# `ConnectedHearts`: A Distributed System for Viewing the Human Heartbeat

Serena Booth[a,1], Michelle Cone[b,1]

[a] *sbooth@college.harvard.edu*
[b] *mcone@college.harvard.edu*

**Abstract**

We present a physical distributed system for viewing the human heartbeat. We modified a medicine cabinet by embedding 13 light bulbs around the frame, as well as by replacing the cabinet's glass with one-way mirror. When a person stands in front of the mirror, a webcam hidden behind the mirror measures their pulse. The 13 bulbs, each a virtual machine, run leader election via Bully. The leader starts pulsating with the captured pulse, and then instructs neighboring virtual machines to pulsate as well. Finally, we ensure the synchronization of the bulbs through a distributed gossip algorithm.

## 1. Introduction

[SERENA: TO DO]

[1] We, Serena Booth and Michelle Cone, affirm our awareness of the standards of the Harvard College Honor Code.

### *1.1. Artistic Inspiration*

### *1.2. As a Distributed System*

## 2. Physical Creation

## 3. System Design

### *3.1. Architecture*

### *3.2. Bully: Leader Election*

### *3.3. Inter-bulb Communication*

### *3.4. Realtime Synchronization via Gossip*

#### *3.4.1. Layout*

As the layout of our system is predetermined, we are able to adjust the gossip algorithm for synchronization in order to ascribe neighboring processes with scores of trustworthiness. As the leader bulb initiates the heartbeat pulsation, the leader is the single most trustworthy bulb. This has implications for all other bulbs, however: as bulbs increase in distance from the leader, a known measure based on the system's fixed layout, their trustworthiness decreases. Thus when a bulb is receiving contradictory gossip from its two neighboring bulbs, it prioritizes the message which was sent by the bulb closer to the leader.

## 4. Synchronization Testing & Analysis

## 5. Conclusion

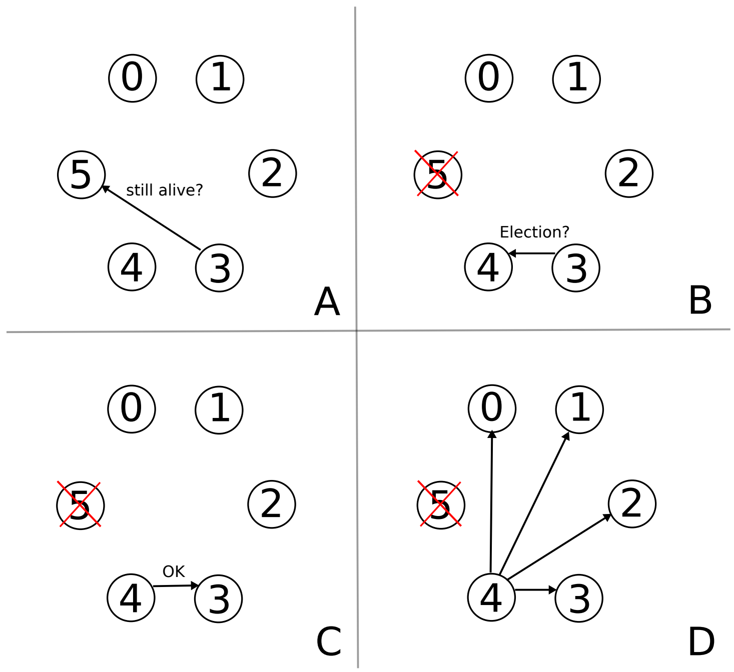


Figure 1: In the above graphic, we demonstrate the bully algorithm for determining a leader. In subimage A, node 3 attempts to confirm that the known leader, node 5, is still alive. In sumbimage B, node 5 does not respond to node 3 for a set period of time—a time set by node 3. Hence node 3 initiates an election by contacting all nodes with a higher id than itself. In subimage C, node 4 responds to node 3's request to initiate an election by confirming it remains alive. Finally, in subimage D, node 4 broadcasts a message to all nodes to inform them of its leader status.

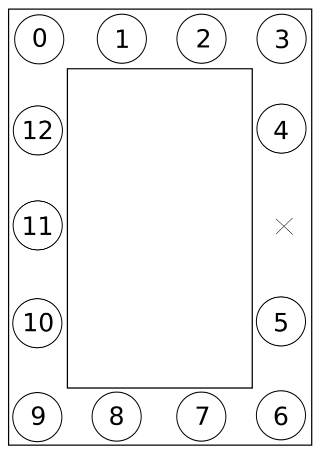


Figure 2: The system layout. While each bulb is assigned a 64-bit UUID during the program, each bulb also has an id, shown above, ranging from $0 - 12$, which allows the system to consistently converge toward the leader bulb's rate of pulsation during the synchronization routine.

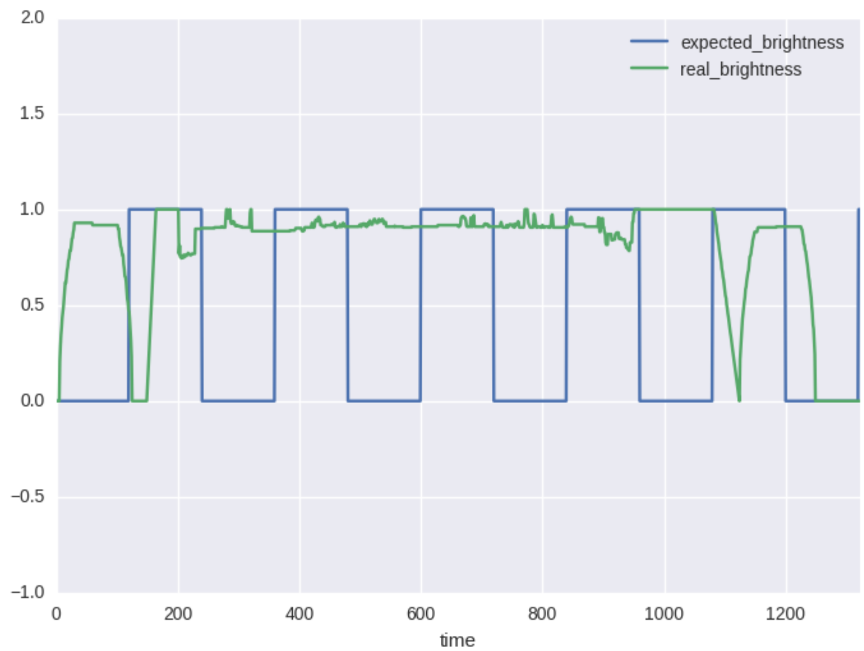


Figure 3: We demonstrate the synchronization of our system as a function of time, as compared with an ideal synchronization system. The green timeseries corresponds to the physical system while the blue timeseries corresponds to an idealized system.