2020 IEEE International Conference on Robotics and Automation (ICRA)
31 May - 31 August, 2020. Paris, France

Reality as a simulation of reality: robot illusions, fundamental limits, and a physical demonstration

Dylan A. Shell and Jason M. O'Kane

***Abstract*—We consider problems in which robots conspire to present a view of the world that differs from reality. The inquiry is motivated by the problem of validating robot behavior *physically* despite there being a discrepancy between the robots we have at hand and those we wish to study, or the environment for testing that is available versus that which is desired, or other potential mismatches in this vein. After formulating the concept of a convincing illusion, essentially a notion of system simulation that takes place in the real world, we examine the implications of this type of simulability in terms of infrastructure requirements. Time is one important resource: some robots may be able to simulate some others but, perhaps, only at a rate that is slower than real-time. This difference gives a way of relating the simulating and the simulated systems in a form that is relative. We establish some theorems, including one with the flavor of an impossibility result, and providing several examples throughout. Finally, we present data from a simple multi-robot experiment based on this theory, with a robot navigating amid an unbounded field of obstacles.**

"Truth is beautiful, without doubt; but so are lies."— Ralph Waldo Emerson

I. Motivation and Overview

Robotics papers usually include evidence of algorithms or controllers that have been executed or evaluated on some kind of system, typically comprising either physical robots or a substitute. But what constitutes a robot demonstration, exactly? One division is generally drawn between software simulation and real robots. This is, at best, a rather rough distinction for there is a spectrum of simulators spanning a wide range of fidelities. What is certain is that there are more choices, between full software simulation and full physical implementation, than are generally recognized or garner attention. Inasmuch as this is critical for robotics as a scientific enterprise, it is perhaps curious that there has been little formal treatment of representativeness or verisimilitude beyond the complete hardware and software extremes, and their consideration. This paper's *raison d'être* is to initiate a close, systematic examination of these other options. Fig. 1 gives a simple example of robots engaging in fakery of the kind we are interested in.

We want to understand how one physical system may be used to mimic the behavior of another. By system, we are considering a setting where observations are made (via sensors) and used to choose actions that are effected (via actuators) and this unfolds over time. We begin with a simplified discrete-time setting (Definition 1) where we can contemplate exact emulation (Definition 2), rather than considering approximate or imprecise imitation. The central features which distinguish the approach from other formalisms of emulation between robot systems (see Section II) are the possibility of variable time expansion (somewhat akin to Milner's weak bisimulation [19]) and a narrow focus on mimickry only up to the perceptual capabilities of the system under emulation.

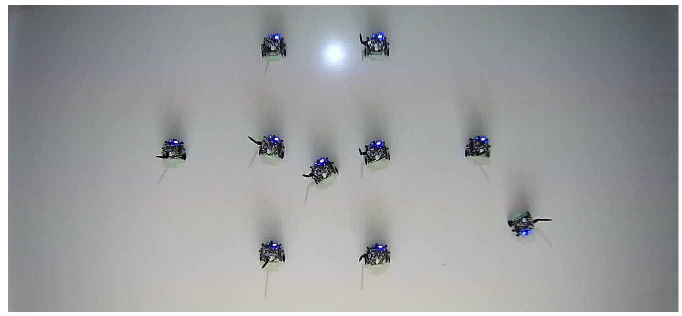

Fig. 1. A collection of 10 robots conspiring to give the illusion of a single robot moving through a field of obstacles. Robot 1 (center) remains motionless. Eight other robots play the roles of eight nearby obstacles. A tenth robot (far right) remains out of view. This allows Robot 1 to receive sensor readings consistent with motion along a path traversing a very, very large field of obstacles. (This illusion is examined, later, as Example 5.)

We then formulate some particular questions, such as: "What are the resources involved, how do we quantify resource requirements, and relate them?" (Definition 3, Theorem 4), "How do we compose or nest such systems?" (Theorem 2), "What happens to these things when systems are modified (Theorem 3)", etc.

In terms of immediate utility for the practitioner, the present paper shows how to conduct a novel sort of emulation with real hardware where sensors, rather than being faked out of whole cloth—as is usually done with computational or mathematical models that are highly idealized coarse approximations—provide real signals. As the instances we study herein show, there may be considerable freedom in choosing different ways to emulate one system with another, with implications for future robotic laboratory infrastructure.

II. Related work

A. Animal studies: The inspiration for the present work

For decades, biologists have sought to chart the perceptual limits of organisms and to understand how informational mismatches affect behavior [8, 38]. Recent years have seen virtual and augmented reality technologies being used in this quest [39]. The animals studied range from small mammals [11] down to insects [35], being studied both while walking [34] and flying [9, 10, 16]. As a concrete example, Takalo et al. constructed a laboratory apparatus comprising a spherical projection surface and a track ball that enables the detailed study of the walking behavior of the cockroach *(periplaneta americana)* by giving it synthetic visual stimuli [34].

978-1-7281-7395-5/20/$31.00 ©2020 IEEE

B. Practical simulation in software

Software simulations are an inescapable part of the current robotics research landscape, with the community devoting much time and attention to related questions, including through the biennial SIMPAR conference. The software traces out some element of a robot's execution in a virtual (rather than physical) world, generating artificial sensor readings (or sometimes state information), and evolving the robot system forward in time.

Center to most discussions about software simulation are considerations of fidelity: How closely does the simulator mimic the real world? High-fidelity simulation software like Gazebo [13] has been developed to account for many of the complications experienced by the complex robots native to many research labs. But fidelity may be traded for other features as some efforts strike a "useful balance between fidelity and abstraction" [37]. Other simulators, designed for specific robot types [5, 6, 12, 25, 28, 31], optimization/control schemes [36], and application domains [2, 21, 29], exist.

This work is partly a generalization of the traditional notion of robot simulation, but with elements of the simulation conducted physically rather than virtually. Closely related work includes endeavors that alter aspects of the physical world using mixed or augmented reality techniques [1, 3, 4, 32]. The distinguishing feature here is that modifications of the world are made by robots and for robots.

C. Simulation as theoretical concept

Relating systems by the fact that they can simulate each other, for some definition of simulation, is a recurring theoretical theme. The symmetric notion, where two systems are each able to match the other, yields the concept of bisimulation, which is an equivalence relation. Bisimilarity was identified independently in modeling concurrent systems [18] and in modal logic [24]; it also has a game theoretic interpretation [33].

Closer to home in robotics, invariants among sensori-computation circuits of Donald [7], and the dominance relation between robot systems introduced by O'Kane and LaValle [23], bear parallels to the notion of illusion we introduce here, particularly in the use of one system, or re-arrangements of the resources contained therein, to emulate certain properties of another. In this paper, the emphasis on perceptual equivalence for the robots participating in the illusion is fresh.

III. Preliminary definitions

A. Systems

We wish to talk about relationships between pairs of systems of robots. First, then, we need to define the notion of a system. Because henceforward we shall consider systems consisting of possibly many robots, we jump directly into definitions that consider (potentially) multiple robots. Superscripts in parentheses denote robot indices; subscripts are time indices.

Definition 1. *A* deterministic multi-robot transition system *is a 7-tuple* (n, X, U, f, Y, h, x_0)*, in which*

1) *n is a positive integer identifying the number of robots,*
2) *$X = X^{(1)} \times \cdots \times X^{(n)}$ denotes a state space, composed of individual state spaces for each robot,*
3) *$U = U^{(1)} \times \cdots \times U^{(n)}$ denotes an action space, composed of individual action spaces for each robot,*
4) *$f : X \times U \to X$ is a state transition function, defined in terms of transition functions $f^{(1)}, \ldots, f^{(n)}$ for each robot, so that*
$$f\left((x^{(1)}, \ldots, x^{(n)}), (u^{(1)}, \ldots, u^{(n)})\right) = \left(f^{(1)}(x, u^{(1)}), \ldots, f^{(n)}(x, u^{(n)})\right).$$
5) *$Y = Y^{(1)} \times \cdots \times Y^{(n)}$ denotes an observation space, composed of individual observation spaces,*
6) *$h : X \to Y$ is an observation function, defined in terms of observation functions $h^{(1)}, \ldots, h^{(n)}$ for each robot, so that $h(x) = \left(h^{(1)}(x), \ldots, h^{(n)}(x)\right)$,*
7) *$x_0 \in X$ is the system's initial state.*

Such a system evolves in discrete time steps subject to:

$$x_{k+1} = f(x_k, u_k), \quad \text{and} \quad y_k = h(x_k).$$

A few simple examples, to be revisited later, illustrate the idea.

Example 1. Consider a caravan of n autonomous vehicles moving down a long single-lane roadway. Suppose each robot can control its own velocity, subject to some upper and lower bounds, and can also measure the distance to the other robots immediately in front of and behind itself. See Fig. 2. We describe this as a deterministic multi-robot transition system

$$S_{n,v_{\min},v_{\max}} = (n, \mathbb{R}^n, [v_{\min}, v_{\max}]^n, f, \mathbb{R}^+ \times \mathbb{R}^+, h, x_0), \quad (1)$$

for which we'll give the state transition function f and observation h shortly. Here elements of the state space $X = \mathbb{R}^n$ encode the position, along the one-dimensional roadway, of each of the n robots. At each time step k, the action $u_k^{(i)} \in [v_{\min}, v_{\max}]$ of robot i denotes the velocity of that robot at that time. Thus, we may define $f(x, u) = x + u$. We assume that $v_{\min} < v_{\max}$. Each observation $y_k^{(i)} \in \mathbb{R}^+ \times \mathbb{R}^+$ is a pair of integers indicating the distance to the closest other robot, if any, in each direction:

$$y_k^{(i)} = h^{(i)}(x_k) = \left(\min\left(\left\{|x_k^{(j)} - x_k^{(i)}| \,\middle|\, x_k^{(j)} < x_k^{(i)}\right\} \cup \{\infty\}\right), \min\left(\left\{|x_k^{(j)} - x_k^{(i)}| \,\middle|\, x_k^{(j)} > x_k^{(i)}\right\} \cup \{\infty\}\right)\right).$$

To refer to the individual measurements in a single observation, we use the symbols b and a for the distances behind and ahead, so that $y_k^{(i)} = (b_k^{(i)}, a_k^{(i)})$. Finally, the initial state x_0 is some known but arbitrary state.

Notice that (1) is, in fact, defining an infinite family of systems, parameterized by the number of vehicles in the system and the ranges of allowable velocities.

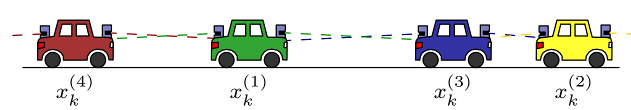

Fig. 2. An example of the sort of system in Example 1, with $n = 4$. At each time step k, each robot is at some point $x_k^{(i)}$ along the roadway moving with velocity $u_k^{(i)}$, and measures the distances $y_k^{(i)}$ to the adjacent robots.

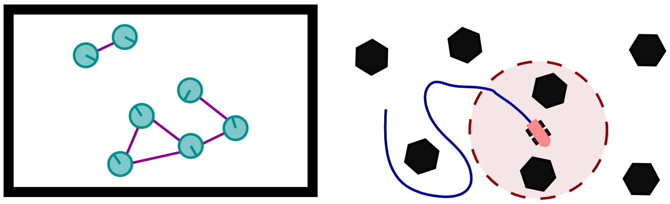

Fig. 3. [left] A team of simple robots in a bounded environment, as in Example 2. [right] A single robot moving in a unbounded field of obstacles, as in Example 3.

This is, of course, a heavily idealized model of caravaning autonomous vehicles, crafted as an elementary illustration of Definition 1. Richer models might, for example, expand X to model multi-lane roadways or the robots' lateral positions within the lanes, enrich U and f to model the dynamics of some physical system more faithfully, or modify Y and h to model, say, a LIDAR sensor with greater fidelity.

Example 2. Consider a system in which many small disk-shaped differential drive robots move in a shared, bounded, planar workspace, with each robot aware of the relative positions of the other robots within some small sensor range. Refer to Fig. 3[left]. One might realize this kind of system using, for example, Khepera [20], r-one [17], or GRITSbot [26] robots. We can model such a system by choosing the number of robots n, the rectangular workspace $W \subseteq \mathbb{R}^2$, the maximum wheel velocity $v_{\max}$, and the sensor range r. We then define

$$S_{\text{disks}} = (n, X_{\text{disks}}, U_{\text{disks}}, f_{\text{disks}}, Y_{\text{disks}}, h_{\text{disks}}, x_0)\,, \tag{2}$$

in which the states in $X_{\text{disks}} = (W \times S^1)^n$, the actions in $U_{\text{disks}} = [-v_{\max}, v_{\max}]^2$ denote the left and right wheel velocities for each robots, the state transition function f_{disks} encodes the well-known kinematics for differential drive robots, the observations in $Y_{\text{disks}} = \cup_{i=0}^{n-1}(\mathbb{R}^2)^n$ are lists of between 0 and $n-1$ planar positions, the observation function $h^{(i)}_{\text{disks}}$ for each robot i returns a list of the relative positions of any other robots within distance r of robot i, and the initial state $x_0 \in X_{\text{disks}}$ is a known but arbitrary state.

Example 3. Definition 1 is also suitable for describing single-robot systems as a particular case with $n = 1$. For example, a velocity-controlled robot moving in a very large field of nearly-identical static obstacles, with a sensor to detect those obstacles when they are nearby, might be modelled as

$$S_{\text{single}} = (1, X_{\text{single}}, U_{\text{single}}, f_{\text{single}}, Y_{\text{single}}, h_{\text{single}}, x_0)\,, \tag{3}$$

with $X_{\text{single}} = \mathbb{R}^2$, $U_{\text{single}} = [-v_{\max}, v_{\max}]$, and $f_{\text{single}}(x, u) = x + u$. The observation space Y_{single} and h_{single} may be defined to return the locations of the center points of each obstacle. See Fig. 3[right].

B. Policies

In the model, a robot operates by choosing actions to execute, a concept detailed via a policy. The essential question in formalizing policies is to determine what information is used by the robot in considering its action. Now, to define the policy concept, we adopt the style of LaValle's book [15].

We begin, first, with something simple that will turn out to be inadequate for our needs. If robot i, at time step k, has sufficient information that it can determine its state, i.e., it is a fully observable problem, then its policy $\pi^{(i)}$ might be defined as a function of that state:

$$x_k^{(i)} \xmapsto{\pi^{(i)}} u_k^{(i)}. \tag{4}$$

More likely, the robot will only have access to its history of actions and observations to select its action

$$u_0^{(i)}, \ldots, u_k^{(i)}, y_0^{(i)}, \ldots, y_k^{(i)} \xmapsto{\pi^{(i)}} u_k^{(i)}. \tag{5}$$

In what follows, one robot system will seek to present some view of the world to match a description as will be seen by some other, secondary system. This primary system must know some aspects of that other system to fool it effectively. That is, the primary system must be aware of the 'fourth wall' and know some of the expectations and qualities on the other side of it. Throughout, we use a notational convention: we distinguish the primary system (initially best thought of as the physical system) by placing a hat over its variables; all variables for the secondary are bare. Now, returning to our formalization of the policy concept, we must generalize the notation so far in order for it to present information about the primary system and a secondary one, partitioned like such

$$\underbrace{\widehat{u}_0^{(i)}, \ldots, \widehat{u}_k^{(i)}}_{\substack{i\text{'s action}\\ \text{history}}}, \underbrace{\widehat{y}_0^{(i)}, \ldots, \widehat{y}_k^{(i)}}_{\substack{i\text{'s observation}\\ \text{history}}}, \underbrace{x_0, \ldots, x_\ell}_{\substack{\text{Whole other}\\ \text{system's state}\\ \text{history}}} \xmapsto{\widehat{\pi}^{(i)}} \widehat{u}_k^{(i)}. \tag{6}$$

Note that the hatted variables in the domain are labelled from 0 to k, while the naked variables extend to ℓ. This models the fact that the primary and second systems may operate at different time scales. Immediately, one sees other variations that are possible, such as instances when $\widehat{\pi}^{(i)}$ uses only the last element (x_ℓ) of the secondary system's state. Or, when the primary robots may communicate, the (i) superscripts may be dropped when we consider the multi-robot system globally. For simplicity, we restrict our attention in this paper only to the basic case. In what follows, the term *robot policy* refers to a function of the form in (6).

C. Illusions

Definition 2. *For deterministic multi-robot transition systems $S = (n, X, U, f, Y, h, x_0)$, and $\widehat{S} = (\widehat{n}, \widehat{X}, \widehat{U}, \widehat{f}, \widehat{Y}, \widehat{h}, \widehat{x}_0)$, and integer $0 < m \leq n$, we say that $\widehat{S}$ is an m-*illusion *of S if there exist*

(i) *robot policies $\widehat{\pi}^{(1)}, \ldots, \widehat{\pi}^{(\widehat{n})}$ in $\widehat{S}$,*
(ii) *a strictly increasing function $z : \mathbb{Z}^+ \to \mathbb{Z}^+$, and*
(iii) *an infinite series of functions $\rho_k : \mathbb{Z}_m \to \mathbb{Z}_{\widehat{n}}$,*

for any robot policies $\pi^{(1)}, \ldots, \pi^{(n)}$ in S, such that for all $k \geq 0$ and all $1 \leq i \leq m$, we have

$$h^{(i)}(x_k) = \widehat{h}^{(\rho_k(i))}\left(\widehat{x}_{z(k)}\right). \tag{$\star$}$$

Further, if $\widehat{S}$ is an m-illusion of S, then a tuple of robot policies, mapping functions, and a time scaling function

$(\widehat{\pi}, (\rho_1, \rho_2, \ldots), z)$ *that ratifies the definition of illusion is called a* witness *to that illusion.*

The preceding definition warrants some dissection.

1) We understand the system S to be the secondary one, i.e., the one that we intend to emulate. The system $\widehat{S}$ is the physical system whose execution will be orchestrated to appear, in the perception of some of its robots, to operate in the same manner as S.[1]
2) The positive integer parameter m is the number of robots in S that are recipients of the illusion, whom we dub the participant robots. To simplify the notation, we will assume without loss of generality that the first m robots in S, according to their indices, are the participants. (One might also expect for $m \leq \bar{n}$ always, as it seems that the number of participant robots cannot exceed the number of robots in the system; in fact, this need not be so.)
3) The $\widehat{S}$ robot policies $\widehat{\pi}^{(i)}$ described in condition (i) govern the movements of the robots in that system.
4) The function z from condition (ii) establishes the relationship between the time scales of the two systems, so that $z(k)$ defines the physical time step in $\widehat{S}$ corresponding to time step k in S.
5) The functions ρ_k from condition (iii) indicate, for each time step k of the execution in S, which robots of $\widehat{S}$ play the roles of each of the participant robots in S.

Pulling these elements together, the constraint marked $(\star)$ requires, at each time step in S, that every participant robot is mapped, via the ρ function for that time step, to a robot in $\widehat{S}$ that experiences the same observation in that system as the mapped robot should experience in S. A few examples follow.

Example 4. Recall the autonomous caravan systems introduced in Example 1. For any such system $S = S_{n,v_{\min},v_{\min}}$, we can form a 1-illusion from any system of the form $\widehat{S} = S_{3,\widehat{v}_{\min},\widehat{v}_{\max}}$. This holds regardless of the number n of robots in S and of the range of actions $[\widehat{v}_{\min}, \widehat{v}_{\max}]$ available to each robot in $\widehat{S}$.

One way to construct such an illusion is to select a policy $\widehat{\pi}$ in which robot 1 moves at a constant speed $(\widehat{v}_{\min} + \widehat{v}_{\max})/2$. The other two robots, knowing the desired observation $y_k^{(1)} = (a_k^{(1)}, b_k^{(1)})$ from S, position themselves on opposite sides of robot 1, moving as fast as possible at each stage in $\widehat{S}$ toward positions where $\widehat{x}_{\widehat{k}}^{(1)} - \widehat{x}_{\widehat{k}}^{(2)} = b_k^{(1)}$ and $\widehat{x}_{\widehat{k}}^{(3)} = \widehat{x}_{\widehat{k}}^{(1)} = a_k^{(1)}$. To satisfy the remaining conditions of Definition 2, define z to return the time when robots 2 and 3 in $\widehat{S}$ have reached their target positions, and the sequence of mapping functions $\rho_k : \{1\} \to \{1, 2, 3\}$ as a constant series of functions, under which $1 \stackrel{\rho_k}{\longmapsto} 1$ for all k. See Fig. 4.

Example 5. Recall the system S_{single} introduced in Example 3. Suppose there exists an upper bound m on the number of obstacles visible from—that is, within distance r of—any position that the robot might reach. Then S_{disks}, from Example 2, is a 1-illusion for S_{single}, provided that it has at least $m+1$ robots, its workspace W is large enough to contain a circle of radius r, and the sensing range in S_{disks} is no smaller than the sensing range in S_{single}.

One way to achieve this illusion is to select robot 1 in S_{disks} to act as the recipient of the observations as required by $(\star)$. This robot remains motionless at the center of the physical workspace W. At each stage k in S_{single}, the desired observation $y_k^{(1)}$ is a list of positions at which robot 1 should perceive obstacles. We choose a policy $\widehat{\pi}$ that directs the some of the remaining $\widehat{n} - 1$ robots to those positions relative to robot 1, and directs the remaining robots to positions beyond its sensing range. See Fig. 5. Many different policies, with varying degrees of time efficiency, can achieve this.

[1] Occasionally human illusionists opt for for a certain type of stereotypical headwear (🎩). Likewise, our convention uses notation with hats ($\widehat{\ }$) to refer to systems whose robots are performing an illusion. The parallel is unintentional but perhaps nonetheless a useful aid to understanding.

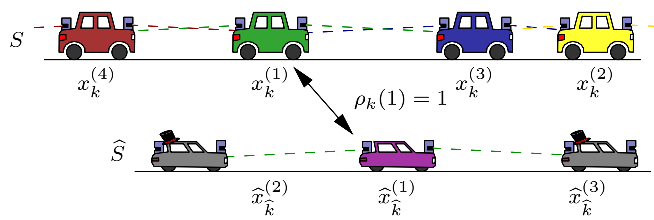

Fig. 4. An illustration of Example 4. A system of three vehicles reproduces the observations expected in a system with potentially many more robots.

Next, we consider the execution time in the primary system as a resource cost in which we are interested.

Definition 3. *If $\widehat{S}$ is an m-illusion of S with witness $(\widehat{\pi}, (\rho_1, \rho_2, \ldots), z)$, then the illusion is an (m, τ)-illusion if the sequence*

$$z(2) - z(1), z(3) - z(2), z(4) - z(3), \ldots$$

is bounded above by τ. The constant τ, which we can take to be an integer owing to the definition of z, is called the slowdown *of the illusion.*

In broad terms, we may then consider $\frac{1}{\tau}$, the inverse slowdown, to be time efficiency of an illusion.

Example 6. Recall Example 4. That illusion has slowdown $\lceil 2(v_{\max} - v_{\min})/(\widehat{v}_{\max} - \widehat{v}_{\min}) \rceil$.

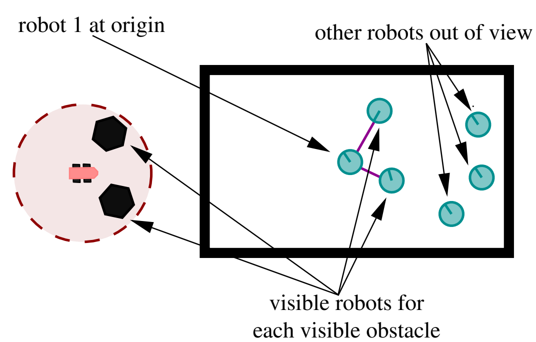

Fig. 5. A 1-illusion of S_{single} using S_{disks}.

IV. Basic Properties of Illusions

Definition 2 provides a foundation for understanding the notion of one system presenting an illusion of another. Next, we present some results that follow from that definition. As an initial sanity check, we show that a system does indeed present a faithful and efficient illusion of itself.

Theorem 1 (identity)**.** *A deterministic multi-robot transition system $S = (n, X, U, f, Y, h, x_0)$ is an (n,1)-illusion of S.*

Proof: We observe that, if z and ρ_k are taken as identity functions, then ($\star$) holds when $\widehat{\pi}^{(i)} = \pi^{(i)}$. ∎

Considering the preceding theorem, one might wonder whether a stronger statement ought to be made, to the effect that every S can provide an m-illusion of itself for any $m < n$. That statement is absent because it is false. Supposing $m + p = n$ with $p > 0$, then there are p robots that may show up under h. Additional properties of h are needed to ensure that the p robots can be made invisible.

With additional assumptions on the dynamics of S, i.e., if the system can be made to either loiter or affect state changes more slowly, then an (n,j)-illusion with $j > 1$ is also possible.

Rather more interesting is the nesting of systems:

Theorem 2 (composition)**.** *If $\widehat{\widehat{S}} = (\widehat{\widehat{n}}, \widehat{\widehat{X}}, \widehat{\widehat{U}}, \widehat{\widehat{f}}, \widehat{\widehat{Y}}, \widehat{\widehat{h}}, \widehat{\widehat{x}}_0)$, is an $(\widehat{n}, \widehat{\tau})$-illusion of $\widehat{S} = (\widehat{n}, \widehat{X}, \widehat{U}, \widehat{f}, \widehat{Y}, \widehat{h}, \widehat{x}_0)$, and $\widehat{S}$ is an (m, τ)-illusion of $S = (n, X, U, f, Y, h, x_0)$, then $\widehat{\widehat{S}}$ is an $(m, (\tau \cdot \widehat{\tau}))$-illusion of S.*

Proof: Omitted due to space limitations. See [30].

Note that, in (iii), function composition requires that $\widehat{\widehat{S}}$ be an $\widehat{n}$-illusion of $\widehat{S}$ in order for the types to agree. If $\widehat{\widehat{S}}$ were only an $\widehat{m}$-illusion of $\widehat{S}$ with $\widehat{m} < \widehat{n}$, then the $\widehat{n} - \widehat{m}$ extra robots are needed to create an illusion for S. This arises because we do not talk of some subset of robots in one system sufficing to provide an illusion of another system, since all the primary robots need to participate to ensure the illusion succeeds, even if participating constitutes moving to ensure they're unobserved, ruining the illusion otherwise.

Illusions hold up to the set of observations made in the secondary system. One might expect that $Y \subseteq \widehat{Y}$ but, in fact, Y may be larger or smaller, though the pair cannot be disjoint. It is not the range but the image which matters:

Definition 4. *The* perceptual occurrence *of deterministic system $S = (n, X, U, f, Y, h, x_0)$, is the subset of Y, denoted Y^{occ}, that is produced under h via states reachable by some robot policies $\pi^{(1)}, \ldots, \pi^{(n)}$.*

In Definition 2, requirement ($\star$) implies that if $(\widehat{n}, \widehat{X}, \widehat{U}, \widehat{f}, \widehat{Y}, \widehat{h}, \widehat{x}_0)$ is an illusion of (n, X, U, f, Y, h, x_0), then $Y^{\text{occ}} \subseteq \widehat{Y}^{\text{occ}}$.

Now we might inquire as to the implications for illusions under alteration of the robots' sensors. We model potential degradation, or preimage coarsening, of sensors via a function in the observation space, where non-injective transformations will conflate things that were distinguishable formerly.

Theorem 3 (coarser observations)**.** *If $(\widehat{n}, \widehat{X}, \widehat{U}, \widehat{f}, \widehat{Y}, \widehat{h}, \widehat{x}_0)$ is an illusion of (n, X, U, f, Y, h, x_0), then, for any function $\kappa : Y \cup \widehat{Y} \to Z$, we have that $(\widehat{n}, \widehat{X}, \widehat{U}, \widehat{f}, Z, \kappa \circ \widehat{h}, \widehat{x}_0)$ is an illusion of $(n, X, U, f, Z, \kappa \circ h, x_0)$.*[2]

Proof: The original witness ratifies the new illusion, since

$$h^{(i)}(x_k) = \widehat{h}^{(\rho_k(i))}\left(\widehat{x}_{z(k)}\right) \implies \kappa \circ h^{(i)}(x_k) = \kappa \circ \widehat{h}^{(\rho_k(i))}\left(\widehat{x}_{z(k)}\right),$$

in which the left equality needs to hold over Y^{occ} only. ∎

It may seem, intuitively, that if S's sensors are weakened, then that should only make illusionability more feasible. But for an illusion to be passable, the definition requires that it appear identical to S, which thus prohibits the robot's sensors from operating with implausibly high fidelity. We note that, though beyond the scope of this work, if one may alter the secondary robot system, then the story changes. One could apply $\kappa(\cdot)$ computationally, degrading after the sensor's signals *ex post facto*, by introducing a small software shim.

V. The Limits of Illusion

Why have two definitions (Definition 2 and 3) to separate m-illusions from (m, τ)-illusions? The next result establishes that pairs of systems exist where the primary system is sufficiently powerful to conjure an illusion of the second, but the gap in relative speeds has no limit. Put another way, for any execution in the one, the other can create a faithful illusion, but no bound exists on the illusion's slowdown (i.e., there is no finite τ such that it is an (m, τ)-illusion). The result is that it is impossible for the primary system to present *any* illusion of the secondary system satisfying Definition 3.

Theorem 4 (Illusions with no bounded τ)**.** *There exist deterministic multi-robot transition systems S and $\widehat{S}$ where the latter is an m-illusion of the former, but for which no τ exists such that it is an (m, τ)-illusion.*

Proof roadmap: We give constructions for both S and $\widehat{S}$, then show that $\widehat{S}$ is indeed a 1-illusion of S (Lemma 1); and also, that any desired bound placed on the slowdown will be surpassed (Lemma 2).

Construction 1 (S_{thirds})**.** We define the following deterministic multi-robot transition system

$$S_{\text{thirds}} = \left(1, \mathbb{R}^+, \left\{-\tfrac{1}{3}, 0, \tfrac{1}{3}\right\}, f_{\text{add}}, \{\bot\} \cup \mathbb{Z}^+, h_{\text{sqz}}, x_0 = 0\right),$$

where $f_{\text{add}}(x, u) = f^{(1)}_{\text{add}}(x, u) = x + u$, and, dubbed *squeeze*,

$$h_{\text{sqz}}(x) = \begin{cases} q & \text{if } \exists q \in \mathbb{Z}^+ \text{ s.t. } -\frac{1}{4 \cdot 2^q} \le x - \frac{q}{3} \le \frac{1}{4 \cdot 2^q}, \\ \bot & \text{otherwise.} \end{cases}$$

This describes a robot that lives on the positive x-axis and which moves along in discrete steps, each with size $\frac{1}{3}$ units. This is shown as the green robot in the top diagram in Fig. 6. The robot is equipped with a stylized range sensor that measures a quantized distance to an obstacle at the

[2] This theorem holds for a slightly broader, albeit more obscure, class of functions. One may take the disjoint union as the domain, $\kappa : Y \sqcup \widehat{Y} \to Z$, so long as there is agreement on the function restrictions up to perceptual occurrence in the secondary system, i.e., $\forall y \in Y^{\text{occ}}$, $\kappa|_{Y^{\text{occ}}}(y) = \kappa|_{\widehat{Y}^{\text{occ}}}(\widehat{y})$.

origin (the blue information in the diagram). The sensor's precision increases (geometrically) with increasing x, with readings outside stripes of increasing precision return a generic reading, $\perp$. (The sensor's behavior here is essentially arbitrary for the construction, the symbol emphasizes its insignificance.)

Construction 2 ($\widehat{S}_{\text{binary}}$)**.** Next, consider deterministic multi-robot transition system

$$\widehat{S}_{\text{binary}} = \left(1, \mathbb{R}^+, \left\{\tfrac{1}{2^p} \mid p \in \mathbb{Z}\right\}, f_{\text{add}}, \{\perp\} \cup \mathbb{Z}^+, h_{\text{sqz}}, \widehat{x}_0 = 0\right),$$

where f_{add} and h_{sqz} are as in the preceding construction.

This robot also lives on the positive x-axis and moves in steps. It has rather more options for its movement, it moves in either direction with steps that are negative powers of two. This is shown as the red robot in the bottom part of Fig. 6, where the arrows show 'hops' of length $-\frac{1}{4}, -\frac{1}{8}, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}$, these being a sample of some actions available to the robot.

Lemma 1. *$\widehat{S}_{binary}$ is a* 1*-illusion of S_{thirds}.*

Lemma 2. *For any* 1*-illusion of S_{thirds} by $\widehat{S}_{binary}$, and any finite T, the constant policy $u_k = \frac{1}{3}$ for the robot in S_{thirds} implies that some $0 < N_T$ exists where*

$$T < \max_{k \in \{1,\ldots,N_T\}} \{z(k+1) - z(k)\}.$$

Proof: The two Lemmas prove Theorem 4. See [30]. ∎

VI. Physical Demonstration in the Robotarium

As a proof-of-concept, we implemented the illusion described in Example 5 both in simulation and on a physical robot testbed. Simulations were conducted using an implementation in Python; physical experiments were conducted in the Robotarium [27]. Fig. 1 shows a snapshot of the execution. Refer also to the supplemental video.

Note that Example 5 calls for the complicit robots to assume certain positions, but does not prescribe which robots should take which roles. We implemented three distinct strategies:

(i) A **naïve** matching strategy, in which robots are assigned to roles from left to right, in order of their indices.

(ii) The **Hungarian** algorithm [14, 22] for optimal task assignment, wherein some robots are assigned to obstacle roles and the remaining robots travel to the nearest location outside of the sensor range. The matching is selected to minimize the total travel time.

(iii) An enhancement of the **Hungarian** strategy with a **heuristic** that directs the offstage robots to the locations of the nearest obstacles that are not yet visible.

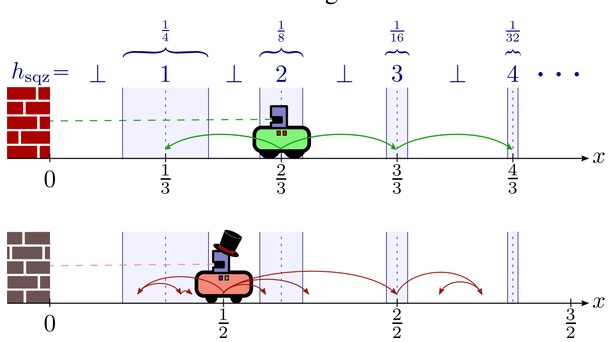

Fig. 6. A visual representation of the two systems in Constructions 1 and 2. The green robot at the top of the figure is S_{thirds}, the red robot below is $\widehat{S}_{\text{binary}}$. Both measure the environment with h_{sqz}, yielding a sensor whose preimage information is diagrammed in blue.

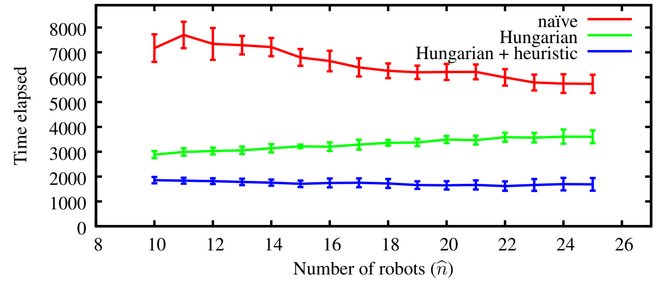

Fig. 7. Simulation results showing the impact of the number of robots and the policy on time efficiency.

One might expect, in this context, that the time efficiency of the illusion might be impacted both by the number of robots employed in the physical system and by the policy used in that system to carry out the illusion. To test this hypothesis, we performed a series of simulations of the policies described above. We conducted 10 trials, each using a distinct randomly-generated path for the robot in S. For each, we executed each of the three illusions described above and measured the amount of real time in $\widehat{S}$ needed to execute the policy in S.

Several notable trends appear in the results, which are shown in Fig. 7. Most plainly, the relative efficiency between the three algorithms matches what one might expect: Better use of more information leads to a more time-efficient illusion. For the two methods based on Hungarian matching, opposite trends appear as the number of robots increases: the basic Hungarian approach loses efficiency as robots are added, presumably due to interference from avoiding collisions between the robots. In contrast, the heursitic that positions robots near locations where new obstacles are likely to appear in the future is better able to take advantage of additional robots waiting 'in the wings' to take on roles when needed, leading to improvements in efficiency as the number of robots increases.

VII. Conclusion

There can be an immense variety of very different means to realize the same illusion. The single lesson that emerges most clearly from our demonstration implementation—both the more thorough simulation trials and the physical instance on the robotarium, where a time cut-off is imposed—is that distinct approaches may have time efficiencies that differ considerably. Even the more efficient curve in Fig. 7 has a slowdown factor of about 9, which is likely an impediment when producing an illusion of robots that one has direct access to. But consider an illusion for the system in Example 3 where the field of obstacles is unbounded: it simply can't be achieved physically. Moreover, if the 1-illusion has both the participant and the obstacles moving, it is possible to present an illusion for a robot that is faster than any we own. Judging the value of the idea by an early implementation is probably unwise.

REFERENCES

[1] Peter Amstutz and Andrew H. Fagg. Real time visualization of robot state with mobile virtual reality. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Washington, DC., USA, May 2002.

[2] Stefano Carpin, Mike Lewis, Jijun Wang, Stephen Balakirsky, and Chris Scrapper. Usarsim: a robot simulator for research and education. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1400–1405, 2007.

[3] Ian Yen-Hung Chen, Bruce Macdonald, and Burkhard Wünsche. Mixed reality simulation for mobile robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Kobe, Japan, May 2009.

[4] Toby H. J. Collett and Bruce Macdonald. Augmented reality visualisation for player. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Orlando, FL., USA, May 2006.

[5] Jeff Craighead, Robin Murphy, Jenny Burke, and Brian Goldiez. A survey of commercial & open source unmanned vehicle simulators. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 852–857, 2007.

[6] Oscar M. Curet, Ibrahim K. AlAli, Malcolm A. MacIver, and Neelesh A. Patankar. A versatile implicit iterative approach for fully resolved simulation of self-propulsion. *Computer Methods in Applied Mechanics and Engineering*, 199(37-40):2417–2424, 2010.

[7] Bruce Randall Donald. On Information Invariants in Robotics. *Artificial Intelligence — Special Volume on Computational Research on Interaction and Agency, Part 1*, 72(1–2):217–304, January 1995.

[8] David B. Dusenbery. *Sensory Ecology: How Organisms Acquire and Respond to Information*. W. H. Freeman, 1992.

[9] Steven N. Fry, Pie Müller, H.-J Baumann, Andrew D. Straw, Martin Bichsel, and Daniel Robert. Context-dependent stimulus presentation to freely moving animals in 3D. *Journal of Neuroscience Methods*, 135(1):149–157, May 2004. doi: 10.1016/j.jneumeth.2003.12.012.

[10] Steven N. Fry, Nicola Rohrseitz, Andrew D. Straw, and Michael H. Dickinson. Trackfly: Virtual reality for a behavioral system analysis in free-flying fruit flies. *Journal of Neuroscience Methods*, 171(1):110–117, June 2008. doi: 10.1016/j.jneumeth.2008.02.016.

[11] Christian Hölscher, Alexander Schnee, Hansjuergen Dahmen, Lokesh Setia, and Hanspeter A. Mallot. Rats are able to navigate in virtual environments. *Journal of Experimental Biology*, 208:561–569, January 2005. doi: 10.1242/jeb.01371.

[12] Nick Jakobi. Evolutionary Robotics and the Radical Envelope-of-Noise Hypothesis. *Adaptive Behavior*, 6(2): 325–368, 1997.

[13] Nathan P. Koenig and Andrew Howard. Design and use paradigms for gazebo, an open-source multi-robot simulator. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 4, pages 2149–2154, 2004.

[14] Harold W. Kuhn. The Hungarian Method for the Assignment Problem. *Naval Research Logistic Quarterly*, 2:83–97, 1955.

[15] Steven M. LaValle. *Planning Algorithms*. Cambridge University Press, Cambridge, U.K., 2006. Available at http://planning.cs.uiuc.edu/.

[16] Tien Luu, Allen Cheung, David Ball, and Mandyam V. Srinivasan. Honeybee flight: a novel 'streamlining' response. *Journal of Experimental Biology*, 211:2215–2225, June 2011. doi: 10.1242/jeb.050310.

[17] James McLurkin, Joshua Rykowski, Meagan John, Quillan Kaseman, and Andrew J Lynch. Using multi-robot systems for engineering education: Teaching and outreach with large numbers of an advanced, low-cost robot. *IEEE Transactions on Education*, 56(1):24–33, 2013.

[18] Robin Milner. A calculus of communicating systems. *Lecture Notes in Comput. Sci. 92*, 1980.

[19] Robin Milner. Four combinators for concurrency. In *Proceedings of the first ACM SIGACT-SIGOPS symposium on Principles of distributed computing*, pages 104–110. ACM, 1982.

[20] Francesco Mondada, Edoardo Franzi, and Andre Guignard. The development of khepera. In *Proceedings of the First International Khepera Workshop*, pages 7–14, 1999.

[21] Francesco Mondada, Giovanni C. Pettinaro, Andre Guignard, Ivo W. Kwee, Dario Floreano, Jean-Louis Deneubourg, Stefano Nolfi, Luca Maria Gambardella, and Marco Dorigo. Swarm-bot: A new distributed robotic concept. *Autonomous robots*, 17(2-3):193–221, 2004.

[22] James Munkres. Algorithms for the Assignment and Transportation Problems. *Journal of the Soc. for Industrial and Applied Math.*, 5(1):32–38, March 1957.

[23] Jason M. O'Kane and Steven M. LaValle. On Comparing the Power of Robots. *International Journal of Robotics Research*, 27(1):5–23, January 2008. doi: 10.1177/0278364907082096.

[24] David Park. Concurrency and automata on infinite sequences. In *Theoretical Computer Science*, pages 167–183. Springer, 1981.

[25] David Pekarek and Todd D. Murphey. A backwards error analysis approach for simulation and control of nonsmooth mechanical systems. In *Proceedings of the IEEE International Conference on Decision and Control*, pages 6942–6949, 2011.

[26] Daniel Pickem, Myron Lee, and Magnus Egerstedt. The GRITSBot in its natural habitat: A multi-robot testbed. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2015.

[27] Daniel Pickem, Paul Glotfelter, Li Wang, Mark Mote,

Aaron Ames, Eric Feron, and Magnus Egerstedt. The Robotarium: A remotely accessible swarm robotics research testbed. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2017.

[28] Carlo Pinciroli, Vito Trianni, Rehan O'Grady, Giovanni Pini, Arne Brutschy, Manuele Brambilla, Nithin Mathews, Eliseo Ferrante, Gianni Di Caro, Frederick Ducatelle, et al. Argos: a modular, multi-engine simulator for heterogeneous swarm robotics. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5027–5034, 2011.

[29] Adolfo Rodríguez, Luis Basanez, J. Edward Colgate, and Eric L. Faulring. A framework for the simulation and haptic display of dynamic systems subject to holonomic constraints. *International Journal of Robotics Research*, 29(4):336–352, 2010.

[30] Dylan A. Shell and Jason M. O'Kane. Reality as a simulation of reality: robot illusions, fundamental limits, and a physical demonstration, September 2019. arXiv:1909.03584.

[31] Michael C. Stanley and J. Edward Colgate. Computer simulation of interacting dynamic mechanical systems using distributed memory parallel processors. *Advances in Robotics*, 42:55–61, 1992.

[32] Mike Stilman, Philipp Michel, Joel Chestnutt, Koichi Nishiwaki, Satoshi Kagami, and James Kuffner. Augmented reality for robot development and experimentation. Technical Report CMU-RI-TR-05-55, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA,

[39] Klaudia Witte, Stefanie Gierszewski, and Laura

2005.

[33] Colin Stirling. Local model checking games. In *Proceedings of the International Conference on Concurrency Theory (CONCUR'95)*, pages 1–11. Springer-Verlag, 1995.

[34] Jouni Takalo, Arto Piironen, Anna Honkanen, Mikko Lempeä, Mika Aikio, Tuomas Tuukkanen, and Mikko Vähäsöyrinki. A fast and flexible panoramic virtual reality system for behavioural and electrophysiological experiments. *Scientific Reports*, 2(324), March 2012. doi: 10.1038/srep00324.

[35] Graham K. Taylor, Marko Bacic, Richard J. Bomphrey, Anna C. Carruthers, James Gillies, Simon M. Walker, and Adrian L. R. Thomas. New experimental approaches to the biology of flight control systems. *Journal of Experimental Biology*, 221:258–266, January 2008. doi: 10.1242/jeb.012625.

[36] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033, 2012.

[37] Richard Vaughan. Massively multi-robot simulation in stage. *Swarm intelligence*, 2(2-4):189–208, 2008.

[38] Talbot H. Waterman. *Animal navigation*. W. H. Freeman, 1989.

Chouinard-Thuly. (Guest Editorial) Virtual is the new reality. *Current Zoology—Special Issue: Computer Animations and Virtual Reality in Animal Behavior Research*, 63:1–4, February 2017. doi: 10.1093/cz/zow119.