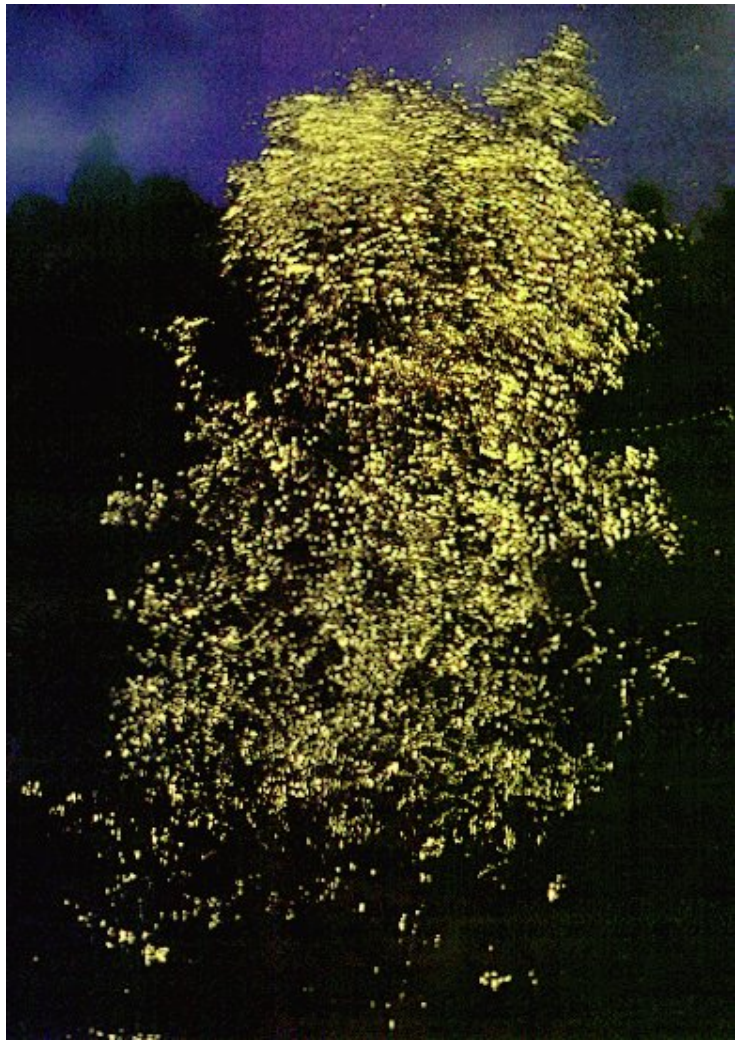
Swarm Intelligence Course—INFO-H-414

Carlo Pinciroli and Manuele Brambilla

IRIDIA — Université Libre de Bruxelles, Belgium

April 25, 2013

Synchronization



Two video examples:

- `http://youtu.be/sROKYelaWbo`
- `http://youtu.be/a-Vy7NZTGos`

Synchronization: Idea

AIM: All the robots are synchronized, that is, lighting up at the same time.

Synchronization: Idea

AIM: All the robots are synchronized, that is, lighting up at the same time.

HOW?

Synchronization: Idea

AIM: All the robots are synchronized, that is, lighting up at the same time.

HOW? **(propose something!)**

Synchronization: Basics

METHOD: each robot emits a signal every T_{max} steps

Synchronization: Basics

METHOD: each robot emits a signal every T_{max} steps

T_{max} is the same for all the robots $\rightarrow$ **same frequency**

Synchronization: Basics

METHOD: each robot emits a signal every T_{max} steps

T_{max} is the same for all the robots $\rightarrow$ **same frequency**

Problem: the robots start at a different moment $\rightarrow$ **different period**

Synchronization: Basics

METHOD: each robot emits a signal every T_{max} steps

T_{max} is the same for all the robots $\rightarrow$ **same frequency**

Problem: the robots start at a different moment $\rightarrow$ **different period**

To synchronize the robots change their signaling period when they receive a signal from a neighbour

Synchronization: Basics

METHOD: each robot emits a signal every T_{max} steps

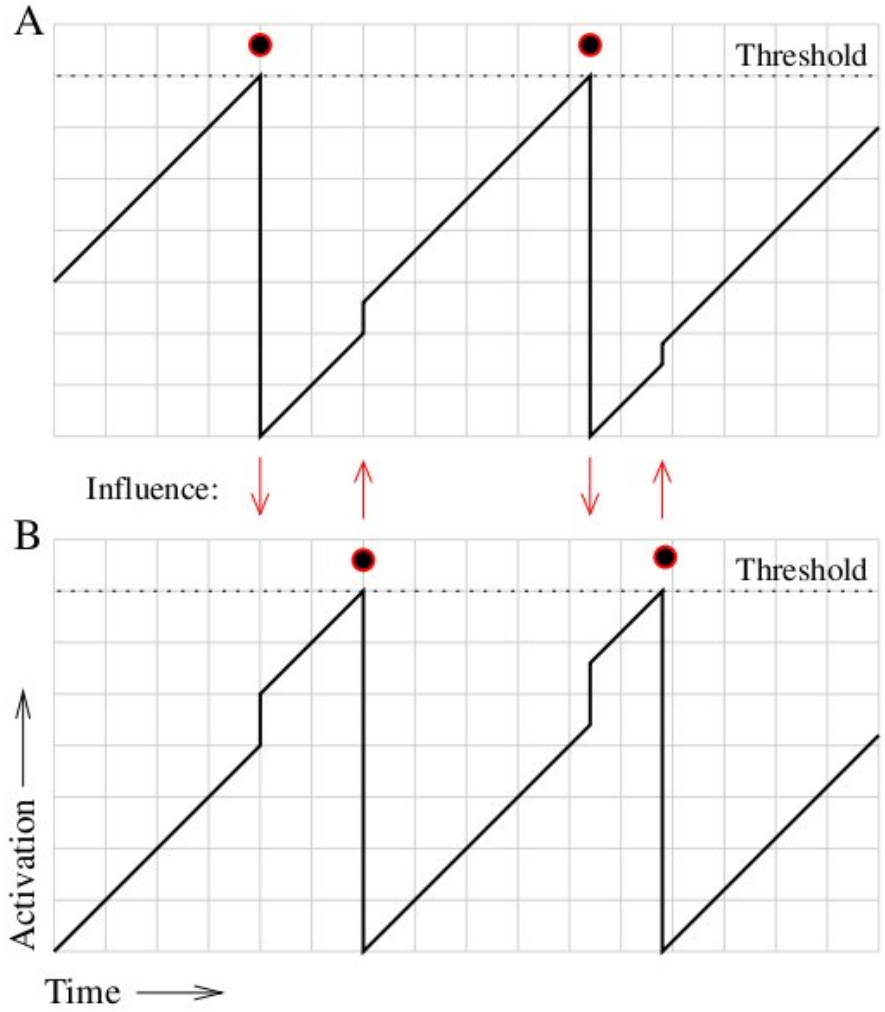
T_{max} is the same for all the robots $\rightarrow$ **same frequency**

Problem: the robots start at a different moment $\rightarrow$ **different period**

To synchronize the robots change their signaling period when they receive a signal from a neighbour

After some time all the robots have the same period $\rightarrow$ **synchronization!**

Synchronization: In Practice



Aggregation: Implementation

```
increase counter

# signaling part
if counter > t_max then
   signal[range_and_bearing and leds]
end
```

Aggregation: Implementation

```
increase counter

# signaling part
if counter > T_max then
   signal
end

#synchronization
if message_received then
  counter = counter + alpha * counter/beta
end
```

Aggregation: Notes

Suggested values:
$T_{max} = 100$
$alpha = 2$
$beta = 10$

Other notes:

- remember to init the counter to a random value ($< T_{max}$)
- signal with both the range and bearing (for the robots) and the LEDs (for you to see)
- it might be necessary to have an ignore phase after the signal
- there is always a one time-step delay when transmitting

Aggregation: Extra

- What happens if the robots are moving? Is it better? Worse? Why?
- What if the robots have a probability P of missing a message? Does it work? In other words, is this behavior robust to noise?