

# Using Physical Stigmergy in Decentralized Optimization Under Multiple Non-separable Constraints: Formal Methods and an Intelligent Lighting Example

Theodore P. Pavlic  
School of Life Sciences  
Arizona State University  
Tempe, AZ, USA  
Email: tpavlic@asu.edu

**Abstract**—In this paper, a distributed asynchronous algorithm for intelligent lighting is presented that minimizes collective power use while meeting multiple user lighting constraints simultaneously and requires very little communication among agents participating in the distributed computation. Consequently, the approach is arbitrarily scalable, adapts to exogenous disturbances, and is robust to failures of individual agents. This algorithm is an example of a decentralized primal-space algorithm for constrained non-linear optimization that achieves coordination between agents using stigmergic memory cues present in the physical system as opposed to explicit communication and synchronization. Not only does this work make of stigmergy, a property first used to describe decentralized decision making in eusocial insects, but details of the algorithm are inspired by classic social foraging theory and more recent results in eusocial-insect macronutrient regulation. This theoretical analysis in this paper guarantees that the decentralized stigmergically coupled system converges to within a finite neighborhood of the optimal resource allocation. These results are validated using a hardware implementation of the algorithm in a small-scale intelligent lighting scenario. There are other real-time distributed resource allocation applications that are amenable to these methods, like distributed power generation; in general, this paper means to provide proof of concept that physical variables in cyberphysical systems can be leveraged to reduce the communication burden of algorithms.

**Keywords**—Optimization algorithms, Resource allocation, Pareto optimality, Constrained optimization, Intelligent lighting, Bioinspiration, Agents and autonomous systems, Decentralized control, Distributed optimization, Stigmergy.

## I. INTRODUCTION

Conventional distributed primal- or dual-space algorithms for solving constrained optimization problems with non-separable constraints require some coordination between agents [1]–[9]. Here, a novel numerical approach for non-linear optimization under constraints is described that implements this on-line coordination using *stigmergy*, which is a term that has gained recent popularity in the cooperative robotics literature [e.g., 10] but was originally coined by Grassé [11] to describe the indirect communication among

insects through modifications of the environment. The algorithms discussed in this paper are designed specifically for optimal resource allocation problems where it is guaranteed that the desired solution will exist on a constraint boundary, and these constraint boundaries will provide the environmental cues that guide the collective motion of the distributed actuators in lieu of formal communication between them.

This class of constrained resource allocation problems can be used to describe energy problems such as intelligent lighting and economic power dispatch. However, they can also serve as a description of eusocial insect foraging under nutrient constraints. So rather than simply mimicking biology as in the stigmergy-related robotics literature, this resource allocation model can be used as a general framework to describe both artificial and natural autonomous agents. This paper uses intelligent lighting as a running example, but the general concept of using physical variables as shared memory may enhance many other distributed algorithms running within cyberphysical systems.

This paper is organized as follows. The relevant constrained optimization problem is presented in Section II as well as example applications; among those applications is an intelligent lighting case that serves as a focal example throughout the paper. In Section III, the proposed stigmergic distributed algorithm is given along with theoretical results describing its convergence and ultimate boundedness. Next, experimental results shown in Section IV validate the efficacy of the algorithm in an intelligent lighting case. Finally, brief concluding remarks are included in Section V.

## II. FOCAL PROBLEMS

This section describes the general optimization problem of interest and the motivating applications for this problem and the proposed solution method. The focal example of intelligent lighting is described in detail in Section II-C1, and the model system presented there will be used throughout the paper. A salient feature of the general optimization problem and all of the example applications is that an “all-off” solution is less costly than all other solutions but is infeasible due to operational constraints. For example, it is energetically favorable for a room to always be dark,

but the occupants require at least some light in the room. Consequently, like a shelf supporting a potted plant, the operational constraints are reactive surfaces that support an ever-present attraction to the origin of the resource space.

### A. Focal Optimization Problem

Consider  $n \in \mathbb{N}$  continuous resources where, for each  $i \in \{1, \dots, n\}$ , resource  $i$  can be allocated along a continuum from lower level  $\underline{x}_i \in \mathbb{R}_{\geq 0}$  to an upper level  $\bar{x}_i \in \mathbb{R}_{\geq 0}$ . The convex and continuously differentiable function  $F : \prod_{i=1}^n [0, \bar{x}_i] \rightarrow \mathbb{R}$  maps the resource space  $\mathcal{X} \triangleq \prod_{i=1}^n [\underline{x}_i, \bar{x}_i]$  to a cost which the optimal allocation will minimize. It is assumed that function  $F$  is not constant and that the gradient of  $F$  increases along each dimension of the resource space; that is, for each  $i \in \{1, \dots, n\}$  and  $\vec{x} \in \mathcal{X}$  and  $\delta \in \mathbb{R}_{> 0}$ ,  $\nabla_i F(\vec{x} + \delta \vec{e}_i) > \nabla_i F(\vec{x})$  where  $\vec{e}_i$  is the elementary vector along the  $i^{\text{th}}$  direction. So although  $F$  is convex, it need not be strictly convex.

Next, take  $m \in \mathbb{N}$  to be the number of linear inequality constraints. In particular, for each  $j \in \{1, \dots, m\}$ , a feasible  $\vec{x} \in \mathcal{X}$  is such that  $\vec{a}_j^\top \vec{x} \geq c_j$  where  $\vec{a}_j \in \mathbb{R}_{\geq 0}^n$  and  $c_j \in \mathbb{R}_{\geq 0}$  are parameters representing the  $j^{\text{th}}$  operational constraint. For each constraint  $j \in \{1, \dots, m\}$ , the constraint boundary  $\{\vec{x} \in \mathcal{X} : \vec{a}_j^\top \vec{x} = c_j\}$  can be drawn as a hyperplane in resource space  $\mathcal{X}$ , and the vector  $\vec{a}_j$  is normal to that hyperplane and points toward allocations that continue to meet the constraint. To prevent the pathological case where all constraints are trivially met by the origin of the resource space, it is additionally assumed that there is some constraint  $j \in \{1, \dots, m\}$  such that  $\sum_{i=1}^n a_{ji} > 0$ .

Thus, the focal optimization problem is to

$$\begin{aligned} & \text{minimize} && F(\vec{x}) \\ & \text{subject to} && A\vec{x} \geq \vec{c} \\ & && E\vec{x} \geq [\underline{x}_1, \underline{x}_2, \dots, \underline{x}_n]^\top \end{aligned} \quad (1)$$

with elementary matrix  $E \triangleq [\vec{e}_1, \dots, \vec{e}_n]^\top$ , constraint matrix  $A \triangleq [\vec{a}_1, \dots, \vec{a}_m]^\top$ , and constraint vector  $\vec{c} \triangleq [c_1, \dots, c_m]^\top$ .

### B. Characterization of Optimal Solutions

An optimal solution to equation (1) must exist; however, there may be several optima. Let  $\vec{x}^* \in \mathcal{X}$  be an optimal solution. By the Karush–Kuhn–Tucker (KKT) conditions [12], there must exist a scalar Lagrange multiplier  $\lambda_j^* \in \mathbb{R}_{\geq 0}$  for each constraint  $j \in \{1, \dots, m\}$  and scalar Lagrange multipliers  $\mu_i^*, \nu_i^* \in \mathbb{R}_{\geq 0}$  respectively for the lower and upper bounds  $\underline{x}_i, \bar{x}_i$  for each  $i \in \{1, \dots, n\}$  such that

$$\begin{aligned} \nabla F(\vec{x}^*) &= \lambda_1^* \vec{a}_1 + \dots + \lambda_m^* \vec{a}_m \\ &+ (\mu_1^* - \nu_1^*) \vec{e}_1 + \dots + (\mu_n^* - \nu_n^*) \vec{e}_n \end{aligned} \quad (2)$$

where  $\vec{e}_i$  is an elementary vector for  $i \in \{1, \dots, n\}$ . Thus, when  $\vec{x}^*$  is in the interior of  $\mathcal{X}$ , it is the case that the multipliers  $\mu_i^* = 0$  and  $\nu_i^* = 0$  for all  $i \in \{1, \dots, n\}$ , and the gradient  $\nabla F(\vec{x}^*)$  is a conical combination of the vectors

normal to each active constraint. Using the example of the potted plant on a shelf from earlier, the gradient plays the role of the force of gravity and the constraint vectors play the role of normal forces supporting the potted plant from below. Moreover, the additional constraints provided by the upper and lower bounds are like bookends on that bookshelf.

### C. Example Applications

The non-linear optimization problem in equation (1) can be shown to be a more general case of existing problems in engineering and theoretical ecology. In the following examples, a resource is being optimally distributed across to a set of tasks (e.g., power allocated to lights for minimum luminance, power allocated across generators to supply required demand, foragers allocated to food patches to supply minimum nutrients). In each case, the constrained optimization problem is equivalent to the projection of the origin (i.e., a supremely favorable operating point) onto the constraint set (i.e., requirements on the system that prevent operating at minimum cost). Consequently, each active constraint may be viewed as a surface supporting a descending particle, and each inactive constraint may be viewed as an even lower surface whose interaction with the particle is prevented by the active surfaces above it.

1) *Intelligent Lighting:* In the built environment, there are usually several overhead sources of artificial light as well as windows that provide variable levels of natural light. Intelligent lighting systems have been proposed that use feedback control of individual lights to meet occupant preferences measured at sensors distributed throughout the environment. Although some schemes are designed to control the color of light [13], most approaches are ostensibly if not explicitly designed to reduce the power used by artificial sources [9], [14]–[18]. However, few power-saving methods are constructed within an optimization framework; instead, they save power solely by eliminating preference surpluses. In these over-actuated systems (i.e., many more lights than lighting constraints), there is a continuum of resource allocations that can meet all constraints, but the power demanded by each of these constraint-equivalent allocations can vary greatly. Consequently, simply eliminating preference surpluses does not guarantee minimal power use. The algorithms that are sensitive to measurements of power or energy are either heuristic [15] or assume optimal solutions are provided by an exogenous procedure [17]. Moreover, the optimality criteria chosen are often *ad hoc* and either lack physical significance or lead to impractical equilibrium solutions. So intelligent lighting research is presently a vacuum for rigorous distributed optimization results.

Here, an intelligent lighting optimization problem similar to the one used by Wen and Agogino [17] is presented, but our approach uses inequality constraints to match the form of equation (1). By using inequality constraints, different

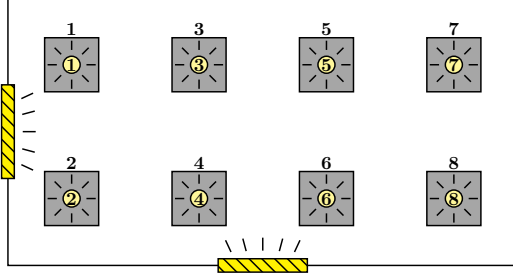


Figure 1. Top view of prototypical lighting system with  $n = 8$  lights and  $m = 8$  sensors. Lights, which are shown as circles around their indexes, are mounted in the ceiling of the room. Sensors, which are shown as patches underneath their indexes, are located at a distance beneath the lights. There are two disturbance sources (e.g., windows) shown as hatched rectangles. In general,  $m \neq n$  and sensors need not be placed directly beneath lights.

solutions activate different constraints and the optimization problem always has feasible solutions. This approach is similar to the minimal-power inequality-constrained problem motivating the recent distributed lighting controller of Caicedo and Pandharipande [9]. However, that sophisticated controller requires explicit communication of control variables with local neighbors, and it converges to a sub-optimal neighborhood of the optimal solution that cannot be made arbitrarily small. The lighting controller proposed here requires no explicit communication and converges to a suboptimal neighborhood which can be made arbitrarily small by selection of a parameter that reduces the speed of the system. This intelligent lighting case will be used as a motivating example throughout the rest of the paper.

Let there be  $n \in \mathbb{N}$  lights positioned above  $m \in \mathbb{N}$  sensors. The lights and sensors can be in any position, but one possible configuration is the room depicted in Figure 1 where eight lights are shown as circles, eight sensors are shown as squares, and two disturbance sources (e.g., windows) are shown as rectangles in the perimeter of the room; this picture shows  $n = m$ , but typically  $n > m$ . In general,

- Each light  $i \in \{1, \dots, n\}$  is actuated by control  $x_i \in [\underline{x}_i, \bar{x}_i] \subseteq \mathbb{R}_{\geq 0}$  where  $\infty > \bar{x}_i \geq \underline{x}_i \geq 0$ .
- For each sensor  $j \in \{1, \dots, m\}$ , if the control for light  $i \in \{1, \dots, n\}$  increases from  $x_i$  to  $x_i + \delta_i$  with  $\delta_i \in \mathbb{R}_{> 0}$  while all other lighting controls remain constant, the reading of sensor  $j$  increases by  $a_{ji}\delta_i$ .
- Associated with each sensor  $j \in \{1, \dots, m\}$  is a minimum desired sensor level  $\gamma_j \in \mathbb{R}_{\geq 0}$  and a positive sensor bias  $b_j \in \mathbb{R}_{\geq 0}$  due to an exogenous light source (e.g., the sun); it is taken for granted that  $b_j$  is constant or changes at a slow rate relative to the speed of the control system. Assuming that weight-vector  $\vec{a}_j$  and the present value of control variables  $\vec{x}$  is available to sensor  $j \in \{1, \dots, m\}$  [e.g., 19], the value of bias  $b_j$  can be estimated by subtracting the expected sensor value from the actual sensor value. Thus, to harvest

the injected daylight, the effective minimum constraint  $c_j \triangleq \gamma_j - b_j$  for each sensor  $j \in \{1, \dots, m\}$ . So the intelligent lighting system must

$$\begin{aligned} & \text{minimize} && F(\vec{x}) \\ & \text{subject to} && a_{11}x_1 + \dots + a_{1n}x_n \geq \gamma_1 - b_1, \\ & && \vdots \\ & && a_{m1}x_1 + \dots + a_{mn}x_n \geq \gamma_m - b_m, \end{aligned} \quad (3)$$

which is equivalent to equation (1). This paper considers a general class of objective functions; however, a natural choice is to minimize the instantaneous power  $F(\vec{x}) = \|\vec{x}\|_2^2$ . Mathematically, this choice projects the origin (i.e., all lights off) onto the constraint set using the Euclidean distance. Moreover, for each  $i \in \{1, \dots, n\}$ , if  $x_i$  represents a control voltage (e.g., the RMS voltage of a dimmed AC power signal) across a linear lighting element, this policy will minimize the power used by a group of identical lights. Alternatively, for each  $i \in \{1, \dots, n\}$ ,  $x_i$  may simply be a set-point reference for a lower-level control system on the individual light (e.g.,  $x_i$  may be the input to a dimmable fluorescent ballast or LED driver electronics). In that case, minimization of  $\|\cdot\|_2^2$  will create diffuse aesthetically pleasing pools of light as opposed to single bright point sources that would be induced by 1-norm minimization.

2) *Economic Power Dispatch*: The basic economic dispatch problem in power engineering is summarized by Bergen and Vittal [20]. There are  $n \in \mathbb{N}$  generators that supply the  $P_D \in \mathbb{R}_{\geq 0}$  power demanded by a given community. At each generator  $i \in \{1, \dots, n\}$ , the cost  $C_i(P_i)$  of supplying  $P_i \in \mathbb{R}_{\geq 0}$  units of power is assumed to come from the convex increasing function  $C_i: \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ . Thus, the optimal allocation  $\vec{P}^* = [P_1, P_2, \dots, P_n]^\top$  is the solution to the problem

$$\begin{aligned} & \text{minimize} && \sum_{i=1}^n C_i(P_i) \quad \text{subject to} \quad \sum_{i=1}^n P_i \geq P_D \end{aligned} \quad (4)$$

So the methods from this paper are amenable to the distributed solution of the economic power dispatch problem. Moreover, the problem can be extended for multiple power demand constraints. For example, consider  $m = 2$  communities, and let community  $j \in \{1, 2\}$  demand  $P_{Dj}$  units of power and receive  $a_{ji}$  fraction of its power from generator  $i \in \{1, \dots, n\}$ . Due to network effects (e.g., distance, connectivity), the vector  $\vec{a}_1 \neq \vec{a}_2$ . Thus, the extension of equation (4) for this problem is to

$$\begin{aligned} & \text{minimize} && \sum_{i=1}^n C_i(P_i) \\ & \text{subject to} && a_{11}P_1 + \dots + a_{1n}P_n \geq P_{D1}, \\ & && a_{21}P_1 + \dots + a_{2n}P_n \geq P_{D2}, \end{aligned} \quad (5)$$

where one or both constraints may be active. When both constraints are active, the optimal solution to this problem

cannot be solved using numerical methods described by Bergen and Vittal. However, this problem does have the form of equation (1) and is applicable to the methods in this paper. Moreover, the decentralized algorithm discussed in this paper is well suited for spatially distributed generation of power.

3) *Ideal Free Distribution and Nutrient Foraging*: The ideal free distribution (IFD) of social foraging theory was originally introduced by Fretwell and Lucas [21], and a review of recent biological advances in the theory is given by Stephens et al. [22, Box 10.1]. Engineering extensions of IFD theory have lately been used to solve distributed resource allocation problems in application areas such as temperature control, municipal water distribution, and autonomous-air-vehicle surveillance [23]–[28]. Here, the IFD is shown to be a degenerate single-constraint case of equation (1), and so the methods presented in this paper can also be used to find distributed IFD solutions. However, because the framework in this paper generalizes the IFD, it can provide more practicality for engineering applications and more explanatory power in social foraging analysis.

In the basic IFD model, each of  $N \in \mathbb{N}$  foragers is free to move among  $n$  food patches. At each food patch, food arrives at some rate and then is distributed among the foragers within the patch. So whenever a new forager enters a patch, the food available to each of the previous patch occupants decreases per unit time. In the IFD model, it is assumed that each forager that is free to move among these  $n$  patches has perfect knowledge of a continuous and decreasing suitability function  $s_i : [0, N] \rightarrow \mathbb{R}_{>0}$  that maps the number of foragers in patch  $i \in \{1, \dots, n\}$  to a subjective patch-quality valuation related to decreasing food availability. When suitability of one patch is higher than the suitability of another, some small number of animals immediately move from the higher-suitability patch to the lower-suitability patch so that the imbalance between all suitabilities decreases to its minimum. This search for suitability balance is analogous to the numerical computation of a Lagrange multiplier for a singly constrained separable convex optimization of a differentiable cost function. Consequently, the IFD can be described by the solution to the problem

$$\text{minimize } \sum_{i=1}^n \int_0^{x_i} \frac{1}{s_i(\tau)} d\tau \text{ subject to } \sum_{i=1}^n x_i \geq N \quad (6)$$

which matches equation (1) and is applicable to the methods in this paper. However, the IFD can be generalized to a more useful model that is also applicable to methods in this paper.

Quantitative predictions of an IFD require that population size  $N$  is known *a priori*. Consequently, conventional bio-inspired resource allocation techniques based on the IFD [25], [28] introduce an artificial constraint corresponding to this fixed population size. For example, over a bank of temperature controllers, the sum of the actuator voltages is fixed at

a constant population size, and that population of voltages distributes itself across actuators in different temperature zones. In this case, the suitability in a zone represents the temperature error from the desired set point – a temperature zone is relatively more suitable for entry if it is farther from its desired temperature set point. So as the IFD-inspired algorithm operates, foraging voltages accumulate in temperature zones, suitabilities equalize, and temperatures approach their set points. If the desired set points can be met without the total population being allocated, a fictitious temperature zone is introduced in software that provides a region for the surplus population to be stored. So the population constraint adds extra complications to the implementation of these resource allocation techniques. Moreover, although set points are reached and less power is used than conventional distributed regulation techniques (e.g., each zone having a PID controller operating independently in parallel with the controllers in the other zones), there is no guarantee that the regulating resource allocation also achieves minimal power use. So it is more desirable to remove the fixed population requirement and introduce sensitivity to meaningful optimization metrics, as is described in Section II-C1 for minimal-power lighting.

Not only is the constant-population assumption impractical for bio-inspired engineering applications, but it is also unnatural for the analysis of social foraging in some cases. For colonial foraging (e.g., an ant colony), foragers are a team that do not simply maximize caloric intake but instead simultaneously regulate the intake of several macronutrients (e.g., protein and carbohydrates) to colony-specific set points [29], and that team has the ability to adapt its size based on the richness of the environment. The solution to

$$\begin{aligned} \text{minimize } & \sum_{i=1}^n \int_0^{x_i} \frac{1}{s_i(\tau)} d\tau \\ \text{subject to } & a_{11}x_1 + \dots + a_{1n}x_n \geq c_1, \\ & \vdots \\ & a_{m1}x_1 + \dots + a_{mn}x_n \geq c_m, \end{aligned} \quad (7)$$

matches the IFD case in equation (6) with  $m = 1$  and  $\vec{a}_1 = \vec{1}$ , but it also allows the single population constraint to be replaced with possibly many macronutrient constraints. So it is a better description of the colonial foraging problem in general. In this colonial case, the suitability function represents the desired spread of foraging effort to meet the requirements. For example, letting  $s_i(\tau)^{-1} = 1$  for each  $i \in \{1, \dots, n\}$  minimizes the sum of the foragers required at each patch, which leads to clumps of foragers at only the richest patches. Letting  $s_i(\tau)^{-1} = \tau$  for each  $i \in \{1, \dots, n\}$  increases the number of foragers required to meet the constraints, but the foraging distributions are spread across more patches, and thus is more risk averse. So the number of required foragers (i.e., resources being allocated)

scale with nutrient constraints, richness of the environment, and desired spread. These are desirable features for explaining allocations in nature and they are useful in designing resource allocation strategies for engineering applications. Likewise, the on-line optimization method described later in this paper may be viewed as a model of nutrient-constrained foraging dynamics or a guide for decentralized strategies to solving this problem.

### III. NATURE-INSPIRED DECENTRALIZED ALGORITHM

Using Section II-C1 as a motivating example, a distributed optimization algorithm is developed here for intelligent lights. This algorithm was originally inspired by the ideal free distribution (IFD) [21] of social foraging theory, which is described in Section II-C3. Like other uses of the IFD in engineering [23]–[28], each step of this algorithm causes abstract “animals” (e.g., marginal units of a fixed power budget or members of a vehicle) to move from less suitable “food” patches to more suitable ones. However, unlike this algorithm, those algorithms endow information-processing agency to the actuators. That is, either the patches (e.g., heaters across rooms) or the resources themselves (e.g., vehicles) gather information from sensors and nearby patches, and resources are passed from one patch to another. Existing work in intelligent lighting takes a similar actuator-centric approach [9], [13]–[18], [30] – agents sit at the level of lights and adjust their local lighting level based on communication with sensors and surrounding lights. Not only do those approaches require direct communication between lights, but they require significant system-level reconfiguration when constraints are added, removed, or shifted.

*A Sensor-Centric Approach:* The algorithm here puts information-processing agency at the level of each sensor. While the important details of the algorithm are given below, the salient features of this algorithm are depicted graphically in Figure 2. In this sensor-centric paradigm, the actuators have naive but consistent behaviors in isolation – they all reduce allocation levels at a uniform rate, as seen in Figure 2(b). This behavior will inevitably cause a constraint to be violated at one of the sensors. At that point, the sensor commands a subset of lights to increase according to a particular non-uniform profile. Because lights decrease uniformly and are commanded to non-uniformly increase, the result causes resources to accumulate more in some lights than others – like animals shifting to different patches in an IFD. Sensors giving these commands do not communicate with each other. However, because they share access to the artificial light, their actions become correlated in a way that ensures that the sum of the non-uniform increase commands will be uniform at the optimal lighting allocation. The uniform upward sum will be precisely balanced by the uniform lighting decay, and thus the system comes to equilibrium, as shown in Figures 2(b) and 2(c). In this regard, the action of a system with multiple constraints is more

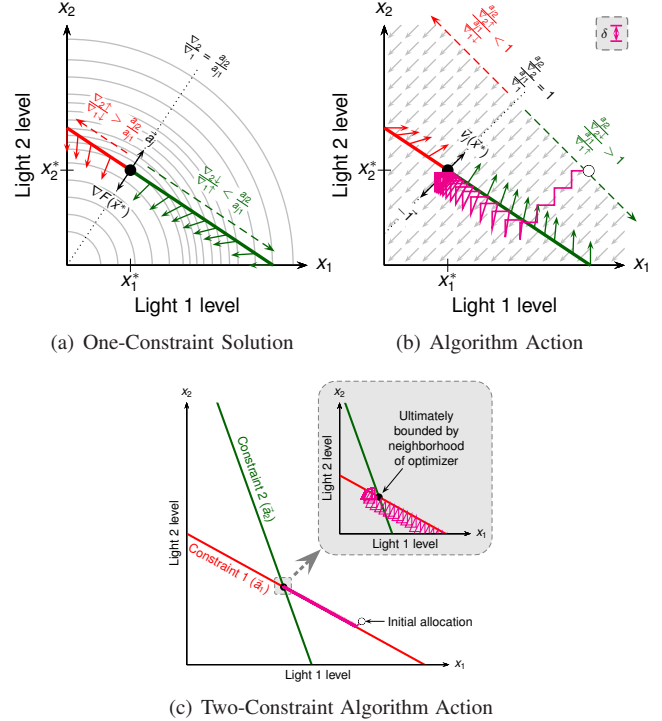


Figure 2. Graphical depiction of algorithm action for  $n = 2$  lights with levels  $\vec{x} = [x_1, x_2]^T$ . The light gray curves in (a) are level sets of a cost function  $F(\vec{x}) = \|\vec{x}\|_2^2 = x_1^2 + x_2^2$ , and the oblique line shows the zero-surplus  $\vec{x}$  allocations that meet the constraint  $\vec{a}_j^T \vec{x} \geq c_j$  with strict equality. The minimal-cost solution  $\vec{x}^*$  is the solid dot occurring where the constraint line is tangent to the cost function. At that point, vector  $\vec{a}_j$  is co-linear with cost gradient  $\nabla F(\vec{x}^*)$ . Due to constraint convexity and cost-gradient monotonicity, the gradient is steeper (shallower) to the left (right) of  $\vec{x}^*$  when sliding along the constraint boundary. In (b), the algorithm action for (a) is shown. The gray diagonal arrows show the mean action of two lights that regularly reduce illumination level by value  $\delta$ . The arrows pointing upward from the constraint boundary show each marginal increase  $\vec{v}_j$  commanded by the sensor when the constraint is violated at point  $\vec{x}$ . For light  $i \in \{1, \dots, n\}$ , coordinate  $v_{ji}(\vec{x}) = a_{ji} / \nabla_i F(\vec{x})$ . Thus, the slope of  $\vec{v}_j$  here is  $v_{j2}/v_{j1} = (a_{j2}/a_{j1}) / (\nabla_2 F(\vec{x}) / \nabla_1 F(\vec{x}))$ , which is unity only at the optimal solution  $\vec{x}^*$  because the constraint vector  $\vec{a}_j$  and gradient  $\nabla F(\vec{x}^*)$  are co-linear. The unity-slope increase balances the unity-slope decrease from the naive decaying action of the lights. A sample trajectory for the system is shown originating at the open circle and moving toward the closed dot at the equilibrium. It becomes asymptotically closer to a set of points inside a  $\delta$ -sized square anchored near the solution  $\vec{x}^*$ . In (c), a sample trajectory is shown for a case with two active constraints – the optimal solution is the intersection of the two constraint lines, which is found by the algorithm as movement along the second constraint is balanced by opposite movement along the first constraint.

like a macronutrient-constrained social-insect colony with foragers that continually retire from foraging and then are re-allocated independently when each macronutrient constraint is violated.

*Notation:* In the following, let there be  $n \in \mathbb{N}$  lighting agents and  $m \in \mathbb{N}$  sensor agents that each have access to the  $n$  lighting control variables  $\vec{x} \triangleq [x_1, x_2, \dots, x_n]^T \in \mathcal{X}$ . For example, the lighting levels can be learned by the sensors through observing broadcasts over radio or even visual-light

communication [19] embedded within the light itself. As in Section II-C1, there are  $m \in \mathbb{N}$  luminescence constraints of the form  $\vec{a}_j^\top \vec{x} \geq c_j$  for each  $j \in \{1, \dots, m\}$ . There is also a sufficiently small parameter  $\delta \in \mathbb{R}_{>0}$  that is chosen such that  $\delta < c_j/(\vec{a}_j^\top \vec{1})$  for all  $j \in \{1, \dots, m\}$ . In the algorithm below, this parameter can be tuned to increase convergence speed (i.e., high  $\delta$ ) or increase the accuracy of the equilibrium solution (i.e., low  $\delta$ ).

#### A. Algorithm Action

The algorithm implemented on each agent operates asynchronously and results in the system as a whole jumping from one state (i.e., set of lighting levels) to another state at discrete events. Lighting agents act regularly to decrease allocations while sensor agents only act to increase lighting levels when corresponding constraints cease to be met. In the following, assume that  $\nabla_i F(\vec{x}) > 0$  for each  $i \in \{1, \dots, n\}$ ; this assumption is met if the lower bound  $\underline{x}_i > 0$  for all  $i \in \{1, \dots, n\}$ . For simplicity, the upper and lower bounds of the lighting levels are not discussed here; an actual implementation would truncate lighting-level changes to their bounds.

- At regularly scheduled events, each lighting agent  $i \in \{1, \dots, n\}$  will decrease its lighting level by  $d_i$ , which may in general depend on state  $x_i$  or time, but will be taken as  $d_i \equiv \delta$  unless otherwise noted. Thus, the lighting agent may operate without knowledge of the gradient nor states of other lights. In the social-insect analogy, this action corresponds to regular attrition of foragers allocated to a particular patch. An individual may transition out of foraging due to either a behavioral timeout or because she is unable to detect a global signal, such as a macronutrient-specific pheromonal cue that decays at a constant rate.
- When its corresponding constraint is violated, sensor  $j \in \{1, \dots, m\}$  causes the  $n$  lighting levels in vector  $\vec{x}$  to transition to new levels  $\vec{x}^{\text{next}}$  such that

$$\vec{x}^{\text{next}} = \vec{x} + \begin{cases} \sigma_j \vec{v}_j & \text{if } \vec{a}_j^\top \vec{x} \leq c_j, \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

where state-dependent direction

$$\vec{v}_j \triangleq \left[ \frac{a_{j1}}{\nabla_1 F(\vec{x})}, \frac{a_{j2}}{\nabla_2 F(\vec{x})}, \dots, \frac{a_{jn}}{\nabla_n F(\vec{x})} \right]^\top \quad (9)$$

and scalar  $\sigma_j > 0$ . For each  $j \in \{1, \dots, m\}$ , the scalar  $\sigma_j$  may depend on  $\vec{x}$ . Unless otherwise noted, we take

$$\sigma_j \triangleq \frac{c_j - \vec{a}_j^\top \vec{x}}{\vec{a}_j^\top \vec{v}_j}, \quad (10)$$

which ensures that  $\vec{a}_j^\top \vec{x}^{\text{next}} = c_j$  if sensor  $j \in \{1, \dots, m\}$  acts alone. This scalar can also be related to the index proposed by Yamamoto [31] for ordering active and inactive constraints in a linear programming

problem. It is assumed that each sensor has access to broadcasted values of the state vector  $\vec{x}$ ; however, no sensor has any information about any other sensor. Coordination between sensors is stigmergic and makes use of the state variables (i.e., the lighting levels available to all sensors simultaneously) as shared memory in the system. In the social-insect analogy, this action corresponds to colony-wide deficits in one macronutrient triggering an increase in foraging allocation biased to bring in more of that macronutrient from appropriate patches.

*Improving Transient Characteristics:* When components of  $\vec{x}$  are sufficiently large, lights will go for many update cycles without receiving signals from sensors. Thus, decay  $d_i$  of light  $i \in \{1, \dots, n\}$  can be temporarily increased during times of little sensor contact in order to improve transient response. Similarly, to reduce system overshoot from simultaneous action of many sensors, each sensor  $j \in \{1, \dots, m\}$  can temporarily reduce scalar  $\sigma_j$  when it is far under its constraint. In this respect, the action of the algorithm is similar to the action of interior-point optimization algorithms that drive the state toward constraint boundaries. From the perspective of sliding mode control [32], these far-from-boundary actions merely establish the constraint boundary a sliding mode of the system.

#### B. Ultimate Boundedness and Stability

As with many primal-space algorithms, there is no single equilibrium point where this algorithm will come to rest. Here, it is shown that the motion of this algorithm can be eventually constrained to a ball around the optimal solution of equation (1); moreover, the size of this ball can be made arbitrarily small by decreasing the size of  $\delta$ , albeit at the cost of decreasing convergence speed. Using the IFD interpretation, for each  $i \in \{1, \dots, n\}$  and  $j \in \{1, \dots, m\}$ , the component  $v_{ji} = a_{ji}/\nabla_i F(\vec{x})$  represents the suitability of light  $i$  induced by the constraint at sensor  $j$ . The motion of the algorithm will continually move resources from low suitability lights to relatively higher suitability lights. Because of the coordinate-level monotonicity of  $\nabla F$ , a decrease (increase) in a lighting level will increase (decrease) its suitability. Thus, the algorithm causes all suitabilities to collapse onto a single common suitability. At that point  $\vec{x}^*$ , the gradient  $\nabla F(\vec{x}^*)$  is co-linear with the constraint vector  $\vec{a}_j$ , which implies that the system arrives at an optimal equilibrium (Figure 2(a)). For multiple constraints, actuator bounds, or weakly convex optimization functions, this motion continues until balanced by actuator truncation or collision with motion from another sensor.

*1) Simplifying Assumptions:* For simplicity, the timing of events is slightly constrained in this analysis. Agents update asynchronously in general, and some agents may update simultaneously, but the order of events must be fixed. This assumption is met, for example, if all agents are triggered

by local clocks that have been manufactured to tick at the same rate. Formally, it is assumed that there is an  $M \in \mathbb{N}$  such that the set of agents triggered at time  $k$  must be equal to the set of agents triggered at time  $k + M$ . The state of the system at time  $k$  is denoted  $\vec{x}[k]$ .

Here, consider sensor  $j \in \{1, \dots, m\}$  and assume that the system is operating in a region that meets constraints of all other sensors. Consequently,  $j$  is the only sensor that will be activated in this analysis. Likewise, it is assumed that the system light levels are operating sufficiently far from their upper and lower bounds so as to not activate truncation constraints. The analysis starts at a time  $k$ , and sensor  $j$  will have its influence on state  $\vec{x}$  at time  $k + 1$ .

2) *A Period- $M$  Invariant Sliding Mode:* If  $\vec{x}[k]$  is such that  $\vec{a}_j^\top \vec{x}[k] > c_j$ , then sensor  $j$  will be inactive because it will be far over its constraint, but the persistent decaying action of the lighting agents will eventually cause constraint  $j$  to be violated. Hence, assume that  $\vec{x}[k]$  is such that  $\vec{a}_j^\top \vec{x}[k] \leq c_j$ . Then, at the time  $k + M$  when sensor  $j$  and all lighting agents have also executed,

$$\begin{aligned} \vec{a}_j^\top \vec{x}[k + M] &= \vec{a}_j^\top \left( \vec{x}[k] + \underbrace{\frac{\sigma_j \vec{v}_j \text{ from sensor } j}{\vec{a}_j^\top \vec{v}_j[k]} \vec{v}_j[k]}_{\text{combined lighting decay}} - \vec{1}\delta \right) \\ &= c_j - \vec{a}_j^\top \vec{1}\delta. \end{aligned} \quad (11)$$

That is, after violating the constraint at time  $k$ , the state  $\vec{x}[k + M]$  will return to the set  $\mathcal{P}_j \triangleq \{\vec{x} \in \mathcal{X} : \vec{a}_j^\top \vec{x} = c_j - \vec{a}_j^\top \vec{1}\delta\}$ , which is parallel to the constraint boundary for sensor  $j$ . In fact,  $\mathcal{P}_j$  is the set of allocations that satisfy the sensor  $j$  constraint if it is reduced slightly from  $c_j$  to  $c_j - \vec{a}_j^\top \vec{1}\delta$ . So  $\mathcal{P}_j$  is positively invariant with respect to a version of the system re-sampled at period  $M$  after the time  $k$ . Moreover, at all times between these period- $M$  samples, the system is confined to

$$\mathcal{C}_j \triangleq \{\vec{x} \in \mathcal{X} : \vec{a}_j^\top \vec{x} = c_j - \delta \vec{a}_j^\top \sum_{i \in \mathcal{A}} \vec{e}_i, \mathcal{A} \subseteq \{1, \dots, n\}\},$$

and so  $\mathcal{C}_j$  is positively invariant with respect to the full system. Surfaces  $\mathcal{C}_j$  and  $\mathcal{P}_j$  are reduced-order sliding modes [32] of the full and period- $M$  sampled systems, respectively.

3) *Optimality of Period- $M$  Equilibria:* In Section II-B, it was shown that the optimal solution  $\vec{x}^*$  for a single active constraint will be such that  $\nabla F(\vec{x})^* = \lambda_j^* \vec{a}_j$  for some  $\lambda_j^* > 0$ . Consequently, the corresponding  $v_{ji} = a_{ji}/\nabla F(\vec{x}^*) = 1/\lambda_j^*$ , and  $\vec{v}_j \propto \vec{1}$ . In the IFD interpretation, if  $v_{ji}$  represents the suitability of patch  $i$ , then the IFD represents the equal-suitability distribution. Assuming that  $\vec{v}_j[k + M] \propto \vec{1}$  and  $\vec{x}[k + M] \in \mathcal{P}_j$ , then

$$\begin{aligned} \vec{x}[k + M + M] &= \vec{x}[k + M] + \frac{\delta \vec{a}_j^\top \vec{1}}{\vec{a}_j^\top \vec{v}_j[k]} \vec{v}_j[k] - \delta \vec{1} \\ &= \vec{x}[k + M] + \delta \vec{1} - \delta \vec{1} = \vec{x}[k + M] \end{aligned}$$

Consequently, the optimal solution for the system when constraint  $c_j$  is reduced to  $c_j - \vec{a}_j^\top \vec{1}\delta$  is also a period- $M$  fixed point for the system. In fact, any period- $M$  fixed point has  $v_j \propto \vec{1}$ . Thus, the full system has a period- $M$  steady state that oscillates near the vicinity of the solution  $\vec{x}^*$ .

4) *Stability By Suitability Range Compression:* Here, it is shown that each iteration of the algorithm increases the allocation to the patch with the highest suitability and decreases the allocation to the patch with the lowest suitability. So long as the identity of the highest and lowest suitability eventually stabilizes, the system will converge because all suitabilities will collapse onto a single global suitability and  $\vec{v} \propto \vec{1}$ . For brevity, the discussion here establishes a very low upper bound on  $\delta$  that ensures that the identities of the highest-and lowest-suitability patch are fixed for all time. It is left for future work to show that a tight upper bound on  $\delta$  is much higher.

Let  $u, \ell \in \{1, \dots, n\}$  such that  $a_{ju}/\nabla_u F(\vec{x}) > a_{j\ell}/\nabla_\ell F(\vec{x})$ . Because  $\nabla F$  is continuous and  $x_i \in [\underline{x}_i, \bar{x}_i] \subset (0, \infty)$ , there is some  $\bar{\Delta} \in \mathbb{R}_{>0}$  such that  $a_{ju}/\nabla_u F(\vec{x} + \Delta) > a_{j\ell}/\nabla_\ell F(\vec{x} - \Delta)$  for all  $\Delta \in (0, \bar{\Delta}]$ . Next, assume the period- $M$  system is within  $\mathcal{P}_j$  at time  $k$  but not at a fixed point. Let  $\bar{i} = \arg \max_{i=1}^n v_{ji}[k]$  and  $\underline{i} = \arg \min_{i=1}^n v_{ji}[k]$ . Here it will be shown that, with appropriate choice of  $\delta$ ,  $0 < x_{\bar{i}}[k + M] - x_{\bar{i}}[k] < \bar{\Delta}$  and  $0 > x_{\underline{i}}[k + M] - x_{\underline{i}}[k] > -\bar{\Delta}$ . Thus,  $\bar{i}$  and  $\underline{i}$  do not vary while the system slides along  $\mathcal{P}_j$ . The minimum and maximum suitabilities,  $v_{j\bar{i}}$  and  $v_{j\underline{i}}$ , always correspond to the same two patches, and the difference between these suitabilities continually decreases. So the system converges to the  $\vec{v}_j \propto \vec{1}$  fixed point.

Because  $\vec{x}[k] \in \mathcal{P}_j$ ,

$$x_{\bar{i}}[k + 1] - x_{\bar{i}}[k] = \delta \left( \frac{a_{j1} + \dots + a_{jn}}{a_{j1} \frac{v_{j\bar{i}}}{v_{j\underline{i}}} + \dots + a_{jn} \frac{v_{j\underline{i}}}{v_{j\bar{i}}}} - 1 \right).$$

However,  $v_{ji}/v_{j\bar{i}} \leq 1$ . Consequently,  $x_{\bar{i}}[k + 1] - x_{\bar{i}}[k] > 0$  and can be made arbitrarily small by choice of  $\delta$ . By similar reasoning, it can be shown that  $x_{\underline{i}}[k] - x_{\underline{i}}[k + 1] > 0$  and can also be made arbitrarily small by choice of  $\delta$ . Thus, by the reasoning above, suitabilities tend to equalize, and the period- $M$  re-sampled system converges.

5) *Ultimate Bounds of Suboptimality:* It has been shown that for sufficiently small  $\delta$ , the period- $M$  re-sampled system converges to a suboptimal allocation  $\vec{x}^+$  that is  $F$ -optimal subject to the modified constraint that  $\vec{a}^\top \vec{x} \geq c_j - \vec{a}^\top \vec{1}\delta$ . Moreover, it will not deviate from this result by more than  $\delta$  in each coordinate. The distance between  $\vec{x}^+$  and  $\vec{x}^*$  varies with the cost function in general. For example, for  $F(\vec{x}) = \|\vec{x}\|_2^2$ , both  $\vec{x}^*$  and  $\vec{x}^+$  are proportional  $\vec{a}_j$  and each other. In this case, it is possible to show that the steady state of the system oscillates within a  $2\delta$  max-norm distance of  $\vec{x}^*$ . On the other hand, internally increasing the constraint by  $\vec{a}^\top \vec{1}\delta$  ensures that  $\vec{x}^+ = \vec{x}^*$ , which decreases max-norm suboptimality to  $\delta$ .

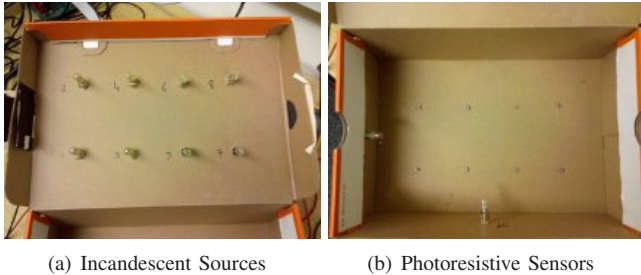


Figure 3. Small-scale Lighting Testbed Similar to Figure 1.

This discussion assumes that  $\bar{x}^*$  only involves the activation of a single constraint  $j$ . As shown in Figure 2(c), the sliding motion along one constraint can collide with the sliding motion along another. It can be shown that the algorithm described here will reach equilibria that rest on multiple constraint boundaries, and it can be shown that the suboptimal solutions can also be made arbitrarily close to the optimal solution by appropriate choice of  $\delta$ .

#### IV. EXPERIMENTAL VALIDATION

Experimental validation of the distributed-lighting algorithm was performed using an 8-light–8-sensor hardware-in-the-loop apparatus resembling Figure 1. In particular, incandescent lights and Cadmium Sulfide (CdS) photoresistor sensors were installed inside a shoe box sealed from outside light sources (Figure 3). Algorithms were implemented on a single dSPACE RTI1104 DSP, and so distributed controls were simulated on a single embedded controller using stochastic delays and scheduling. The controller received analog inputs from the sensors and responded with analog control signals fed to a series of Darlington-style drivers providing voltage control of the incandescent bulbs. In the experiments here, only sensors 3 and 7 shown in Figures 1 and 3(b) are used; inputs from the six other sensors are ignored. So in the following, there are  $n = 8$  lights and  $m = 2$  sensors. The cost function used was the 2-norm of the resource allocation, which is a proximate for the total instantaneous power as described in Section II-C1. That is, the cost function  $F(\vec{x}) \triangleq \|\vec{x}\|_2^2 = \sum_{i=1}^n x_i^2$ .

In practice, conventional multiple-access communication techniques can be used so that each sensor can estimate the illumination contribution of nearby lights continuously during normal operation [e.g., 19]. For simplicity, a separate commissioning process is used here during the first sixteen seconds of each run. That is, over two seconds per each of the eight lights, the light assumes two discrete output levels while all other lights are off. Sensors then fit an affine influence model to each light. After this commissioning process, there is no explicit coordination between lights and no information provided to lights or sensors about the relative location of other agents.

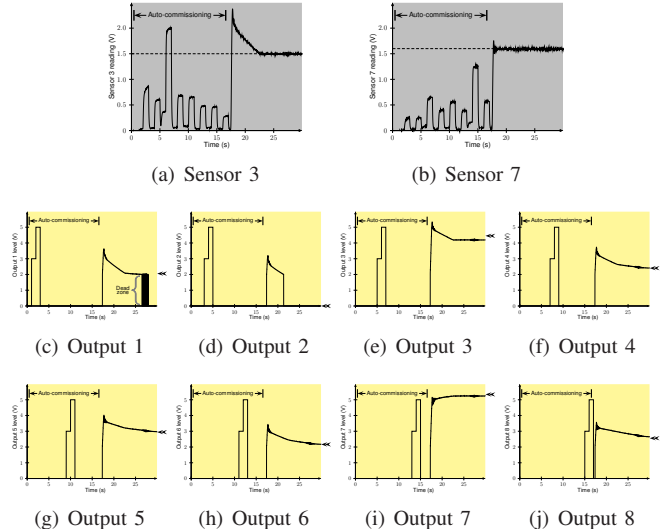


Figure 4. Results for distributed algorithm. For sensor readings shown in (a) and (b), the corresponding constraint level is shown with a dashed line. In the first sixteen seconds of the experiment, an auto-commissioning process fits a linear model of the influence of the lights on the sensors. Immediately after commissioning, the power-minimization algorithm starts. Based on the fitted model, the predicted optimal allocations are shown with double arrowheads on each of (c)–(j). In (c), the actual allocation is truncated to zero by an unmodeled dead-zone non-linearity added to disable the actuator outside of its normal operating range.

#### A. Results

Results for a sample distributed power-minimization experiment are shown in Figure 4. The double arrowheads adjacent to each of Figures 4(c)–4(j) indicate the optimal open-loop output allocation predicted from a static optimization based on the linear model fitted from the commissioning process. To mitigate overshoot effects at the end of the commissioning process, the step size  $\sigma_j$  for each sensor  $j \in \{3, 7\}$  is less aggressive than the example in equation (10). The decay parameter  $d_i$  for each light  $i \in \{1, \dots, 8\}$  could also be adjusted to tune the transient behavior of the algorithm. The chattering across the 2 V output level seen in Figure 4(c) is due to an artificial dead zone introduced on each actuator to disable the incandescent light at the bottom of its linear region of operation when it ceases to produce visible light.

For comparison, the results of a centralized dual-space optimization procedure is shown in Figure 5. This centralized procedure continuously re-calculates its allocation based on new sensor information. As before, the double arrowheads in each of Figures 5(c)–5(j) indicate the optimal open-loop allocation predicted from a static optimization based on the linear model fitted from the commissioning process. Because the centralized optimization procedure may not converge in real time, intermediate allocation estimates are used at each real-time clock event based on the current Lagrange multiplier estimates. Moreover, the dual-space algorithm always uses the most recently available sensor information.



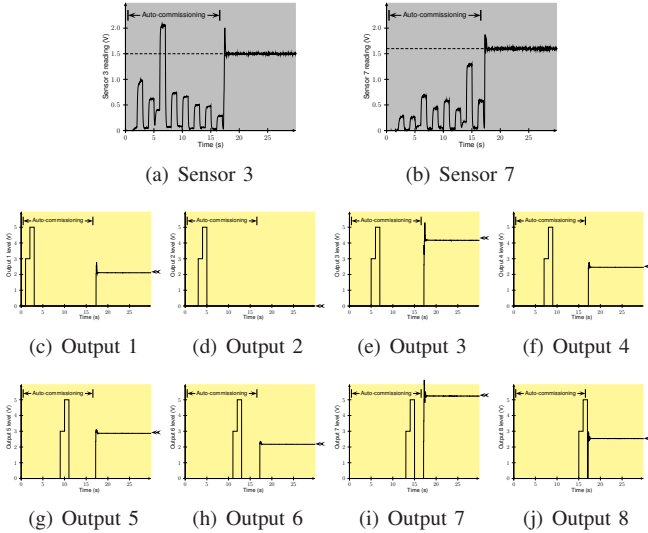


Figure 5. Benchmark results from centralized algorithm. For sensor readings shown in (a) and (b), the corresponding constraint level is shown with a dashed line. In the first sixteen seconds of the experiment, an auto-commissioning process fits a linear model of the influence of the lights on the sensors. Immediately after commissioning, the power-minimization algorithm starts. Based on the fitted model, the predicted optimal allocations are shown with double arrowheads on each of (c)–(j).

Otherwise, the algorithm is prone to significantly more chattering due to the relatively slow sampling time that would be induced by waiting for convergence.

### B. Discussion

The equilibrium behavior of the decentralized and centralized allocations match. That is, the distributed algorithm with no coordination between agents reaches the same constrained optimum as the centralized algorithm. The decentralized allocation in Figure 4(e) is noticeably lower than the predicted allocation from static optimization; however, the centralized allocation in Figure 5(e) is also less than predicted by the static model. Thus, the noticeable variation in Figure 4(e) is due to model error as opposed to an error in the distributed algorithm. Moreover, if a sensor is added or removed from the system, the centralized algorithm would have to be re-configured and re-started. However, the distributed algorithm will automatically adapt to the change.

## V. CONCLUSIONS

A decentralized algorithm for solving constrained non-linear optimization problems with linear constraints has been outlined. This algorithm is particularly tailored for physical resource allocation problems. It uses heterogeneous agents in two classes: one class that continuously decreases resource allocations in the direction of decreasing cost, and another class that reacts when an operational constraint is violated. In these cases, the physical resource itself (e.g., light, temperature) is used as a stigmergic shared memory implicitly communicating information between constraints.

Thus, rather than communicating directly along explicit networks, the agents coordinate through the use of a shared physical medium in which they all reside.

In applications like intelligent lighting, additional reductions in power use may be possible with predictive models of disturbances (e.g., seasonal patterns, cloud motion, occupancy changes) and demand changes. The algorithms presented in this paper are not meant to replace conventional methods like model predictive control (MPC) for this task. Instead, a hierarchal control scheme may be used where control methods like MPC adjust the schedule of how set points change over relatively slow time scales while the algorithms presented here can regulate to those set points quickly using minimal power and low communication bandwidth. Moreover, the physical stigmergy used in these algorithms is uniquely amenable to the physical isolation induced by visible light communication. So there is an opportunity for conventional and stigmergic distributed resource allocation to complement each other.

## ACKNOWLEDGMENTS

Thanks to Kevin M. Passino for his helpful comments and laboratory support. Swanand Phadke assembled much of the intelligent-lighting testbed. Ideas for some of the example engineering applications were conceived during discussions with Jin Wang, Roger Buelow, Keith Kazenski, and Laszlo Takacs. Thanks to the Mathematical Biosciences Institute which facilitated discussions with Audrey Dussutour and Stephen Simpson about the overlap between this work and nutrient regulation. Under the supervision of Stephen C. Pratt, some of the eusocial-insect nutrition predictions of this work are currently being empirically tested in eusocial Hymenoptera. Aspects of these experiments were influenced by discussions with Ian M. Hamilton, Arianne J. Cease, and Rebecca M. Clark. Thanks to Thomas A. Waite for introducing me to the interesting problems in behavioral ecology many years ago. Finally, thanks to three anonymous referees for helpful comments on this manuscript.

## REFERENCES

- [1] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Belmont, Massachusetts: Athena Scientific, 1997.
- [2] D. Blatt, A. O. Hero, and H. Gauchman, "A convergent incremental gradient method with a constant step size," *SIAM J. Optim.*, vol. 18, no. 1, pp. 29–81, January 1, 2007.
- [3] E. Compongara, D. Jia, B. H. Krogh, and S. Talukdar, "Distributed model predictive control," *IEEE Control. Syst. Mag.*, vol. 22, no. 1, pp. 44–52, February 2002.
- [4] B. Johansson, M. Rabi, and M. Johansson, "A simple peer-to-peer algorithm for distributed optimization in sensor networks," in *Proceedings of the 46th IEEE Conference on Decision and Control*, New Orleans, LA, December 12–14, 2007, pp. 4705–4710.

- [5] J. Li, E. Elhamifar, I.-J. Wang, and R. Vidal, "Consensus with robustness to outliers via distributed optimization," in *Proceedings of the 49th IEEE Conference on Decision and Control*, Atlanta, GA, December 15–17, 2010, pp. 2111–2117.
- [6] A. Nedić and A. Ozdaglar, "Constrained consensus and optimization in multi-agent networks," *J. Optim. Theory Appl.*, vol. 142, no. 1, pp. 205–228, April 2009.
- [7] —, "Distributed subgradient methods for multi-agent optimization," *IEEE Trans. Autom. Control.*, vol. 54, no. 1, pp. 48–61, January 2009.
- [8] A. Nedić, A. Ozdaglar, and P. A. Parrilo, "Constrained consensus and optimization in multi-agent networks," *IEEE Trans. Autom. Control.*, vol. 55, no. 4, pp. 922–938, February 2, 2010.
- [9] D. Caicedo and A. Pandharipande, "Distributed illumination control with local sensing and actuation in networked lighting systems," *IEEE Sensors J.*, vol. 13, no. 3, pp. 1092–1104, March 2013.
- [10] C. R. Kube and E. Bonabeau, "Cooperative transport by ants and robots," *Robot. Auton. Syst.*, vol. 30, no. 1–2, pp. 85–101, January 2000.
- [11] P.-P. Grassé, "La reconstruction du nid et les coordinations interindividuelles chez *Bellicositermes natalensis* et *Cubitermes sp.* la théorie de la stigmergie: essai d'interprétation du comportement des termites constructeurs," *Insectes Sociaux*, vol. 6, no. 1, pp. 41–80, 1959.
- [12] D. P. Bertsekas, *Nonlinear Programming*. Belmont, Massachusetts: Athena Scientific, 1995.
- [13] M. Ashibe, M. Miki, and T. Hiroyasu, "Distributed optimization algorithm for lighting color control using chroma sensors," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, 2008. SMC 2008*, Singapore, 2008.
- [14] J. Granderson, Y.-J. Wen, A. M. Agogino, and K. Goebel, "Towards demand-responsive intelligent lighting with wireless sensing and actuation," in *Proceedings of the IESNA (Illuminating Engineering Society of North America) 2004 Annual Conference*, Tampa, FL, 2004, pp. 265–274.
- [15] M. Miki, E. Asayama, and T. Hiroyasu, "Intelligent lighting system using visible-light communication technology," in *Proceedings of the 2006 IEEE Conference on Cybernetics and Intelligent Systems*, 2006, pp. 1–6.
- [16] M.-S. Pan, L.-W. Yeh, Y.-A. Chen, Y.-H. Lin, and Y.-C. Tseng, "A WSN-based intelligent light control system considering user activities and profiles," *IEEE Sensors J.*, vol. 8, no. 10, pp. 1710–1721, October 2008.
- [17] Y.-J. Wen and A. M. Agogino, "Wireless networked lighting systems for optimizing energy savings and user satisfaction," in *Proceedings of Wireless Hive Networks Conference*, Austin, TX, USA, 2008.
- [18] M. T. Koroglu and K. M. Passino, "Illumination balancing algorithm for smart lights," *IEEE Trans. Control. Syst. Technol.*, vol. 22, no. 2, pp. 557–567, March 2014.
- [19] J.-P. M. G. Linnartz, L. Feri, H. Yang, S. B. Colak, and T. C. W. Schenk, "Communications and sensing of illumination contributions in a power LED lighting system," in *Proceedings of the 2008 IEEE International Conference on Communications (ICC 2008)*, Beijing, China, May 19–23, 2008, pp. 5396–5400.
- [20] A. R. Bergen and V. Vittal, *Power Systems Analysis*, 2nd ed. Upper Saddle River, NJ: Prentice Hall, 2000.
- [21] S. D. Fretwell and H. L. Lucas, Jr., "On territorial behavior and other factors influencing habitat distribution in birds: I. Theoretical development," *Acta Biotheor.*, vol. 19, no. 1, pp. 16–36, March 1969.
- [22] D. W. Stephens, J. S. Brown, and R. C. Ydenberg, Eds., *Foraging: Behavior and Ecology*. Chicago, IL: University of Chicago Press, 2007.
- [23] J. Finke and K. M. Passino, "Stable cooperative vehicle distributions for surveillance," *J. Dyn. Syst., Meas., Control.*, vol. 129, no. 5, pp. 597–608, 2007.
- [24] B. J. Moore, J. Finke, and K. M. Passino, "Optimal allocation of heterogeneous resources in cooperative control scenarios," *Automatica*, vol. 45, no. 3, pp. 711–715, March 2009.
- [25] A. Pantoja, N. Quijano, and S. Leirens, "A bioinspired approach for a multizone temperature control system," *Bioinspiration & Biomimetics*, vol. 6, 2011.
- [26] N. Quijano and K. M. Passino, "The ideal free distribution: theory and engineering application," *IEEE Trans. Syst., Man, Cybern., Part B: Cybern.*, vol. 37, no. 1, pp. 154–165, February 2007.
- [27] —, "Honey bee social foraging algorithms for resource allocation: theory and application," *Eng. Appl. Artif. Intell.*, vol. 23, no. 6, pp. 845–861, September 2010.
- [28] E. Ramirez-Llanos and N. Quijano, "A population dynamics approach for the water distribution problem," *Int. J. Control.*, vol. 83, no. 9, pp. 1947–1964, September 2010.
- [29] A. Dussutour and S. J. Simpson, "Communal nutrition in ants," *Curr. Biol.*, vol. 19, no. 9, pp. 740–744, May 12, 2009.
- [30] D. Tran and Y. K. Tan, "Sensorless illumination control of a networked led-lighting system using feedforward neural network," *IEEE Trans. Ind. Electron.*, vol. 61, no. 4, pp. 2113–2121, April 2014.
- [31] Y. Yamamoto, "A new method for solving a linear programming problem," in *Proceedings of the 49th IEEE Conference on Decision and Control*, Atlanta, GA, December 15–17, 2010, pp. 3890–3895.
- [32] K. D. Young, V. I. Utkin, and Ü. Özgüner, "A control engineer's guide to sliding mode control," *IEEE Trans. Control. Syst. Technol.*, vol. 7, no. 3, pp. 328–342, May 1999.