Hark! Who goes there? Concurrent Association of Communication Channels for Multiple Mobile Robots

Plamen Ivanov Dylan A. Shell

April 28, 2016

Abstract

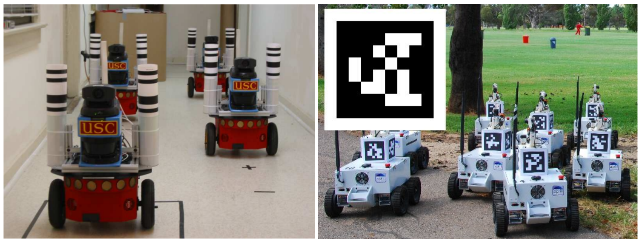
Robots working in teams can benefit from recruiting the help of nearby robots. But, while robots are typically aware of their neighbors' relative positions through information sensed locally (e.g., range and bearing), a robot does not necessarily know the network identifiers (IDs) of its neighbors directly from observation. In this work robots use a simple visual gesture, paired with wireless messages, to rapidly and effectively establish a one-to-one association between the relative positions (local, visual IDs) of neighboring robots and their network addresses (global, wireless IDs).

We formalize the *channel association problem* and explore its structure from an information filter perspective. Under an idealized communication model, we investigate two simple probabilistic algorithms and contribute analyses of performance in terms of parameters, such as robot density, communication range, and movement speed, Branching Processes are used to predict the macroscopic performance of the algorithms, producing models that characterize the channel association behavior, given parameters that describe the multi-robot system. The approach also allows parameters to be fine-tuned when designing a system so that its performance meets some specified threshold.

1 Introduction

Whilst operating as part of a team, robots may recruit the help of those around them. One robot, sensing another in a useful place, might send the request: "Will the robot to my right help me move this piano?" But, whereas robots locate others with cameras, laser range finders, or other (spatial) sensors, they use specialized communication devices, like Wi-Fi radios, to send messages to (logical) recipients. Since the same robot will have an identifier in the spatial medium that differs from its identifier in the communicative medium, an association between the two is needed if the message is to target a specific recipient. *How does a robot learn the network addresses of those it wishes to communicate with?*

A common solution is to compile a set of pairs by hand prior to deployment of the robots. Each pair connects a network address with the corresponding visual identifier and remains static once established. Practical implementations typically use visual marker or fiducial systems with high saliency. Influential examples include that of Howard et al. [2006], Olson [2011], Garrido-Jurado et al. [2014], the first two are shown in Fig. 1. In this paper, we tackle the question of relating a local, relative view of a robot to an identifier which can be used to address the robot directly. The multi-robot systems we study are composed of anonymous robots which need not be visually distinguishable from one another. Also, the robots in the system need not share any common spatial reference frame.[1]



(a) Visual barcode fiducial markers as described in Howard et al. [2006].

(b) The AprilTag markers as described in Olson [2011].

Figure 1: Examples of visual markers used for identifying robots.

In the preceding examples, the Wi-Fi and camera are two independent means by which information is exchanged. When robots can produce some visually identifiable sign, the camera can form part of a visual communication channel. The visual fiducials in the photos are excellent for communicating presence of a marker, though a pre-arranged gesture has the benefit of being dynamic. A protocol using just such a visual communication channel is given by Dieudonné et al. [2009]. A second, rather more practical example, is that of Batalin and Sukhatme [2002] who employ a behavior called *Dance*, which constitutes a gesture to send information visually.

We term the problem of making an association between visual and networked identifiers the *communication channel association problem*, or *association problem* for short. This work studies simple probabilistic algo-

[1] The systems in Fig. 1 also established a shared spatial representation and the visual markers were helpful in that regard too—the present work is concerned solely with the association problem.

rithms that solve the association problem without relying on global information or external markers. Instead, the robots use a simple, visually identifiable gesture (like a light being turned on-and-off or a *Dance* behavior) and wireless communication to solve the association problem quickly and concurrently. The identifiers of robots in the visual channel (visual IDs) are locally assigned by each observer (it is helpful to think of the typical, representative case of the IDs being the range and bearing to each visible robot). The network IDs of the robots in the wireless channel are assigned to the robots globally and do not depend on other robots' perspectives.

This paper's focus is on understanding and predicting the expected behavior of this association process for large multi-robot systems. We analyze the problem using an idealized model of the inter-agent communication, which enables characterization of the performance in terms of very few parameters. Starting with simulations, three key system parameters were identified: robot density, communication range, and robot velocity. The effect that each has on performance is examined in detail, and a model using the theory of Branching Processes to predict the expected macroscopic performance of such systems constructed.

The paper is organized as follows. Section 2 examines related work. In Section 3 we formalize the communication channel association problem, describing the communication model we study, and introducing the information space view. Section 4 addresses the problem with stationary robots and then, Section 5, the mobile case. The models are validated by comparing their predictions to the simulation results. We also show that a Markov chain analysis is infeasible when applied to the association problem. The infeasibility of standard analysis, even under assumptions of perfect communication, helps exculpate the simplified model we employ. But, because the results obtained are for this idealized communication model, they may be best viewed as *bounds* for systems involving erroneous packet transmissions. Nevertheless, Section 6 returns to our modeling assumptions, suggesting that several real-world circumstances are approximated reasonably well. Section 7 concludes.

2 Related Work and Background

2.1 Related Work

The notion of situated communication is central to solving the channel association problem. Støy [2001] distinguishes *abstract* channels, which carry meaning only in the message contents, from *situated* channels. In situated communication some property, inherent to the channel, provides additional meaning to the received messages. In our case, the visual channel is situated and the added information is the relative position of the transmitter. A more general concept, encompassing the situated property, is that of indexical knowledge [Agre and Chapman, 1987, Lespérance and Levesque, 1995]. Indexical knowledge refers to knowledge, which is relative to the observer. It is contrasted by objective knowledge, which is independent from the perspective of the observer.

One may categorize robotic systems informally based on the number of communication channels that are available to them. Most common are one- and two-channel systems (e.g. idealized wireless in the first case, and infrared communication devices in the second). In Dieudonné et al. [2009], the authors examine a scenario where the robots can only communicate through movement-signals. Their work shows how an explicit communication channel can be established using robot sensors and robot gestures — the gestures consist of specific motion of a robot through the environment.

A multi-robot system falls under the two-channel category if its robots are equipped with two independent communication devices such as wireless and infrared-based communication. In such systems, typically, both communication channels are capable of transmitting a robot's ID reliably. In addition, the infrared communication channel is also a situated one, it can be used to detect range and bearing to other robots (*cf.* Gutiérrez et al. [2008]). Since each channel can transmit a robot's ID, this means that the association problem can be solved by directly sending a message containing the ID. Robots which receive the message now know the robot's identity and can address them directly in either channel.

As Dieudonné et al. [2009] and Batalin and Sukhatme [2002] show, a communication channel can be established through the use of any robot sensor which can detect a change in the state of another robot. In our work, the second communication channel is assumed to be established through a camera and a light (vision beacon).

Multi-robot systems in which there is no explicit second communication channel, such as Howard et al. [2006], Garrido-Jurado et al. [2014], Olson [2011], must still solve the association problem. As mentioned above, a common approach in such systems is to use passive markers along with a fixed pre-compiled list of marker-network address pairs. This sort of static association has the limitation of requiring maintenance by some mechanism external to the multi-robot system. Additionally, such markers can encode a limited number of unique identifiers, may suffer from reliability issues, and may impose restrictions on the maximum distance at which the markers can be identified. Markers may also increase the computational or hardware required by the system.

This work addresses algorithmic aspects of the association problem for systems where the second channel does not suffice to transmit IDs directly. Most closely related is the recent work of Mathews et al. [2012, 2015] which introduces the concept of *spatially targeted communication.* They use the idea of a situated channel to target communication to a receiver based on their location in space. Both the model and solution in their work have similarities with that which we describe below. But their work establishes an association between a pair of robots in the system, while we establish an association between all robots in the system concurrently and without restricting the use of the wireless and visual communication channels for any robot.

Finally, we note that mutual or collective localization in a multi-robot system, as exemplified by Franchi et al. [2009] and Fox et al. [2000] is related to the association problem because it solves a harder problem, the solution to which can provide a common reference frame for robots in a system. However, the cost of localization of all robots may be greater than is desirable simply for targeted communication.

2.2 Background

The *information space* and *information state* concepts explained by LaValle [2009] underpin our understanding of the association process. An information state encapsulates the abstract information which a robot maintains to keep track of the world around it. The information space can be seen as the set of all information states and a transition function which describes how one information state is transformed into another, given an observation.

We also use techniques based on discrete-time absorbing Markov chain [Berend and Tassa, 2010] and discrete-time Branching Processes [Haccou et al., 2007] to analyze and predict the macroscopic behavior of a multi-robot system solving the association problem.

An earlier version of this work appeared in our conference paper, Ivanov and Shell [2014]. This paper expands the analysis substantially and offers an extended algorithm for solving the association problem for a system composed of mixture of stationary and mobile robots.

3 General Description of the Association Problem

Consider a large number of robots spread randomly and uniformly through an open and empty environment. Each robot in the system can directly observe (and distinguish from the environment) other robots using a camera or array of cameras with a 360° field of view. The robots' operation proceeds in discrete, synchronous time steps. Each robot has a light which can be on or off during any time step. The camera and light form a situated visual communication channel and robots may also broadcast messages through a wireless channel. Both channels have limited range. Every robot is assigned a wireless ID and itself assigns visual IDs to robots it can detect with its camera. The following is a list of assumptions about the motivating system.

Assumption 1. *Perfect communication*: Messages are neither lost, degraded, nor noisy. A message broadcast by a robot will be received by all robots within range during the time step it was transmitted.

Assumption 2. *Message size*: The wireless communication medium can transmit the full binary representation of a robot's ID: $log_2(n)$ bits. The visual channel can transmit a message of size 1 bit.

Assumption 3. *Unlimited bandwidth and message processing*: Robots can receive and process any number of messages during each time step.

The preceding, idealized model describes the system used in our simulation experiments and serves as the basis of our formal model as well as subsequent analysis. Observe that we do not consider communication errors or robot failures. Section 6.3 discusses potential approaches for dealing with communication noise and error while Section 6.4 provides an informal discussion on the feasibility of Assumption 3.

3.1 Formal Definition

Definition 1. We consider a *multi-robot system* of robots capable of communicating via a radio communication channel (channel 1) and a physically situated communication channel (channel 2), and described by a tuple: $\langle G_1 = \langle V, E_1 \rangle, G_2 = \langle V, E_2 \rangle, C_1, C_2, f_1, f_2 \rangle$ where

1. V is a set of vertices, each vertex represents a robot.
2. $G_1 = \langle V, E_1 \rangle$ and $G_2 = \langle V, E_2 \rangle$ are two directed graphs representing the connectivity of the system over the two channels. Edge $e \equiv (v_i, v_j) \in E_k$ is interpreted as robot i being able to receive a one-hop message from robot j over communication channel k.
3. $E_2 \subseteq E_1$ — This is Property 2 (see below).
4. C_1 and C_2 are the label sets for channels 1 and 2.
5. f_1, f_2 are labeling functions $f : V \times V \to C$ which are applied to channels 1 and 2, respectively. C is a set of labels.

The labeling function for each channel maps robots to a set of labels, corresponding to addresses (wireless and visual IDs, respectively). In the motivating system, the labeling functions will provide unique IDs for the wireless channel (IP addresses) and locally unique (defined precisely next) IDs for the visual channel. Whatever produces the ID assignment, it must satisfy Property 1.

Property 1. *Labeling function property*:

$$\forall x, y, z \in V, e_y = (y, x) \in E_{ch} \land e_z = (z, x) \in E_{ch} \implies f_{ch}(e_y) \neq f_{ch}(e_z).$$

Thus, if a robot can receive a message from two robots, then the IDs of the two robots will differ. This ensures that each neighborhood will have non-repeating IDs, which we call *locally unique IDs*. In the motivating system, it applies to both the wireless and visual IDs and, moreover, any labeling that provides globally unique IDs satisfies Property 1.

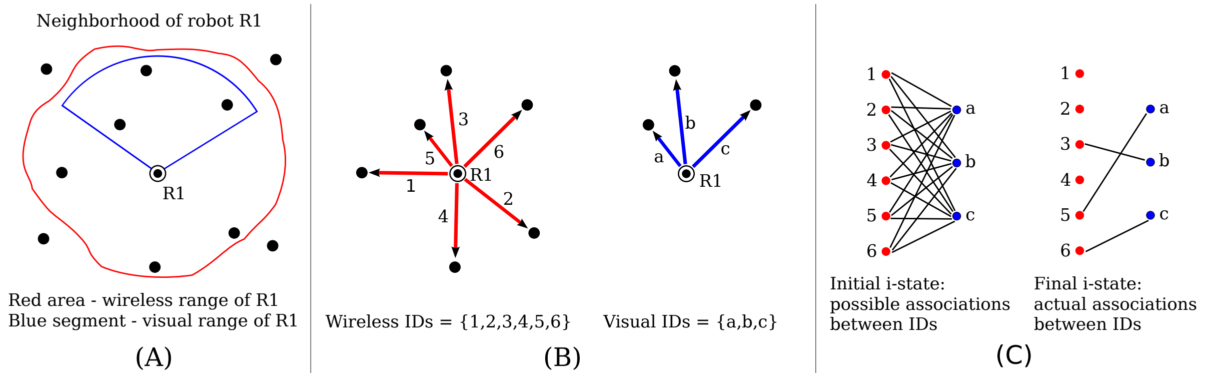


Figure 2: A robot and several neighboring robots as part of a multi-robot system are shown in (A). The wireless communication channel (in red) connects robot $R1$ to some of the robots surrounding it. The visual communication channel (in blue) has a more restricted view of robots in the system. In (B), we can see the neighborhood of robot $R1$ as given by Def. 2. The vertices correspond to robots, the edges to the connectivity of $R1$ to its neighbors in each channel, and the label sets C_1^{R1}, C_2^{R1} are composed of the wireless and visual IDs. Two possible i-state bipartite graphs, as given by Def. 6 can be seen in (C).

Property 2. *Channel inclusion property*:
When a robot can receive a channel 2 message from another robot then it is guaranteed to be able to receive a channel 1 message from that robot as well.

Property 2 ensures that a robot can receive a wireless message from robots in its visual range. Actually, this property is not required for solving the association problem, but it simplifies the analysis by restricting the problem to a subset guaranteed to have a match for every visual ID within range. The algorithms and analysis can be extended to relax this property.

To be precise about the information available from the perspective of an individual robot, several additional concepts are introduced next. Consider a robot, whose perspective we shall examine, the *observer* or *observing robot*. Graphs representing the state of an individual robot and its neighborhood are depicted in Fig. 2.

Definition 2. A *neighborhood* is a subset of the system centered on an individual robot i, and can be described by the tuple $\langle V^i, E_1^i, E_2^i, f_1,\ f_2, C_1^i, C_2^i \rangle$ where

1. i is the observer at the center of the neighborhood.
2. V^i are those v_j for which edge $(v_i, v_j) \in E_1$ exists.
3. E_1^i and E_2^i contain all directed edges in E_1 and E_2 which have robot i as source.
4. C_1^i, C_2^i are the labels given to edges in E_1^i and E_2^i by f_1 and f_2, respectively.

Definition 3. An *activation* describes the communication activity of a neighborhood for a given time step t. It is given by the tuple $A_t^i = \langle C_{1At}^i, C_{2At}^i \rangle$ where

1. C_{1At}^i is the set of labels of robots which are in range of and from which a message was sent to robot i on channel 1 during time step t.
2. C_{2At}^i is defined similarly for channel 2.

Definition 4. An *observation* is an individual robot's perception of an activation for a given time step t and is described by the tuple $O_t^i = \langle C_{1Ot}^i, C_{2Ot}^i \rangle$ where

1. C_{1Ot}^i are labels of robots from which a message was received by robot i on channel 1 during the time step.
2. C_{2Ot}^i is defined analogously.

Definition 5. The *local view* of robot i is the information directly available to it, and is given by tuple $\langle n, C_1^i, C_2^i, O^i = [O_1^i, O_2^i, ...O_t^i] \rangle$ where

1. n is the maximum possible robots in the system.
2. C_1^i and C_2^i are label sets as previously defined.
3. $O^i = [O_1^i, O_2^i, \cdots, O_t^i]$ is a sequence of observations (*Def. 4*) of the robots in the neighborhood of i, and t is the time step when the observation was made.

Given the perfect communication assumptions, activations and their corresponding observations are identical.

Definition 6. Each robot maintains an *information state* [LaValle, 2009], or *i-state*, which represents its knowledge of its neighborhood. The i-state is described by the bipartite graph $S^i = \langle C_{1t}^i, C_{2t}^i, E_{1-2} \rangle$ where

1. C_{1t}^i and C_{2t}^i are subsets of C_1^i and C_2^i. The subsets consist of all labels from which a message was received in any time step up to t. These two subsets represent the two distinct vertex sets of the graph.
2. E_{1-2} is the edge set of the graph, representing possible associations between elements of each label set.

Each robot starts in complete ignorance of the actual one-to-one association of labels. As observations arrive, the observing robot updates its i-state by removing edges from the graph.

Definition 7. The *communication channel association problem* for each robot i is:

Given: A local view with an observation sequence at time t.

Output: A complete or partial one-to-one association between channel 1 labels and channel 2 labels for the label tuple $\langle C_1^i, C_2^i \rangle$ corresponding to i's neighborhood at time t.

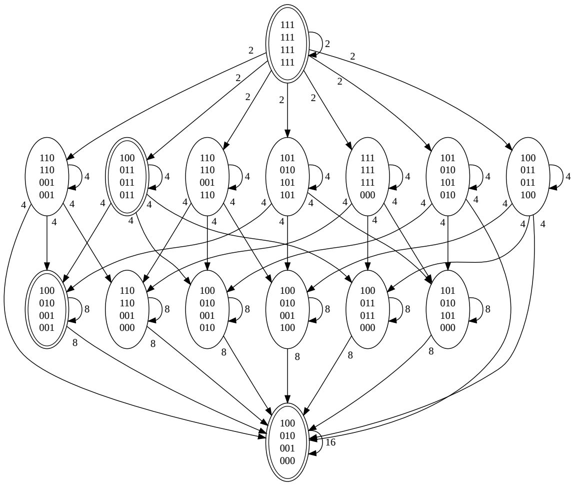


Figure 3: I-state transition graph for the case of $w = 4$ wireless, $v = 3$ visual IDs. Each node contains an $w \times v$ i-state in matrix form and each edge is labeled with the number of observations which triggers the transition. The total number of observations for this case is 16. A specific path through the graph has been highlighted (with double lined ovals) and is described in greater detailed in Fig. 4

3.2 Information Space Interpretation

The bipartite graph is a natural representation of the i-state and observations of a robot. The wireless IDs within range of an observing robot are one set of vertices, the visual IDs which are in range make up the other set. Let w and v denote the size of wireless and visual vertex sets, respectively. The edges of the i-state graph represent possible associations between a wireless ID and a visual ID. The edges can be subdivided into two categories, false edges and true edges. *True edges* correspond to correct associations: they connect the wireless and visual IDs of the same robot. *False edges* on the other hand connect the IDs of two different robots and are therefore incorrect. An edge in the observation graph represents a potential association between a wireless and a visual ID which have the same activity status (active/inactive) during the current observation. If an edge in the i-state graph has no corresponding edge in the observation graph, the edge has been identified as false and is removed from the i-state.

Each robot's initial i-state has all the edges as no knowledge of association is known. The i-states and observations for a given neighborhood form the *directed i-state transition graph.* The i-states form the vertex set of the transition graph and the directed edges correspond to the set of observations which lead from one i-state to another. A version of the transition graph can be seen in Fig. 3. The transition graph is a representation of the information space of the association problem.

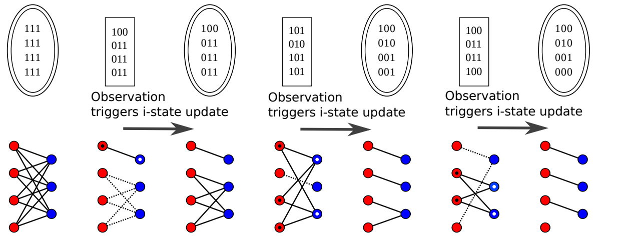


Figure 4: An i-state path through the transition graph: the matrices in ovals are i-states, and matrices in squares are observations, along with the corresponding bipartite graphs. The column of 4 vertices are wireless IDs and the column of 3, visual ones. Black/white marks in the observation graph denote transmission during that time step. The solid edges in the observation graph connect currently active IDs, while the dashed edges connect inactive IDs. This path is highlighted in the full i-space graph in Fig. 3.

It is important to note that the robots do not explicitly store, compute or use the information space which can be built for their current neighborhood. But the information space idea is only used for understanding the algorithms and predicting their performance. Given w and v, the number of possible i-states is given by Bell's number [Berend and Tassa, 2010] — it counts the number of ways a set of a given size can be partitioned. The number of possible observation graphs is 2^w which corresponds to the number of ways w wireless IDs can be active or inactive. Each observation partitions the vertices of the i-state graph into two disjoint subsets — active and inactive IDs. A sequence of observations, via elimination of edges, splits the initial i-state into disconnected bipartite graphs. Observations need not always result in elimination of edges. The final i-state is a disconnected bipartite graph composed of subgraphs of size 2 or 1. A subgraph of size 2 associates one wireless and one visual ID. Subgraphs of size 1 indicate wireless IDs of robots outside of visual range. This process of elimination of false edges allows an observer to solve the association problem.

4 Stationary Robots

First we consider a system of robots which do not move (or else move very slowly) and thus their relative positions do not change enough to affect their connectivity in either communication channel, that is, there is no change in the set of robots an observer can receive messages from. For the stationary case, we make the following additional assumption: every robot knows the number of robots in its neighborhood. In general this can be made to hold by adding an initial time step in which all robots transmit a message on both channels. The assumption is not required for the operation of the algorithm, but permits more uniform analysis of the performance (by establishing the same starting i-state for all cases).

4.1 Probabilistic Algorithm for Solving the Stationary Case

The reader is directed to Algorithm 1 for detailed pseudo-code of the algorithm. The i-states and observations for a single run of the algorithm (for one neighborhood) are depicted in Fig. 4.

All robots construct a $w \times v$ i-state matrix where w and v represent the number of robots in direct communication over channel 1 (C_{1t}^{i} from S^i) and channel 2 (C_{2t}^{i} from S^i), respectively. Cells in the matrix correspond to edges in E_{1-2} from S^i and are filled with 1's. The wireless and visual IDs (labels) of all robots within communication range are associated to activity arrays (C_{1Ot}^{i} from O_t^i for wireless and C_{2Ot}^{i} from O_t^i for visual) that keep track of which robot transmitted a message in the current time step (1 denotes transmission). Wireless and visual IDs are indexed with $h \in [0, w-1]$ and $s \in [0, v-1]$, respectively. The i-state matrix contains the observer's current information about the matching between wireless and visual IDs.

Each time step, all robots emit a *message pair* (simultaneous transmission in both channels) with probability 0.5. Every observer constructs a $w \times v$ observation matrix where cell $[h][s]$ is a 1 if both the h^{th} wireless ID and s^{th} visual ID were active or inactive during the current time step; otherwise it is 0. Once constructed, the observation matrix is used to update the i-state matrix by element-wise multiplication. When the i-state matrix contains only v 1's, the association problem for the observer is complete. The non-zero $[h][s]$ entries in the matrix indicate that the h^{th} wireless ID and the s^{th} visual ID belong to the same robot.

The algorithm scales with the size of robots' neighborhoods, not with the total system size. Every robot solves the association of its neighborhood concurrently.

Algorithm 1 Stationary Robots Association

```
procedure AssociateFilter
    broadcast(messagePair)
    $w$                          ▷ number of wireless IDs in range
    $v$                          ▷ number of visual IDs in range
    $istate[w][v] \leftarrow 1's$
    $obs[w][v]$
    $w\_activity[w]$             ▷ 0 – inactive, 1 – active
    $v\_activity[v]$             ▷ 0 – inactive, 1 – active
    for each time step do
        if $fairCoinFlip() = heads$ then
            $broadcast(messagePair)$
        end if
        $w\_activity \leftarrow receivedWifiMsgs()$
        $v\_activity \leftarrow receivedVisMsgs()$
        $obs \leftarrow fillObservation(w\_activity, v\_activity)$
        $istate \leftarrow updateState(obs)$
        if $istate.isFinal()$ then
            $ready \leftarrow$ true
        end if
    end for
end procedure
```

4.2 Performance Metrics

Two metrics for quantifying the algorithm's performance are useful. The *expected time to completion* is the number of time steps the algorithm takes to complete the association problem for a given neighborhood. And *fraction matched* is the fraction of visual IDs which have been unambiguously associated (matched) with a wireless ID for a given time step.

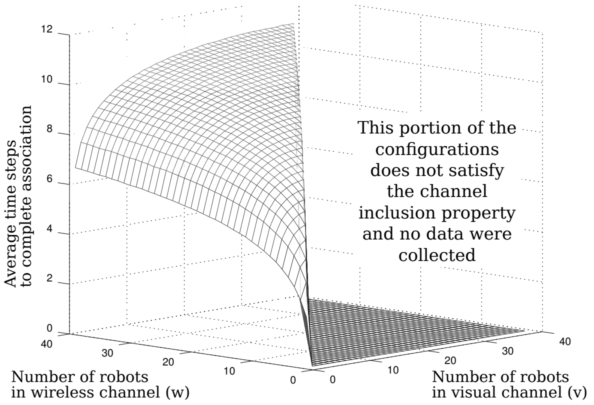


Figure 5: A plot of the average number of time steps to complete the association problem for all $w \times v$ neighborhoods for which $w \in [2, 40]$ and $v \in [1, w-1]$. Every neighborhood was simulated 50,000 times.

4.3 Expected Time to Completion

4.3.1 Information Space and Markov Chain

Algorithm 1's path through the information space graph is not determined by the specific IDs in a given neighborhood and it can include any valid transition in the i-space. Since robots running the algorithm transmit a message pair with probability 0.5, all possible observations are equally likely. The information space graph has the necessary properties to be considered an absorbing Markov chain [Grinstead and Snell, 1997]—it has an absorbing state (the final state) and this state can be reached from any other state in the graph (transient states). The absorbing Markov chain yields a computation of the expected number of steps from any state in the graph to the absorbing state given the transition matrix — this would be equivalent to the expected time to completion for the probabilistic algorithm.

Unfortunately, the expense of this approach is prohibitive. The number of i-states for a given neighborhood is given by Bell's number: B_w. Berend and Tassa [2010] give an upper bound of Bell's number which is $O(w!)$ and $\Omega(2^w)$ for a set of size w. A simple application of all possible observations to all possible i-states can naïvely generate the probability matrix in $O(2^w B_w)$ time. The matrix inversion needed in the Markov chain computation will take $O({B_w}^3)$. The absorbing Markov chain method, therefore, runs in $O(w!^3)$. A cheaper prediction of the time to completion is clearly required.

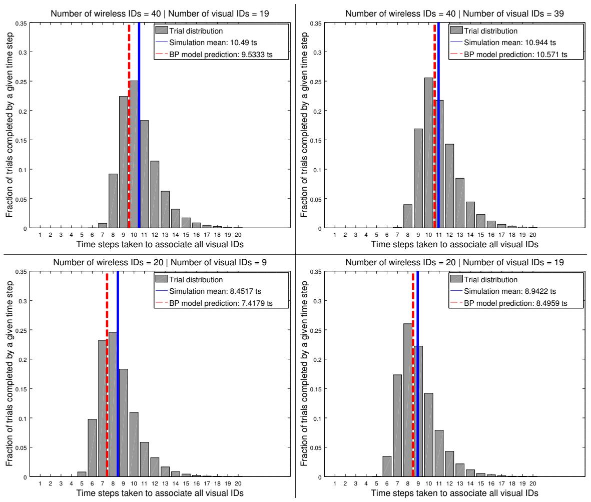


Figure 6: Plots of the distribution of the different times to completion of the association problem for combinations of w and v. The blue vertical line is the mean number of time steps in the simulation, and the red vertical line the Branching Processes model.

4.3.2 Simulation Results

To get an idea of the time to completion of the algorithm, we simulated a number of neighborhoods and gathered statistics on the time it took the algorithm to complete the association problem. We simulated all neighborhood size combinations of $w \in [2, 40]$ and $v \in [1, w-1]$. To validate the data from the simulation we also calculated the precise expected value for all combinations of $w \in [2, 8]$ and $v \in [1, w-1]$ using the absorbing Markov chain method, results from which matched the simulation up to the precision we used during the calculations (10^{-4}). The results appear in Fig. 5. More detail of the distribution of times to completion for four w and v combinations appear in Fig. 6.

From the data we can see that there is a minimum time to completion equal to $log_2(v)$. This is the smallest number of steps needed to split a set into singletons, splitting every subset at each step.

4.3.3 Branching Processes Model

Discrete time Branching Processes (BPs) are used to model populations of individuals which go through reproductive cycles at given intervals [Haccou et al., 2007]. Individuals reproduce by creating offspring in the next reproductive cycle, their number being described by some known distribution. The model can answer such questions as: Will this population go extinct? If so, how long will it take? What is the expected number of individuals in a given reproduction cycle after the start of the population?

We treat the potential associations in a robot's i-state as a population. Each wireless to visual ID pairing is thus an individual which can either produce 1 offspring (itself, as it has not been eliminated by observation) or 0 offspring (be discarded as it has been eliminated by an observation). Once a population reaches some number of individuals (the number of visual IDs which the current robot can observe) the association problem for the neighborhood has been solved.

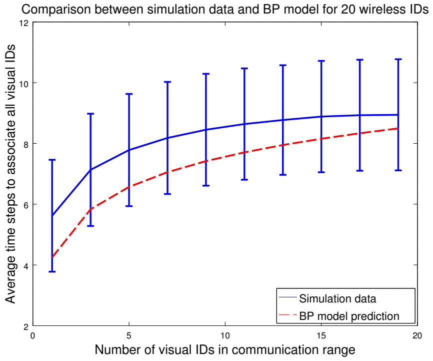


Figure 7: The mean and one standard deviation of the expected time steps to completion for the association problem for $w = 20$ and odd $v \in [1, w-1]$. The Branching Process model's prediction is given in dashed red.

One complexity is that not all potential associations have the same likelihood of producing offspring—true edges always produce 1 offspring. This introduces some inaccuracy in calculating the time to extinction. Extending the model by introducing multiple types within a population, each with a different distribution, allows for a closer treatment of the actual process.

The mean number of individuals is simply Nm^t, where N is the initial population, m is the mean number of offspring and t is the current generation. For the false edges, $N = wv - v$. The value of m is determined by the probability of two unrelated IDs having different activity statuses in a given step, which is 0.5, and hence $m = \frac{1}{2}$. For true edges, $N = v$ and $m = 1$. Taken together, this gives the expected number of false edges for each step.

Haccou et al. [2007] gives the following approximation to the expected time to extinction, *i.e.,* the number of generations until the population has disappeared completely, applicable to our setting:

$$E_N[\tau] = \frac{\ln(N)}{|\ln(m)|} = \frac{\ln(wv - v)}{|\ln(\frac{1}{2})|}. \tag{1}$$

(Note that Eq. 1 is continuous, whereas the system is discrete. The error caused by this simplification is worse for fewer robots.)

Figs. 7 and 8 compare simulation data with the approximation for two numbers of wireless IDs and several possible visual ID counts. The BP approximation underestimates the actual values, but falls well within the bulk of the solution lengths and is closer for higher values of v. Despite incurring some inaccuracy, the BP model provides a prediction that is cheap to compute.

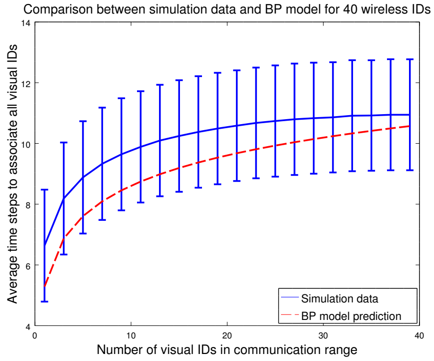


Figure 8: Data as in Fig. 7 but for $w = 40$.

4.4 Fraction Matched Metric

4.4.1 Simulation Results

Keeping the simulation setup as before, an average of the fraction matched metric over several simulation runs, for neighborhoods of different sizes, was computed and appears in Fig. 9. In addition, to the time to completion, we kept track of the number of visual IDs unambiguously associated with a wireless ID. From this we computed the expected fraction of visual IDs solved at each time step.

4.4.2 Fraction Matched Model

To predict fraction matched we set $w = v$ since empirical observations showed that, under our assumptions, there is no dependence on v. The starting number of false edges is $wv - v$ and, thereafter, one can compute the expected number of false edges via Eq.(2) to give a value for each time step.

$$F_t = F_{\text{init}} \times m^t. \tag{2}$$

Next, for every visual ID, we calculate the probability that it will have zero false edges assigned to it, as this represents an unambiguously associated visual ID. The value is equal to the probability that all false edges have been assigned to any of the rest of the visual IDs, which is given by Eq. (3). The same equation also produces the fraction of unambiguously associated visual IDs.

$$P_{matched} = \left(\frac{v-1}{v}\right)^{F_t}. \tag{3}$$

The following simplifications have been introduced to boiling this probabilistic, combinatorial problem in such a concise characterization. First, we assume each visual ID is independent from the other visual IDs. Second, when looking from the perspective of a single visual ID, that there is no limit on how many false edges may be assigned to the other visual IDs. A comparison between the model's prediction and simulation results can be seen in Fig. 9. This simple model is startlingly effective.

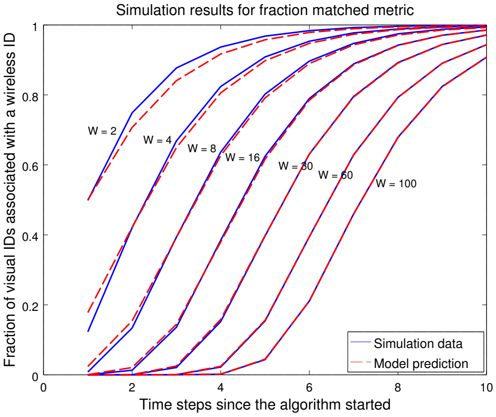


Figure 9: Predictions of the model compared with simulation performance for the fraction matched metric across different numbers of wireless IDs (w).

5 Moving Robots

Next, we examine the channel association problem in which a subset of the robots move fast enough to cause connectivity changes during the execution.

5.1 Information Space and States

In the mobile case, the associated i-state and i-space structures and their respective sizes may change as the robots move. We assume that this change may occur every time step of the algorithm, thus, the association problem can be imagined as a two part process. One subprocess acts multiplicatively to reduce the number of false edges as in the stationary robot case. But now a second subprocess acts additively, introducing new edges to the process. Eq. 4 models these two processes:

$$T_t = (T_{t-1} \times m) + N_t. \tag{4}$$

Here T_t is the total number of edges for time step t, T_{t-1} denotes the number of edges in the previous step, and N_t is the number of newly added edges, with m, a multiplicative factor $m < 1$, giving the dependence of edges from one time step to the next.

5.2 Algorithm for Mobile Robots

The algorithm in Section 4.1 needs some modification to handle the dynamic nature of the moving robot's neighborhood. The evolving information state must distinguish between new and old IDs, and old IDs which are no longer in range should be removed. Unlike the stationary case, we do not assume that the full number of robots (IDs) which are in range is known in advance.

A new wireless ID enters an observer's range as soon as a message is received from it, which can only happen when the ID is both in range and active. Visual IDs, on the other hand, can enter the range of an observer before they activate since the robot can distinguish a neighbor even when it is not transmitting through the visual channel. But since adding a visual ID before it is activated brings no practical information, we may ignore visual IDs which are inactive and only add them to the observer's i-state once they activate. This, given our earlier assumptions, guarantees that there will be at least one candidate wireless ID to be associated with the visual ID.

Visual IDs can simply be removed when they leave the robot's range. This does not apply to wireless IDs since an observing robot cannot detect when a robot has left its wireless range. Thus, the observer needs to keep track of how long a wireless ID has been inactive. If, after a pre-set number of time steps, an ID has been inactive, it is dropped from the list. By choosing an appropriate drop time, the robot limits the number of IDs it tracks and also guards against losing too much information by dropping IDs too soon.

We note that it is possible for the algorithm to operate without dropping wireless IDs, but this runs the risk of creating unnecessary work with little or no benefit. For example, if a robot is likely to see most of the IDs in the system, but will only be in communication range with relatively few, then there is no need to keep old wireless IDs around. Since the algorithm only adds potential edges between active IDs, a wireless ID which has been encountered before but subsequently dropped, will be added back as soon as it is detected. Furthermore, visual IDs are dynamically assigned to the robots in range by the observing robot. This means that a wireless ID which re-enters communication range is unlikely to be matched with the same visual ID, making keeping the old wireless ID around even less useful.

A major change to Algorithm 1 is a procedure which modifies the i-state based on the list of new and removed IDs before it updates the i-state using the current time step's observation. The procedure removes rows and columns from the i-state corresponding to wireless and visual IDs which have exited their respective ranges. The procedure also adds new rows and columns into the i-state for new wireless and visual IDs. Once new IDs have been added, the appropriate edges between vertices need to be added to the i-state as well. Edges are added as follows. An edge is added between all new wireless and new visual IDs. In the case of equal wireless and visual range, no more edges are added. In the case of the wireless range being greater than the visual range, new edges are added between all new visual IDs and old, not yet associated, active wireless IDs. This is necessary because a robot can enter wireless communication range much sooner than it will enter the visual communication range of an observer. This means that a newly added visual ID may be matched with a wireless ID which has already been included in the i-state of the observing robot. In this work, we only examine the case where the wireless range is equal to the visual range. The pseudo-code for the mobile version of algorithm can be seen in Algorithm 2.

Algorithm 2 Mobile Robot Association

```
procedure PROBABILISTIC MOBILE
    $w; v$                         $\triangleright$ number of Wireless/Visual IDs in list
    $w\_ids; v\_ids$               $\triangleright$ list of IDs in i-state
    $w\_new; v\_new$               $\triangleright$ newly added IDs
    $w\_solved; v\_solved$         $\triangleright$ Solvedness Status of IDs
    $w\_activity; v\_activity$     $\triangleright$ Activity status of IDs
    $w\_inactivity;$               $\triangleright$ Time since last activation
    $drop\_time$                   $\triangleright$ Time steps before a W ID is dropped
    $w\_current; v\_current$       $\triangleright$ All IDs which are currently active
    $v\_lost$                      $\triangleright$ V IDs which exited the visual range
    $istate[h][s] \leftarrow 1's$
    $observ[h][s]$
    for each time step do
        if $fairCoinFlip() = heads$ then
            $broadcast(messagePair)$
        end if
        $w\_current \leftarrow receivedWifiMsgs()$
        $v\_current \leftarrow receivedVisMsgs()$
        $v\_lost \leftarrow exitedRangeVisIDs()$
        $removeVisIDs(v\_lost)$
                                   $\triangleright$ Remove the V IDs from data structures
                                   $\triangleright$ Mark any matched W ID as unsolved
        $removeWifIDs(drop\_time)$
                                   $\triangleright$ Remove W IDs which reach drop time limit
                                   $\triangleright$ Mark any matched V ID as unsolved
        $updateExisitingVisIDs(v\_current)$
                                   $\triangleright$ Update activity of existing V IDs
        $addNewVisIDs((v\_current)$    $\triangleright$ Add new V IDs to the structures
        $modifyIStateVis(v\_new)$      $\triangleright$ Expand the istate
        $updateExisitingWifIDs(w\_current)$
                                   $\triangleright$ Update activity of existing W IDs
        $addNewWifIDs(w\_current)$     $\triangleright$ Add new W IDs to data structures
        $modifyIStateWif(w\_new)$      $\triangleright$ Expand istate matrix
                                   $\triangleright$ Add 1's between new W IDs & new V IDs
        $modifyObservation$            $\triangleright$ Expands the observation matrix to match the istate
        $observ \leftarrow fillObs(wifi\_activity, vis\_activity)$
                                   $\triangleright$ update Observation matrix
        $istate \leftarrow updateIState(istate, observ)$
                                   $\triangleright$ Multiply the i-state and observation (by element) to create current i-state
    end for
end procedure
```

5.3 Performance Metrics

Since, in the mobile case, edges are added to the association process, the *expected time to completion* metric is not useful: the i-state will likely never achieve fully associated status. Instead we introduce a new metric—the *ratio of total to true edges*—giving a measure of how ambiguous the current i-state is by counting the average

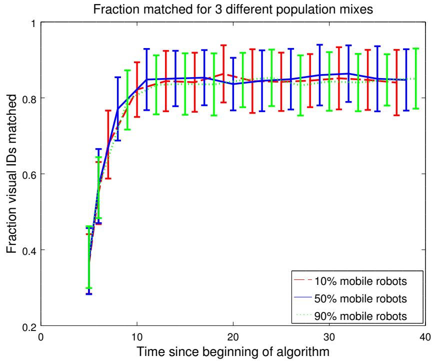


Figure 10: Plots of the fraction matched metric with time for 3 different population mix values. All other parameters are equal to the default values.

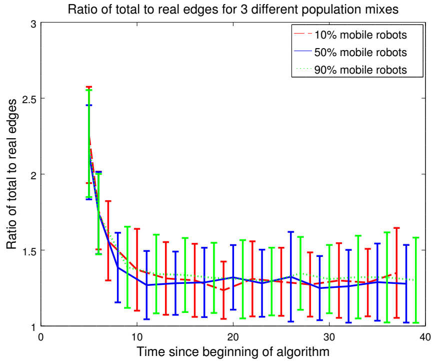


Figure 11: Plots of the ratio of total to true edges metric with time for 3 different population mix values. All other parameters are equal to the default values.

number of false edges per visual ID. A ratio of 1 indicates no ambiguity while a higher ratio means greater ambiguity. The *fraction matched* metric is still useful.

5.4 Experimental Setup

The multi-robot system we simulate is set up as a mixture of two types of robots: stationary and mobile. The majority of the robots are stationary, with fewer than 10% are mobile. They are placed on a grid in the environment with regular separation. The stationary robots are also uniformly perturbed from their grid position within half a grid spacing in either direction along each axis. This creates a random distribution of robots but maintains a uniform density in the environment. The mobile robots are also positioned uniformly and randomly in the entire environment and are given a random direction in which they move. All mobile robots share the same speed. They continue moving for the duration of the simulation. When they encounter the environment boundary, they change direction by bouncing off it. The mobile robots are used to collect data about the performance of Algorithm 2. Every other robot simply sends a message pair with 0.5 probability each time step.

We chose this population mix so that we can have a uniform density of robots in the environment and be able to model the path of robots through an observer's communication range as a straight path. This eases modeling of the association problem, and the few mobile-mobile robot interactions are ignored in the modeling process. (But we note that one example of a realistic multi-robot system with a similar population mix is in heterogeneous teams with fast aerial and slow surface-based vehicles.) For all the following experiments, the wireless and visual communication ranges are equal.

The initial conjecture that varying the fractions of mobile robots would have a significant impact on the algorithm's performance was incorrect as can be seen from Fig. 10 and Fig. 11 which show simulation data for 3 population mixes: 10%, 50%, 90% mobile robots. We briefly discuss potential causes in Section 6.2, however, in depth exploration of the lack of fractional dependence, is beyond the scope of the present paper.

5.5 System parameters & performance

Algorithm 2's performance is influenced by motion parameters because the multiplicative subprocess is affected by how much time the algorithm has had to operate on the edges and how many edges there are. The longer an observing robot can observe its neighborhood without any changes, the less ambiguity it will encounter. Conversely, the additive subprocess is affected by how many new IDs there are in range of a robot at every time step. As the number of new IDs increases the the i-state ambiguity increases. The following are key parameters:

1. Communication range size.
2. Robot density in the environment.
3. Robot speed through the environment.

For our experiments, we hold all but one parameter fixed and examine its effect for various values. Distances

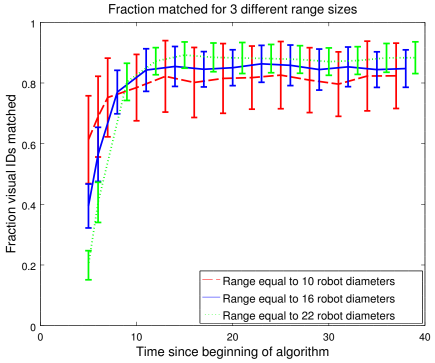


Figure 12: Plots of the fraction matched metric with time for 3 different range sizes. Range is given in robot diameters. All other parameters are equal to the default values. Red dashed line corresponds to 10 diameters, blue solid line to 16, and green dotted line to 22.

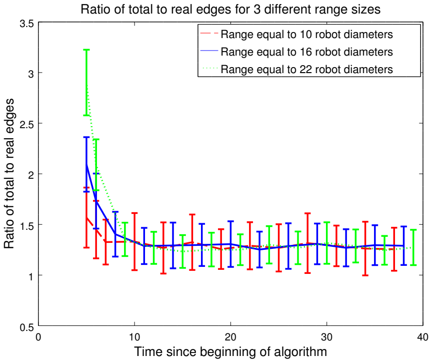


Figure 13: Plots of the ratio of total to true edges metric with time for 3 different range sizes. Range is given in robot diameters. All other parameters are equal to the default values. Red dashed line corresponds to 10 diameters, blue solid line to 16, and green dotted line to 22.

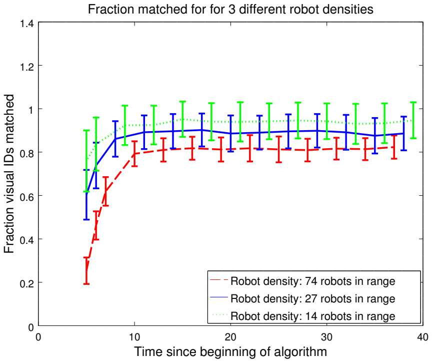


Figure 14: Plots of the fraction matched metric with time for 3 different density values. Density is given by the resultant average number of robots withing communication range. All other parameters are equal to the default values. Red dashed line corresponds to 74 robots in range, blue solid line to 27, and green dotted line to 14.

are given in multiples of the robot diameter. The following are the default values for each parameter:

- Robot speed — 1.4 robot diameters per time step.
- Robot communication range radius — 16 robot diameters for both wireless and visual range.
- Robot density in the environment — based on an average separation between robots' centers of 4 robot diameters.

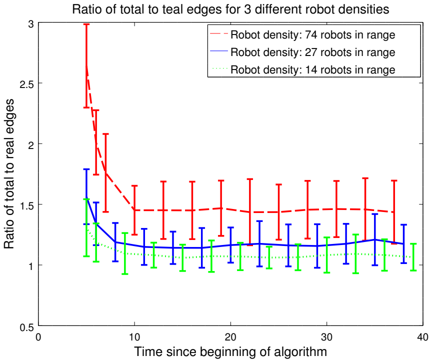


Figure 15: Plots of the total to true edges metric with time for 3 different range sizes. Density is given by the resultant average number of robots withing communication range. Red dashed line corresponds to 74 robots in range, blue solid line to 27, and green dotted line to 14.

The algorithm displays two distinct phases: a *transient initialization* phase and a *steady-state* phase. The transient phase results from the initial introduction of IDs when the robots wake up and start running the algorithm. The steady-state phase occurs once each robot has observed its neighborhood for a few time steps and the new IDs entering the i-state come only from a section at the front of the communication range.

Fig. 12 and Fig. 13 show the evolution of the two performance metrics for three different range sizes. The range sizes which have been plotted on the two figures are 10, 16 and 22 robot diameters.

Fig. 14 and Fig. 15 show the evolution of the two performance metrics when varying the robot density in the environment. The range radius for this set of experiments is 16 robot diameters, which combined with the average separation between robots of 8, 5, and 3.2 robot diameters gives an average of 14, 27 and 74 robots in range for each density tested.

Fig. 16 and Fig. 17 show the evolution of the two performance metrics as robot speed is varied. The robot speeds plotted in the two figures are $\frac{1}{8}^{th}$, $\frac{1}{11}^{th}$, and $\frac{1}{20}^{th}$ of the communication radius per time step. Equivalently these values are 2, 1.4 and 1.25 robot diameters per step.

5.6 Macroscopic Model

Next, we construct a macroscopic model for predicting the behavior of a moving mobile robot; the model is restricted to the case where the wireless range is equal to the visual range, and further work is needed to address cases where the wireless range exceeds the visual range. Fixing the communication range ratio to 1 removes the interaction between old and new IDs in the i-state. In effect the IDs which enter the i-state of an observing robot simultaneously form an independent smaller i-state. If we allow the wireless range to be larger than the visual, then the wireless ID corresponding to a robot may enter the i-state of an observer several time steps before its corresponding visual ID. This necessitates the addition of edges between all new visual and old but yet unassociated active wireless IDs.

5.6.1 Underlying Processes for the Mobile Case

The robots we model have circular communication ranges centered around them. Each time step the robot moves a particular distance d, which changes the area which is covered by the communication range. If the distance is less than the diameter of the range, there will be overlap between the areas covered by the communication range in two consecutive time steps. See Fig. 18

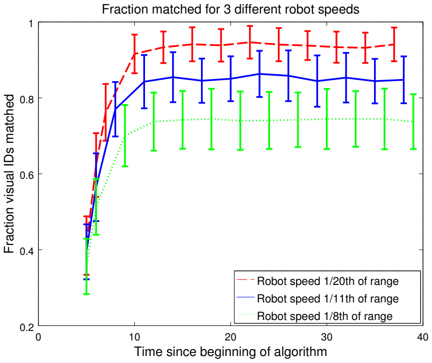


Figure 16: Changes in the fraction matched metric with time for 3 different robot speeds. Robot speed is given in fractions of the communication range radius. All other parameters are equal to the default values. Red dashed line corresponds to 1/20th of range, blue solid lines to 1/11th, and green dotted lines to 1/8th.

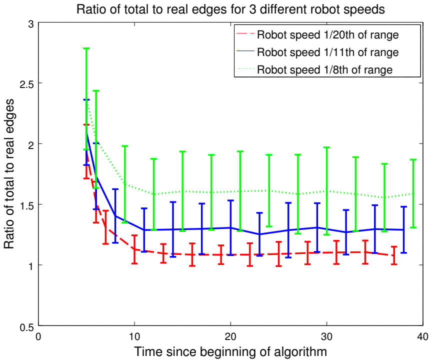


Figure 17: The changes in the total to true edges metric with time for 3 different robot speeds. Robot speed is given in fractions of the communication range radius. All other parameters are equal to the default values. Red dashed line corresponds to 1/20th of range, blue solid line to 1/11th, and green dotted line to 1/8th.

for a graphic representation of the overlap. This overlap can be computed as:

$$A_O = 2r^2 \cos^{-1}\frac{d}{2r} - d\sqrt{r^2 - \left(\frac{d}{2}\right)^2}. \tag{5}$$

The area newly covered by the communication range, which we call the *frontier*, can then be computed simply by subtracting the overlap area from the full circle area. The area which was covered in the previous time step but not in the current is called the *exit* area. It has the same size as the frontier, assuming the robot is moving in a straight line and does not change direction.

As the robot continues on its path, the region that its communication disk covers keeps changing. From the perspective of the robots, the previously covered area moves backward. If the distance covered each time step is small, there will be several regions in the communication range which would have spent different time *in range* of the observing robot. These areas are illustrated in Fig. 19. We term these areas, *age regions*. The age of a region indicates how many time steps it has been in range of the observer and how many times the algorithm has been applied to the IDs originating in that region.

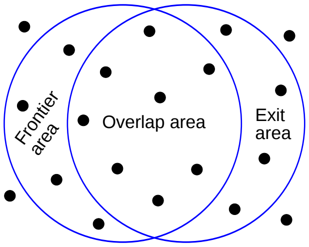


Figure 18: A schematic showing the overlap of two circles separated by a small displacement. The small circles represent robots. The frontier corresponds to a section of the environment which is newly within communication range. The exit area, on the opposite side, is a region that is no longer within communication range at the end of the time step.

The area of the age regions can be computed by applying Eq. (5) with different multiples of d to find each area. The area and the expected robot density in the environment can then be used to compute the expected population of robots in each region. This, in turn, helps yield the mean number of IDs which will enter each time step through the frontier region. We call the process of new IDs entering the communication range *positive migration*. The positive migration determines the size of the additive subprocess in the association problem. The IDs which exit the communication range determine the *negative migration*.

5.7 Fraction Matched Model

One may apply the stationary case fraction matched model to the mobile case when the two communication ranges are equal. As mentioned earlier, each region constitutes an independent i-state and its fraction of unambiguously associated visual IDs can be approximated using the *fraction matched* model. To get an idea of how many matched visual IDs there are in total in the full range, we just need to compute the fraction of visual IDs which have been unambiguously associated for each age region and then sum the results. Figures 20, 21, and 22

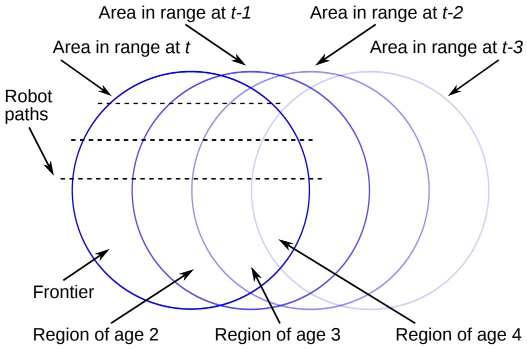


Figure 19: Geometry of the overlap between the area covered by the communication range of a moving robot at different time steps and the resulting distribution of regions by age. The blue dashed lines show the varying length of the paths of robots.

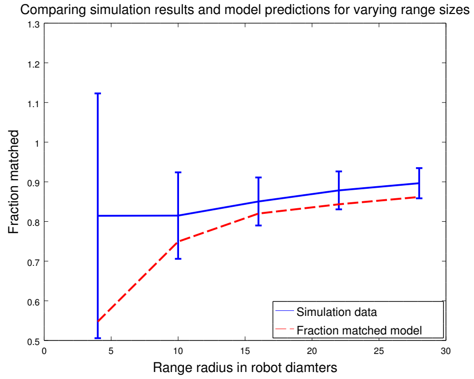


Figure 20: A comparison between simulation data for several range size values and the fraction matched model (in red, dashed line) prediction.

show how the predictions made by the *fraction matched* model compare with the results of the simulated trials.

It is evident that there is some inaccuracy in applying the *fraction matched* model to the mobile case of the association problem. One possible contribution to the inaccuracy is that the actual numbers of edges which enter the i-state at different time steps have some distribution around the mean, which affects the fraction visual IDs that get unambiguously associated. Another possible contribution is the loss of IDs from the i-state as the age region to which they belong shrinks. Further work is needed to explain why the *fraction matched* model underestimates by such a degree.

5.8 Ratio of Total to True Edges Model

5.8.1 Extending the Branching Processes Model

To predict the expected ratio of total to true edges, we extend the BP model from Section 4.3.3 to handle the additional additive subprocess. To achieve this one must incorporate both the positive and negative migration in the model. The Branching Processes literature offers ways of handling migration as well.

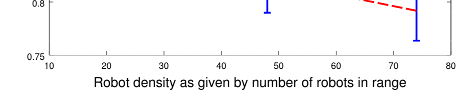


Figure 21: A comparison between simulation data for several density values and the fraction matched model (in red, dashed line) prediction.

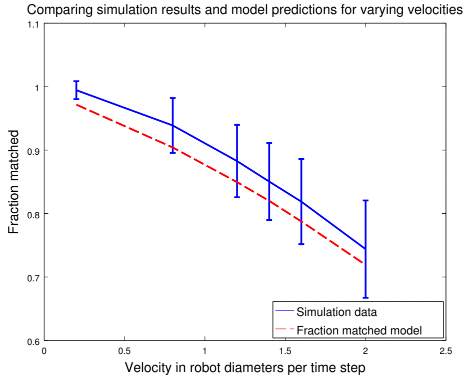


Figure 22: A comparison between simulation data for several velocity values and the fraction matched model (in red, dashed line) prediction.

As in the stationary case, we need to take into account the mean reproduction rate of the two types of edges we have. These will be handled by the variables m_F and m_T. In the mobile case, the values of the mean reproduction rate are no longer 0.5 and 1 for the false and true edges, respectively. Now we need to add the contribution due to negative migration. This will be computed as a correction factor to the mean reproduction rates based on how IDs exit the communication range. The two negative migration factors will be denoted with p_{F-} and p_{T-}, which are the probability of an edge *migrating out*. This is determined by the probability of one or both of the IDs composing the edge moving out of communication range.

In addition to the reproduction rate of the individual edges, we also need to incorporate the positive migration of edges. These will be denoted with λ_{F+} and λ_{T+} for positive migration of false and true edges.

The above parameters can be used to construct the *mean matrix* M of the branching process, Eq. (6). This matrix can be used to compute the expected number of individuals of each type given a starting population P_i and a number of generations (time steps) t, Eq. (7).

The mean matrix is constructed as follows:

$$M = \begin{bmatrix} 1 & \lambda_{T+} & \lambda_{F+} \\ 0 & m_T(1-p_{T-}) & 0 \\ 0 & 0 & m_F(1-p_{F-}) \end{bmatrix}. \tag{6}$$

Here a population vector is defined as follows:

$$P_i = \begin{bmatrix} 1 & e_T & e_F \end{bmatrix}, \tag{7}$$

where:

- the first element in the matrix and population vector stands for a *migrator individual*, which is a mathematical tool used to introduce the additive positive migration,
- e_T is the number of true edges in the i-state,
- e_F is the number of false edges in the i-state,

- i is the time step during which the population was counted.

The mean matrix is applied to the initial population to get the resulting population after t time steps as follows:

$$P_{i+t} = M^t \times P_i. \tag{8}$$

The Branching Processes literature also offers an approximation for the expected number of individuals of each type if the mean migration and reproduction rates remain constant. The approximation for a single individual type with $m < 1$ is summarized in the following equation:

$$E[P_t] = \frac{\lambda}{1-m}. \tag{9}$$

5.8.2 Converting System Parameters to Branching Processes Parameters

Next, we expand upon the construction of two specific Branching Processes models and describe how to convert the robot and environment parameters into Branching Processes parameters. In Section 5.6.1 we explained how the motion of a robot through the environment creates areas inside the communication range which have spent different times *in range* of an observing robot. We specifically distinguished the frontier region, Fig. 19. This section describes two BP models built around the idea of different treatment of the remaining age regions. The first model ignores the distinction between age regions and treats all areas which are not the frontier as one region — we call this region the *rest*. The second model explicitly models each age region and the transition of robots and IDs from one region to the next. We call the first model the *two-region model* and the second the *multi-region model*.

In the two-region model we get a mean matrix as described in Eq. (6) of size 3×3. The mean positive migration for each edge type is calculated using the area of the frontier region and the mean density of robots in the environment. For the false edges $\lambda_{F+} = w_{\text{new}} v_{\text{new}} - v_{\text{new}}$ and for the true edges $\lambda_{T+} = v_{\text{new}}$. In order for the model to give accurate results, one must take into account the variance of w_{new} and v_{new}. In the case of equal wireless to visual range the two numbers are equal and their variance is identical. So, the value of the expected positive migration of false edges, after accounting for variance, becomes:

$$\lambda_{F+} = \mathrm{Var}[v_{\text{new}}] + E[v_{\text{new}}]^2 - E[v_{\text{new}}]. \tag{10}$$

We gathered data from the simulation for the variance of the expected number of new visual and wireless IDs. This can also be done by modeling the spatial variance in the position of the robots in the environment, though this modeling was not attempted in this work.

The negative migration probability is calculated as:

$$p_{T-} = p_{F-} = \frac{v_{\text{exit}}}{v_{\text{rest}}} + \frac{w_{\text{exit}}}{w_{\text{rest}}} - \frac{v_{\text{exit}} w_{\text{exit}}}{v_{\text{rest}} w_{\text{rest}}}. \tag{11}$$

This is motivated by the fact that each edge in the *rest* area will migrate out if one or both of its IDs migrate out. There are v_{rest} IDs of each type in the *rest* area and, on average, about v_{exit} IDs of each type exit the range. The number of exiting IDs is computed in way similar to how the population of frontier IDs is computed. One obtains a probability of exiting the *rest* area which is given by the mean number of IDs exiting the range over the mean number of IDs in the *rest* area.

In the multi-region model we explicitly account for the movement of IDs and edges through each region. As can be seen in Fig. 19, the size of each age region shrinks with increasing age. Since false and true edges are independent of one another, we can split the mean matrix into one dedicated for each edge type. In addition, the mean matrix now has a larger size, being equal to the total number of age regions plus 1 (for the migrator). One gets a correction factor for the mean reproduction rate based on how the area of two adjacent region compares. For example, the mean reproduction rate for edges going from an age region with an area $A_i = 10$ to an age region with an area $A_j = 9$ will need to be adjusted by a factor of $\frac{A_j}{A_i} = 0.9$. The positive migration for the multi-region model is still accounted for by the number of new IDs in the frontier region. Also, the positive migration is added only to the frontier region's count of edges. At each time step, the surviving edges (those which were not eliminated by observation and those whose IDs remained in range) of each age region are moved to the next age region. All the edges in the last age region exit the i-state once the robot moves.

An example mean matrix for false edges in the multi-region model can be seen in Eq. 12. The false edge population of the communication is defined in Eq. 13. A similar matrix and population can be created for the true edges:

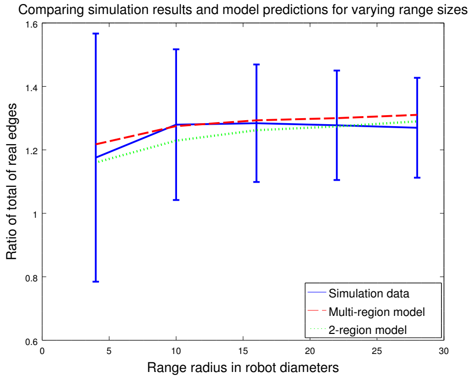


Figure 23: A comparison between simulation data for several range size values and 2-region model (in green, dotted line) and multi-region model (in blue, dashed line) predictions.

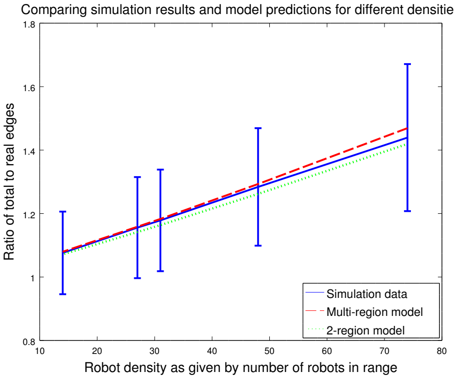


Figure 24: This figure shows a comparison between simulation data for several density values and 2-region model (in green, dotted line) and multi-region model (in blue, dashed line) predictions.

$$\begin{bmatrix} 1 & \lambda_{F+} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{A_2}{A_1} m_{F-} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{A_3}{A_2} m_{F-} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{A_4}{A_3} m_{F-} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{A_5}{A_4} m_{F-} \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{12}$$

$$P_i' = \begin{bmatrix} 1 & e_{A1} & e_{A2} & e_{A3} & e_{A4} & e_{A5} \end{bmatrix}. \tag{13}$$

When computing the steady-state results for the performance metrics we did not use the approximations from the Branching Processes literature, instead we chose a starting population and applied the mean matrix to it for a (large) predetermined number of time steps. Then we used the values to which the population tends with time as our steady-state prediction. These match well with the approximation results.

Figures 23, 24, and 25 show the predictions of the two models (green for the two-region model, and blue for the multi-region model) and how they compare to the data gathered from simulation. Both models track the simulation mean closely and fall well within 1 standard deviation from the mean.

6 Future Work & Discussion

6.1 Larger Wireless than Visual Range

This paper only examines the case of equal wireless and visual communication ranges, allowing the mobile case algorithm to take advantage of properties which exist only in this restricted instance. More specifically, wireless and visual IDs belonging to the same robot enter the i-state during the same time step. When the assumption of equal ranges is relaxed, a wireless ID may enter the i-state much sooner than the corresponding visual ID. This causes IDs from different age regions to interact and form edges—an edge must be added between all active unassociated wireless IDs and newly discovered visual IDs. Additional work is required to model how new edges are added in this case.

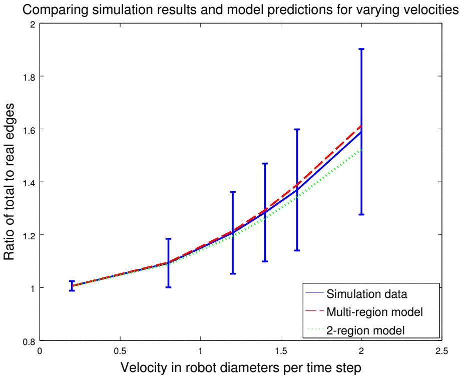


Figure 25: This figure shows a comparison between simulation data for several robot velocity values and 2-region model (in green, dotted line) and multi-region model (in blue, dashed line) predictions.

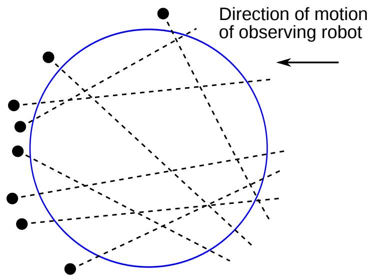


Figure 26: A diagram of potential trajectories of mobile robots passing through the communication range of a mobile observing robot. The smaller circles correspond to robots moving randomly through the environment and the greed lines denote their potential trajectories. The length of these trajectories no longer depends simply on the distance between the robot and the line of motion of the observer.

6.2 Different Population Mixes

The simulation results show that varying the ratio of stationary to mobile robots in the environment has little impact on the association performance as measured by the steady-state mean and variance. Stationary robots travel through the communication range of a mobile robot in parallel lines along the direction of motion of the observer. The trajectories of mobile robots through the communication range of an observer are much more varied as is illustrated in Fig. 26. Additionally, the time that a mobile robot spends in range of an observer now depends on the vector sum of their speeds. Modeling why the different geometry of the paths of robots through the communication disk of an observer for mobile robots produces similar results to the stationary robots requires further research.

6.3 Noise and Errors in Communication

The communication of the multi-robot system in this work is assumed to be ideal *i.e.,* there is no noise, dropped messages or errors (Assumption 1). A more practical algorithm for solving the association problem would require taking into account the uncertainty of observed robot activity. The edges corresponding to potential associations can no longer be eliminated based on a single or even multiple observations. In the case of ideal communication, true and false edges have different ratios of being active to being inactive, which results in a clear distinction between them. This distinction exists because a true edge's activity depends on 1 random event, while that for a false edge depends on 2 independent random events. In our case, true edges will be active on average during $\frac{1}{2}$ of the time steps, while false edges will be $\frac{1}{4}$ of the time.

One practical strategy for distinguishing between true and false edges is to simply make observations for some period of time. This strategy may be effective in the case of unreliable communication, but has the disadvantage of being costly (in terms of time).

The expected average time for the ideal case is better used to find a bound so that, along with information on the noise, one may compute thresholds that distinguish between true positives and true negatives, appropriately. Combining knowledge of the uncertainty in the communication with an understanding of the expected performance in the ideal case, can give good results in predicting the expected performance of a system executing the algorithm. Our work is a step which lays the foundation for analysis of this approach. Further work is certainly needed to apply the analysis and models presented in this work to systems with unreliable communication.

6.4 Time Step Duration & Channel Bandwidth

We offer a brief informal discussion of the time step duration and the feasibility of unlimited bandwidth and message processing (Assumption 3). Mathews et al. [2015] use a 200 ms time step and Pulli et al. [2012] define real-time visual processing in the 30–40 ms range. Hence we can conservatively set our time step to 500 ms. The algorithms only need to send 100 bytes: 16-byte IPv6 address, 60-byte IP header, 24-byte TCP header. Additional padding may double the size to 200 bytes. Kwak et al. [2005] provide asymptotic behaviors of the binary exponential backoff algorithm. Equation (34) with $r = 2$ gives us an asymptotic normalized throughput of 0.347 and equation (36) gives us a dependence between the median access delay and the number of active wireless nodes. A 2003, 802.11g and a low-power 802.15.4 card have up to 54Mbit/s and 250 kbit/s bandwidths, respectively. If we require that the median access delay is half of the available throughput, we get the following results. During each 500 ms time step the the first card can handle up to 2000 transmissions successfully while the low-power card can handle up to 9.

6.5 Message Transmission Probability

In our work we use a probability of 0.5 for sending a message pair. A different choice can be made, but the probability has multiple subtle impacts. First, the probability determines the average number of messages sent each time step; lowering the probability reduces the load on the communication channels. Second, the probability determines how likely one is to split the observed robots into even groups based on active/inactive status. The further the probability from 0.5, the lower the chance of an even split, slowing the edge elimination process. Third, the probability determines the average ratio of active time steps to total time steps for true and false edges. The difference between the two ratios (and the ease with which they can be distinguished in the presence of noise) is greatest when the transmission probability is 0.5.

7 Summary and Conclusion

Robots as part of a multi-robot team often need to address conveniently located neighbors. A common approach to achieve this is to give the robots external markers which are paired to a network address. This work identified and provided a detailed formal description of the communication channel association problem in which a robot associates its neighbors to their network addresses algorithmically, without the use of external markers or global information. The algorithms we described for computing this association use visually identifiable gestures (such as a light being turned on or off) and wireless messages to quickly establish the one-to-one association between the wireless IDs of neighboring robots and their locally assigned visual IDs. The gestures along with the sensors used to detect them form a situated communication channel. The approach exploits the physically situated property of this channel to create a meaningful to the sensing robot association between neighboring robots and their wireless IDs.

This paper identified key system parameters —density of robots in the environment, communication range, and robot speed— and explored their effect on the performance of the algorithms. We also discovered that a fourth system property —ratio of stationary to mobile robots in the environment— has, somewhat surprisingly, no significant effect on performance. We showed that a common modeling approach for similar problems, using absorbing Markov chains, is too expensive to apply to realistic instances of the association problem. So, instead, we introduced a set of models based partly on Branching Processes to predict the macroscopic behavior of a multi-robot system running our algorithms. These models can be used to evaluate the feasibility of using the probabilistic algorithms to solve the association problem for a multi-robot system with given parameters or during the process of designing a system to achieve some desired level of performance.

Acknowledgements

This work was partially supported by National Science Foundation under award number IIS-1453652, for which we are grateful.

References

P. E. Agre and D. Chapman. Pengi: An implementation of a theory of activity. In *Proceedings of the Sixth National Conference on Artificial Intelligence - Volume 1*, AAAI'87, pages 268–272, Palo Alto, CA, USA, Jul 1987. AAAI Press.

M. A. Batalin and G. S. Sukhatme. Spreading out: A local approach to multi-robot coverage. In *Distributed Autonomous Robotic Systems 5*, pages 373–382. Springer Japan, Tokyo, Japan, Jun 2002.

D. Berend and T. Tassa. Improved bounds on Bell numbers and on moments of sums of random variables. *Probability and Mathematical Statistics*, 30(2): 185–205, 2010.

Y. Dieudonné, S. Dolev, F. Petit, and M. Segal. Deaf, dumb, and chatting asynchronous robots. In *Proceedings of the 13th International Conference on Principles of Distributed Systems*, OPODIS'09, pages 71–85, Berlin, Germany, Dec 2009. Springer-Verlag.

D. Fox, W. Burgard, H. Kruppa, and S. Thrun. A probabilistic approach to collaborative multi-robot localization. *Autonomous Robots*, 8(3):325–344, Jun 2000.

A. Franchi, G. Oriolo, and P. Stegagno. Mutual localization in a multi-robot system with anonymous relative position measures. In *Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IROS'09, pages 3974–3980, Piscataway, NJ, USA, Oct 2009. IEEE Press.

S. Garrido-Jurado, R. Muñoz Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280–2292, Jun 2014.

C. M. Grinstead and J. L. Snell. *Introduction to Probability*, chapter 11: Markov Chains. American Mathematical Society, Providence, RI, USA, Jul 1997.

Á. Gutiérrez, A. Campo, M. Dorigo, D. Amor, L. Magdalena, and F. Monasterio-Huelin. An open localization and local communication embodied sensor. *Sensors*, 8(11):7545–7563, Nov 2008.

P. Haccou, P. Jagers, and V. A. Vatutin. *Branching Processes: Variation, Growth, and Extinction of Populations*. Cambridge University Press, Cambridge, UK, Nov 2007.

A. Howard, L. E. Parker, and G. S. Sukhatme. The sdr experience: Experiments with a large-scale heterogeneous mobile robot team. In *Experimental Robotics IX: The 9th International Symposium on Experimental Robotics*, pages 121–130. Springer Berlin Heidelberg, Heidelberg, Germany, Feb 2006.

P. Ivanov and D. A. Shell. Associating nearby robots to their voices. In *Proceedings of The Fourteenth International Conference on the Synthesis and Simulation of Living Systems (ALIFE 14) 2014*, pages 274–281, Cambridge, MA, USA, Jul 2014. MIT Press.

B. Kwak, N. Song, and L. E. Miller. Performance analysis of exponential backoff. *IEEE/ACM Transactions on Networking*, 13(2):343–355, Apr 2005.

S. M. LaValle. Tutorial: Filtering and planning in information space. In *Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IROS'09, Piscataway, NJ, USA, Oct 2009. IEEE Press.

Y. Lespérance and H. J. Levesque. Indexical knowledge and robot action - a logical account. *Artificial Intelligence*, 73(1–2):69–115, Feb 1995.

N. Mathews, A.L. Christensen, R. O'Grady, and M. Dorigo. Spatially targeted communication and self-assembly. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IROS'12, pages 2678–2679, Piscataway, NJ, USA, Oct 2012. IEEE Press.

N. Mathews, G. Valentini, A.L. Christensen, R. O'Grady, A. Brutschy, and M. Dorigo. Spatially targeted communication in decentralized multirobot systems. *Autonomous Robots*, 38(4): 439–457, Apr 2015.

E. Olson. Apriltag: A robust and flexible visual fiducial system. In *2011 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3400–3407, Piscataway, NJ, USA, May 2011. IEEE Press.

K. Pulli, A. Baksheev, K. Kornyakov, and V. Eruhimov. Real-time computer vision with opencv. *Communications of the ACM*, 55(6):61–69, Jun 2012.

K. Støy. Using situated communication in distributed autonomous mobile robotics. In *Proceedings of the Seventh Scandinavian Conference on Artificial Intelligence*, SCAI'01, pages 44–52, Amsterdam, The Netherlands, Feb 2001. IOS Press.