

Technical paper

Collaborative surface patterning through cascaded ergodic control and automatic task allocation

Malachi Landis[✉], Annalisa T. Taylor[✉], Todd D. Murphey[✉], Ping Guo^{✉*}

Department of Mechanical Engineering, Northwestern University, 2145 Sheridan Road L286, Evanston, IL, 60208, USA

ARTICLE INFO

Keywords:

Multi-robot systems
Ergodic control
Surface patterning

ABSTRACT

Coordinating groups of heterogeneous agents to accomplish multiple sequential surface patterning objectives is challenging and may be difficult or impossible to achieve manually when the number of agents does not evenly partition the tasks. One important sequential surface patterning task is the application of friction-reducing textures, where beneficial indentations are first formed in a surface then the resulting raised burr is removed via polishing, producing a surface with lower sliding friction under submerged lubrication. Here, we introduce cascaded ergodic control to enable automatic sequential surface modification by teams of mobile robots. The trajectory history of a group of leader robots is taken as the target for a group of follower robots, allowing implicit task allocation without explicit sharing of task information. We demonstrate, in both simulation and experiment, that this method matches the performance of existing ergodic control schemes and enables flexible assignment of agents to different tasks without directly communicating objectives to follower robots. Teams of indentation and polishing robots work together in physical experiments to pattern acrylic workpieces, demonstrating a process scalable to patterning of massive workpieces like airplanes or cargo ships with greater flexibility than existing methods.

1. Introduction

Manufacturing often requires sequential operations performed by heterogeneous agents with different capabilities [1]. Coordinating these agents is often complex and inflexible, particularly when multiple objectives must be accomplished simultaneously [2,3]. One common sequential process is surface patterning, or the addition of features to a surface which enhances its functionality. Laser ablation, stamping, polishing, or etching, among other processes, can be used in some combination to impart drag reduction [4], anti-icing [5], biofouling resistance [6], and more, and so could greatly benefit applications like airplanes or cargo ships. Traditional, high-precision methods cannot access these large workpieces due to poor scalability. Teams of mobile robotic agents could solve this scalability problem but would require a specialized control strategy designed to overcome the aforementioned complexity.

Parker et al. describe the challenges with using teams of mobile robots to solve coordinated tasks [7]. Sequential surface patterning is especially demanding: work must be split among heterogeneous agents whose number rarely divides the objectives evenly, and robots handling later stages must act on a target that earlier-stage robots are still producing.

Practically, patterning robots must coordinate not only with other robots but must plan their own trajectories as well. Traditional robotic polishing is typically done with a single robot arm and requires designing and executing a path-planning strategy. Numerous works have explored how to generate paths for polishing, but typically default to simple back and forth scanning paths [8], occasionally augmented with modeling of the tool behavior [9] or addition of curvilinear features in Peano-like paths [10]. Others have focused on predicting coverage using contact mechanics [11] or spray paint distributions [12].

Ergodic control offers a route towards solving both problems, group task allocation and individual path generation, by generalizing the path-planning problem for surface patterning. In [13], a single mobile robot, controlled by an ergodic sequence of points generated offline by a central computer, patterned brass and aluminum workpieces with an indentation tool. Ergodic control is an optimal control scheme centered around the ergodic metric, given in [14] and explained in Section 2.1. In short, it seeks to minimize the difference between two distributions in Fourier space, and as such is naturally suited for control of patterning robots whose tasks (often images) are easily represented in terms of frequency components.

In [15], a team of 5 robots patterned meter-scale steel workpieces. Robots worked independently on the patterning objective, yet the

* Corresponding author.

E-mail address: ping.guo@northwestern.edu (P. Guo).

<https://doi.org/10.1016/j.jmsy.2026.09.003>

Received 2 September 2026; Accepted 2 September 2026

Available online 11 September 2026

0278-6125/© 2026 The Society of Manufacturing Engineers. Published by Elsevier Ltd. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

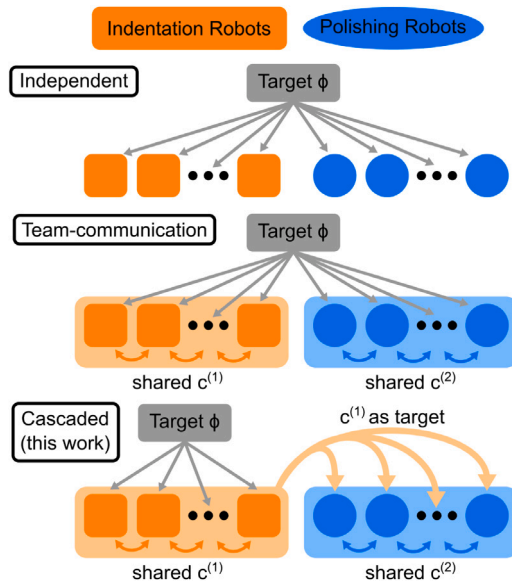


Fig. 1. Independent, team-communication, and cascaded control modes differ in how the target pattern is supplied to the robots. Cascaded ergodic control uses the trajectory statistics of the indentation robots as the target for the polishing robots.

group still accomplished a cohesive goal. The same robot then worked within a team of 4, showing that introducing communication through a shared trajectory representation could enable task division without explicit external guidance [16]. Additionally, rotary tribometer tests confirmed the efficacy of the generated surface patterns.

In this work, we introduce cascaded ergodic control, in which one group of robots takes another group's trajectory history as its target in place of a predefined objective. Cascaded control enables practical adaptability to a priori unknown target distributions without requiring explicit communication of these distributions during patterning. Follower robots allocate themselves automatically, that is, spontaneously divide the patterning target in space, without being explicitly given the objective as required in previous work [16].

We demonstrate this cascaded control approach, shown schematically in Fig. 1, through a practically-motivated indentation and polishing task. We introduce a roughness heuristic to estimate polishing performance in simulation and present results showing that robots polishing without initial knowledge of the desired pattern (i.e., based solely on indentation robot behavior) give performance that matches robots with complete knowledge of the objective. We simulate the case where different objectives are given to each indentation robot and show that the polishing robots successfully adapt without explicit knowledge of the objectives. Finally, we perform physical experiments using two indentation and two newly-modified polishing robots to confirm that cascaded control performs comparably to existing methods while adding task-allocation flexibility those techniques cannot provide.

Section 2 describes the ergodic control scheme, roughness, and the robotic hardware. Section 3 gives several simulated and experimental demonstrations of the proposed technique, along with discussion of tradeoffs and future work. Finally, Section 4 concludes with a summary.

2. Materials and methods

We briefly describe the general control architecture, particularly the cascaded control metric, though implementation details and hardware specifics can be found in previous work [15,16]. Additionally, we describe the roughness heuristic and additional hardware required for polishing.

2.1. Ergodic control scheme

Ergodic control is generally a technique for matching temporal and spatial statistics, typically used in robotics for coverage tasks [17]. The time-history statistics (i.e., how long a robot has been in an area) are controlled to match some desired spatial statistics (i.e., a feature density map). It is an optimal control technique, meaning control inputs are chosen to generally reduce the value of an objective function over time. Our objective, shown in Eq. (1), is a linear combination of an ergodic term $\mathcal{E}$, a barrier term $\mathcal{B}$, and a control term $\mathcal{D}$, along with their respective weights $w_{\mathcal{E}}$, $w_{\mathcal{B}}$, and $w_{\mathcal{D}}$, respectively. The barrier and control terms are described in detail in [15], but they generally restrict robot positions to a desired area and prevent excessive control inputs, respectively.

$$\ell = w_{\mathcal{E}}\mathcal{E} + w_{\mathcal{B}}\mathcal{B} + w_{\mathcal{D}}\mathcal{D} \quad (1)$$

The standard, single-robot ergodic metric given in [17] is shown in Eq. (2). It is a difference between two Fourier space representations, c and ϕ , over all pairs of Fourier basis functions k_x, k_y , weighting each contribution according to Λ . Details on computing c , ϕ , and Λ are given in [15]. The robot trajectory representation, c , is computed by integrating the Fourier basis functions for robot position over a given time horizon. The target coverage representation, ϕ , is computed by integrating the Fourier basis functions for a target distribution over the distribution's domain. These statistics are in Fourier space because they allow analytical derivatives, a critical component for effective optimal control schemes (see [15] for these derivatives).

$$\mathcal{E} = \sum_{k_x=0}^{K-1} \sum_{k_y=0}^{K-1} \Lambda_{k_x, k_y} \left(c_{k_x, k_y}(t) - \phi_{k_x, k_y} \right)^2 \quad (2)$$

In [16], we extended the single-robot ergodic metric to multi-robot systems. Eq. (3) uses the trajectory statistics of the entire group of robots, $c^{(1)}$, computed by averaging c across all robots. This metric allows implicit coordination during patterning where robots will tend to segregate themselves to more efficiently match the target distribution over time. Robots share their position with other robots but are not explicitly assigned to cover a certain area.

$$\mathcal{E}^{(1)} = \sum_{k_x=0}^{K-1} \sum_{k_y=0}^{K-1} \Lambda_{k_x, k_y} \left(c_{k_x, k_y}^{(1)}(t) - \phi_{k_x, k_y} \right)^2 \quad (3)$$

Here, we introduce a cascaded ergodic metric, shown in Eq. (4). Two teams of robots are used, each with their own trajectory representation ($c^{(1)}$ and $c^{(2)}$). The first team uses Eq. (3) to compute control inputs. The second team computes the difference between their statistics $c^{(2)}$ and the first group's statistics $c^{(1)}$, rather than the fixed target representation ϕ . Note that with this metric, the second team of robots does not need to store ϕ . The second team tracks the evolving target $c^{(1)}(t)$ rather than the fixed distribution ϕ used by the other modes. Note that in implementation, values of $c^{(1)}$ are not communicated directly; instead, positions are communicated to all robots which then perform local updates to their own $c^{(1)}$.

$$\mathcal{E}^{(2)} = \sum_{k_x=0}^{K-1} \sum_{k_y=0}^{K-1} \Lambda_{k_x, k_y} \left(c_{k_x, k_y}^{(2)}(t) - c_{k_x, k_y}^{(1)}(t) \right)^2 \quad (4)$$

We compare these metrics across three control modes for the indentation and polishing robots, illustrated in Fig. 1. In "Independent" mode, each robot matches the target ϕ from its own trajectory alone (Eq. (2)). In "Team-Communication" mode, each group shares a single trajectory and matches ϕ together (Eq. (3)), dividing the work implicitly within the group. In "Cascaded" mode, the indentation robots do the same, but the polishing robots instead target the indentation robots' trajectory $c^{(1)}(t)$ (Eq. (4)) rather than a preset ϕ , following what those robots actually produced. Table 1 gives the control parameters used across all experiments such that any performance differences reflect the control formulation rather than tuning. K is the number

Table 1
Control parameters.

K	L	Δt_{sim}	Δt_{MPC}	H	α	w_ε	w_B	w_D
20	500 mm	50 ms	250 ms	20	0.01	10	1	0.1

of basis functions used in each direction and L is the side length of the physical square domain. Δt_{sim} is the timestep for Newton's method when simulating robot behavior. The shooting method implementation of model predictive control (MPC) detailed in [15] has parameters H , the number of simulation timesteps in a rollout, Δt_{MPC} , the timestep used for those rollouts, and α , the coefficient on modifications to the controls. Objective function weights w_ε , w_B , and w_D were chosen by manual tuning in simulation.

2.2. Roughness heuristic

The ergodic metric does not fully capture processing of the workpiece by polishing robots. For that reason, we introduce the roughness heuristic as a proxy for surface processing based on two ideas:

1. Indentation sets surface roughness at a single location instantly to a specified upper bound R_{max} , even if that area was originally polished.
2. Polishing decreases surface roughness over an area at a certain rate down to a lower bound R_{min} . Additional polishing beyond the lower bound has no effect.

These two rules reasonably represent the polishing task and allow the degree of polishing to be estimated within simulation. The two parameters (roughness added per indentation and roughness removal rate from polishing) can be combined into one: polishing time required to remove the roughness added by one indentation. This parameter is then calibrated experimentally by measuring the time required to remove a single indentation's roughness.

Preliminary experiments with this robot hardware showed that 2 s of polishing was sufficient to remove a single indentation's roughness; this choice was verified later by microscope measurements in Section 3.5. In simulation, this metric is implemented using a 2000×2000 grid storing the remaining time steps required to reach the roughness lower bound, shown in Algorithm 1. Indentation adds 40 timesteps of roughness, and the polishing tool affects a radius of 20 grid points. Because the heuristic is evaluated over this spatial grid, it directly quantifies coverage in a way the ergodic metric cannot. The fraction of the target region driven to the roughness lower bound measures how completely the surface has been patterned, and its rate of change gives a coverage rate.

Algorithm 1 Roughness heuristic.

```

1: Initialize roughness grid to  $R_{min} = 0$ 
2: for each simulation timestep do
3:   for each robot do
4:     Find local roughness at  $(x, y)$ 
5:     if robot is indenting then
6:       Set roughness at point to  $R_{max} = 40$ 
7:     if robot is polishing then
8:       if local roughness  $> R_{min}$  then
9:         Decrement local roughness of area by 1
10: Sum over grid to find total roughness

```

As discussed in [15], the total number of indentations required to achieve surface functionality can be determined by the intended application and density distribution. Assuming no or minimal indentation overlap, then the total roughness added to the surface can be calculated

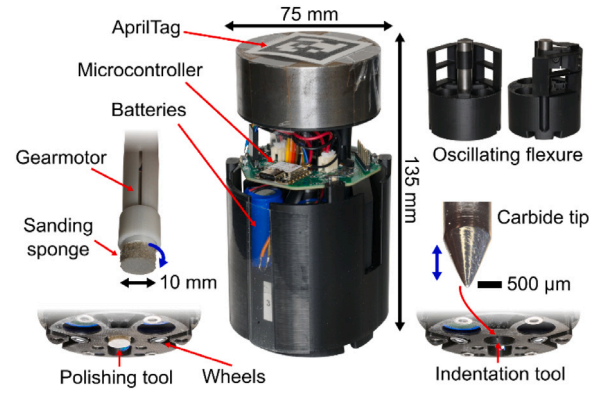


Fig. 2. Mobile robot hardware. The small, two-wheeled robots are equipped with either a polishing or indentation tool. The microcontroller computes control actions onboard and receives position information by communicating with a central camera system tracking each robot using AprilTags.

by multiplying the roughness per indentation by the total number of indentations.

Stopping criteria for each group are computed separately. The total number of required indentations is divided by the indentation production rate, determined by the number of indentation robots and their individual indentation rates, to compute the stop time for the indentation robots. Target roughness can be specified in terms of percentage of total roughness removed, i.e., a 90% reduction in peak roughness; simulations to predict when this specified roughness reduction occurs set the stop time for polishing robots.

2.3. Hardware design

Fig. 2 gives an overview of the robot. The robot is mostly 3D printed, with motors, batteries, and either an oscillating indentation tool or rotating polishing pad contained within a 75 mm diameter, 135 mm tall cylindrical volume. The small size allows multiple robots to work together on a workpiece without significantly disrupting each other. Bluetooth communication and AprilTags enable localization through a central computer and camera. The localization noise was characterized by recording the position estimate of a stationary robot over 200 frames, giving $2\sigma < 0.05$ mm, giving the repeatability of the AprilTag-based tracking. The highest spatial frequency used in this work (20) gives a minimum feature resolution of 25 mm; therefore, this localization method is more than adequate for the application (see Appendix). See [15] for more hardware details, particularly for the indentation tool.

Two of the four robots use a polishing tool (Fig. 2, left) rather than an indentation tool (Fig. 2, right). The polishing tool is formed from a DC gearmotor rotating a piece of aluminum oxide sanding sponge, collectively mounted in a printed flexure which serves to apply consistent downward force. Though simple, it is sufficient for removing the burrs raised when the indentation tool operates on an acrylic workpiece, demonstrated experimentally in Section 3.5.

3. Results and discussion

Two simulation studies, detailed in Sections 3.1 and 3.2, verify the cascaded ergodic control scheme and compare it to independent and team-communication modes. Two experiments, detailed in Sections 3.3 and 3.4, were then performed to validate the simulation results and demonstrate that the algorithms described can run successfully on a realistic robot system. Section 3.5 gives additional measurements and roughness heuristics from the experiments to demonstrate successful processing.

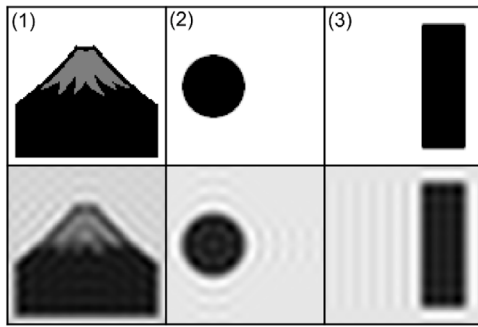


Fig. 3. Target images and their Fourier reconstructions: (1) cartoon of Mt. Fuji, (2) circle, and (3) rectangle. All are 100 pixels square and represented with 20 basis functions.

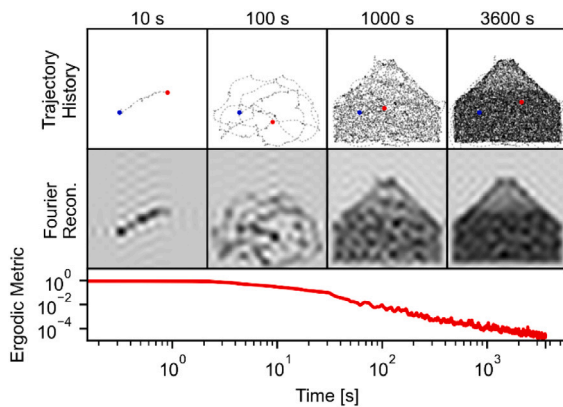


Fig. 4. An illustrative single robot patterning. Reconstruction of the trajectory Fourier representation shows that the robot approximates the desired target distribution over time, quantified by the decreasing ergodic metric.

Target images are normalized to generate a correct probability distribution ϕ . The mode-comparison study in Section 3.1 uses a target image of Mt. Fuji (Fig. 3, 1), chosen for its recognizable profile and lightly shaded peak. The cascaded-mode, multi-objective study in Section 3.2 uses a circle (Fig. 3, 2) for one indentation robot and a rectangle (Fig. 3, 3) for the other and does not supply ϕ to the polishing robots.

An illustrative single-robot simulation is shown in Fig. 4. The robot is initialized to a starting location, then steps forwards in time according to its dynamics and chosen controls. A robot which is behaving ergodically will increasingly approximate the target distribution with its trajectory history, visible qualitatively both in the trajectory history and its Fourier reconstruction, and quantitatively through the improving ergodic metric.

3.1. Simulation 1 – Comparing independent, team-communication, and cascaded ergodic control

The following simulations were repeated 100 times, where the initial positions and orientations of the simulated robots were chosen at random while ensuring no two robots clipped into each other's boundary. Within a simulation, collisions between robots are handled by stepping backwards one timestep, rotating the colliding robots to a random heading, and stepping forward one timestep. This is repeated within a single simulation timestep as needed until the robots move apart and the collision is resolved. The collision count is incremented at most once per robot per timestep, even when resolving a collision takes several sub-steps, and it measures how much the robots physically

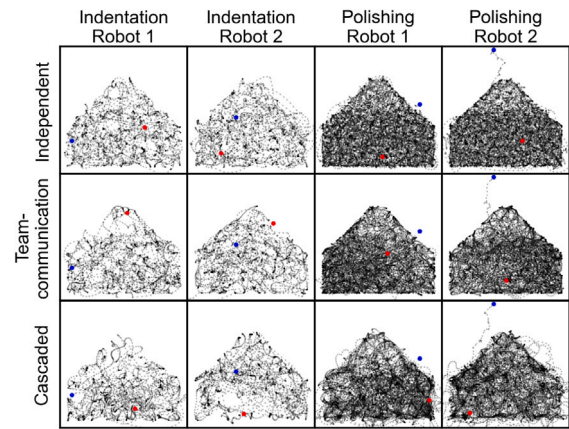


Fig. 5. Trajectories from a representative trial for the different control modes in Simulation 1. The blue dots are the start locations and the red dots are the finish locations. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

interfere with one another. A similar collision resolution scheme is used in experiment.

Each of the three control methods described in Section 2.1 was simulated. The total simulation time was 60 min with the indentation robots deactivated after 15 min. A representative group of trajectories from a single, cascaded ergodic control trial is shown in Fig. 5. Qualitatively, the trajectories of the indentation robots are similar between all three modes. The team-communication and cascaded modes operate identically for indentation robots, so some task division is expected in both as predicted by [16]. The trajectories of the polishing robots are relatively distinct between modes. Independent and team-communication modes cover the lower part of the target well and successfully reproduce the lighter shaded area. The cascaded mode has patchier coverage, and the lighter shaded area is not as obvious. The edges of the cascaded mode coverage are “fuzzier” because they do not have access to the barrier function.

A better comparison can be made quantitatively using the average roughness heuristic, ergodic metric, and collision histories plotted in Fig. 6. For the roughness heuristic and collisions, the shaded area indicates ± 1 standard deviation, while on plots of the ergodic metric, only +1 standard deviation is shown due to the semilog axes. The green lines show the time at which the average roughness curve reaches 90% removal.

The roughness heuristic showed similar trends across all the control modes. Roughness increased up to a peak, then decreased over time. All the modes showed comparable times (less than 45 min) to reach the 90% reduction target. The slope of the curve indicates the coverage rate, which begins high and decreases as remaining roughness becomes more spatially dispersed and robots travel at a finite speed.

The ergodic metric for indentation robots was relatively consistent across all modes. It decreased over time, then held flat after the indentation robots were removed. On the other hand, the polishing robots continued to improve their metric with the independent and team-communication modes, while they asymptotically approached the metric of the indentation robots in the cascaded mode. In addition, there is a lag between the polishing robots and the indentation robots.

This lag and plateau are expected from the nature of the cascaded mode: polishing robots follow the “blueprint” laid out by indentation robots, so their performance can only be as good, not better, and requires time to pass to allow the indentation robots to produce that blueprint. Mathematically, if the indentation robots are operating ergodically, then $c^{(1)}$ approaches ϕ after infinite time (subject to the robot dynamics). For a finite time, such as those in these simulations and experiments, $c^{(1)}$ only approximates ϕ , so $c^{(2)}$ can at best plateau at $c^{(1)}$,

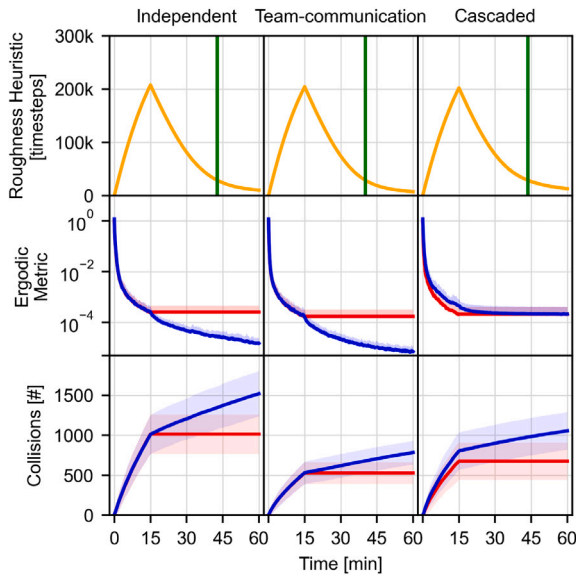


Fig. 6. Performance metrics for the different control modes calculated using 100 trials. Red lines correspond to the indentation robots and blue lines to the polishing robots. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

even after infinite time. Additionally, since $c^{(1)}$ is an imperfect representation of ϕ , its gradients are also an approximation, so polishing robots will not approach ϕ as rapidly, giving a lag. Due to the complexities of the robot dynamics, it is difficult to make predictions on the exact amount of lag or time to plateau between $c^{(1)}$ and $c^{(2)}$.

Fig. 6 also shows the importance of the roughness heuristic. The ergodic metric for cascaded control might suggest polishing stops improving early, while the roughness heuristic shows continued effectiveness. Additionally, the roughness heuristic gives a stopping criteria (i.e., 90% roughness removal) which is not included in the ergodic metric itself; generally, stopping criteria are an open problem in the ergodic control community and often problem-dependent.

3.2. Simulation 2 – Multi-objective flexibility using cascaded ergodic control

Cascaded ergodic control enables different targets to be given to different indentation robots without requiring those targets to be known or supplied to the polishing robots ahead of time. This allows polishing robots to allocate themselves flexibly during run-time.

To showcase this effect, two indentation robots were given separate target distributions (a circle and a rectangle) and the number of polishing robots varied from 1 to 3. Such a scenario would be difficult to handle with manual task allocation: in the case of 1 polishing robot, it could be given the total of both distributions. For 2 polishing robots, each could be assigned one target distribution. However, for 3 polishing robots, it is not clear how those target distributions should be assigned in a way that fairly allocates each robot's time.

The result is shown in Fig. 7. The trajectories of the indentation robots are sharply confined to the target image boundaries due to the barrier function. The trajectories of the polishing robots meander at the beginning, but over time tend to converge to a particular distribution. As expected, 1 polishing robot works on the entire, cumulative task and 2 robots split the task relatively evenly. In the case of 3 robots, there is a mix of behavior: polishing robots 1 and 3 have similar distributions which focus on the circle and top of the rectangle, while polishing robot 2 focuses on the bottom of the rectangle.

Following [16], we computed trajectory heterogeneities over multiple trials to confirm consistent task division across randomized starting

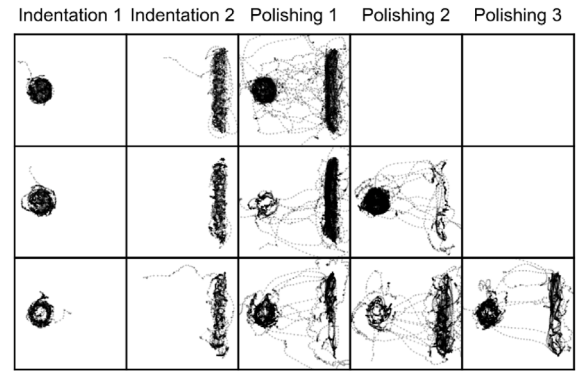


Fig. 7. Trajectory histories from a multi-objective simulation showing the flexibility of cascaded ergodic control. Each row has a different number of polishing robots, none of which are directly informed of the target distributions.

conditions. The ergodic metric is computed pairwise using individual robot histories; a large value indicates highly dissimilar trajectories. For reference, the average final heterogeneity between the polishing robots in Simulation 1 was less than 10^{-4} for independent mode and above 10^{-2} for both team-communication and cascaded modes. Across 100 trials of the uneven, 3 polishing robot case, the heterogeneity was above 10^{-1} , indicating persistent task division robust to starting conditions.

We do not compare to a team-communication, merged-target case (i.e., where targets 1 and 2 are combined) because this would effectively yield the same result as Simulation 1. The behavior of the indentation robots does not change between team-communication and cascaded control. Cascaded control is specifically intended to give greater flexibility, rather than better performance, compared to team-communication.

3.3. Experiment 1 – Team-communication and cascaded ergodic control

The simulations in Section 3.1 suggested that the team-communication mode could provide the fastest time to achieve a 90% reduction in the roughness heuristic and the greatest decrease in the ergodic metric, so it was chosen as the benchmark to compare cascaded ergodic control against in experiment.

Robots were placed in a grid starting arrangement on top of an acrylic sheet with LEDs placed along its periphery (a “lightboard”). Though the robots can apply indentations to harder materials like aluminum, brass, and steel [15], the lightboard allows small features to be seen much more readily since each indentation effectively becomes a point light source. Additionally, the polishing tool leaves a distinct, diffuse trail which also visualizes robot behavior.

All robots started together, then, after 15 min, the indentation robots were deactivated and removed from the lightboard, matching the conditions in the simulation. After 45 min, the experiment was completed. This reduced runtime was based on the time for 90% roughness reduction predicted by the simulation. Fig. 8 shows selected photographs from during the cascaded mode experiment, and Fig. 9 shows the final trajectories, patterning results, and metrics. As in simulation, robots that collided rotated randomly to escape getting stuck.

The robots reduce the ergodic metric over time, though not as effectively as in simulation. This difference can be explained by several discrepancies between simulation and experiment: the bump sensors the robots use to detect collisions require some amount of force to activate, so robots may not detect and respond to collisions immediately, leading to moments where robots are stuck. The wheels of the robots are driven by small, 3.3 V DC gearmotors. At low voltages, the

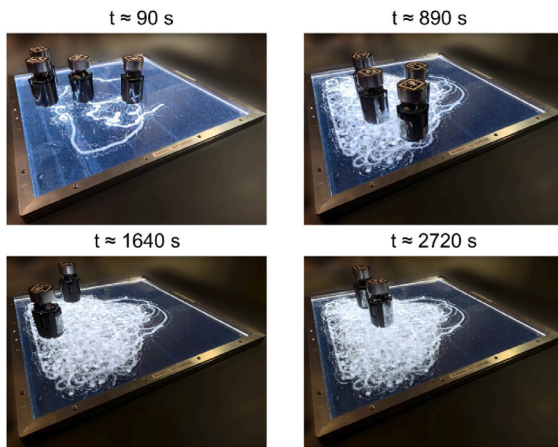


Fig. 8. Oblique view of robots patterning the Mt. Fuji target on the acrylic lightboard using cascaded ergodic control.

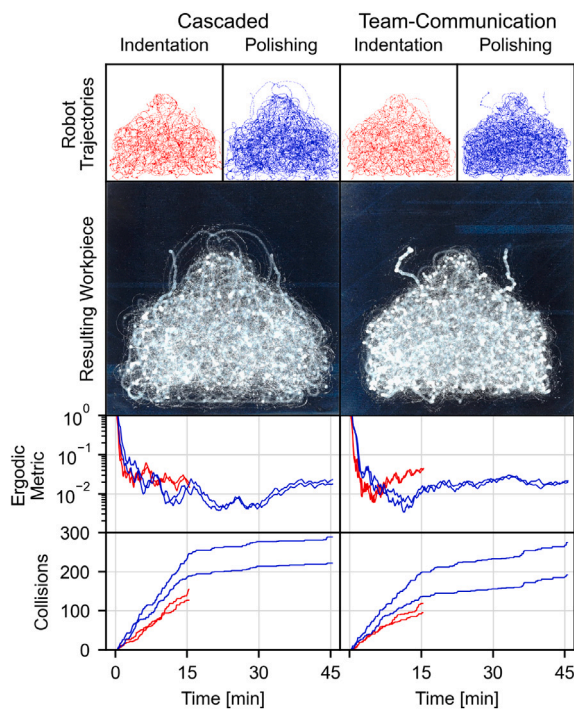


Fig. 9. Experimental results for team-communication and cascaded ergodic control modes. Note the “fuzzy” borders for cascaded control as the robots do not have access to the barrier functions.

motor may not produce the expected rotational velocity, so the actual applied control may differ from the desired control. Additionally, the robots produce dust from both the indentation and polishing processes, and this dust causes slip at the wheels. These differences are explored in simulation in the [Appendix](#). Overall, the robots achieve quantitatively similar performance between team-communication and cascaded ergodic control, both in terms of the ergodic metric and number of collisions.

3.4. Experiment 2 – Multi-objective flexibility with cascaded ergodic control

The case of 2 polishing robots from Simulation 2 was verified in experiment. The experimental conditions were identical to Experiment 1, except that indentation robots 1 and 2 were loaded with the circle and rectangle target distributions, respectively. Selected images from the

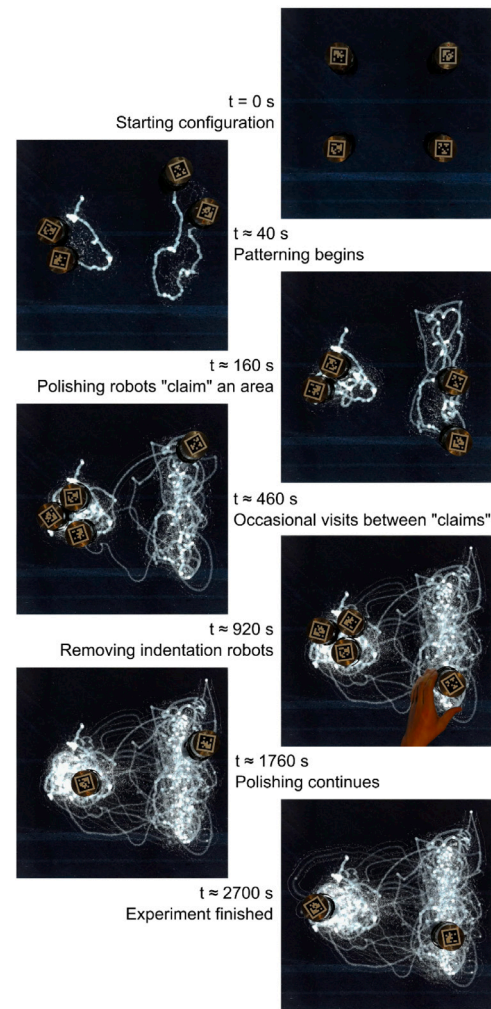


Fig. 10. Timelapse of the multi-objective experiment.

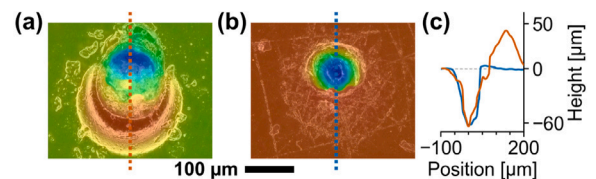


Fig. 11. Representative indentations on acrylic (a) without and (b) with polishing. (c) Height profiles along the dashed lines show burr removed by polishing.

overhead camera are shown in [Fig. 10](#) and show the evolution of the patterning over time. As expected, the indentation robots quickly move to their respective target areas and are then followed by the polishing robots. Each polishing robot tends to stay in one area, though there are times when the polishing robots swap or they momentarily travel to the other area. This behavior is expected because there are no barrier functions to tightly restrict the robots to certain areas. Qualitatively, the robots successfully achieve the patterning objective.

3.5. Experimental polishing performance

The polishing quality was evaluated using both microscope measurements and the roughness heuristic. A Keyence VHX-7000 focus-variation microscope measured the height profile of two representative

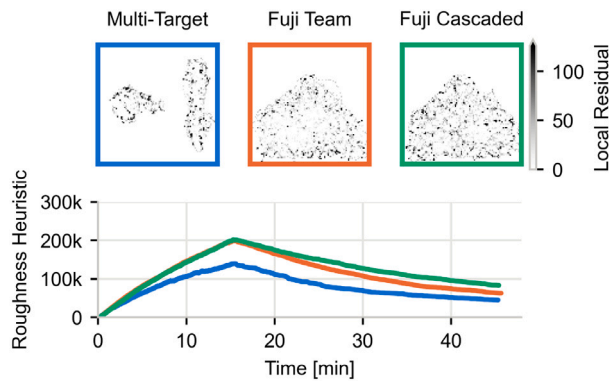


Fig. 12. Roughness heuristic fields for the different experiments (top) and the total over time (bottom).

indentations, one with and one without polishing, shown in Fig. 11. An approximately 50 μm burr is shown around the indentation and is shown to be removed after 2 s of polishing.

Due to the limited measurement area of the microscope used to measure individual indentation profiles, it was not practically feasible to inspect the roughness of the entire surface. For that reason, the roughness heuristic was evaluated on the time-series position data for the robots during the three experimental runs. The result is shown in Fig. 12. The top three fields show the local remaining roughness, binned into a 100×100 grid. The bottom plot shows the total roughness heuristic over time, generally matching the trends from Fig. 6. In Experiment 1, cascaded control polishes slightly worse than team-communication. In Experiment 2, the robots are confined to a smaller area, leading to a smaller peak roughness. The robots likely should have run for longer than the 45 min suggested by simulation to eliminate more of the regions with remaining roughness.

3.6. Discussion

The simulation and experimental results shown in the prior sections confirm that cascaded ergodic control effectively guides polishing robots based on the trajectory history of indentation robots rather than a predefined target distribution. We compare to independent and team-communication ergodic schemes rather than boustrophedon (raster scanning or “lawn mower”) schemes because cascaded ergodic control aims to increase flexibility rather than strict patterning efficiency. Other techniques require pre-specifying additional information, such as number of robots and their starting positions, or cannot handle disruptions like the loss of a robot. The results shown in Figs. 6 and 9 indicate that this flexibility does not come at the cost of performance. Cascaded ergodic control does not reduce collisions, improve the ergodic metric, or improve processing efficiency compared to team-communication control; rather, its advantage is removing the need to communicate the target distribution to the polishing robots, enabling run-time task flexibility demonstrated in Simulation 2 and Experiment 2 that team-communication control cannot provide.

While effective, cascaded ergodic control has tradeoffs. For instance, cascaded ergodic control does not use predefined barrier functions. If robots must always stay within only one target distribution, then cascaded ergodic control is not appropriate. Additionally, cascaded ergodic control requires that the guiding robots have sufficient time to approximate the target distribution. If these robots are removed too early, then the following robots will have a poor representation of their own target. These issues were not a problem in our experiments, where the polishing process is not safety critical and the indentation robots were given enough time to approximate their target distributions. Cascaded control does not guarantee a particular ordering; indentation

robots were removed while polishing robots remained to guarantee that unpolished indentations would stop being produced.

The Fourier representation used here does not inherently reduce patterning quality and can be made arbitrarily high-fidelity by increasing the number of basis functions: an image with $N \times M$ pixels is represented exactly with $N \times M$ basis functions. $K \ll N$ basis functions were used in this work to make computations feasible on the low-cost microcontroller; future robots with greater onboard computational capacity could use more basis functions (see [15] for a detailed discussion on this limitation). The Fourier representation is critical for its analytic gradients that enable optimal control. Even with reduced fidelity, the present work targets flexible, scalable patterning rather than high-precision point-to-point control, specifically applications such as drag-reduction texturing on cargo ships and aircraft which do not require tight local tolerances.

3.7. Future work

Cascaded ergodic control could be extended to 3D workpieces; more complex, curved substrates, like the exteriors of cargo ships, airplanes, or the surfaces of large reflective optics, could be handled with triangular meshed surfaces [18]. This controller considered both the indentation tool and polishing tool to be points, however, the true size and non-uniform material removal characteristics of the polishing tool could be incorporated using volumetric basis functions [19]. These changes would increase the capability of ergodically-controlled polishing and could be incorporated into existing robotic-arm polishing setups.

Additionally, selecting the optimal number of indentation and polishing robots for a given task is complex and dependent on desired speed of completion, indentation rate, polishing rate, robot size, target geometry, and target indentation area density, among other parameters. A heuristic to guide this decision could be developed in future work. Generally, more indentation robots are needed than polishing robots because of the difference in processing rates. If an indentation area density of 10% was targeted, for example in a friction-reduction application, ratios of 5 to 1 may be practically reasonable for the robots demonstrated here.

4. Conclusion

The key contribution of this work is showing that the trajectory information from one group can successfully be used as the input for another group using ergodic control, with sequential processing of functional surfaces demonstrating real-world applicability. Previous works required all targets to be loaded in advance and only demonstrated a single type of processing [15,16]. Cascaded ergodic control allows coordinated patterning by mobile robots without initial knowledge of the desired pattern, so long as some robots can “guide the way”. Robots can therefore self-allocate without explicit task knowledge or manual intervention. Simulation and experiments verify that cascaded ergodic control performs comparably on polishing tasks to existing methods, while adding additional flexibility. Such a technique could be used in the future for the patterning of large workpieces, like cargo ships or airplanes, which are inaccessible to traditional patterning techniques. Beyond surface patterning, any coverage task with heterogeneous agents and multiple, sequential tasks could apply cascaded ergodic control as a flexible control strategy.

CRedit authorship contribution statement

Malachi Landis: Conceptualization, Methodology, Software, Investigation, Writing – original draft, Visualization. **Annalisa T. Taylor:** Methodology, Software. **Todd D. Murphey:** Methodology, Project administration, Funding acquisition. **Ping Guo:** Conceptualization, Methodology, Writing – review & editing, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research was supported by the National Science Foundation under grant number CNS-2229170 and the Army Research Office under grant number W911NF-22-1-0286. Malachi Landis is supported by the National Defense Science and Engineering Graduate (NDSEG) Fellowship under grant number FA9550-23-F-0014. This material is based upon work supported by the Air Force Office of Scientific Research under grant number FA9550-23-F-0014 in the amount of \$134,600.

Appendix. Sensitivity to position measurement and control errors

The robot true positions and wheel velocity inputs are both subject to error, either from robot motion and frame dropping (for the former) or wheel slip/stalling (for the latter). To assess the sensitivity of the general ergodic control algorithm to these effects, we performed simulated sensitivity studies using the single-robot patterning originally shown in Fig. 4 over twice the duration to allow the ergodic metric to reach steady-state. Other parameters are identical.

Two studies were performed. In the first, the effect of position feedback noise was assessed by applying various levels (from 0 to ± 25 mm) zero-mean Gaussian noise to the position used by the robot to compute controls. The results are shown in Fig. 13. Noise of $\sigma = 0.5$ mm (much worse than the experimental AprilTag repeatability) had no significant effect on the ergodic metric, and $\sigma = 10$ mm was required to cause an order-of-magnitude increase. Qualitatively, higher-noise trajectories became darker and shrunken on their borders; this was not observed in experiment.

In the second study, motor slip/stall was approximated by applying a lower limit to motor controls used by the robot. A control u between $-u_{lim}f < u < u_{lim}f$ would be “stalled” to zero, where u_{lim} is the control limit and f is the stall threshold between $0 < f < 1$. This effectively means “fine” motor controls do not move the robot, depending on the chosen threshold, which varied from 0 to 0.9. The results are shown in Fig. 14. Patterning quality degrades with increasing stall threshold,

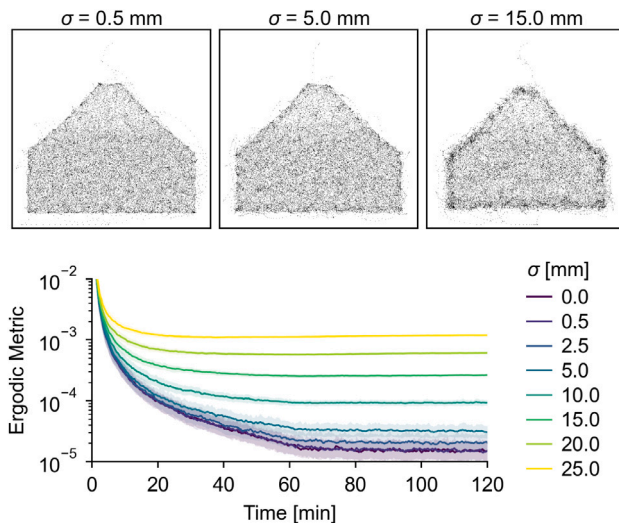


Fig. 13. (top) Representative trajectories for three different applied noise levels. (bottom) The ergodic metric over time showing increasing degradation with added noise.

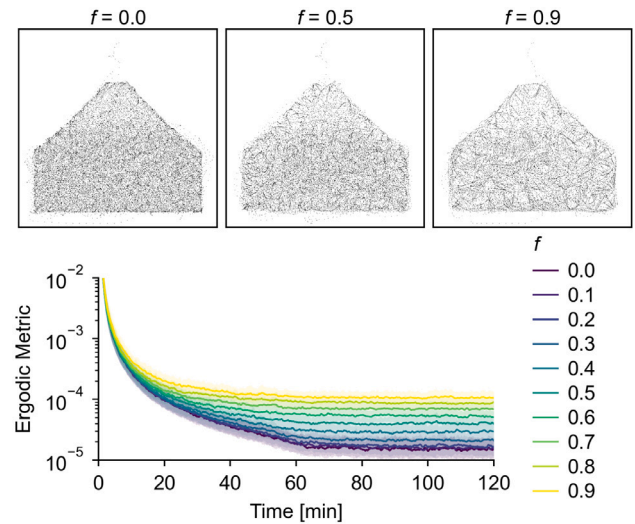


Fig. 14. (top) Representative trajectories for three different applied stall levels. (bottom) The ergodic metric over time showing increasing degradation with increasing stall threshold.

but the overall pattern is still produced; the maximum stall threshold causes a one order of magnitude increase in the metric. Qualitatively, higher stall levels cause “streaks” in the trajectory, appearing less smooth (i.e., high spatial frequency errors); this does seem to appear in experiment.

The multi-scale nature of the ergodic metric (specifically the Λ term) inherently favors low-spatial-frequency contributions. Large velocities will traverse larger distances in a similar amount of time (affecting lower spatial frequencies) compared to lower velocities (affecting higher spatial frequencies). Thus, patterns made without “fine” control can still produce the overall desired patterns but at the cost of high-frequency details.

References

- [1] Li J, Zhang T, Liu X, Guan Y, Wang D. A survey of robotic polishing. In: 2018 IEEE international conference on robotics and biomimetics. ROBIO, IEEE; 2018, p. 2125–32.
- [2] Wang G, Zhang C, Liu S, Zhao Y, Zhang Y, Wang L. Multi-robot collaborative manufacturing driven by digital twins: Advancements, challenges, and future directions. *J Manuf Syst* 2025;82:333–61.
- [3] Zhou J, Zheng L, Fan W. Multirobot collaborative task dynamic scheduling based on multiagent reinforcement learning with heuristic graph convolution considering robot service performance. *J Manuf Syst* 2024;72:122–41.
- [4] Wang Z, Ye R, Xiang J. The performance of textured surface in friction reducing: A review. *Tribol Int* 2023;177:108010.
- [5] Kreder MJ, Alvarenga J, Kim P, Aizenberg J. Design of anti-icing surfaces: Smooth, textured, or slippery? *Nat Rev Mater* 2016;1(1):1–15.
- [6] Halder P, Hossain N, Pramanik BK, Bhuiyan MA. Engineered topographies and hydrodynamics in relation to biofouling control—A review. *Environ Sci Pollut Res* 2021;28:40678–92.
- [7] Parker LE, Rus D, Sukhatme GS. Multiple mobile robot systems. In: Springer handbook of robotics. Springer; 2016, p. 1335–84.
- [8] Tam H-Y, Lui OC-H, Mok AC. Robotic polishing of free-form surfaces using scanning paths. *J Mater Process Technol* 1999;95(1–3):191–200.
- [9] Xiao M, Ding Y, Yang G. A model-based trajectory planning method for robotic polishing of complex surfaces. *IEEE Trans Autom Sci Eng* 2022;19(4):2890–903.
- [10] Tam H-Y, Cheng H, Dong Z. Peano-like paths for subaperture polishing of optical aspherical surfaces. *Appl Opt* 2013;52(15):3624–36.
- [11] Rososhansky M, Xi FJ. Coverage based tool-path planning for automated polishing using contact mechanics theory. *J Manuf Syst* 2011;30(3):144–53.
- [12] Andulkar MV, Chiddarwar SS, Marathe AS. Novel integrated offline trajectory generation approach for robot assisted spray painting operation. *J Manuf Syst* 2015;37:201–16.
- [13] Taylor AT, Landis M, Wang Y, Murphey TD, Guo P. Image to patterning: Density-specified patterning of micro-structured surfaces with a mobile robot. In: 2024 IEEE/RSJ international conference on intelligent robots and systems. IROS, IEEE; 2024, p. 2264–70.

- [14] Mathew G, Mezić I. Metrics for ergodicity and design of ergodic dynamics for multi-agent systems. *Phys D* 2011;240(4–5):432–42.
- [15] Landis M, Jia M, Taylor AT, Murphey TD, Guo P. Large-scale functional patterning using mobile robot swarms and ergodic control. *CIRP Ann — Manuf Technol* 2025;(74):511–5.
- [16] Taylor AT, Landis M, Guo P, Murphey TD. Manufacturing micro-patterned surfaces with multi-robot systems. 2026, arXiv preprint.
- [17] Miller LM, Murphey TD. Trajectory optimization for continuous ergodic exploration. In: 2013 American control conference. IEEE; 2013, p. 4196–201.
- [18] Dong D, Xu A, Gutow G, Choset H, Abraham I. Ergodic exploration over meshable surfaces. In: IEEE international conference on robotics and automation. ICRA, 2025, p. 13088–94.
- [19] Kwon J, Sun MM, Murphey TD. Volumetric ergodic control. 2025, arXiv preprint.