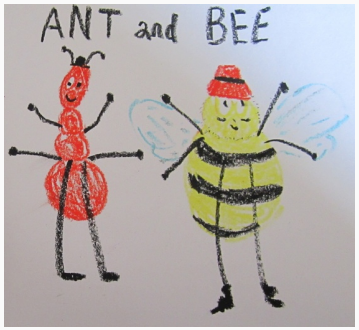



# BIO-INSPIRED AI TECHNIQUES

...and their applications to games

Matthew Bedder

# CONTENTS

I'm going to talk about the **Artificial Bee Colony (ABC) Algorithm** and **Ant Colony Optimisation (ACO)**.

Both of these are **bio-inspired**, **multi-agent**, **optimisation** techniques that currently aren't widely applied to games.

(Or widely applied to any other really complex problems)

# BUZZWORDS

(Pun not intended)

# BUZZWORDS

**Bio-inspired** : *"the use of computers to model the living phenomena, and simultaneously the study of life to improve the usage of computers"* [ WIKIPEDIA ]

**Multi-agent** : *"computerised systems composed of multiple interacting intelligent agents within an environment"* [ WIKIPEDIA ]

**Optimisation technique** : *"a process in which a software system is modified in order to meet some constraints while maximising some value function or modifying some cost function"* [ Me ]

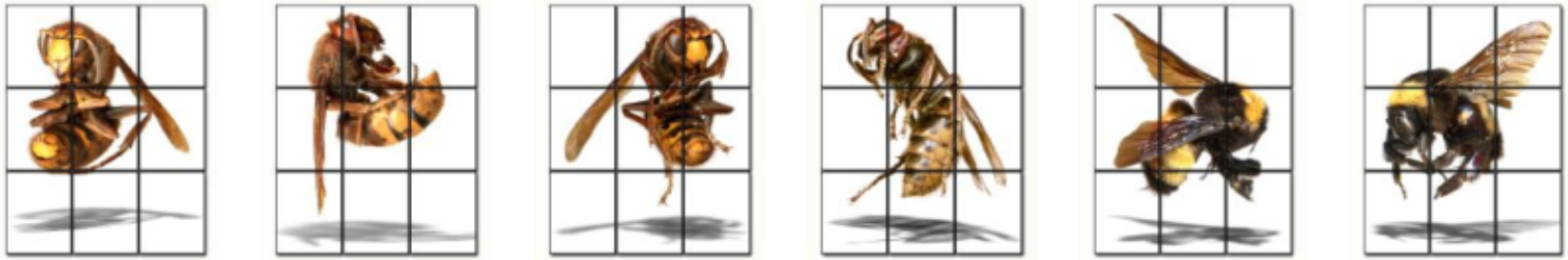
# THE ARTIFICIAL BEE COLONY (ABC) ALGORITHM

From “An Idea Based on Honey Bee Swarm for Numerical Optimization” by Karaboga

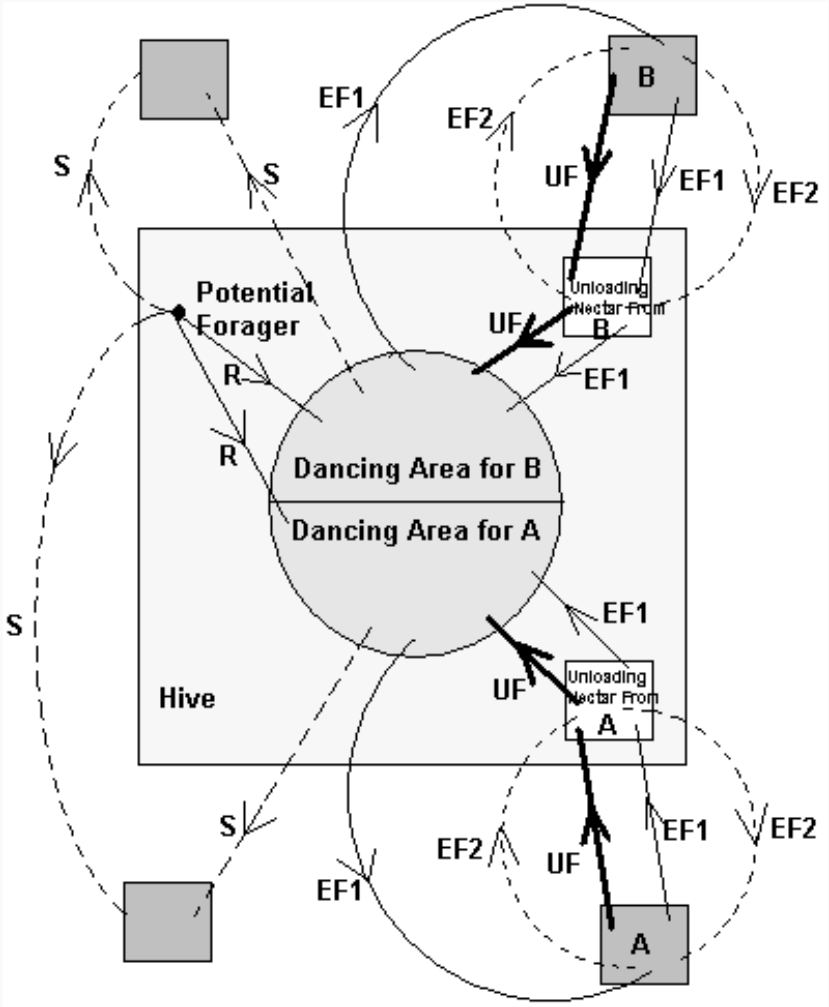
# ARTIFICIAL BEE COLONY - KEY IDEAS

Based on the roles bees take in real bee colonies

- "Employed foragers"
  - Know a lot about a food source (distance and direction from the nest, profitability)
  - "Exploit" this source until it is exhausted or they find a better one
- "Onlookers"
  - Wait in the nest recruited by a forager (using interpretive dance)
- "Scouts"
  - Search the world for new, profitable food sources



# ARTIFICIAL BEE COLONY - KEY IDEAS



# ARTIFICIAL BEE COLONY - ALGORITHM

1: Initialise $n$ scout bees, $n$ onlooker bees
2: **repeat**
3: **for all** scout bees **do**
4: Move to a location in the search space at random
5: **for all** employed forager bees **do**
6: Assess the value of their location
7: **for all** onlooker bees **do**
8: Move to the neighbourhood of an employed forager with probability proportional to its value
9: Assess the value of the new location
10: **for all** employed forager bees **do**
11: **if** Value of location has not improved for a while **then**
12: Become a scout bee
13: **until** Termination condition is met
14: **return** Best location found to date

## We have a **Positive feedback loop**

Good solutions get more onlookers looking in the neighbourhood, so we are likely to optimise those results.

## We have a **Negative feedback loop**

Poor solutions are less likely to get more onlookers, and are more likely to get abandoned

## We have **fluctuations**

Scouts randomly explore (which is cheap but generally gives poor results) so we have a way to find new, better food sources

# Artificial bee colony - usage (general purpose)

ABC (and variants) have been used in areas such as:

- Training feed-forward neural networks
- Structural optimisation of bridges
- Clustering data
- Image Processing

# Artificial bee colony - usage (general purpose)

ABC (and variants) have been used in areas such as:

- Training feed-forward neural networks[1]
- Structural optimisation of bridges[2]
- Clustering data[3]
- Image Processing[4]

[1] Such as in "Artificial bee colony (ABC) algorithm on training artificial neural networks" by Karaboga & Akay

[2] Such as in "Structural optimization using artificial bee colony algorithm" by Hadidi, SK Azad & SK Azad

[3] "An artificial bee colony approach for clustering" by Zhang, Ouyang, & Ning

[4] Such as in "A novel approach to image edge enhancement using artificial bee colony optimization algorithm for hybridized smoothening filters" by Benala, Jampala, Villa, & Konathala

# Artificial bee colony - usage (games)

Potential applications for games:

- Collection of resources in RTSs
- Simulating bees?
- Selecting moves to perform
  - Utilised in the game Hexxagon, achieves promising performance "Introducing Bee Colony Algorithm for Hexxagon Game" by Tătar & Holban

ABC has not been used much in games so far...

# ANT COLONY OPTIMISATION (ACO)

From “Ant Colony Optimisation: A New Meta-heuristic” by Dorigo

# ANT COLONY OPTIMISATION - KEY IDEAS

An optimisation technique for the efficient exploitation of resources

Ants aren't particularly chatty, so communicate about food sources using pheromones

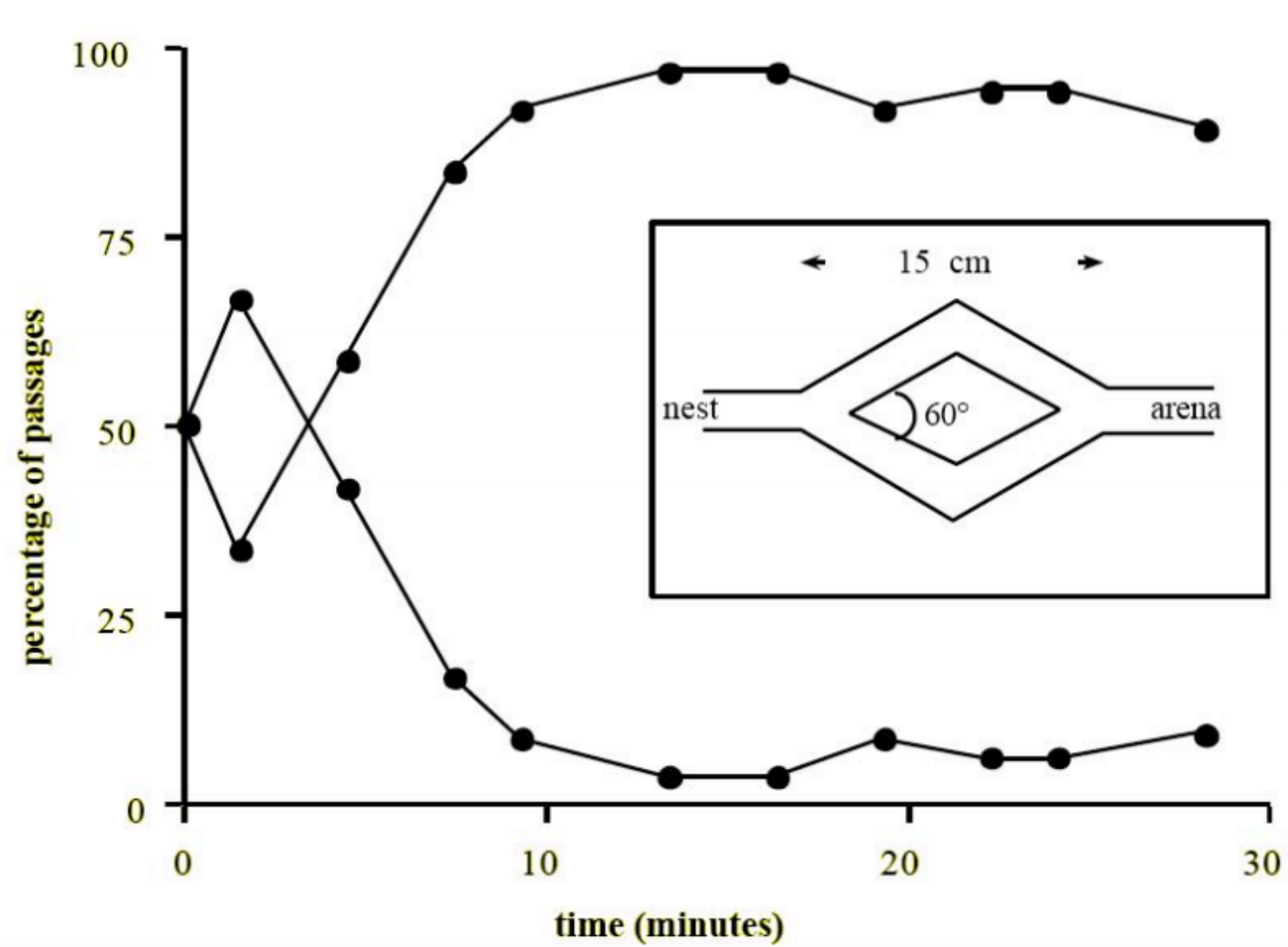
- Foraging ants with no knowledge of food sources move (pseudo-) randomly
- When a foraging ant finds food, it makes its way back to the nest releasing pheromones
- If a foraging ant senses a pheromone trail it will follow it (with the hope of finding food
- Pheromone trails diffuse slightly, and decay over time

# ANT COLONY OPTIMISATION - KEY IDEAS

Provides really simple feedback loops

- A shorter path between food and the nest would get more footfall, so would leave a stronger pheromone trail
- When a food source is used up the pheromone trail doesn't get reinforced, so dies out

# ANT COLONY OPTIMISATION - KEY IDEAS



# ANT COLONY OPTIMISATION - KEY IDEAS

# ANT COLONY OPTIMISATION - ALGORITHM

(Really high-level pseudocode!)

```
initialise ant locations
repeat
  update pheromones
  for all ants do
    select which direction to walk
    if returning to the nest with food then
      release pheromone
until you get bored
```

If you'd like to read a lot about implementation, I'd advise looking at "Ant System: Optimising by a Colony of Cooperating Agents" by Dorigo, Maniezzo, & Colorni

# Ant colony optimisation - usage (general purpose)

ACO (and variants) have been used in areas such as:

- Generating good solutions to the Travelling Salesman Problem
  - For release $n$ ants on the problem travelling according to some heuristic on arc length and pheromone strength until they complete a tour
  - Reinforce the shortest tour found by any ant with extra pheromone (proportionally to the inverse of the tour length)
  - Repeat!
- Scheduling
- Routing
- Edge detection in images
- ...

# Ant colony optimisation - usage (games)

Potential applications to games:

- Exploitation of resources in games with incomplete knowledge
- Simulating ants (duh)
- General game playing

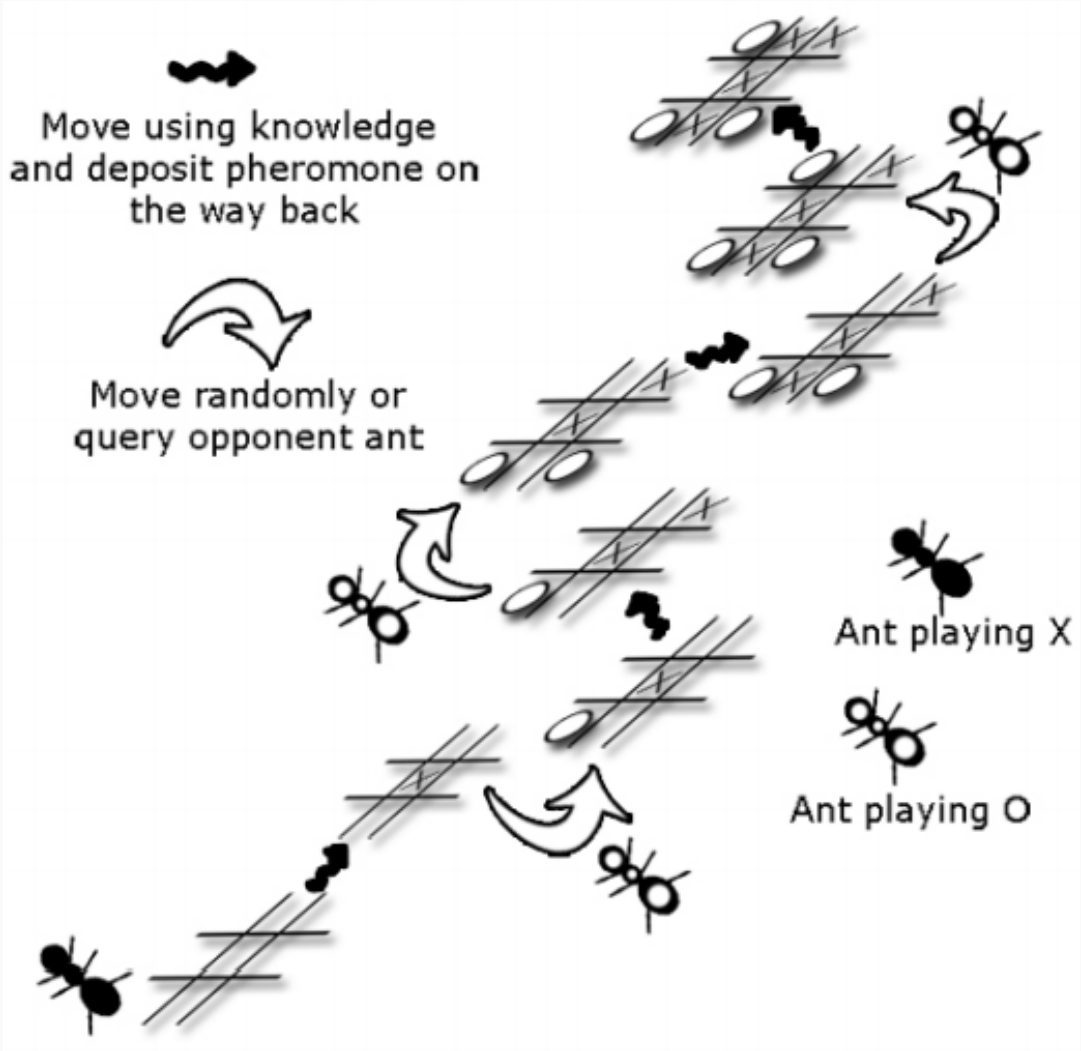
# Ant colony optimisation - usage (games)

ACO for General Game Playing

- Have ants navigate through game trees, where they can only move forward, and move randomly at opponents turns
- OK results exhibited on simple games versus random players, but untested on *realistic* domains

“General Game Playing with Ants” by Sharma, Kobti, & Goodwin

# ANT COLONY OPTIMISATION - USAGE (GAMES)



# CONCLUSIONS

Agent-based bio-inspired algorithms are really cool but aren't currently being used much in games (maybe for a good reason)

Maybe we can convince an applicant for IGGI 2016 to write a proposal for ant-based AI?

# THAT'S ALL, FOLKS!

I'd be more than un happy to answer any sensible questions

(Or we can all start our weekend a bit earlier)