Minimum Time Power-Aware Rendezvous for Multi-Agent Networks*

Tina Setter and Magnus Egerstedt[1]

***Abstract*— This paper discusses the problem of power-aware rendezvous, in which a collection of mobile agents must decide where and when to meet in order to do so in the least amount of time, given different available initial power levels. Moreover, the movements of the individual agents directly affect the power consumption, which means that the agents must meet in such a way that they do not run out of power. This problem is first approached as an optimal control problem and then implemented in a distributed manner on a team of simulated mobile robots.**

I. INTRODUCTION

As a number of multi-agent systems are expected to operate in environments where it may not be easy to replace batteries or refill the fuel, such as hostile environments, e.g., [1], and space, e.g., [2], power consumption must be taken into account already at the design phase. This is especially important for large-scale systems, where a large group of agents is deployed and power is depleted due to the actions that the agents must perform, e.g., moving, sensing, communicating, computing.

Power-aware algorithms have been developed for many applications in the area of multi-agent networks. For example, in the sensor networks community, power-aware algorithms have been developed in order to maximize the lifetime of a network, while still being able to sense as much as possible, e.g., [4], [5], [6], [7], [8]. In these strategies, mobility is not taken into account since the sensors are stationary once deployed.

However, there is a broad class of multi-agent problems which require mobility and in these problems, power-aware strategies are especially important. In fact, as a general observation, mobility is the most "expensive" when it comes to power consumption, whereas communications are less costly and computations and sensing are the least expensive, in comparison [3].

One of these mobile multi-agent problems that has been expanded upon to take the power consumption into account during the design phase includes the classic rendezvous problem [9], in which a group of mobile agents meet at a common location using only relative position information. This problem is explored in [3], where the goal is for the agents to achieve rendezvous, while the dissipation of power of the agents is reflected by a diminishing sensing footprint. In this work, power levels are depleted due to mobility and the sensing footprint, which determines each agent's neighborhood set, is dependent upon that agent's current power level. Similarly, in [10], the power-aware rendezvous problem is again investigated, but this time uses a power model in which power decay is a function of the current power level, as opposed to being dependent on mobility.

Another coordinated multi-agent problem is coverage control, in which a group of mobile agents must be deployed to collectively cover a large environment and collect information across the area. Like the rendezvous problem, this problem has also been explored from a power-aware standpoint, in which the total energy of the agents, taking mobility, sensing and communication costs into account, is minimized [11].

The work presented in this paper also addresses the rendezvous problem, but differs from [3] and [10] in that we do not consider the sensor footprint from a power perspective. In fact, we assume that the agents communicate over a static network, and instead focus on where and when the agents should meet given different initial power levels. This problem is relevant when a large collection of agents are deployed and it is possible that they do not all start with the same amount of power. Instead of the agents with less power simply dropping off when they run out of power, we are interested in designing a strategy that allows the agents to still achieve rendezvous by taking the power levels into account during the design phase. Because the agents lose power as they move, we use a power model that is dependent on mobility, as is done in [3]. Note that even though the paper focuses on rendezvous, this should only be thought of as one particular instantiation of a larger class of power-aware mobility problems.

In this paper, we discuss the optimal way to move

*This work was supported by NSF, under grant CNS-1239225.

[1]Tina Setter and Magnus Egerstedt are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA. tsetter3@gatech.edu, magnus@gatech.edu

agents around so that power consumption is minimized. To this end, we start with the single-agent case in which an agent must move from a desired initial position to a desired final position in a specified time while utilizing the least amount of power. Then we extend this to a group of mobile agents that wish to achieve rendezvous in the least amount of time, where the agents' power levels must be taken into account in order to find the meeting location that minimizes the time required to meet. Once the meeting location and time is decided upon, the control strategy is obtained for each agent to allow it to travel to the meeting location in the given time while minimizing power consumption.

The outline of the paper is as follows: In Section II, the rendezvous problem that we are concerned with is described at a high level. In Section III, the optimal fuel consumption problem for the single-agent case is described and a solution is presented. Section IV describes the extension to the problem in which multiple agents need to decide on where and when to meet and gives a distributed algorithm that solves this problem. Section V concludes and summarizes what was discussed in the paper.

II. PROBLEM OVERVIEW

Consider a network of mobile robots that would like to meet at some spatial location, where each agent starts with some non-negative power level. The problem that we are interested in is finding the location where the agents should meet so that they can meet in the least amount of time. To illustrate, suppose there are two robots that have the same initial power level. In order to meet in the shortest amount of time, a naive approach could be to have the agents meet at the average of their positions. Now instead suppose one of the robots has an initial power level that is double that of the other robot. Now where should they meet? This is the question under consideration in this paper.

For the moment, suppose that we have solved this problem and know where the agents should meet and how long it will take them to meet. There is then the question of *how* the agents should move from their initial positions to the meeting point. There are many different ways that the agents can move between two points, but we are interested in minimizing total power consumption. Thus, we can formulate this problem as an optimal control problem to determine the input that each agent needs to move it from its initial position to the meeting point using the least amount of power.

In summary, we are trying to minimize both power consumption *and* the amount of time required for the agents to meet. If we were just minimizing power consumption, the agents would meet in infinite time, since we could travel infinitely slow, using up "zero" power. On the other hand, if we were just minimizing the time required for the agents to meet with no regard to power consumption, the agents would travel infinitely fast and reach every point in "zero" time, thus making the determination of the meeting point unnecessary.

Now, going back to the rendezvous problem, in order to find the optimal meeting location and time that it will take to meet, we need to know how the agents are moving and how their power levels are dissipating, because each agent needs to be able to reach the meeting point in the meeting time with a final non-negative power level. Clearly, if an agent runs out of power before reaching the meeting point, the agents will not meet. If we look at the problem of finding the meeting location as an optimization problem, this power requirement can be treated as a constraint. However, in order to represent this constraint in a mathematical way, we must deal with the single-agent problem first, in which we find the control signal that allows a single agent to move between two points in a specified amount of time using the least amount of power. This problem will be discussed in the next section.

III. SINGLE-AGENT CASE

The goal in the single-agent case is to find the control input u that minimizes the fuel consumed when an agent moves between prespecified points in a given amount of time, T, starting with zero velocity and ending with zero velocity. We will start by defining the power and mobility models, where we consider agents in one dimension (along a line). This can easily be relaxed to higher dimensions by traveling in straight-line paths.

A. Power and Mobility Models

For this problem, the dynamics of the agent are given by:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= u \\ \dot{x}_3 &= -x_2^2 - \alpha u^2 \end{aligned} \tag{1}$$

Here, $x_1 \in \mathbb{R}$ is the position of the agent, $x_2 \in \mathbb{R}$ is the velocity, and $x_3 \in \mathbb{R}$ is the agent's power level. The acceleration is given by the external input u, which will be designed to minimize power consumption. The dynamics for x_1 and x_2 are given by a double integrator, as is quite standard, e.g. [12], [13], [14], [15].

The dissipation of power is represented by a quadratic function of the velocity and acceleration of the agent, where $\alpha \in \mathbb{R}^+$ is a constant that determines how strongly acceleration affects the dissipation of power in comparison to the effect of the velocity. Clearly, power is lost while the agent is moving, hence we need that $\dot{x}_3 \leq 0$ at all times, which is satisfied by these dynamics. Also, if the agent is not accelerating, but still moving (i.e. has non-zero velocity), the power level should still be decreasing. In this model, we consider power consumed through sensing, communication, and computations to be negligible, since mobility is the most costly in terms of power consumption.

B. Optimal Fuel Consumption

We want to design u to minimize the amount of power consumed throughout the entire move, starting from $t = 0$ and ending at $t = T$, which is represented by $x_3(0) - x_3(T)$. This value can equivalently be represented by the integration of $-\dot{x}_3(t)$. The control problem is formulated as follows:

$$\min_u \int_0^T \left(x_2^2(t) + \alpha u^2(t)\right) \mathrm{d}t \tag{2}$$

where the initial and final conditions are specified by:

$$\begin{aligned} x_1(0) &= x_{10} \\ x_2(0) &= 0 \\ x_3(0) &= x_{30} > 0 \\ x_1(T) &= x_{1T} \\ x_2(T) &= 0 \\ x_3(T) &\geq 0 \end{aligned} \tag{3}$$

Note that x_{10}, x_{30}, and x_{1T} are given parameters representing the initial position, initial power level, and final position, respectively, whereas $x_3(T)$ is free, but must be non-negative.

Theorem 3.1: The control signal $u(t)$ that solves the problem in (2), subject to the agent dynamics in (1) and the initial conditions in (3), is given by:

$$u(t) = -\frac{1}{2\alpha}\left(c_1 \mathrm{e}^{\frac{1}{\sqrt{\alpha}}t} + c_2 \mathrm{e}^{-\frac{1}{\sqrt{\alpha}}t}\right), \tag{4}$$

where

$$\nu_1 = \frac{2(x_{10} - x_{1T})}{T + 2\sqrt{\alpha}\left(\sinh^{-1}\left(\frac{T}{\sqrt{\alpha}}\right) - \tanh^{-1}\left(\frac{T}{\sqrt{\alpha}}\right)\right)}$$

$$\nu_2 = \frac{\sqrt{\alpha}\left(1 - \cosh\left(\frac{T}{\sqrt{\alpha}}\right)\right)}{\sinh\left(\frac{T}{\sqrt{\alpha}}\right)}\nu_1$$

$$c_1 = \frac{1}{2}\mathrm{e}^{-\frac{1}{\sqrt{\alpha}}T}\left(\nu_2 - \nu_1\sqrt{\alpha}\right)$$

$$c_2 = \frac{1}{2}\mathrm{e}^{\frac{1}{\sqrt{\alpha}}T}\left(\nu_2 + \nu_1\sqrt{\alpha}\right)$$

Proof: Letting $L = x_2^2 + \alpha u^2$, the Hamiltonian, H, is given by:

$$\begin{aligned} H &= L + \lambda^\top f \\ &= x_2^2 + \alpha u^2 + \lambda_1 x_2 + \lambda_2 u - \lambda_3 x_2^2 - \lambda_3 \alpha u^2 \end{aligned}$$

and the co-states satisfy

$$\begin{aligned} \dot{\lambda} &= -\left(\frac{\partial H}{\partial x}\right)^\top \\ &= -\begin{bmatrix} 0 \\ 2x_2 + \lambda_1 - 2\lambda_3 x_2 \\ 0 \end{bmatrix}. \end{aligned}$$

This yields $\lambda_1(t) = \nu_1$, where ν_1 is some constant, and $\lambda_3(t) = 0$ due to the fact that $x_3(T)$ is free. As a result, we get $\dot{\lambda}_2(t) = -2x_2(t) - \nu_1$ and let $\lambda_2(T) = \nu_2$, where ν_2 is some constant.

The optimality condition

$$\frac{\partial H}{\partial u} = 2\alpha u + \lambda_2 = 0$$

yields

$$u(t) = -\frac{\lambda_2(t)}{2\alpha}. \tag{5}$$

$\lambda_2(t)$ is found by differentiating $\dot{\lambda}_2(t)$ and then solving the resulting second order differential equation.

$$\begin{aligned} \ddot{\lambda}_2(t) &= -2\dot{x}_2(t) \\ &= -2u(t) \\ &= \frac{1}{\alpha}\lambda_2(t), \end{aligned}$$

with boundary conditions $\lambda_2(T) = \nu_2$ and $\dot{\lambda}_2(0) = \dot{\lambda}_2(T) = -\nu_1$, has solution

$$\lambda_2(t) = c_1 \mathrm{e}^{\frac{1}{\sqrt{\alpha}}t} + c_2 \mathrm{e}^{-\frac{1}{\sqrt{\alpha}}t}, \tag{6}$$

for some constants c_1 and c_2. This solution yields three boundary conditions,

$$\lambda_2(T) = c_1 \mathrm{e}^{\frac{1}{\sqrt{\alpha}}T} + c_2 \mathrm{e}^{-\frac{1}{\sqrt{\alpha}}T} = \nu_2 \tag{7}$$

$$\dot{\lambda}_2(0) = \frac{1}{\sqrt{\alpha}}c_1 - \frac{1}{\sqrt{\alpha}}c_2 = -\nu_1 \tag{8}$$

$$\dot{\lambda}_2(T) = \frac{1}{\sqrt{\alpha}}c_1 \mathrm{e}^{\frac{1}{\sqrt{\alpha}}T} - \frac{1}{\sqrt{\alpha}}c_2 \mathrm{e}^{-\frac{1}{\sqrt{\alpha}}T} = -\nu_1, \tag{9}$$

and using (7) and (9), c_1 and c_2 were found in terms of ν_1 and ν_2 to be

$$c_1 = \frac{\mathrm{e}^{-\frac{1}{\sqrt{\alpha}}T}}{2}\left(\nu_2 - \nu_1\sqrt{\alpha}\right) \tag{10}$$

$$c_2 = \frac{\mathrm{e}^{\frac{1}{\sqrt{\alpha}}T}}{2}\left(\nu_2 + \nu_1\sqrt{\alpha}\right). \tag{11}$$

To solve for ν_1 and ν_2, we first integrate $u(t)$ to find $x_2(t)$, which results in

$$x_2(t) = -\frac{1}{2\alpha}\left[\sqrt{\alpha}c_1 e^{\frac{1}{\sqrt{\alpha}}t} - \sqrt{\alpha}c_2 e^{-\frac{1}{\sqrt{\alpha}}t} + \alpha\nu_1\right],$$

and then integrate $x_2(t)$ to find $x_1(t)$, resulting in

$$x_1(t) = -\frac{1}{2}\left[c_1 e^{\frac{1}{\sqrt{\alpha}}t} + c_2 e^{-\frac{1}{\sqrt{\alpha}}t} + \nu_1 t - c_1 - c_2\right] + x_{10}, \tag{12}$$

which must satisfy $x_1(T) = x_{1T}$. By substituting the expressions for c_1 and c_2 from (10) and (11) into (12) and using the boundary condition, we get

$$\nu_1\left(T - \sqrt{\alpha}\sinh\left(\frac{T}{\sqrt{\alpha}}\right)\right) + \nu_2\left(1 - \cosh\left(\frac{T}{\sqrt{\alpha}}\right)\right) = 2\left(x_{10} - x_{1T}\right). \tag{13}$$

By substituting the expressions for c_1 and c_2 from (10) and (11) into (8), we get

$$\nu_2 = \frac{\sqrt{\alpha}\left(1 - \cosh\left(\frac{T}{\sqrt{\alpha}}\right)\right)}{\sinh\left(\frac{T}{\sqrt{\alpha}}\right)}\nu_1, \tag{14}$$

and by substituting ν_2 from (14) into (13), we solve for ν_1, resulting in

$$\nu_1 = \frac{2(x_{10} - x_{1T})}{T + 2\sqrt{\alpha}\left(\sinh^{-1}\left(\frac{T}{\sqrt{\alpha}}\right) - \tanh^{-1}\left(\frac{T}{\sqrt{\alpha}}\right)\right)},$$

where everything on the right hand side is known. Hence we know ν_2, and then can find c_1 and c_2, which describe $\lambda_2(t)$ completely. By substituting $\lambda_2(t)$ from (6) into (5), we get

$$u(t) = -\frac{1}{2\alpha}\left(c_1 e^{\frac{1}{\sqrt{\alpha}}t} + c_2 e^{-\frac{1}{\sqrt{\alpha}}t}\right).$$

■

Now given T, α, x_{10}, and x_{1T}, the optimal control input, which minimizes the total power consumed throughout the move, can be found. However, this control input does not take the initial power level into account, and thus we can end up with a negative final power level, which isn't feasible in reality since the agents will stop moving when they run out of power. Given α, x_{10}, x_{1T}, and the initial power level (x_{30}), we can determine the minimum time, T_{min}, in which the move can be completed such that the final power level is non-negative. The minimum time in which the move can be completed corresponds to a final power level of zero, because if there is power remaining at the end of the move, it could have been completed faster.

T_{min} is found by solving the following equation:

$$\frac{(x_{10} - x_{1T})^2}{T_{min} + 2\sqrt{\alpha}\left(\frac{1-\cosh\left(\frac{T_{min}}{\sqrt{\alpha}}\right)}{\sinh\left(\frac{T_{min}}{\sqrt{\alpha}}\right)}\right)} - x_{30} = 0 \tag{15}$$

Curves can be created that show how T_{min} relates to the total distance traveled and the initial power level. An example of such curves is shown in Fig. 1.

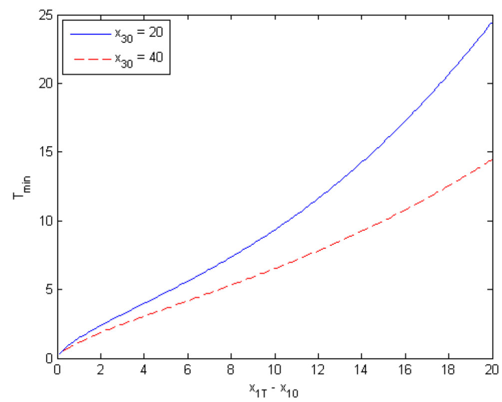

Fig. 1: Depicted are curves that represent the minimum time (T_{min}) required for an agent to travel a specified distance ($x_{1T} - x_{10}$), given that the agent has an initial power level of 20 (solid), and an initial power level of 40 (dashed).

It can be seen in Fig. 1 that T_{min} increases with increasing distance traveled, as expected. Also as expected, an agent with a higher initial power level can travel any specified distance in less time than an agent with a lower initial power level because a higher initial power level means that the agent is able to travel faster.

If we relax the constraint that the final power must be zero and instead let it be greater than or equal to zero, we get

$$x_{30} \geq \frac{(x_{10} - x_{1T})^2}{T + 2\sqrt{\alpha}\left(\frac{1-\cosh\left(\frac{T}{\sqrt{\alpha}}\right)}{\sinh\left(\frac{T}{\sqrt{\alpha}}\right)}\right)} \tag{16}$$

where the expression on the right-hand side is the amount of power lost during the move, which must be less than or equal to the initial power level in order to end up with a non-negative power level at time T. This inequality will be important in the next section when the constrained optimization problem is introduced.

IV. Multi-Agent Rendezvous

Now that the input has been found that minimizes power consumption in the single-agent case, we have to determine where and when the mobile agents should meet so that they meet in the shortest amount of time, while satisfying the requirement that each agent must end up at the meeting location with a non-negative power level.

Given N agents, let $p_i \in \mathbb{R}^n$ be the n-dimensional position where agent i would like to meet the other agents and let $\tau_i \in \mathbb{R}$ be the time at which agent i would like to meet the other agents. Now define $p = [p_1^T, ..., p_N^T]^T \in \mathbb{R}^{Nn}$ and $\tau = [\tau_1, ..., \tau_N]^T \in \mathbb{R}^N$.

Since the agents need to agree upon their meeting point and time, eventually p_i should equal p_j and τ_i should equal τ_j for all $i, j \in \{1, ..., N\}$. This is a well established problem known as consensus and we will combine consensus with our cost to solve this problem.

We want to minimize the amount of time that it takes the agents to meet, while including consensus so that all agents eventually agree on the decision variables. Each agent has a (potentially) different initial power level and thus this problem needs to take power level into account when finding the optimal meeting location and time. This problem of meeting in the least amount of time can be formulated as a constrained optimization problem, given by:

$$\min_{(p,\tau)} \sum_{i=1}^{N} \left(\frac{1}{2}\rho\tau_i^2 + \left(\sum_{j\in N_i} ||p_i - p_j||^2 + ||\tau_i - \tau_j||^2 \right) \right)$$

s.t. $h_i(p_i, \tau_i) \leq 0 \, \forall i = 1, ..., N$.

The first term in the cost ensures that the agents minimize the amount of time that it takes them to meet, where the weighting factor ρ determines how strongly this term affects the cost. The second term enforces consensus of the decision variables across the agents, where N_i is the set representing agent i's neighborhood. Here, h_i is the constraint that ensures that agent i can reach p_i in time τ_i given its initial position and power level. h_i is derived from (16) and is given by:

$$h_i(p_i, \tau_i) = \|(x_{10})_i - p_i\|^2 - (x_{30})_i \left[\tau_i + 2\sqrt{\alpha} \left(\frac{1 - \cosh\left(\frac{\tau_i}{\sqrt{\alpha}}\right)}{\sinh\left(\frac{\tau_i}{\sqrt{\alpha}}\right)} \right) \right]$$

If these constraints are not satisfied, the meeting point and time will not be feasible because one or more agents will run out of power before reaching the desired meeting point.

This problem was solved via the primal-dual gradient laws for constrained optimization, as described in [16], using the dynamics given below, where the co-state, μ_i, dynamics are such that they remain positive, which is necessary because we have inequality constraints.

$$\dot{p}_i = -\sum_{j\in N_i} (p_i - p_j) - \frac{\partial h_i}{\partial p_i}(p_i, \tau_i)\,\mu_i$$

$$\dot{\tau}_i = -\sum_{j\in N_i} (\tau_i - \tau_j) - \rho\tau_i - \frac{\partial h_i}{\partial \tau_i}(p_i, \tau_i)\,\mu_i$$

$$\dot{\mu}_i = \begin{cases} h_i(p_i, \tau_i) & \text{if } h_i(p_i, \tau_i) > 0 \text{ or } \mu_i > 0 \\ 0 & \text{otherwise} \end{cases}$$

Note that this is a decentralized algorithm, in which each agent needs only its own information and that of its neighbors.

This algorithm was implemented in MATLAB for a network of three agents with one-dimensional positions. The agents' initial positions are 10, 20, and 30, with initial power levels of 30, 30, and 5, respectively. The initial meeting point for agent i, p_i, was set to agent i's initial position, $(x_{30})_i$, to indicate that agent i would initially like to meet the other agents at its own starting position. Each agent's initial meeting time, τ_i, was set to 100, so that the power constraint ($h_i(p_i, \tau_i) \leq 0$) would be satisfied initially for all i. Fig. 2 shows the trajectories of the meeting points, p_i for $i = 1, 2, 3$ and Fig. 3 shows the trajectories of the meeting times, τ_i for $i = 1, 2, 3$.

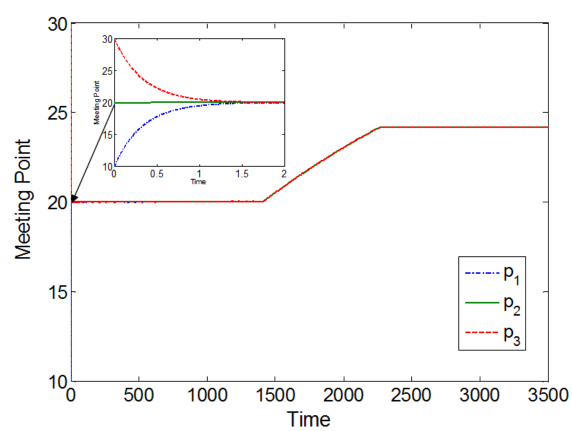

Fig. 2: Depicted is where each of the three agents would like to meet the other agents (p_i corresponds to the desired meeting location of agent i, for $i = 1, 2, 3$), where the initial meeting positions for each agent is that agent's initial position (here the initial positions are 10, 20, and 30). The zoomed in portion shows the p_i trajectories for time $t = 0$ to $t = 2$, in which time the desired meeting location of all agents approaches 20.

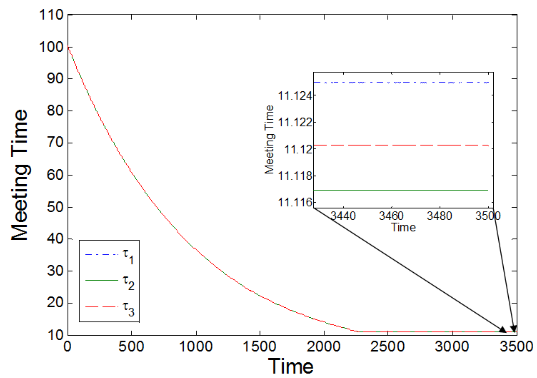

Fig. 3: Depicted are the curves of the times, τ_i, in which each agent would like to meet the other agents, where 100 was used as the initial meeting time for all agents. The zoomed in portion shows the τ_i values for the three agents for time $t = 3440$ to $t = 3500$.

In both of the plots, it is difficult to differentiate between the different trajectories corresponding to the different agents because for the most part, they follow very similar trajectories due to the consensus terms in the cost. The important thing to note is that the agents do indeed end up agreeing to meet closer to the agent that has the lowest initial power level. In this simulation, the final p_i values were 24.1925, 24.2002, and 24.2079, whereas the final τ_i values were 11.1250, 11.1169, and 11.1203, for $i = 1, 2, 3$, respectively.

These values can be compared to those obtained via a centralized method, in which we use an exhaustive search algorithm to find the meeting point that minimizes the time required for all of the agents to meet. This algorithm requires sweeping through all possible values of the meeting point p, so in the given example, p takes on values from 10 to 30. For each p value, we compute the minimum time required for each agent to travel to p, using the fuel-optimal control input, under the given initial positions and power levels. This is done by setting x_{1T} equal to p for every agent and solving for T_{min} using (15).

The solution for T_{min} to (15) using the parameters for agent i will be denoted T_{min_i} for $i = 1, ..., N$. This is the minimum time that it would take agent i to reach point p. The maximum over i of these times is found at each point p, and denoted $T(p)$. Since each agent takes at least T_{min_i} to reach point p, each of them will be able to reach point p in the maximum of these times, $T(p)$. Therefore, $T(p)$ is the time required for all agents to meet at point p. Then we take the minimum of these times over p to find the optimal time, and thus the optimal meeting point.

This procedure is formalized as follows:

$$T(p) = \max_i T_{min_i}(p)$$
$$T^* = \min_p T(p)$$
$$p^* = \arg\min_p T(p),$$

where p^* is the optimal meeting point and T^* is the time required for all agents to reach p^*.

This algorithm resulted in a meeting location of 24.2 in a time of 11.1392. Although there is a slight difference between these numbers and those found via the decentralized method, possibly due to the discrete implementation of continuous time derivatives in the latter, it is important to note that the two algorithms give approximately the same results.

Now that we can find the meeting location that minimizes the time required for the agents to meet, each agent individually can then use (4) to find the input u_i that gets them to the meeting location using the least amount of power.

V. CONCLUSIONS

In this paper, we present a power-aware strategy for allowing a network of agents to achieve rendezvous as quickly as possible. We study both a centralized and a decentralized method for finding the best location for the agents to meet that depends on the initial positions and power levels of the agents. We also present an algorithm that describes how to control a single agent that is moving from some given initial position to some desired final position in a given amount of time such that the overall power consumption (due to mobility) is minimized. This allows us to achieve rendezvous as quickly as possible while also minimizing the amount of power that each agent consumes.

REFERENCES

[1] A. Shia, F. Bastani, and I.-L. Yen, "A highly resilient framework for autonomous robotic swarm systems operating in unknown, hostile environments," in *Autonomous Decentralized Systems (ISADS), 2011 10th International Symposium on*, March 2011, pp. 147–153.

[2] J. Leitner, "Multi-robot cooperation in space: A survey," in *Advanced Technologies for Enhanced Quality of Life, 2009. AT-EQUAL '09.*, July 2009, pp. 144–151.

[3] H. Jaleel, S. Bopardikar, and M. Egerstedt, "Towards power-aware rendezvous," in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, Dec 2011, pp. 6642–6647.

[4] A. Alfieri, A. Bianco, P. Brandimarte, and C. Chiasserini, "Maximizing system lifetime in wireless sensor networks," *European Journal of Operational Research*, vol. 181, no. 1, pp. 390 – 402, 2007.

[5] M. Sarkar and R. Cruz, "An adaptive "sleep" algorithm for efficient power management in wlans," in *Vehicular Technology Conference, 2005. VTC 2005-Spring. 2005 IEEE 61st*, vol. 3, May 2005, pp. 2101–2104 Vol. 3.

[6] R. Subramanian and F. Fekri, "Sleep scheduling and lifetime maximization in sensor networks: fundamental limits and optimal solutions," in *Information Processing in Sensor Networks, 2006. IPSN 2006. The Fifth International Conference on*, 2006, pp. 218–225.

[7] F. Ye, G. Zhong, S. Lu, and L. Zhang, "Peas: a robust energy conserving protocol for long-lived sensor networks," in *Network Protocols, 2002. Proceedings. 10th IEEE International Conference on*, Nov 2002, pp. 200–201.

[8] P. Martin, R. Galvan-Guerra, M. Egerstedt, and V. Azhmyakov, "Power-aware sensor coverage: An optimal control approach," in *Mathematical Theory of Networks and Systems, 19th International Symposium on*, July 2010.

[9] H. Ando, Y. Oasa, I. Suzuki, and M. Yamashita, "Distributed memoryless point convergence algorithm for mobile robots with limited visibility," *Robotics and Automation, IEEE Transactions on*, vol. 15, no. 5, pp. 818–828, Oct 1999.

[10] H. Jaleel and M. Egerstedt, "Power-aware rendezvous with shrinking footprints," in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, Sept 2011, pp. 2772–2777.

[11] Y. Yan and Y. Mostofi, "Efficient communication-aware dynamic coverage using space-filling curves," in *American Control Conference*, June 2014, to appear.

[12] Z. Meng, W. Ren, Y. Cao, and Z. You, "Leaderless and leader-following consensus with communication and input delays under a directed network topology," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 41, no. 1, pp. 75–88, Feb 2011.

[13] W. Ren, "On consensus algorithms for double-integrator dynamics," *Automatic Control, IEEE Transactions on*, vol. 53, no. 6, pp. 1503–1509, July 2008.

[14] W. Yu, G. Chen, M. Cao, and J. Kurths, "Second-order consensus for multiagent systems with directed topologies and nonlinear dynamics," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 40, no. 3, pp. 881–891, June 2010.

[15] J. Qin, W.-X. Zheng, and H. Gao, "Coordination of multiple agents with double-integrator dynamics under generalized interaction topologies," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 42, no. 1, pp. 44–57, Feb 2012.

[16] D. Feijer and F. Paganini, "Stability of primal-dual gradient dynamics and applications to network optimization," *Automatica*, vol. 46, no. 12, pp. 1974–1981, Dec. 2010. [Online]. Available: http://dx.doi.org/10.1016/j.automatica.2010.08.011