Dominating Sets of Agents in Visibility Graphs: Distributed Algorithms for Art Gallery Problems

Evan A. Sultanik
Drexel University
3141 Chestnut Street,
Philadelphia, PA 19104
eas28@cs.drexel.edu

Ali Shokoufandeh
Drexel University
3141 Chestnut Street,
Philadelphia, PA 19104
ashokouf@cs.drexel.edu

William C. Regli
Drexel University
3141 Chestnut Street,
Philadelphia, PA 19104
regli@cs.drexel.edu

ABSTRACT

The Art Gallery Problem asks to find a minimum subset of vertices in a polygon that are sufficient to observe the interior. This problem arises in a variety of multiagent systems, including robotics, sensor networks, wireless networking, and surveillance. Despite the fact that the centralized version of the problem has been extensively studied for the past thirty years, there is relatively little in the literature describing distributed solutions to the problem that have desirable guarantees in both runtime and optimality. We propose and analyze a new distributed algorithm for approximating a solution to this problem and a number of its variants that runs in a linear number of communication rounds with respect to the number of nodes (independent of the topology of the network), and, under assumptions on the embedding of the edge weights, will run in a logarithmic number of communication rounds producing solutions within a constant factor of optimal.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Intelligent agents, Multiagent systems*; G.1.2 [**General**]: Approximation

General Terms

Algorithms, Theory

Keywords

Distributed Problem Solving, Art Gallery Problem, Dominating Set Problem, Multi-Robot Coordination, Sensor Networks, Primal/Dual

1. INTRODUCTION

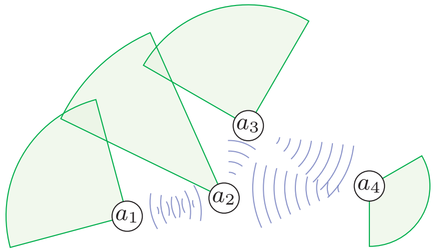
Art gallery problems generally ask to find the minimum number of guards required to observe the interior of a polygonal area [12]. Over the past thirty years since their proposition, these problems have been thoroughly studied by the computational geometry community. Interest in art gallery problems has seen a recent resurgence given their application to a number of areas of multiagent systems. For example, many robotics, sensor network, wireless networking, and surveillance problems can be mapped to variants of the art gallery problem. Since such problems are naturally distributed, a logical approach is to apply the multiagent paradigm (*i.e.*, each guard is an agent).

As a motivating scenario, consider a wireless sensor network such as the one pictured in Figure 1. Since one goal of the network is to maximize survivability, it may be desirable to conserve battery power by having as few sensors active as necessary, especially for sensors with wide overlapping fields of view. The problem is then to find a minimum subset of sensors that need to remain active in order to provide a desirable level of coverage. As another scenario, consider a group of mobile robots each equipped with a wireless access point. The objective of the robots is to maximally cover an area with the wireless network. As the robots are traveling between waypoints, though, it is highly likely that there will be a large amount of overlap in the coverage. Therefore, in order to save power, the robots might want to choose a maximum subset of robots that can lower their transmit power while still retaining coverage. The difficulty in each of these scenarios is for the agents to collectively find the solution without relying on centralization of computation. Centralization is infeasible either due to lack of resources (*i.e.*, no single agent has powerful enough hardware to solve the global problem) or due to lack of time (*i.e.*, centralizing the problem will take at least a linear number of messaging rounds). These problems are $\mathcal{NP}$-COMPLETE and can be modeled as art gallery problems.

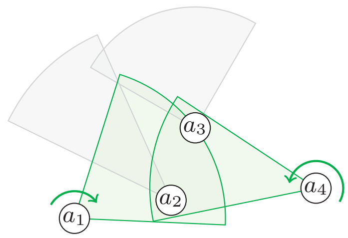
Solving art gallery problems using multiagent systems is not a new idea. We have previously applied the multiagent coordination paradigm of Distributed Constraint Optimization (DisCOP) to a variant of the problem in which a fixed number of robotic guards must patrol a polygonal area [9]. The difficulty with using DisCOP, however, is that all known algorithms that provide a constant bound on the quality of the solution will in the worst case be exponential in either messaging or memory [11]. Whereas DisCOP is a general problem solving paradigm, Ganguli, *et al.*, developed a multiagent algorithm specifically for solving art gallery problems [4]. This algorithm has several desirable properties including optimality, however, there is no theoretical bound on runtime.

The dominating set problem is a generalization of the art gallery problem that asks to find a minimum subset of the vertices in a graph such that every vertex not in the subset

Cite as: Dominating Sets of Agents in Visibility Graphs: Distributed Algorithms for Art Gallery Problems, Evan A. Sultanik, Ali Shokoufandeh, and William C. Regli, *Proc. of 9th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2010)*, van der Hoek, Kaminka, Lespérance, Luck and Sen (eds.), May, 10–14, 2010, Toronto, Canada, pp. 797-804



(a) An *ad-hoc* sensor network wirelessly coordinating to optimize interior coverage.



(b) Dynamically-generated orientation.

Figure 1: In (a), an ad-hoc sensor network must distributedly reorient. In (b), agents a_1 and a_4 rotate to guard the interior.

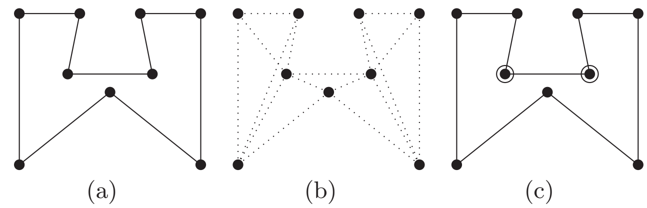


Figure 2: An art gallery, (a), with its associated visibility graph, (b), and an optimal placement of guards, (c). Guard placement is represented by ◉.

has at least one member of the subset in its neighborhood. This problem is also $\mathcal{NP}$-COMPLETE. The dominating set problem has been widely studied in the wireless networking community given its applications to *ad hoc* routing [20]. The majority of the proposed distributed algorithms for the dominating set problem, however, do not have bounds on both runtime and solution quality. In the wireless networking community much emphasis is placed devising algorithms with a constant number of communication rounds. Ruan, *et al.*, propose a one-step greedy algorithm for approximating a solution to the dominating set problem [15], however, the performance ratio is a function of the degree distribution of the graph. Kuhn and Wattenhofer provide a more general result, producing an algorithm that has a variable approximation bound as a function of the number of communication rounds executed. Kuhn and Wattenhofer's approach, however, is likewise tied to the degree distribution of the graph. Finally, Huang, *et al.*, show that with a slightly higher message complexity a solution no worse than 12 times the cost of the optimal solution can be found [7]. This paper introduces an algorithm based on the primal/dual schema that exhibits a lower constant of approximation in worst case linear—but often logarithmic—number of communication rounds.

Proposing distributed algorithms using the primal/dual schema has been the subject of study of a number of recent results [13]. Problems that have been studied using this schema include Steiner problems [16], point-to-point connectivity problems [2], distributed scheduling [14], vertex cover [8, 6], and facility location [17]. To the best of the authors knowledge, however, there is no prior result which theoretically proves a constant bound on the cost of the optimal solution and bounds the number of communication rounds sublinearly.

This paper introduces a novel distributed version of a multiagent approximation algorithm based on the primal/dual schema for solving the distributed art gallery and dominating set problems (§2). We show that this algorithm is correct and complete and bound its runtime with respect to communication rounds (§2.2). We then show through empirical analysis that the algorithm will produce solutions within a constant factor of optimal with high probability (§2.3). We then show that some well known variants of the problem can also be solved with the same algorithm and, under certain reasonable assumptions about the distribution of edge weights, the algorithm will produce a solution no worse than two times optimal (independent of the topology of the problem) (§3). Conclusions and future work are provided in §4.

2. DISTRIBUTED DOMINATING SETS

This section defines an algorithm for solving the distributed dominating set problem, which is equivalent to the original art gallery problem of finding a minimum set of vertices from which the entire polygon is visible.

2.1 Problem Formalization

Given two vertices of a polygon u and v, u is said to be *visible* from v if the line segment between them is contained within the polygon, *& vice versa*. The exterior of the polygon is forbidden for visibility graph edges. Given the vertices of a polygon, V, the *Art Gallery Problem* asks to find a minimum subset of the vertices $D \subseteq V$ such that for every $v \in V$ there is at least one $d \in D$ that is visible. The *visibility graph* of a polygon is constructed by adding an edge between all pairs of vertices that are visible to each other. For example, see Figure 2(b). The Art Gallery Problem therefore reduces to finding a dominating set of the vertices in the polygon's visibility graph. Given a visibility graph $G = \langle V, E \rangle$, the object is to find a $D \subseteq V$ of minimum cardinality such that each $v \notin D$ has at least one $d \in D$ in its neighborhood. An example is given in Figure 2, with an optimal solution depicted in Figure 2(c).

The analysis of the dominating set problem can be simplified by representing it as a connectivity problem. Therefore, let us augment the visibility graph with one special *guard* vertex d_i for each original vertex v_i. Next, add an edge from each vertex to its associated guard vertex with a weight of one. The new overall set of edges is the original set of edges from the visibility graph unioned with the set of new guard edges. All original edges from the visibility graph are given a weight of zero. Let $R = \{v_1, \ldots, v_n\}$ be the set of original vertices in the visibility graph and let $T = \{d_1, \ldots, d_n\}$ be the set of new special guard vertices with the new overall set of vertices $V = R \cup T$. Now the problem reduces to that of finding a minimum weight forest that spans R having the property that the length of the shortest path from any $v \in R$ to a $d \in T$ is no more than two edges. We will hereafter refer to this forest as "the spanning forest". Note, however, that the spanning forest does not necessarily span all of T.

In this new connectivity representation, a vertex v_i will be a part of D (*i.e.*, it will be chosen to become a guard) if the edge from it to its associated guard vertex is a part of the final spanning forest: $\langle v_i, d_i \rangle \in F \implies v_i \in D$. Let $g : 2^V \to \{0,1\}$ be a function defining whether a connected component $S \subseteq V$ satisfies the requirement that each vertex is close to at least one guard. g is defined such that $g(S) = 1$ if and only if there exists an original vertex in S that is not within two edges distance of a guard in S: $g(S) = 1 \iff \exists u \in S \cap R \, \forall v \in S \cap T : |u \rightsquigarrow v| > 2$. A component for which $g(S) = 1$ is said to be *unguarded*. The optimization problem on the augmented graph can be captured as the following integer program:

$$\begin{aligned}
&\text{minimize} \sum_{e \in E} w(e) x_e \\
&\text{subject to:} \qquad\qquad\qquad\qquad\qquad\qquad \text{(IP)}\\
&\qquad x(\delta(S)) \geq g(S), \quad \forall S \subset V : S \neq \emptyset \\
&\qquad x_e \in \{0,1\}, \qquad\qquad \forall e \in E,
\end{aligned}$$

where each variable x_e is an indicator as to whether the edge e is a member of the final spanning forest, $\delta(S)$ is the set of edges having exactly one endpoint in S, and $x(F) \mapsto \sum_{e \in F} x_e$. Therefore, any forest $F \subseteq E$ will be a feasible solution to the problem if $g(S) = 0$ for every connected component S of the forest. Let (LP) denote the linear programming relaxation of (IP) obtained by replacing the integrality restriction with $x_e \geq 0$. The dual of (LP) is

$$\begin{aligned}
&\text{maximize} \sum_{S \subset V} g(S) y_S \\
&\text{subject to:} \qquad\qquad\qquad\qquad\qquad\qquad \text{(D)}\\
&\qquad \sum_{S : e \in \delta(S)} y_S \leq w_e, \qquad \forall e \in E \\
&\qquad y_S \geq 0, \qquad\qquad \forall S \subset V : S \neq \emptyset.
\end{aligned}$$

An edge is *tight* if $w(e) = \sum_{S : e \in \delta(S)} y_S$. Let Z^*_{LP} be the cost of the optimal solution to (LP) and let Z^*_{IP} be the cost of the optimal solution to (IP). It is a folklore result that $Z^*_{\text{LP}} \leq Z^*_{\text{IP}}$.

2.2 The Algorithm

The basic mechanism of the algorithm is quite simple. We start off with an empty forest; each vertex is a member of its own connected component. Every round, each unguarded component greedily chooses to add one of its cut edges in the visibility graph to the forest, merging with the component on the other end of the edge. If the new component becomes guarded as a result of the merger then the new component stops actively growing. This has the effect of first finding a forest that spans the original visibility graph; then each connected component in the forest finds the minimum set of special vertices that is sufficient to be guarded. When all components are guarded the algorithm terminates.

The remainder of this section provides the notation and mathematics required to formally define and model the algorithm. This will later be used to provide formal bounds on the runtime and performance of the algorithm, and also to prove correctness and completeness.

We assume that the communications network provides guaranteed delivery of messages, however, there may be arbitrary latency (*i.e.*, the network is asynchronous [10]). Without loss of generality, we assume that there is one intelligent agent per vertex in the graph. We further assume that all agents are honest and correct and thus need not consider the problem of Byzantine failure. The agents are non-adversarial insofar as their primary goal is to find a feasible solution to the art gallery problem. The collective is therefore a *cooperative multiagent system* [18]. Agents' perceptions of the visibility graph are consistent, possibly through the use of a distributed consensus algorithm [10]. Each agent/vertex has a unique identifier with a globally agreed ordering. This ordering can be used to construct a total ordering over the edges (*e.g.*, by combining the unique identifiers of the incident vertices).

The proposed multiagent algorithm is round-based. The rounds proceed asynchronously between connected components. Therefore, as the connected components grow throughout the execution of the algorithm, the rounds naturally become synchronized.

Let F_t be the partially constructed spanning forest at the beginning of round t. Let $\mathcal{C}_t$ be the set of connected components in F_t. For sake of brevity and simplicity, let $\mu_t : V \to \mathcal{C}_t$ be a function mapping vertices to their associated connected component during round t; therefore, $\mu_t(v) \mapsto C_i \implies v \in C_i (\in \mathcal{C}_t)$. A vertex that is incident to at least one edge in the cut of its connected component is said to be in the *fringe*. Let $b_t : V \to \mathbb{R}$ be a mapping of vertices to a real number during round t. These values represent the amount of slack remaining in the dual variables associated with a vertex.

Let $J_t : V \times V \to \{0,1\}$ be a binary relation defining which edges will become tight during round t. Each unguarded component will choose to add the edge in its fringe that has minimal weight and dual variable slack. Therefore, $J_t(u,v) = 1$ if and only if $g(\mu_t(u)) = 1$ and

$$\langle u, v \rangle = \operatorname*{arg\,min}_{\langle i,j \rangle \in \delta(\mu_t(u))} w(\langle i,j \rangle) - b_t(i) - b_t(j). \tag{1}$$

Ties in the minimization are broken based upon the ordering of the edges. Let J^+ denote the transitive closure of J. Note that J does not commute: $J(u,v) \not\Longrightarrow J(v,u)$. Also note that as long as there exists a feasible solution to (IP) then the minimization ensures that each unguarded connected component must have exactly one edge in the fringe that becomes tight each round: $\forall C \in \mathcal{C}_t : g(C) = \sum_{\langle u,v \rangle \in \delta(C)} J_t(u,v)$.

F_t is the partially constructed spanning forest during round t, initialized to $F_0 = \langle V, \emptyset \rangle$. The forest is updated each round with the set of all edges that became tight during the round: $F_{t+1} = F_t \cup \{\langle u,v \rangle \in E : J_t(u,v) \vee J_t(v,u)\}$.

For a set $S \subseteq V$, let y_S be the dual variable associated with S. Initially all such variables are set to zero. Note that in actuality these variables need not be made part of an implementation of the algorithm; they exist solely for the purpose of proving properties of the algorithm [5]. These dual variables are implicitly updated as follows:

$$y_S \leftarrow \begin{cases} \frac{w(\langle i,j \rangle) - b_t(i) - b_t(j)}{1 + J_t(j,i)} & \text{if } \exists i \in S \in \mathcal{C}_t, j \notin S : J_t(i,j), \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

The b values are initialized such that $\forall v \in V : b_0(v) = 0$.

They are updated each round such that

$$b_{t+1}(v) = b_t(v) + y_{\mu_t(v)}. \tag{3}$$

The value $b_t(v)$ can therefore be interpreted as the amount of slack remaining in the dual variables during round t before an edge incident to vertex v becomes tight.

Let τ be the number of rounds required for the algorithm to reach quiescence. Therefore, τ is the earliest round during which there are no unguarded components:

$$\tau = \operatorname*{arg\,min}_{t \in \mathbb{N}^*} \left(\forall C \in \mathcal{C}_t : g(C) = 0\right). \tag{4}$$

Algorithm 1 Message handlers for Algorithm 2.

```
procedure HANDLE-UPDATE-REQUEST-MESSAGE(UpdateRequest sent by u)
    SEND-MESSAGE(Update⟨v, b(v)⟩) to u
procedure HANDLE-UPDATE-MESSAGE(Update⟨v_u, b_u⟩ sent by u)
    b(v_u) ← b_u
procedure HANDLE-UNION-MESSAGE(Union⟨e_u⟩ sent by u)
    if e = e_u then
        SEND-MESSAGE(Ack⟨Mutual, C⟩)
    else if F is already guarded then
        SEND-MESSAGE(Ack⟨Is-Guarded, ∅⟩)
    else
        SEND-MESSAGE(Ack⟨Not-Mutual, ∅⟩)
        I ← I ∪ {u}
procedure HANDLE-ADDING-MESSAGE(Adding⟨e_a, ε_a, C_a⟩)
    F ← F ∪ {e_a}
    b(v) ← b(v) + ε_a
    C ← C ∪ C_a
```

The performance guarantees of the algorithm are proven in this section. First, Lemmas 1 and 2 lead to Proposition 1 which implies that any solution found by the algorithm is acyclic and thereby a forest, implying that it is primal feasible. Proposition 2 states that under certain common conditions the main loop (line 8 of Algorithm 2) will have a logarithmic number of iterations. Finally, Claim 2 leads to Proposition 3 which states any solution found by the algorithm is dual feasible.

LEMMA 1. *Any cycle in the intersection graph*[1] *of F_{t+1} formed from $\mathcal{C}_t$ must consist solely of edges along the cuts between unguarded components.*

PROOF. Assume, on the contrary, that there exists a cycle containing an edge that is incident to at least one guarded component. Let $\langle u, v\rangle$ be such an edge and assume $\mu_t(v)$ is guarded. (1) implies that v's connected component has no outgoing edges,

$$\forall i \in \mu_t(v) : (\neg\exists j \in V : J_t(i,j)),$$

which contradicts the fact that $\langle u, v\rangle$ is in a cycle. □

The *potential cost* of an edge is the fractional quantity associated with ϵ on line 11 of Algorithm 2.

LEMMA 2. *Any cycle in the intersection graph of F_{t+1} formed from $\mathcal{C}_t$ must consist of edges of equal potential cost.*

[1]An *intersection graph* is formed from a family of sets $\mathcal{C} = \{C_1, C_2, C_3, \ldots\}$ by creating one vertex v_i for each set C_i and connecting two vertices v_i and v_j by an edge whenever their corresponding sets have a nonempty intersection, producing the edge set $\{\langle v_i, v_j\rangle : \mu(v_i) \cap \mu(v_j) \neq \emptyset\}$.

PROOF. Let $e_1 = \langle u_1, v_1\rangle$ be an edge in a cycle. (1) implies that all edges in a cycle must be cuts between existing connected components. Therefore, $\mu_t(u_1) \neq \mu_t(v_1)$. Furthermore, there must be another edge in the cycle, $e_2 = \langle u_2, v_2\rangle$, such that $\mu_t(v_2) = \mu_t(u_1)$. It must also be true that $J_t(u_1, v_1) = J_t(u_2, v_2) = J_t^+(u_1, v_2) = 1$. By Lemma 1 all components in the cycle are unguarded. Therefore, applying (1) gives

$$w(e_1) - b_t(u_1) - b_t(v_1) \leq w(e_2) - b_t(u_2) - b_t(v_2).$$

In general, this inequality will hold for the incoming and outgoing edges of any connected component in the cycle. Therefore, by transitivity,

$$\begin{aligned} w(e_1) - b_t(u_1) - b_t(v_1) &\leq w(e_2) - b_t(u_2) - b_t(v_2) \\ &\leq w(e_1) - b_t(u_1) - b_t(v_1), \end{aligned}$$

implying that

$$w(e_1) - b_t(u_1) - b_t(v_1) = w(e_2) - b_t(u_2) - b_t(v_2).$$

□

PROPOSITION 1. *The intersection graph of F_{t+1} formed from $\mathcal{C}_t$ is acyclic.*

PROOF. Assume, on the contrary, that there is a round t during which a cycle of length ℓ is formed. Since the graph is simple, $\ell > 1$. By Lemma 2, all of the edges in the cycle must be of equal potential cost. Therefore, each connected component will have had a tie between two fringe edges which must have been broken using the edge ordering. Therefore, either $\ell = 1$ or there are two edges with the same unique identifier, both of which are contradictions. □

COROLLARY 1. *$F_0, \ldots, F_\tau$ are all acyclic.*

PROOF. Since $F_0 = \langle V, \emptyset\rangle$, the base case is acyclic. Induction over Proposition 1 then proves the corollary. □

It is easy to see that $\tau =$ the diameter of the visibility graph $= O(n)$ since every acyclic subgraph has $O(n)$ edges and the algorithm adds at least one edge per round. This upper bound can in fact be tightened for many common cases, which we shall now demonstrate. Let $A_f(t) = A(t)$ be an upper bound on the number of unguarded components at the beginning of round t. Similarly, let $L_f(t) = L(t)$ be an upper bound on the total number of components at the beginning of round t. Clearly,

$$\begin{aligned} A(t) &\geq |\{C \in \mathcal{C}_t : g(C) = 1\}|, \text{ and} \\ L(t) &\geq |\mathcal{C}_t| \geq A(t). \end{aligned}$$

In general, every unguarded component will union with another component during each round. Regardless of whether such a component chooses to union with a guarded or unguarded component, the total number of components will decrease by one half the number of unguarded components. Therefore $L(t) = L(t-1) - A(t-1)/2$. Now let us consider the extrema for the change in the number of unguarded components. If all unguarded components choose to union with other unguarded components and all unions are pairwise, then we have $A(t) = A(t-1)/2$. On the other hand, if as many unguarded components union with guarded components as possible, then $A(t) \leq \min(A(t-1), L(t-1) -$

Algorithm 2 The distributed art gallery/dominating set algorithm. Message handlers are defined in Algorithm 1.

1: **procedure** DISTRIBUTED-ART-GALLERY(v)
Require: v is the vertex associated with the location of this agent.
Ensure: v will become a guard if $\langle v, d_v\rangle \in F$.
2: $C \leftarrow \emptyset$ /* The other fringe vertices in our component. */
3: $N \leftarrow \delta(\{v\}) \cup \{\langle v, d_v\rangle\}$ /* The neighborhood of v along with the special guard vertex d_v */
4: $F \leftarrow \emptyset$ /* The spanning forest of our component. */
5: **for all** $i \in \delta(C) \cup \{v\}$ **do**
6: $b(i) \leftarrow 0$
7: $I \leftarrow \emptyset$
8: **while** F is unguarded **do**
9: BROADCAST-MESSAGE(`UpdateRequest`) to all $u \in N$
10: Block until we have received and handled all `Update` messages from N.
11: Find an edge $e = \langle v, u\rangle \in N$ such that $u \notin C$ and $\epsilon = w(e) - b(v) - b(u)$ is minimized.
12: BROADCAST-MESSAGE(`Potential`$\langle\epsilon\rangle$) to all $c \in C$
13: Listen for all broadcast `Potential` messages from the fringe
14: **if** ϵ is the smallest in the fringe and ties are broken in our favor **then**
15: $N \leftarrow N - \{u\}$
16: **if** $u = d_v$ **then**
17: $C_m \leftarrow \{u\}$ /* v is to become a guard. */
18: **else**
19: SEND-MESSAGE(`Union`$\langle e\rangle$) to u
20: Wait for an `Ack`$\langle m, C_m\rangle$ message from u
21: **if** $m =$ `Mutual` **then** /* u also chose to make edge e tight */
22: $\epsilon \leftarrow \frac{\epsilon}{2}$
23: **else**
24: **if** $m =$ `Not-Mutual` **then** /* this means u is not yet guarded */
25: Block until we have received and handled an `Adding`$\langle e_a, \epsilon_a, C_a\rangle$ message from u
26: $C_m \leftarrow C_a$
27: BROADCAST-MESSAGE(`Adding`$\langle e, \epsilon, C_m\rangle$) to all $c \in C \cup I$
28: $I \leftarrow \emptyset$
29: $C \leftarrow C \cup C_m$
30: $b(v) \leftarrow b(v) + \epsilon$
31: $F \leftarrow F \cup \{e\}$
32: **else**
33: Block until we have received and handled an `Adding` message from another fringe member

$A(t-1)$). Therefore, assuming pairwise unions, the general recurrences for $A(t)$ and $L(t)$ are:

$$A(t) = \max\left(\frac{A(t-1)}{2}, \min\Big(A(t-1), L(t-1) - A(t-1)\Big)\right), \quad (5)$$

$$L(t) = L(t-1) - \frac{A(t-1)}{2}.$$

The initial conditions for the recurrences are clearly

$$\begin{aligned} A(0) &= |\{C \in \mathcal{C}_0 : g(C) = 1\}| = |R|, \\ L(0) &= |\mathcal{C}_0| = |R| + |T| = 2|R|. \end{aligned}$$

CLAIM 1. *$A(t-1)/2$ will always dominate in the maximization in* (5).

Validation of this claim will be given in the proof of the following proposition.

PROPOSITION 2. *The algorithm will terminate after a logarithmic number of rounds if all component unions are pairwise (i.e., iterations of the main loop on line 8 of Algorithm 2): $\tau = O(\log n)$.*

PROOF. This follows from the fact that the algorithm will terminate once the number of unguarded components is zero:

$$\forall t \in \mathbb{N}^* : A(t) = 0 \implies t \geq \tau.$$

Therefore, the burden of this proof is to show that, the $A(t)$ recurrence will converge exponentially, implying that $\tau = O(\log n)$.

If Claim 1 holds, then it is clear that the $A(t)$ recurrence will converge exponentially:

$$\begin{aligned} A(t) &= \frac{A(0)}{2^t}, \\ L(t) &= 2|R| - \sum_{i=0}^{t} \frac{A(0)}{2^i}. \end{aligned}$$

Let $k = \frac{A(0)}{2|R|}$ and observe that $k = \frac{1}{2}$. Substituting $2k|R|$ for $A(0)$ ensures that the minimization in $A(t)$ will always evaluate to $L(t-1) - A(t-1)$ because

$$\begin{aligned} \forall t \in \mathbb{N}^* : A(t) &\geq L(t) - A(t). \\ 2|R|\frac{k}{2^t} &\geq 2|R|\left(1 - \left(\sum_{i=0}^{t} \frac{k}{2^i}\right) - \frac{k}{2^t}\right) \\ \frac{2k}{2^t} &\geq 1 - \sum_{i=0}^{t} \frac{k}{2^i} \\ k &\geq \frac{2^t}{1 + 2^{t+1}}, \end{aligned}$$

which is true because $2^t/(1+2^{t+1})$ is bounded above by $\frac{1}{2}$. Therefore, provided Claim 1 holds, (5) can be simplified to

$$A(t) = \max\left(\frac{A(t-1)}{2}, L(t-1) - A(t-1)\right).$$

Claim 1 obviously holds for the base case of $t = 1$ because $A(0)/2 = 2k|R|$ is bounded below by $L(0) - A(0) = 2|R| -$

$\frac{2|R|}{2}$. Therefore, Claim 1 will hold as long as

$$\frac{A(t)}{2} \geq L(t) - A(t).$$

This equates to

$$\begin{aligned} k &\geq 2^{t+1}\left(1 - \left(\sum_{i=0}^{t} \frac{k}{2^i}\right) - \frac{k}{2^t}\right) \\ &\geq \frac{2 \times 4^t}{2^t + 4^{t+1}}, \end{aligned}$$

which must be true because $(2 \times 4^t)/(2^t + 4^{t+1})$ is bounded above by $\frac{1}{2}$. $\square$

CLAIM 2. *Let t' be the round during which an edge $e = \langle u, v\rangle$ is added to the spanning forest. Then e will not be in the cut of any component in a subsequent round: $\forall t > t', C \in \mathcal{C}_t : e \notin \delta(C)$.*

PROOF. $\mu_{t'+1}(u) = \mu_{t'+1}(v) = \mu_{t'}(u) \cup \mu_{t'}(v)$. Therefore, in all rounds subsequent to t' both endpoints of e are in the same component and therefore cannot be in the fringe. $\square$

PROPOSITION 3. *The vector y is a feasible solution to* (D) *and has the property*

$$\sum_{e \in F_\tau} w(e) \leq \sum_{e \in F_\tau} \sum_{S : e \in \delta(S)} y_S.$$

PROOF. The fact that y is a feasible solution to (D) is a straightforward result of the fact that y is initially zero and is updated according to (2). Let t be the round during which an edge $e = \langle u, v\rangle \in F_\tau$ was added to the forest. From (3), note that

$$\left(b_t(u) = \sum_{i=0}^{t-1} y_{\mu_i(u)}\right) \bigwedge \left(b_t(v) = \sum_{i=0}^{t-1} y_{\mu_i(v)}\right).$$

Furthermore, at the beginning of round t the potential for e is $\epsilon = w(e) - b_t(u) - b_t(v)$. Once e is added to F_t, the dual variables $y_{\mu_t(u)}$ and $y_{\mu_t(v)}$ are updated according to (2). Then there are three possible cases:

1. $g(\mu_t(u)) = g(\mu_t(v)) = J_t(u, v) = J_t(v, u) = 1$;
2. $g(\mu_t(u)) = g(\mu_t(v)) = J_t(u, v) + J_t(v, u) = 1$; or
3. $g(\mu_t(u)) + g(\mu_t(v)) = 1$.

In case 1,

$$y_{\mu_t(u)} + y_{\mu_t(v)} = \frac{\epsilon}{1 + J_t(v, u)} + \frac{\epsilon}{1 + J_t(u, v)} = \epsilon,$$

implying that

$$w(e) = \sum_{i=0}^{t} \left(y_{\mu_i(u)} + y_{\mu_i(v)}\right). \tag{6}$$

For case 2, assume without loss of generality that $J_t(u, v) = 1$ and $J_t(v, u) = 0$. For case 3, assume without loss of generality that $g(\mu_t(u)) = 1$ and $g(\mu_t(v)) = 1$. Then for both of these cases note that

$$y_{\mu_t(u)} = \frac{\epsilon}{1 + J_t(v, u)} = \epsilon,$$

implying that

$$w(e) = y_{\mu_t(u)} + \sum_{i=0}^{t-1} \left(y_{\mu_i(u)} + y_{\mu_i(v)}\right). \tag{7}$$

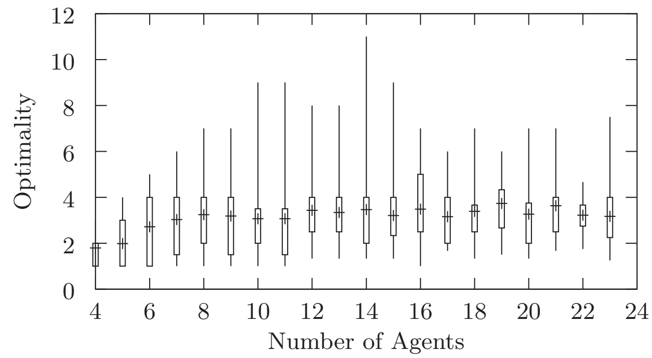


Figure 3: Solution quality of the algorithm for art gallery problems of various size. The x-axis is the size of the polygon (*i.e.*, the number of agents) and the y-axis is the constant of approximation. Each column is the distribution over 32 randomly generated polygons of a specific size. Boxes surround the middle two quartiles. The mean of each distribution is depicted as "+".

Claim 2 implies that the summations in (6) and (7) comprise all sets that cut e, thus completing the proof. $\square$

2.3 Empirical Analysis

We have thus far proven that Algorithm 2 produces a feasible solution to both (IP) and (D) in a linear—and often logarithmic—number of rounds. It therefore only remains to analyze the quality of the solution.

A series of n-gons were randomly generated by connecting n uniformly distributed vertices in the unit square of the Cartesian plane according to the "Two Peasants" method. 32 random polygons were created for each value of n. An agent was instantiated at each vertex of each randomly generated polygon and the algorithm run. The optimal dominating set was also calculated.

Figure 3 presents the distribution of optimality as a function of polygon size. Y-axis values represent the constant of approximation; lower values are better, with 1.0 being the optimal solution. Boxes represent the second and third quartiles of each distribution. The overall mean constant of approximation is 3.13 with a standard deviation of 0.36. Therefore, we can say with high probability that the algorithm will produce a solution with a constant approximation bound regardless of the problem size.

3. ART GALLERY VARIANTS

In this section we will show that some variants of the art gallery problem can be solved using the same approach as described above. In fact, some harder problems can be approximated with a constant *theoretical* bound on solution quality. For example, one popular variant is what is dubbed the "Treasury Problem" [3], in which treasures dispersed in the polygon are what need to be guarded. As another variant, one might need to minimize the distance between a guard and that which he or she is guarding (*e.g.*, due to a limited view distance of the sensor, or due to the mobility of a robot). This variant is in fact equivalent to the treasury problem in which each treasure has weighted importance [1].

In order to model this variant, we need only to embed the edge weights of the augmented graph into the proper metric space. As long as all of the edges in the augmented graph are weighted a metric space with a bijection in the range $\left[1, \frac{3}{2}\right]$ then we will show that the algorithm as defined above will produce a solution that is no more than a factor of $2 - \frac{2}{|R|}$ away from optimal. This can easily be done by parameterizing the relative cost between covering a vertex/treasure and the distance between a guard and a vertex/treasure. To prove this claim, we use a technique of defining an invariant over the weights of the edges added to the forest that can ultimately be bounded by the average vertex degree of the forest. The basic intuition of our result is that the average degree of a vertex in a forest of at most n vertices is at most $2 - \frac{2}{n}$. This technique is exactly the same as that first used in a proof due to Goemans and Williamson in [5, Theorem 3.6], in which they show that certain connectivity problems can be sequentially 2-approximated in polynomial time. Our result in fact generalizes that of Goemans and Williamson by proving that, with a slight change to the potential function (and thereby the invariant), the approximation guarantee can be maintained even if multiple edges are added per round (allowing for parallelism/distribution).

LEMMA 3 (WILLIAMSON, *et al.* [19, THEOREM 3.6]). *Let H be the intersection graph of the final spanning forest F_τ formed from $\mathcal{C}_t$. Remove all isolated vertices in H that correspond to components in $\mathcal{C}_t$ that are guarded. Then no leaf in H corresponds to a guarded component.*

PROOF. This is a transcription of the proof, reproduced here for completeness using our notation in the specific domain of art gallery problems. Assume the contrary: Let v be a leaf, let C_v be its associated guarded component, let e be the edge incident to v, and let $C \subseteq V$ be the component of F which contains C_v. Let N and $C - N$ be the two components formed by removing edge e from the edges of component C. Without loss of generality, say that $C_v \subseteq N$. The set $N - C_v$ is partitioned by some of the components of the current round; call these $C_1, \ldots, C_k$. Since vertex v is a leaf, no edge in F_τ connects C_v to any C_i. Thus by the construction of F_τ, $\forall i \in \{1, \ldots, k\} : C_i$ is guarded. Since C_v is also guarded, it follows that N must be too. Clearly, if two components S and B are both guarded and $B \subseteq S$, then the component $S - B$ must also be guarded. Since we know that C is guarded then $N - C$ must as well, and thus by the construction of F_τ, $e \notin F_\tau$, which is a contradiction. $\square$

PROPOSITION 4. *The cost of the final spanning forest F_τ is bounded above by $\left(2 - \dfrac{2}{|R|}\right) Z^*_{\mathrm{IP}}$.*

PROOF. Without loss of generality, assume $y_S > 0 \implies g(S) = 1$. This property ensures that $\sum_{S \subset V} y_S \leq Z^*_{\mathrm{LP}}$. Since it is clear that $Z^*_{\mathrm{LP}} \leq Z^*_{\mathrm{IP}}$, we then have

$$\sum_{S \subset V} y_S \leq Z^*_{\mathrm{LP}} \leq Z^*_{\mathrm{IP}}.$$

Proposition 3 ensures that the weight of F_τ is

$$\sum_{e \in F_\tau} w(e) \leq \sum_{e \in F_\tau} \sum_{S : e \in \delta(S)} y_S = \sum_{S \subset V} y_S |F_\tau \cap \delta(S)|.$$

To prove this theorem we will show by induction over the construction of F_τ that

$$\sum_{S \subset V} y_S |F_\tau \cap \delta(S)| \leq \left(2 - \frac{2}{|R|}\right) \sum_{S \subset V} y_S. \tag{8}$$

The base case certainly holds at round zero since all y_S are initialized to zero. Let A be the set of edges added to the spanning forest during an arbitrary round t. For each edge $e = \langle u, v \rangle \in A$, let ϵ_e denote the potential value associated with that edge: $\epsilon_e = w(e) - b_t(u) - b_t(v)$. Now sort A according to descending potential value, such that $e_i \in A$ is edge with the i^{th} largest potential. At the end of a round t, the left-hand side of (8) will increase by at most

$$\begin{aligned} &\sum_{C \in \mathcal{C}_t : g(C) = 1} y_C |F_\tau \cap \delta(C)| \\ &= \sum_{C \in \mathcal{C}_t : g(C) = 1} \sum_{\langle u, v \rangle \in A : u \in C} \frac{J_t(u, v)\epsilon_e}{1 + J_t(v, u)} |F_\tau \cap \delta(C)|. \end{aligned} \tag{9}$$

If we can prove that this increase is bounded above by the increase of the right-hand side, namely

$$\left(2 - \frac{2}{|R|}\right) \sum_{i=1}^{|A|} \epsilon_{e_i} \times i, \tag{10}$$

then we will be done.

First, observe that (9) can be bounded above by

$$\left(\max_{e \in A} \epsilon_e\right) \sum_{C \in \mathcal{C}_t : g(C) = 1} \sum_{\langle u, v \rangle \in A : u \in C} J_t(u, v) |F_\tau \cap \delta(C)|. \tag{11}$$

Next, observe that (10) can be bounded below by

$$\left(\min_{e \in A} \epsilon_e\right) \left(2 - \frac{2}{|R|}\right) \left(\frac{|A|}{2} + \frac{1}{2}\right) |A|. \tag{12}$$

Now let H be the intersection graph of the final spanning forest F_τ formed from $\mathcal{C}_t$. Remove all isolated vertices in H that correspond to guarded components in $\mathcal{C}_t$. Notice that H is a forest, and by Lemma 3 no leaf in H corresponds to a guarded component. Let N_a be the set of vertices in H that correspond to unguarded components:

$$N_a = \{C \in \mathcal{C}_t : g(c) = 1\},$$

and let N_i be the set of vertices in H corresponding to guarded components. The degree of a vertex v in H corresponding to component C, denoted d_v, must be $|\{e \in \delta(C) : e \in F_\tau\}|$. Then the summation of (11) can be rewritten as

$$\begin{aligned} \sum_{v \in N_a} d_v &= \sum_{v \in N_a \cup N_i} d_v - \sum_{v \in N_i} d_v \\ &\leq 2(|N_a| + |N_i| - 1) - 2|N_i| \\ &= (2|N_a| - 2). \end{aligned}$$

This inequality holds since H is a forest with at most $|N_a| + |N_i| - 1$ edges, and since each vertex corresponding to a guarded component has degree at least 2. Substituting this result back into (11) we have

$$\begin{aligned} (9) \leq (11) &\leq \left(\max_{e \in A} \epsilon_e\right) 2(|\{C \in \mathcal{C}_t : g(C) = 1\}| - 1) \\ &\leq \left(\max_{e \in A} \epsilon_e\right) \left(2 - \frac{2}{|R|}\right) |A|, \end{aligned}$$

since the number of unguarded components is always no more than $|R|$. Therefore, (9) $\leq$ (12) $\leq$ (10) if during every round the following invariant holds:

$$\max_{e \in A} \epsilon_e \leq \left(\frac{|A|}{2} + \frac{1}{2}\right) \min_{e \in A} \epsilon_e,$$

which is clearly true because all of the non-zero edge weights are equal to one. Hence the theorem is proven. □

Assuming all messages can be both unicast and broadcast in a constant number of messaging rounds then, by the same argument as in the proof of 2, the main loop of the distributed algorithm on line 8 can run in a logarithmic number of iterations and thereby will have a logarithmic number of messaging rounds. If this assumption does not hold—for example, if *ad hoc* routing is required—then the algorithm can be trivially extended to support the BROADCAST-MESSAGE function itself. To do this, the algorithm will use the partially constructed spanning trees within each connected component for multicast.

The most expensive operations in the distributed algorithm are (1) determining the fringe edge with minimal potential; and (2) merging two connected components once an edge between them becomes tight. Should efficient broadcast be unavailable, one way of implementing these operations is to have broadcast messages convergecasted up the partially constructed spanning tree in the component. This method was used to solve a similar problem in [16]. The root of the tree (*e.g.*, the vertex that was added the earliest and is of highest unique identifier) can then perform the operation and unicast the result back down to the relevant fringe member(s).

4. CONCLUSIONS AND FUTURE WORK

This paper has introduced a distributed algorithm for the art gallery and dominating set problems that is guaranteed to run in a number communication rounds on the order of the diameter of the visibility graph. The algorithm produces a solution whose cost is no worse than a constant factor of optimal with high probability. For art gallery variants in which the distances between guards and treasures/vertices must also be minimized, the algorithm is proven to be a 2-optimal approximation, provided that the edge weights are embedded in the proper metric space. Ultimately, we have furthered the results of Panconesi [13] by showing that the primal/dual optimization scheme proposed by Goemans and Williamson [5] can be successfully distributed into a multiagent algorithm with not only bounds on approximation, but also on runtime. This suggests that other multiagent coordination problems might yield to the same approach.

5. REFERENCES

[1] S. Carlsson and H. Jonsson. Guarding a treasury. In *Proceedings of the 5th Canadian Conference on Computer Geometry*, pages 85–90, 1993.

[2] R. C. Corrêa, F. C. Gomes, C. A. S. Oliveira, and P. M. Pardalos. A parallel implementation of an asynchronous team to the point-to-point connection problem. *Parallel Computing*, 29(4):447–466, May 2002.

[3] L. L. Deneen and S. Joshi. Treasures in an art gallery. In *Proceedings of the 4th Canadian Conference on Computer Geometry*, pages 17–22, 1992.

[4] A. Ganguli, J. Cortés, and F. Bullo. Visibility-based multi-agent deployment in orthogonal environments. In *Proceedings of the American Control Conference*, pages 3426–3431, New York, July 2007.

[5] M. X. Goemans and D. P. Williamson. A general approximation technique for constrained forest problems. *SIAM Journal on Computing*, 24:296–317, 1995.

[6] F. Grandoni, J. Könemann, A. Panconesi, and M. Sozio. Primal-dual based distributed algorithms for vertex cover with semi-hard capacities. In *Proceedings of the twenty-fourth annual ACM symposium on Principles of distributed computing*, pages 118–125, Las Vegas, NV, USA, 2005.

[7] C. Huang, C. Qin, and Y. Xian. A distributed algorithm for computing connected dominating set in ad hoc networks. *International Journal of Wireless and Mobile Computing*, 1(2):148–155, 2006.

[8] S. Khuller, U. Vishkin, and N. Young. A primal-dual parallel approximation technique applied to weighted set and vertex cover. *Journal of Algorithms*, 17(2):280–289, October 1994.

[9] R. N. Lass, M. J. Grauer, E. A. Sultanik, and W. C. Regli. A decentralized approach to the art gallery problem. In *Proceedings of the 17th Fall Conference on Computational Geometry*, November 2007.

[10] N. A. Lynch. *Distributed Algorithms*. Morgan Kaufmann, 1997.

[11] A. Meisels. *Distributed Search by Constrained Agents: Algorithms, Performance, Communication*. Springer-Verlag, London, 2008.

[12] J. O'Rourke. *Art Gallery Theorems and Algorithms*. Oxford University Press, 1987.

[13] A. Panconesi. *Fast Distributed Algorithms Via Primal-Dual (Extended Abstract)*, volume 4474, pages 1–6. Springer, Heidelberg, 2007.

[14] A. Panconesi and M. Sozio. Fast distributed scheduling via primal-dual. In *Proceedings of the twentieth annual symposium on Parallelism in algorithms and architectures*, pages 229–235, Munich, Germany, 2008. ACM.

[15] L. Ruan, H. Du, X. Jia, W. Wu, Y. Li, and K.-I. Ko. A greedy approximation for minimum connected dominating sets. *Theoretical Computer Science*, 329(1–3):325–330, 2004.

[16] M. Santos, L. M. A. Drummond, and E. Uchoa. *A Distributed Primal-Dual Heuristic for Steiner Problems in Networks*, volume 4525, pages 175–188. Springer, 2007.

[17] M. Sozio. *Efficient Distributed Algorithms via the Primal-Dual Schema*. PhD thesis, "La Sapienza" University, Rome, September 2006.

[18] K. Sycara, K. Decker, A. Pannu, M. Williamson, and D. Zeng. Distributed intelligent agents. *IEEE Expert: Intelligent Systems and Their Applications*, 11(6):36–46, 1996.

[19] D. P. Williamson, M. X. Goemans, M. Mihail, and V. V. Vazirani. A primal-dual approximation algorithm for generalized steiner network problems. *Combinatorica*, 15(3):435–454, September 1995.

[20] J. Wu and H. Li. On calculating connected dominating set for efficient routing in ad hoc wireless networks. In *Proceedings of the 3rd International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, pages 7–14, New York, NY, USA, 1999. ACM.