$
Interaction time	$T_w = 10.00\text{s}$	$T_{hb} = 10.00\text{s}$	$T_r = 4.00\text{s}$

5 Results

The numerical optimization routine finds an optimal parameter set for minimizing model prediction error with respect to experimental measurements after 300 to 600 function evaluations (each involving the calculation of model prediction for four different team sizes). We used values from Table 1 as initial guess for θ, and provided upper and lower bounds on θ (all encountering probabilities are forced to be in the interval between 0.00001 and 1, whereas the interaction times are bounded by $8 \leq T_b \leq 12$, $2 \leq T_r \leq 20$, and $2 \leq T_w \leq 30$). Optimal parameters (θ_n) are given in Table 2. Inspection
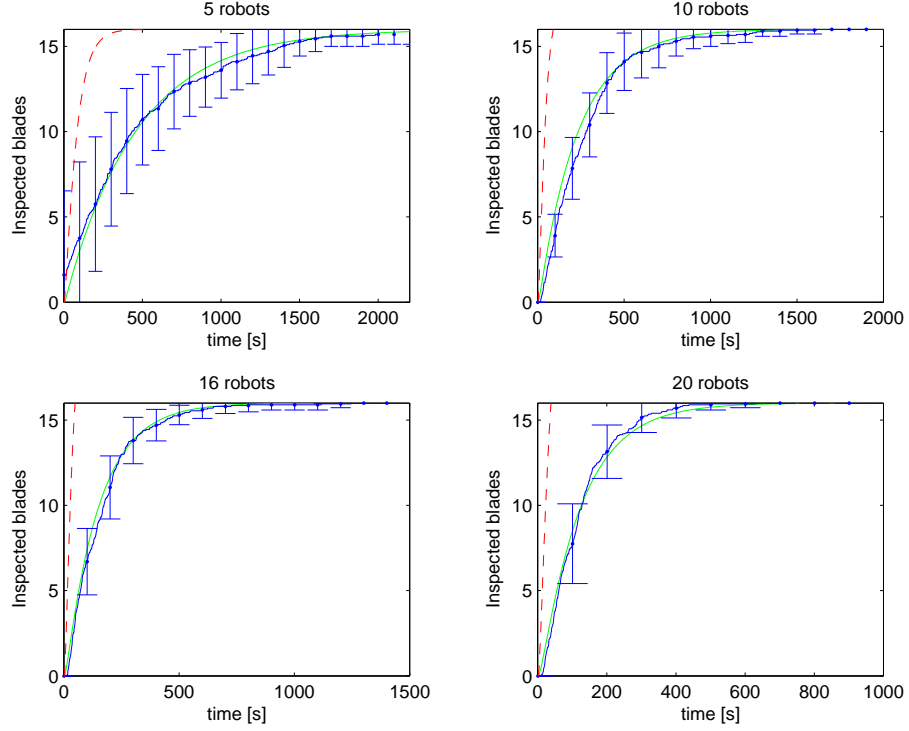
Fig. 2. Average inspection progress for different team sizes and model predictions before (dashed curve) and after optimization (continuous curve) when compared to experimental data (continuous curve with error bars). Error bars show the standard deviation of the measurements.

progress $M_v(k)$ (mean and standard deviation), as well as the model estimates before $(\hat{M}_v(k, N_0, \theta_0))$ and after optimization $(\hat{M}_v(k, N_0, \theta_n)$ for different team sizes $(N_0 = 5, 10, 16, 20)$ are given in Figure 2.

Table 2. Model parameters after optimization, time discretization of the model T=1s

	Wall	Blade	Robot
Encountering probability	$p_w = 0.18899$	$p_b = 0.001543$	$p_r = 0.009363$
Interaction time	$T_w = 11.16556s$	$T_{hb} = 8.0666s$	$T_r = 3.829s$

6 Discussion

Model prediction based on the initial estimate θ_0, predicted the inspection to be roughly twice as fast as reality [5]. Using optimization procedures to complement the system identification process allowed for a good match between prediction and experimental data, for the inspection time metric as well as for the inspection progress. As a consistent pattern over all runs (with different initial conditions and parameter bounds), one observes that the encountering probability for detecting a blade is estimated to be an order of magnitude lower than the resulting probability after calibration. This result matches our expectations from [5] where we observed a nonuniform distribution of the robots over the arena (induced by the geometry of the blades and the robots's controllers), which yields a blade encountering probability that is dependent on the location in the arena.
This case study is an example where optimization averages out more complex effects, such as the drift phenomenon, and allows us to capture them with a relatively simple model. In order to reach this level of accuracy in a different way, the level of detail in our model would need to be increased at the cost of analytical tractability.
As formally shown in section 3, a system where not all of the state variables are measured is likely to be under-determined if the number of degrees of freedom are not varied separately. Indeed, although the predictions for the robot-to-robot encountering probability are consistent over all runs (the system was run with different numbers of robots), it is unclear how the optimization procedure should distribute robots between the wall avoidance state, and the inspection state in order to achieve the desired inspection delay. This is reasonable, as the differences between model prediction and real experiments can not only be explained by the robots' spatial distribution, but also by the fact that the robots may get stuck on a wall or prematurely leave blades. These artifacts are not explicitly captured by the model considered here, and therefore need to be accounted elsewhere in the model's structure. In order to reduce the number of degrees of freedom in the system and increase identifiability [6], one could either measure more state variables, or perform other sets of experiments (in our case, possibly an experiment in a larger arena or with more blades).

7 Conclusion

We have shown how our methodology for system identification of swarm robotic systems, consisting of deriving the system dynamics from the controller of the individual robot, and calibration of model parameters using simple heuristics, can be complemented by macroscopic data-fitting with parameter constraints. We have also shown how this additional step can not only lead to quantitatively correct models for systems that do not comply with the assumptions of our calibration procedure, but also give additional insight into

the whole system. We note that the proposed methodology is suitable for a wide range of swarm robotic scenarios, in which dynamics can be captured by probabilistic models.

Concerning our case study swarm robotic inspection, optimization results show that the structure of the model is sufficient to accurately reproduce the system dynamics, and indicates critical points where the model reaches its limitations. Indeed, the probability of encountering a blade, which is an order of magnitude lower than the calibration estimate, suggests further research into capturing the spatio-temporal distribution of the robots, potentially by using diffusion models borrowed from statistical physics [10].

Acknowledgments

References

1. R. Arkin. *Behavior-Based Robotics*. The MIT press, Cambridge, MA, USA, 2000.
2. E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. SFI Studies in the Science of Complexity, Oxford University Press, New York, NY, USA, 1999.
3. G. Caprari and R. Siegwart. Mobile micro-robots ready to use: Alice. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3295–3300, 2005.
4. N. Correll and A. Martinoli. Modeling and optimization of a swarm-intelligent inspection system. In *Proceedings of the 7th Symposium on Distributed Autonomous Robotic System (DARS)*. To appear in Springer Distributed Autonomous Systems VII, 2004.
5. N. Correll and A. Martinoli. Collective inspection of regular structures using a swarm of miniature robots. In *Proceedings of the 9th International Symposium of Experimental Robotics (ISER)*, pages 375–385. Springer Tracts for Advanced Robotics (STAR), Vol. 21., 2006.
6. Rolf Johansson. *System Modeling and Identification*. Prentice Hall, 1993.
7. K. Lerman and A. Galstyan. Mathematical model of foraging in a group of robots: Effect of interference. *Autonomous Robots*, 2(13):127–141, 2002.
8. Lennart Ljung. *System Identification — Theory for the User*. Prentice Hall, 1999.
9. A. Martinoli, K. Easton, and W. Agassounon. Modeling of swarm robotic systems: A case study in collaborative distributed manipulation. *Int. Journal of Robotics Research*, 23(4):415–436, 2004.
10. A. Prorok. Multi-level modeling of swarm robotic inspection. Master's thesis, SWIS-MP1, École Polytechnique Fédérale Lausanne, 2006.
11. K. Sugawara, M. Sano, I. Yoshihara, and K. Abe. Cooperative behavior of interacting robots. *Artificial Life and Robotics*, 2:62–67.