Better than average: analyzing distributions to understand robot behavior in a multi-agent area coverage scenario

David Buckingham, Giordano B. S. Ferreira and Matthias Scheutz

Tufts University, Medford, MA 02155
{david.buckingham, giordano.ferreira, matthias.scheutz}@tufts.edu

Abstract

Building robots, even for performing simple tasks, requires the designer to assess performance using various parameters. However, sometimes the best solution is not the one that performs best on average. Hence, other ways of evaluating performance are necessary. We ran a broad parameter sweep for an agent-based simulation of a robotic area coverage task with very simple agent controllers in four different task environments. Analysis of the results emphasizes the importance of considering the entire distribution across randomized starting conditions, and not just the mean overall performance, when assessing the effectiveness of parameter settings. Our findings indicate the potential for robotic system designers to constrain or specify the qualities of system performance distributions.

Introduction

Due to the complex and often unforeseen interactions between robotic systems and their environments, analytic methods may be insufficient to parameterize optimally the controllers of such systems. Even for extremely simple systems, computer simulations can help to determine the effects of various parameter settings, in particular, performance according to some metric. However, evaluating the performance of the system with some parametrization by measuring its mean performance over multiple simulation runs might give a skewed picture of the performance profile. We argue that when a system is evaluated over many different initial conditions, consideration of the *performance distribution* can yield important insight beyond what is indicated by the performance mean.

In order to study this proposal, we have constructed a simulation of a simple robotics task. By conducting a suite of experiments with this simulation, we attempt to gain insight into the relevance of distribution analysis to assessing robot performance. Our results offer evidence of the importance of considering the entire distribution of results when evaluating a robotic system whose behavior varies across some conditions.

Our simulation is a discrete-time, continuous-space, online, agent-based model of a generic Area Covering Problem (ACP) task. The ACP requires agents to move within some domain area until the entire area has been explored. The agents in our simulation have very simple controllers. An agent moves in a straight line for δ timesteps, changes its heading by a random angle, drawn uniformly from $\{-\theta \dots \theta\}$ radians and then repeats. If the agent collides with a wall or obstacle, it selects a new heading uniformly from $\{0, 2\pi\}$ radians. This continues for all agents until either entire arena has been completely explored or 10,000 timesteps have elapsed. We ran simulations across a broad parameter sweep to analyze the effect of varying δ, θ, and the number of agents deployed, n, in three different arenas.

Related Work

Experimenters in diverse fields have benefited from analyzing system performance distributions instead of just mean performance. For example, Ramsli (1991) analyzed the repeatability of actuator positioning in industrial robots. He found that, despite the common assumption of normality, repeatability usually does not follow a Gaussian distribution.

In the field of *artificial life*, Schneider et al. (2015) compared the distributions of performance metrics between simulated and physical systems to determine how well simulation predicts performance in a physical environment. That model assumes that performance metrics follow normal distributions, and the authors suggest that future work should consider non-normal distributions and outliers.

Thill and Pearce (2007) ran a stochastic optimization algorithm 50 times with a biological model of *C. elegans* gradient navigation. They used the distribution of the scores of the resulting solutions to construct a definition of solution optimality. They then applied the optimization algorithm to recursively selected subspaces of the model parameter space and analyzed the resulting solutions to identify features that are common to optimal solutions from across the parameter space.

Nevertheless, we are unaware of previous work that explicitly cautions against relying on performance mean and motivates the analysis of performance distributions across repeated stochastic simulation runs when evaluation robotic simulations.

Carole Knibbe et al, eds., Proceedings of the ECAL 2017, Lyon, France, 4-8 September 2017,
(Cambridge, MA: The MIT Press, ©2017 Massachusetts Institute of Technology).

Previous work (Zhu and Ting, 2000) developed a general design technique for analytically describing system performance with respect to parameter variations. A design was considered robust when perturbations to parameters did not significantly alter system performance. Assuming known analytical relationships between parameters and performance, this technique was shown to obtain robust systems. However, the relationships between parameters and behavior are often unknown, or cannot be described analytically. This is the case, for example, in stochastic robotic systems. Therefore, our discussion of Future Work will consider how system parameters can be determined according to specifications about the desired distribution of outcomes when the relationships between parameters and behaviors are determined only experimentally.

Area Coverage Problem

How best to cover a two-dimensional area is a computational problem that arises in many practical applications in agriculture (Oksanen and Visala, 2009), industrial painting (Atkar et al., 2005), planetary exploration, floor cleaning (Schmidt and Hofner, 1998), and military uses such as de-mining (Acar et al., 2003). Two surveys provide excellent background on coverage path planning methods. Choset (2000) presented an overview of theory and techniques. More recently, Galceran and Carreras (2013) reviewed the various approaches to area coverage and presented several relevant works.

The ACP is also known as the “Lawn Mowing Problem”, the “Geometric Milling Problem”, the “Covering Tour Problem”, and “Coverage Path Planning”. While the importance of different task and performance aspects vary depending on the application domain, the ACP can generally be defined as follows: Given a two-dimensional spatial environmental E (possibly non-contiguous and possibly containing obstacles and regions not to be covered) and a set of agents $a \in A$ each having a two-dimensional fixed-shape effector, determine paths p_a for each a such that there no is no point in E that is not bound by f_a for some a at some time. The effector f_a could be the body of the agent as in the case of a vacuum cleaner, a component or appendage of the agent as in the case of a harvesting machine, or the range of a sensor. Agents thus have to move through the environment so that their fixed shape effector explores all points in E.

Randomized control algorithms have proved successful in many practical ACP settings, e.g., commercial floor cleaning robots such as Roomba by iRobot, RC3000 by Karcher, and Trilobite by Electrolux (Palacin et al., 2005). However, they can be inefficient, especially for large areas, wasting time and energy resources on redundant exploration. It has been argued that randomized approaches may not scale effectively to vast areas (Galceran and Carreras, 2013). Nevertheless, randomized approaches that do not rely on sensors for localization or detection and require only minimal storage and computational capacities may admit robots that are much simpler than those used in other approaches. These robots may be cheaper, smaller, lighter, less fragile, or more disposable, and they do not suffer the dead-reckoning error associated with the simultaneous localization and mapping (SLAM) problem. Thus, our simulation uses simple randomized controllers because the advantages of such approaches may outweigh the disadvantages of decreased speed or efficiency.

Methods

In this section we first describe how we set up our simulation and specify the experiments we conducted. Then we explain how we evaluated performance and how we compared performance distributions to test our hypothesis that performance distributions can distinguish experimental conditions even when performance means cannot.

Our simulation is intentionally simple: robots move with constant speed, there is no inter-agent collision detection, etc. The simulation is not intended as a realistic simulation of a physical robotic system, but as a computer implementation of an abstract multi-agent area coverage problem. Given bounded computational resources and time, we have limited the computational complexity of the simulation in order to conduct a broad sweep of the parameter space with enough repetition to produce reliable performance distributions.

Agents

In our model, agents have square bodies of area S_A and fixed speed v. An agent's effector f_a always covers the same space as its body. The values of S_A and v, constant within each experiment and shared by all agents, are determined according to the total area of the S_E:

$$S_A = S_E/1000$$

$$v = 0.15\sqrt{2S_E}$$

Thus, agent behavior is normalized by arena area. We chose the constant 0.15 to be large enough for feasible simulation speed but small enough to avoid collision detection anomalies.

Agents store the coordinates $\langle x_a, y_a \rangle$ of the center of their body at a given time in relation to the 2-dimensional environment. Agent controllers are parameterized by two values. The running distance, δ, is how many timesteps the agents move in a straight line before setting a new heading. The turning angle range, θ, bounds in radians the headings from which the agent's new heading is drawn. Algorithm 1 specifies the operation of the simulation, including the agent controllers.

Arena Environments

We used three different arenas, shown in Figure 1. The *Simple* arena is an empty square. The other arenas were selected

Carole Knibbe et al, eds., Proceedings of the ECAL 2017, Lyon, France, 4-8 September 2017,
(Cambridge, MA: The MIT Press, ©2017 Massachusetts Institute of Technology).

Algorithm 1 Simulation agent control

```
Simulation(seed, E, θ, δ, n)
A ← {}
for all i ∈ (1, ..., n) do
    a ← new agent
    randomly select ⟨x, y⟩ in E
    x_a ← x
    y_a ← y
    A ← A ∪ {a}
    m_a ← 0
end for
for all t ∈ (1, ..., 100,000) do
    if ratioCovered == 1.0 then
        return ⟨ratioCovered, t⟩
    end if
    for all a ∈ A do
        newx ← x_a + (v · cos h_a)
        newy ← y_a + (v · sin h_a)
        if ⟨newx, newy⟩ is not in E then
            h_a ← uniform(0, 2π)
            m_a ← 0
        else
            x_a ← newx
            y_a ← newy
            m_a ← m_a + 1
            if m_a == δ then
                h_a ← uniform(h_a − (θ/2), h_a + (θ/2))
                m_a ← 0
            end if
        end if
    end for
end for
return ⟨ratioCovered, t⟩
```

from a range of relevant works in order to construct a diverse test suite. We selected the *Walls* (Wagner et al., 2000) arena because we expected the tight corridor that partitions the arena into two regions to be difficult for our agents to navigate. If all agents start in the same region, then in order to cover the entire arena some agents must enter and pass through the corridor to the other region. We selected the *Spiral* arena (Wagner et al., 2008) for it's circumnavigable obstacles and spiral-shaped wall.

Experiments

Each experiment we ran is described by a tuple $\langle E, \theta, \delta, n\rangle$ where $E \in \{Simple, Spiral, Walls\}$ is the arena, $\theta \in \{0.125, 0.25, 0.375, \ldots, 2.0\}$ is the turning angle range, $\delta \in \{20, 40, 60, 80, 100\}$ is the running distance, and $n \in \{1, 2, 3, 4, 5, 10\}$ is the number of agents. We created an experimental condition for each combination of parameters. For each experimental condition we ran the simulation 100 times, each time randomizing the starting positions of the agents. Therefore, we ran the simulation with 144,000 different initial conditions.

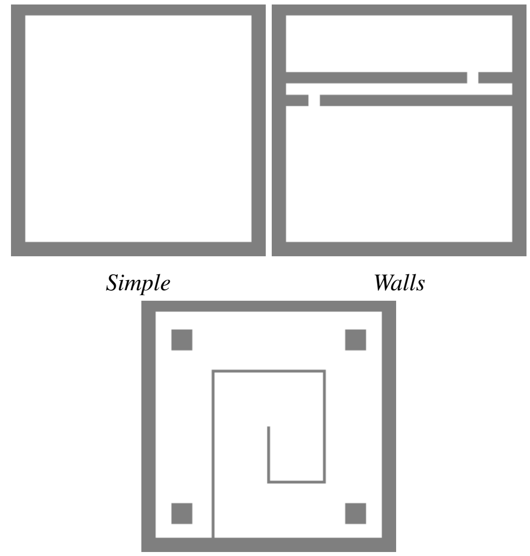


Figure 1: The environmental arenas. Note that because agents are scaled by the arena's area, the three arenas are effectively equal in area.

Performance Metrics

We use two metrics to assess the performance of a set of simulation parameters. The *performance sum* metric P_s expresses how much of the area is covered at each timestep, and is defined as

$$P_s = \sum_{t=1}^{100,000} C_t$$

where C_t is the total area that has been covered at timestep t. The *performance timestep* metric P_t expresses how long it takes to cover half of the arena, and is defined as

$$P_t = argmin_{t'}(t' \leq t \wedge C_{t'}/S_E \geq 0.5 \wedge C_{t'-1}/S_E < 0.5) \quad \forall t > 0$$

where C_t is the total area that has been covered at timestep t and S_E is the total area. Note that higher P_s values indicates better performance than lower P_s values, while the inverse is true for P_t.

As shown in Table 1, there were 17 experimental conditions, all of which place a single agent in the *Walls* arena, for which one or more runs had undefined P_t because the arena did not become 50% covered within 100,000 timesteps.

Pairwise Tests

In order to assess the value of using result distributions instead of just result means when determining optimal param-

Carole Knibbe et al, eds., Proceedings of the ECAL 2017, Lyon, France, 4-8 September 2017,
(Cambridge, MA: The MIT Press, ©2017 Massachusetts Institute of Technology).

Experiment	Number of runs
$\langle walls, 1.25, 20, 1\rangle$	2
$\langle walls, 1.75, 40, 1\rangle$	1
$\langle walls, 0.5, 40, 1\rangle$	1
$\langle walls, 1.625, 20, 1\rangle$	3
$\langle walls, 1.75, 20, 1\rangle$	5
$\langle walls, 1.25, 60, 1\rangle$	1
$\langle walls, 2.0, 20, 1\rangle$	4
$\langle walls, 1.5, 20, 1\rangle$	3
$\langle walls, 0.875, 60, 1\rangle$	1
$\langle walls, 0.125, 40, 1\rangle$	1
$\langle walls, 0.25, 100, 1\rangle$	1
$\langle walls, 1.5, 40, 1\rangle$	1
$\langle walls, 1.125, 20, 1\rangle$	1
$\langle walls, 1.875, 20, 1\rangle$	2
$\langle walls, 0.375, 20, 1\rangle$	1
$\langle walls, 0.875, 20, 1\rangle$	2
$\langle walls, 1.375, 100, 1\rangle$	1

Table 1: Experimental runs having undefined P_t values.

eters, we conducted pairwise statistical tests of the P_s results of all experiments in arenas $E \in \{$*Simple*, *Spiral*, *Walls*$\}$.

For each pair of experiments, we ran a paired two-sample *t*-test with the null hypothesis that the means of the populations from which the two P_s samples were drawn have the same mean, and a two-sample Kolmogorov-Smirnov (KS) test with the null hypothesis that the samples were drawn from populations having the same probability density function (pdf). We conducted these tests to identify pairs of experiment results for which the *t*-test did not find a difference in the means but the KS-test did find a difference in the distributions. It was not possible to apply this approach to the P_t metric because the standard KS-test is only valid for continuous distributions.

We define the function $f(x)$ to be the set of experiment pairs that differ only by the single environmental parameter x such that the *t*-test finds no difference in the P_s performance mean but the KS-test does find a difference in the performance pdf.

$$X = \{E, \theta, \delta, n\}$$
$$A = \langle E_a, \theta_a, \delta_a, n_a\rangle$$
$$B = \langle E_b, \theta_b, \delta_b, n_b\rangle$$
$$x \in X \cup \{\emptyset\}$$

$$f(x) = \{\langle A, B\rangle : y_a = y_b \forall y \in X \setminus \{x\} \wedge \mathrm{p}_{t-test(A,B)} \geq \alpha \wedge \mathrm{p}_{ks-test(A,B)} < \alpha \; \forall \alpha \in \{0.001, 0.01, 0.05, 0.1\}\}$$

Thus, for any one of the four system parameters, represented by the variable x, the function $f(x)$ represents all of the pairs of experimental conditions, represented by the variables A and B, for which the following three conditions hold:

1. The values of all parameters except for x are the same for A and B.

2. Our KS-tests using each value of α all indicate that the performance distributions of A and B are different.

3. Our *t*-tests using each value of α all indicate that the performance means of A and B are not different.

Results

We have ranked the arenas according to relative difficulty by comparing the overall mean performance for experiments in each arena (Table 2). For both metrics, we can rank the arenas from easiest to hardest as *Simple*< *Spiral*< *Walls*.

Arena	P_t	P_s
Simple	1259.493 ± 4.650	98161.03 ± 6.692
Spiral	2236.093 ± 13.170	96533.97 ± 17.650
Walls	2948.231 ± 31.414	94501.48 ± 35.047

Table 2: Average results in all three arenas for both evaluation metrics, $\pm$ 1 standard deviation.

However, what constitutes a difficult environment may depend not only on the metric of evaluation but also on n. Figure 2 shows how the relative difficulty of two arenas changes as n increases. Each point in the figure is the average performance of a set of n agents in the *Walls* arena divided by the average performance of the same number of agents in the *Spiral* arena. Thus, for the P_t metric a value above 1 means that on average, that number of agents took more time to cover 50% of the *Walls* arena than to cover the same area in the *Spiral* arena. On the other hand, for the P_s metric a value below 1 represents that this set of agents covered, on average, a smaller area at each timestep of the simulation in the *Walls* arena than in the *Spiral* arena. With $n \leq 3$, *Walls* is harder than *Spiral* according to both metrics. However, for $n > 3$ the *Walls* arena produced better P_t results than *Spiral* (depicted as the red line crossing the cutoff of 1). We hypothesize that this divergence is caused by the increasing likelihood that some agents will start off in the larger region of the *Walls* arena as n increases.

According to the P_s metric, on the other hand, *Walls* persists as being harder than *Spiral* across all values of n, although the difference lessens as n increases. In general, as the number of agents deployed increases, the performance distribution gets tighter, a phenomenon that is most pronounced in the more difficult arenas.

Carole Knibbe et al, eds., Proceedings of the ECAL 2017, Lyon, France, 4-8 September 2017,
(Cambridge, MA: The MIT Press, ©2017 Massachusetts Institute of Technology).

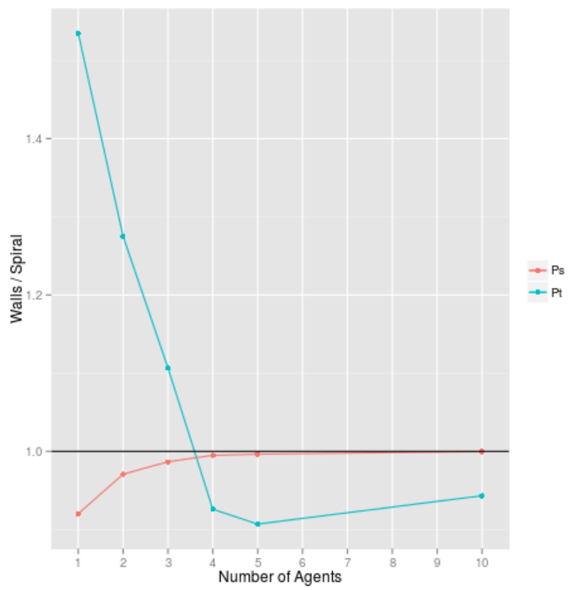


Figure 2: Ratio of performance in the *Walls* arena to that in the *Spiral* arena across all values of n.

Table 3 shows the results of the pairwise statistical test comparison. The first column indicates which pairs of experiment were tested by listing the parameter that varied between pairs. The second column shows the number of experiment pairs that differed by only one of the parameters in $\{E, \theta, \delta, n\}$ where the KS-test rejected the null hypothesis but the t-test did not for all values of α in $A = \{0.001, 0.01, 0.05, 0.1\}$. There were 35,869 experiment pairs in total where the ks-test rejected the null hypothesis but the t-test did not for all $\alpha \in A$. The third column shows the total number of pairs whose parameters differed only by the given parameter, and the fourth column is the ratio of the second and third columns. Figure 3 shows four comparisons of experiment pairs that differed by only one parameter with P_s results having the same mean but different distributions.

x	$\|f(x)\|$	N	$\|f(x)\|/$ N
E	5	1440	0.0035
θ	131	10800	0.0121
δ	19	2880	0.0066
n	10	3600	0.0278
$\emptyset$	35,869	1,036,080	0.0346

Table 3: Number and ratio of experiment pairs that differed by only one of the parameters in $\{E, \theta, \delta, n\}$ where the KS-test rejected the null hypothesis but the t-test did not for all values of α in $A = \{0.001, 0.01, 0.05, 0.1\}$.

We performed an analysis of variance (ANOVA) for each evaluation metric. With the P_s metric, we found significant main effects for all independent variables as well as for two, three and four way-interactions. With the P_t metric, there were significant main effects for all variables, and for two-way and three-way interactions, but not for a four-way interaction.

Discussion

Our results emphasize the benefits of considering the entire distribution of results when evaluating a parametrization of our simulation. Our ANOVA results demonstrate that E, θ, δ, and n all influence performance. What optimal values should be assigned to these variables, given our simulation results? The value of n may be determined partly by factors like cost, i.e., there may be material limits to how many agents can be deployed. The values of θ and δ, on the other hand, should be determined according to performance.

One option is to consider the performance averaged across the 100 runs of each experiment. However, deeper insight may be gained by analyzing the distribution of results for each experimental condition. For example, a distribution could contain outliers with performance significantly worse than the mean. In that case, the overall performance could be considered unacceptable due to the worst-case outliers, even though mean performance is satisfactory.

We do not know to what extent our methods and results will generalize to other multi-agent systems. However, our findings indicate that it may be fruitful to analyze the performance distributions of other systems in order to gain insight into the relationships between performance pdfs, performance means, and system parameters.

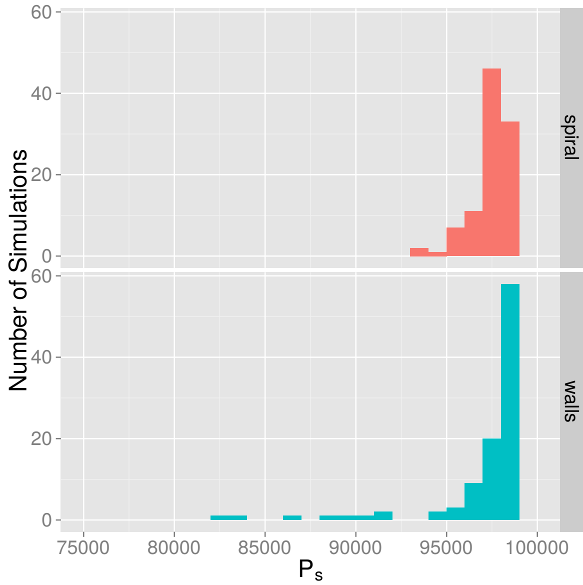
Pairwise Distribution Comparisons with P_s

When do changes to a single simulation parameter alter the distribution, but not the mean, of performance results? Our pairwise statistical tests address this question by identifying pairs of experiments that differ by a single parameter for which a *t*-test found no significant difference in means but a KS-test did find a significant difference in pdfs. Overall, the ratio of pairs differing by a single parameter that caused this divergence was small, less than 3% of for any parameter.

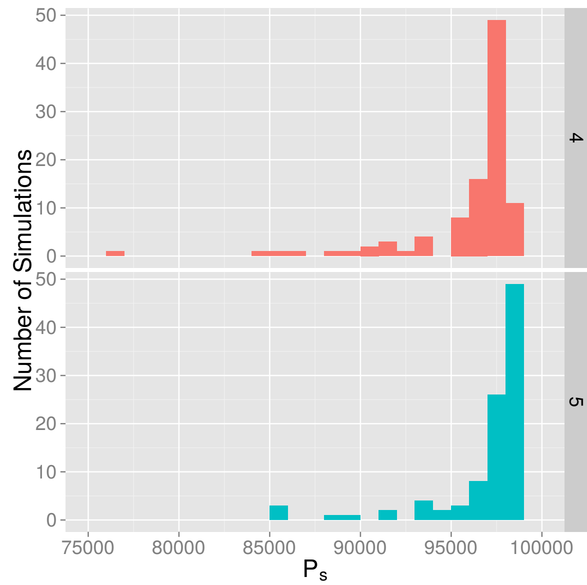
For each parameter, we selected for further analysis one pair of experiments identified by the pairwise distribution comparisons. Our goal is to identify qualities of the simulation performance that would be missed by only considering performance means.

Figure 3a shows histograms for the P_s metric for $\langle$*Walls*$, 1.125, 20, 5\rangle$ and $\langle$*Spiral*$, 1.125, 20, 5\rangle$. Performance is similar in the two different arenas, *Walls* and *Spiral*. However, there are some runs in the *Walls* arena with very bad performance. By visually observing the simulation during some of these runs we have found that agents often remain on one side of the separating wall for a long time before crossing to the other area, leading to long completion times.

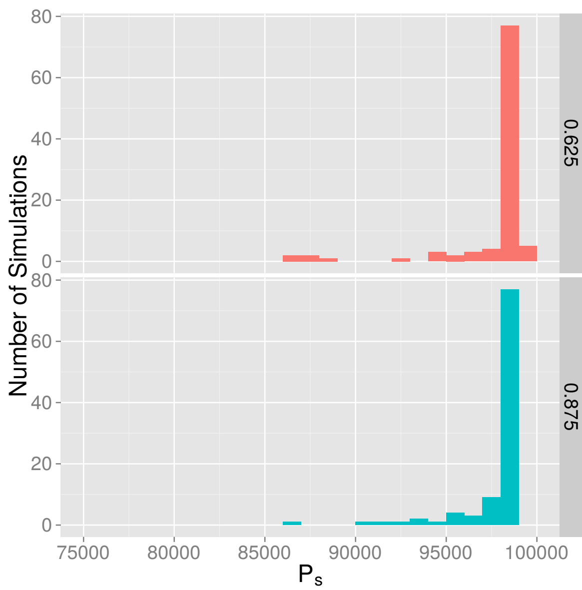
Carole Knibbe et al, eds., Proceedings of the ECAL 2017, Lyon, France, 4-8 September 2017,
(Cambridge, MA: The MIT Press, ©2017 Massachusetts Institute of Technology).



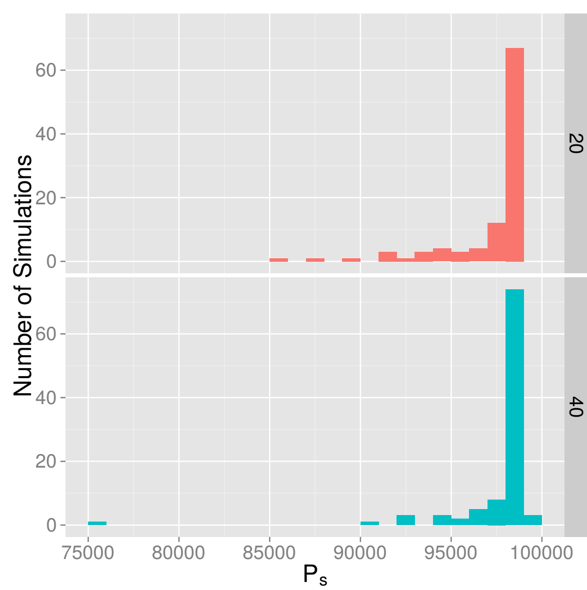
(a) Histograms for P_s metric in $\langle Spiral, 1.125, 20, 5\rangle$ (top) and $\langle Walls, 1.125, 20, 5\rangle$ (bottom)



(b) Histograms for P_s metric in $\langle Walls, 1.25, 20, 4\rangle$ (top) and $\langle Walls, 1.25, 20, 5\rangle$ (bottom)



(c) Histograms for P_s metric in $\langle Walls, 0.625, 20, 5\rangle$ (top) and $\langle Walls, 0.875, 20, 5\rangle$ (bottom)



(d) Histograms for P_s metric in $\langle Walls, 1.875, 20, 5\rangle$ (top) and $\langle Walls, 1.875, 40, 5\rangle$ (bottom)

Figure 3: Example of distribution pairs with only one distinct parameter for which the KS-test rejected the null-hypothesis but the *t*-test did not.

How robust is the performance distribution to a decrease in the number of agents assigned to the task, or to the failure of a subset of the agents? In the majority of comparisons between experiments where only n differs, mean performances were significantly different: decreasing the number agents lowers performance. However, there were ten experiment pairs for which varying n altered the performance distribution, but not the mean. Figure 3b depicts the distributions from one of those cases. The KS-test found a difference in the result distribution for $\langle Walls, 1.25, 20, 4\rangle$ and $\langle Walls, 1.25, 20, 5\rangle$, but the *t*-test found no difference in their means. Visual examination of the distribution reveals not only that $\langle Walls, 1.25, 20, 5\rangle$ had more simulation runs at the very top of the performance distribution, but also that

Carole Knibbe et al, eds., Proceedings of the ECAL 2017, Lyon, France, 4-8 September 2017,
(Cambridge, MA: The MIT Press, ©2017 Massachusetts Institute of Technology).

$\langle Walls, 1.25, 20, 4\rangle$had a poorly performing outlier. Thus, there is a notable difference in performance as n changes between these two experiments, even though the means are not statistically different.

Although we ran simulations in arenas with and without obstacles, agents in general performed better with more straight movements and less turning. Specifically, low values of θ, with agents making small changes to their heading, tended to result in better performance. High values of δ tended to improve performance as well, with agents moving for a longer distance before changing heading.

We visually compared the distributions of $\langle Walls, 0.875, 20, 5\rangle$ and $\langle Walls, 0.625, 20, 5\rangle$ because their result means were not significantly different. Figure 3c shows that there were 5 simulations of $\langle Walls, 0.625, 20, 5\rangle$ that performed performed better than any run in $\langle Walls, 0.875, 20, 5\rangle$. This corroborates our general findings that smaller values of θ improve performance.

Figure 3d shows histograms for a pair of experiments that vary only in the δ parameter. In this case, increasing δ does not significantly change mean performance. However, three of the runs $\langle Walls, 1.875, 40, 5\rangle$ performed better than all of the runs of $\langle Walls, 1.875, 20, 5\rangle$, corroborating our general finding that straighter movement is generally better. Nevertheless, a single outlier in $\langle Walls, 1.875, 40, 5\rangle$ has extremely poor performance.

Distribution Comparison with P_t

Even though we were unable to use the pairwise statistical tests to examine results according to P_t, a comparison of the P_t results for experiments $\langle Walls, 0.625, 60, 1\rangle$ and $\langle Walls, 0.625, 100, 1\rangle$ provides another example. Figure 4 shows histograms of P_t for both experiments.

The mean P_t across all 100 runs of $\langle Walls, 0.625, 60, 1\rangle$ is 7548.99. However, one run performed much worse than the mean, with $P_t = 86,982$. Figure 5 and Figure 6 compare the progress of the best and worst runs from this experiment. In the best-performing case, the agent started in the large open region and 50% of the arena was covered by timestep 2833. Between timesteps 2000 and 3000 the agent moved to the smaller region and covered it almost entirely, returning to the bottom region before timestep 20,000. By timestep 30,000, the agent in the best run had almost completely covered the arena, with only small spots remaining, while the agent in the worst run had not yet entered the lower region. Between timesteps 30,000 and 40,000 the agent in the worst run entered the lower region, covered a small part of it, but then returned to the upper region, remaining there until timestep 60,000.

With $\delta = 100$ instead of $\delta = 60$, $\langle Walls, 0.625, 100, 1\rangle$ achieved a mean P_t of 7840.27, worse than $\langle Walls, 0.625, 60, 1\rangle$. However, in this case there were fewer outliers with exceptionally poor performance. Thus, analysis of the experimental distributions suggests that the best parameters for a given arena might not be those with the best performance on average.

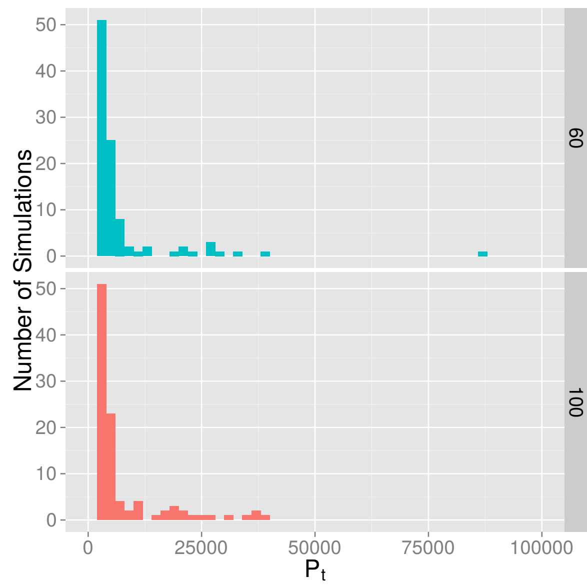


Figure 4: Histograms for P_t metric in $\langle Walls, 0.625, 60, 1\rangle$ and $\langle Walls, 0.625, 100, 1\rangle$.

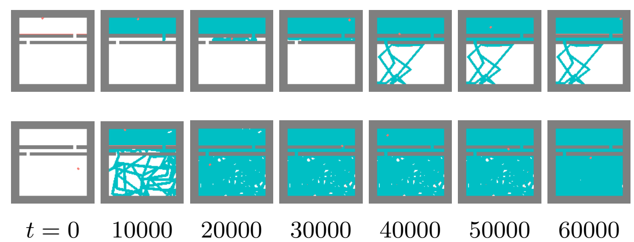


Figure 5: Time series showing a single evaluation of $\langle Walls, 0.625, 60, 1\rangle$. The top row shows the worst solution, with $P_t = 86982$. The bottom row shows the solution with the lowest P_t, 2833.

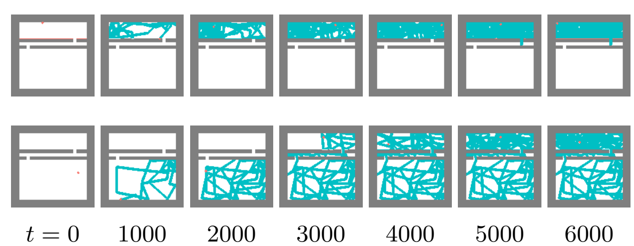


Figure 6: Timeseries every 1000 timesteps for the worst (top row) and the best (bottom row) runs for experiment $\langle Walls, 0.625, 60, 1\rangle$.

Carole Knibbe et al, eds., Proceedings of the ECAL 2017, Lyon, France, 4-8 September 2017,
(Cambridge, MA: The MIT Press, ©2017 Massachusetts Institute of Technology).

Generalization of Techniques

Our simulation was simple enough to permit the evaluation of a broad parameter sweep over many repetitions using reasonable computational resources. Such a comprehensive exploration of the relationships between parameter space and performance distribution may not be possible in all scenarios. With physical robotic systems, or with simulations that have more parameters or demand more computational resources than ours, it may be infeasible to collect all the necessary data. Nevertheless, it is possible even in many such cases to devise techniques for performance analysis that go beyond a comparison of means to consider distributions of performance. For example, as discussed in our Related Work section, Thill and Pearce (2007) used a recursive technique to explore performance distributions over a 12-dimensional parameter space.

For some systems, analysis of performance distributions may yield little insight beyond that offered by their means. However, that cannot be determined without some knowledge or assumptions about the shape of those distributions, begging the original question: What is the distribution of performance results across multiple iterations of the system?

As another example of how our findings may apply more broadly, we propose that system specifications with respect to performance pdfs could guide the selection of system parameters, either by a parameter sweep like the one we conducted or by a search process. These specifications could take the form of explicit constraints on the distribution, bounding, for example, the mean, minimum, variance, skewness, or kurtosis. Alternatively, it could be specified that the distribution should be unimodal, or should fit some closed form pdf, such as a uniform, normal, or Poisson, distribution. Finally, one could specify constraints on the relationships between parameters and the performance distribution. For example, a robustness specification could require a decrease in the number of agents deployed not to significantly alter the performance pdf. As evidenced by our results, such specifications could distinguish sets of system parameters whose average performances are statistically equivalent.

Conclusions

We have described a simulation of the area coverage problem with agents controlled by a simple, randomized algorithm. The results of a suite of experimental runs over a broad parameter sweep indicate the significance of all of our simulation parameters. Analysis of these results has shown that some parameter changes alter the distribution of the simulation performance over many starting conditions, even if the mean performance does not change. Thus, using the entire distribution of behaviors to assess the effectiveness of a set of simulation parameters can provide insight beyond what is possible by simply comparing averages.

Acknowledgements

This work was supported in part by NSF grant #1257815. D.B. acknowledges support by NSF IGERT grant #1144591.

References

Acar, E. U., Choset, H., Zhang, Y., and Schervish, M. (2003). Path planning for robotic demining: Robust sensor-based coverage of unstructured environments and probabilistic methods. *The International Journal of Robotics*, 22(7-8):441–466.

Atkar, P. N., Greenfield, A., Conner, D. C., Choset, H., and Rizzi, A. A. (2005). Uniform coverage of automotive surface patches. *The International Journal of Robotics Research*, 24(11):883–898.

Choset, H. (2000). Coverage for robotics – a survey of recent results. *Annals of Mathematics and Artificial Intelligence*, 31(1):113–126.

Galceran, E. and Carreras, M. (2013). A survey on coverage path planning for robotics. *Robotics and Autonomous Systems*, 61(12):1258 – 1276.

Oksanen, T. and Visala, A. (2009). Coverage path planning algorithms for agricultural field machines. *Journal of Field Robotics*, 26(8):651–668.

Palacin, J., Palleja, T., Valganon, I., Pernia, R., and Roca, J. (2005). Measuring coverage performances of a floor cleaning mobile robot using a vision system. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 4236–4241.

Ramsli, E. (1991). Probability distribution of repeatability of industrial robots. *The International Journal of Robotics Research*, 10(3):276–283.

Schmidt, G. and Hofner, C. (1998). An advanced planning and navigation approach for autonomous cleaning robot operations. In *Intelligent Robots and Systems, 1998. Proceedings., 1998 IEEE/RSJ International Conference on*, volume 2, pages 1230–1235 vol.2.

Schneider, E., Sklar, E. I., Azhar, M. Q., Parsons, S., and Tuyls, K. (2015). Towards a methodology for describing the relationship between simulation and reality. In *Proceedings of the European Conference on Artificial Life 2015*, pages 562–569.

Thill, S. and Pearce, T. C. (2007). Understanding complex behaviors by analyzing optimized models: C. elegans gradient navigation. *HFSP Journal*, 1(4):263–273.

Wagner, I. A., Altshuler, Y., Yanovski, V., and Bruckstein, A. M. (2008). Cooperative cleaners: A study in ant robotics. *The International Journal of Robotics Research*, 27(1):127–151.

Wagner, I. A., Lindenbaum, M., and Bruckstein, A. M. (2000). Mac versus pc: Determinism and randomness as complementary approaches to robotic exploration of continuous unknown domains. *The International Journal of Robotics Research*, 19(1):12–31.

Zhu, J. and Ting, K.-L. (2000). Performance distribution analysis and robust design. *Journal of Mechanical Design*, 123(1):11–17.

Carole Knibbe et al, eds., Proceedings of the ECAL 2017, Lyon, France, 4-8 September 2017,
(Cambridge, MA: The MIT Press, ©2017 Massachusetts Institute of Technology).