Copy when Uncertain

Bumblebees rely on Social Information when Rewards are highly variable

Copy when uncertain...

Bumble bee A

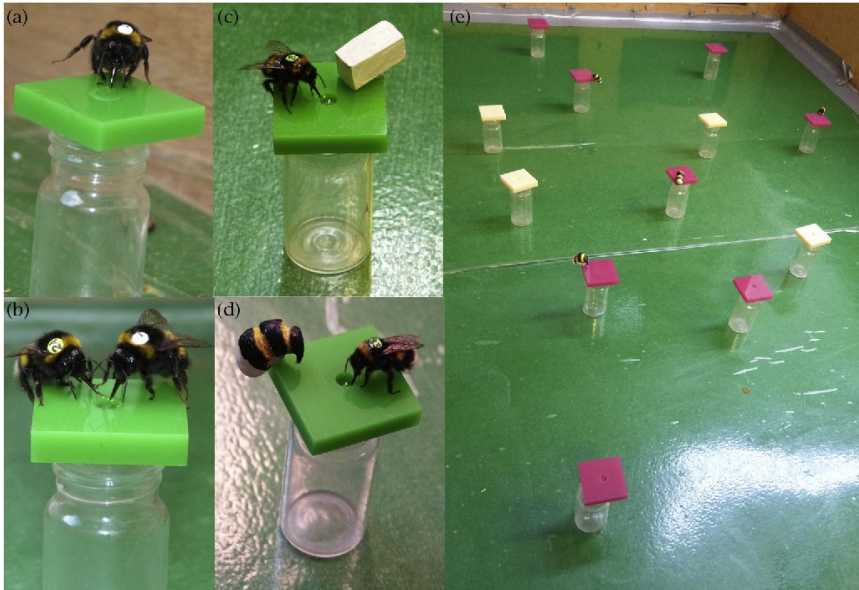
Bumble bee B

Simulating Copy when Uncertain

To understand the relative benefits of social and personal information use in foraging decisions.

I developed an agent-based model of social learning that predicts:

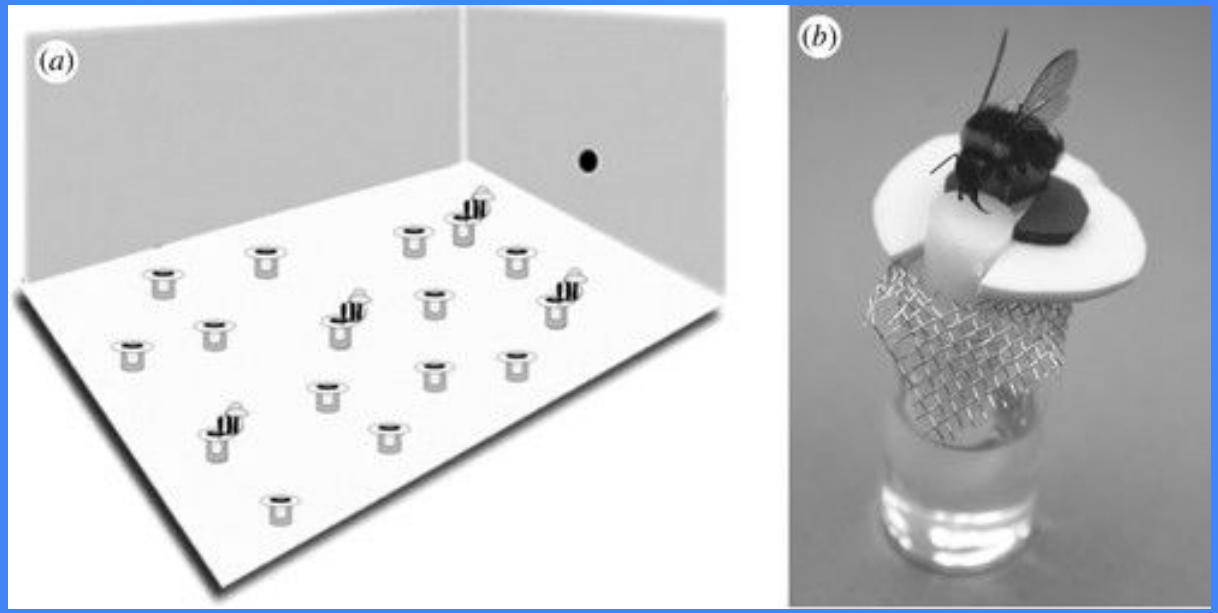
1. Social information should be more adaptive where resources are highly variable and
2. Personal information should be more preferable where resources vary little



To understand the simulation

We need to understand the following:

1. Patches
2. Bees
3. Step by step rundown of the code
4. Results



PATCHES

Patches

We model an environment of **100 flowers** (called patches) that store sugared solution and attract bees. There are **two modes** of environments:

1. No Variance mode:
   Each patch has an equal amount of sugared solution (8.3 in this case)
2. High Variance mode:
   Patches have different amounts of sugared solution. We use a gamma distribution with k = 0.183 and theta = 45.45 (G-index = 0.81)

The patches also change with the **probability** of 0.0001 every timestep.

Therefore each patch is defined by **number of flowers**, **variance mode** and c**hange probability** of patches.

Bumble bees

Bees

We use bumblebees as agents to simulate foraging. We use a population of 33 bees that look for flowers with high resource and collect the sugared solution. There are two types of bees:

1. Social Learners: they learn about the patch resources by observing other bees that are exploiting the patch. (trust others)
2. Individual Learners: they learn about the patch resources by individually observing a single patch. (trust themselves)

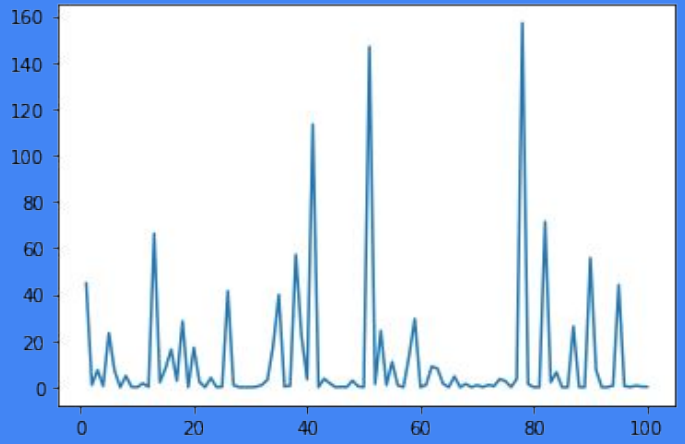
Each bee, for every timestep, can either explore the patches (socially/individually) or exploit the resources based on their exploration (highest anticipated reward).

Bees

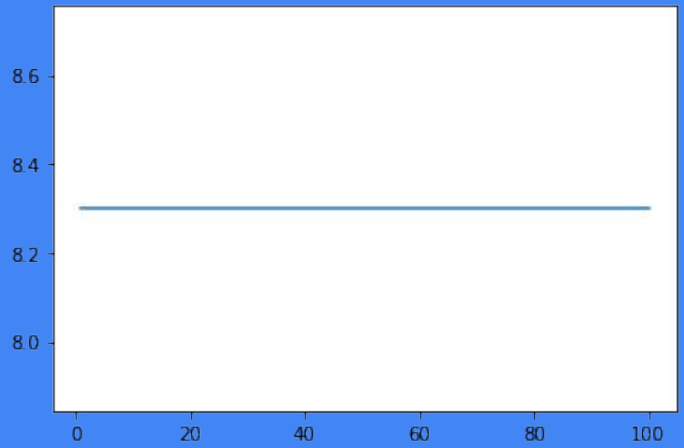
If multiple bees sit on the same patch, they also share the resources thus simulating competition amongst the bees.

Bees update their memory (anticipated reward from each patch) either by exploring based on their type or by exploiting and finding a different amount of reward until they die. Each bee has a 0.02 chance of death for every timestep. They also die after 100 timesteps by default.

Each dead bee is replaced by a newborn bee, thus maintaining a constant population. The type of the newborn bee is decided based on the average fitness of the types which is equal to the average of net collected resources of that type.



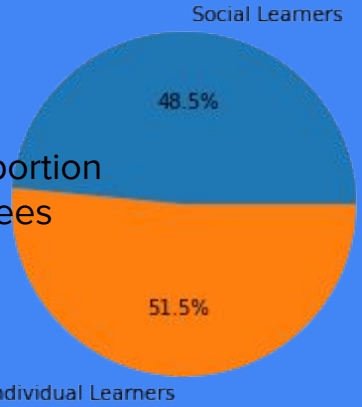
High Variance Distribution



No Variance Distribution

Step by step

Proportion of bees

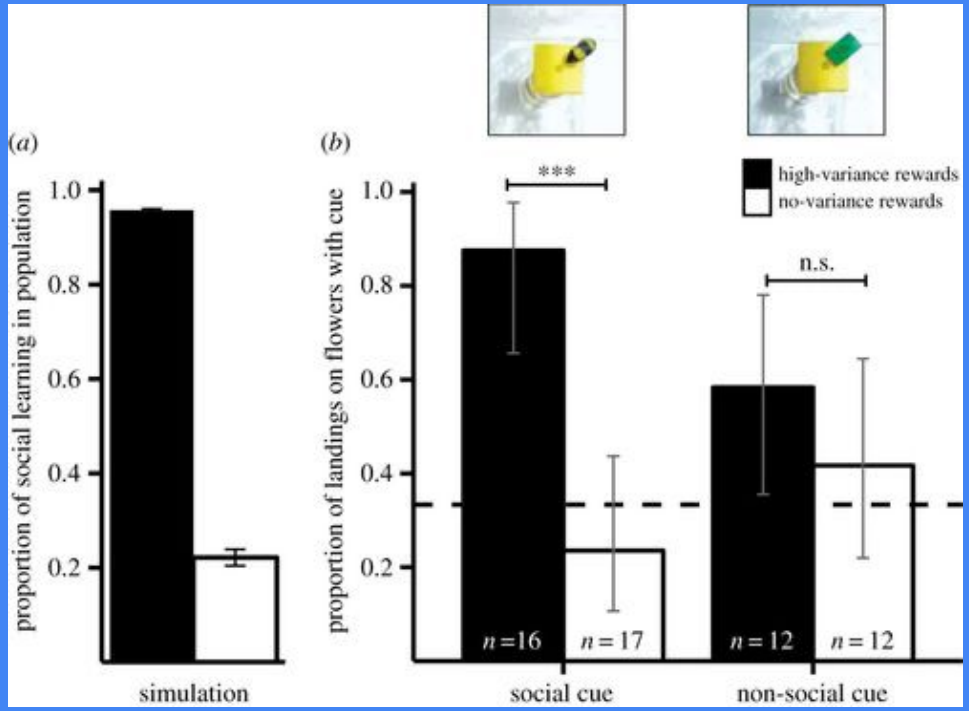


Steps

Here are the different events in the simulation:

1. Initialize patches - high or no variance
2. Initialize bees
3. Run simulation for 1000 time steps. For each step:
    a. For each bee:
        i. Does bee explore (20%) or exploit (80%)?
        ii. Explore based on type / Exploit resources
        iii. Update memory
    b. Calculate average fitness based on resources collected (for Social and Individual bees)
    c. Find Dead bees based on probability (0.02) or timesteps (100)
    d. Create new bees for dead bees, using mean fitness ratio as probability
    e. Check if patches distribution needs to be changed based on probability (0.0001)

Results


(a)
proportion of social learning in population
1.0
0.8
0.6
0.4
0.2
simulation
(b)
proportion of landings on flowers with cue
1.0
0.8
0.6
0.4
0.2
***
n.s.
high-variance rewards
no-variance rewards
n = 16
n = 17
n = 12
n = 12
social cue
non-social cue

Results - High Variance

ITERATION: 0

Net Resources:	659
Collected Resources:	0
Resource Efficiency:	0
Resources Breakdown:	0
-->Social Collected Resources:	0
-->Individual Collected Resources:	0
Bees Population Breakdown:	33

-->Social Bees: 16 -->Individual Bees: 17

Bee proportion --> Social / Individual :
0.9411764705882353

ITERATION: 999

Net Resources:	659
Collected Resources:	446
Resource Efficiency:	67
Resources Breakdown:	446
-->Social Collected Resources:	**242**
-->Individual Collected Resources:	**203**
Bees Population Breakdown:	33

-->Social Bees: 17 -->Individual Bees: 16

Bee proportion --> Social / Individual :
1.0625

Results - No Variance

ITERATION: 0

Net Resources:	829
Collected Resources:	0
Resource Efficiency:	0
Resources Breakdown:	0
-->Social Collected Resources:	0
-->Individual Collected Resources:	0
Bees Population Breakdown:	33

-->Social Bees: 16 -->Individual Bees: 17

Bee proportion --> Social / Individual :
0.9411764705882353

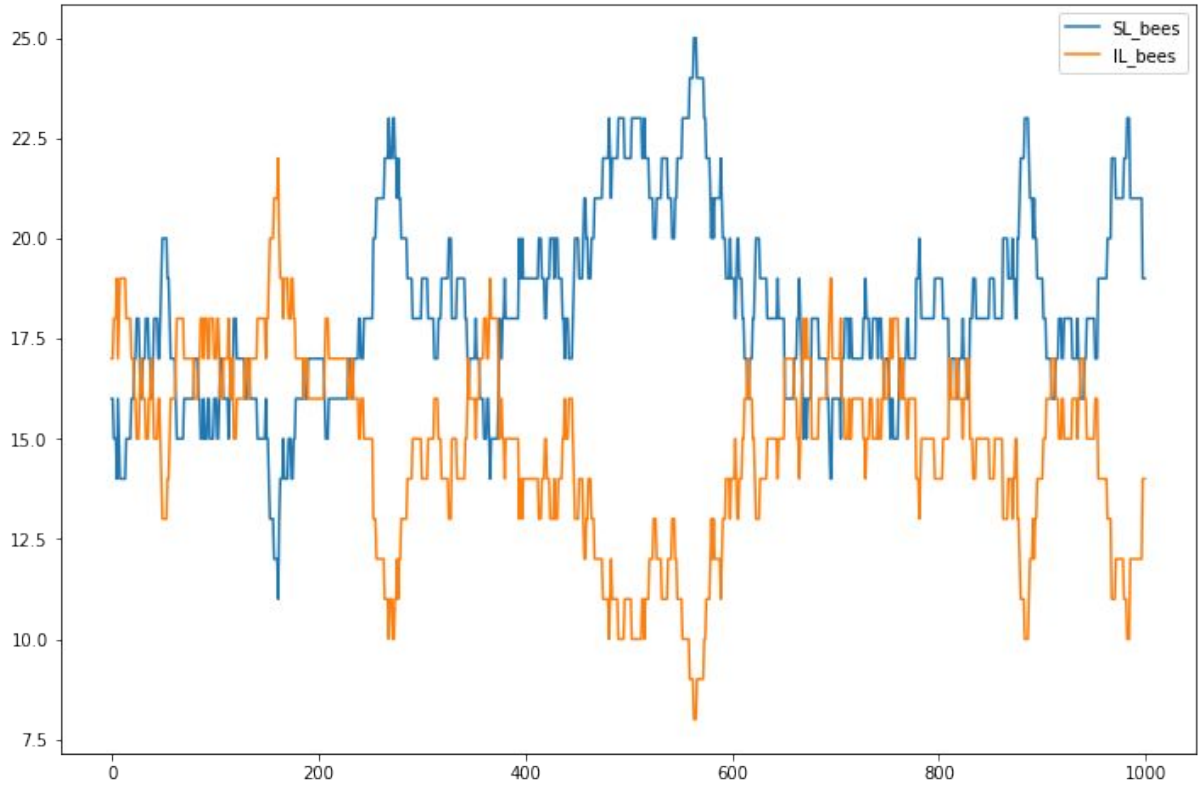
ITERATION: 999

Net Resources:	829
Collected Resources:	190
Resource Efficiency:	23
Resources Breakdown:	190
-->Social Collected Resources:	**74**
-->Individual Collected Resources:	**116**
Bees Population Breakdown:	33

-->Social Bees: **14** -->Individual Bees: **19**

Bee proportion --> Social / Individual :
0.73684210526

Results - High Variance



Population of

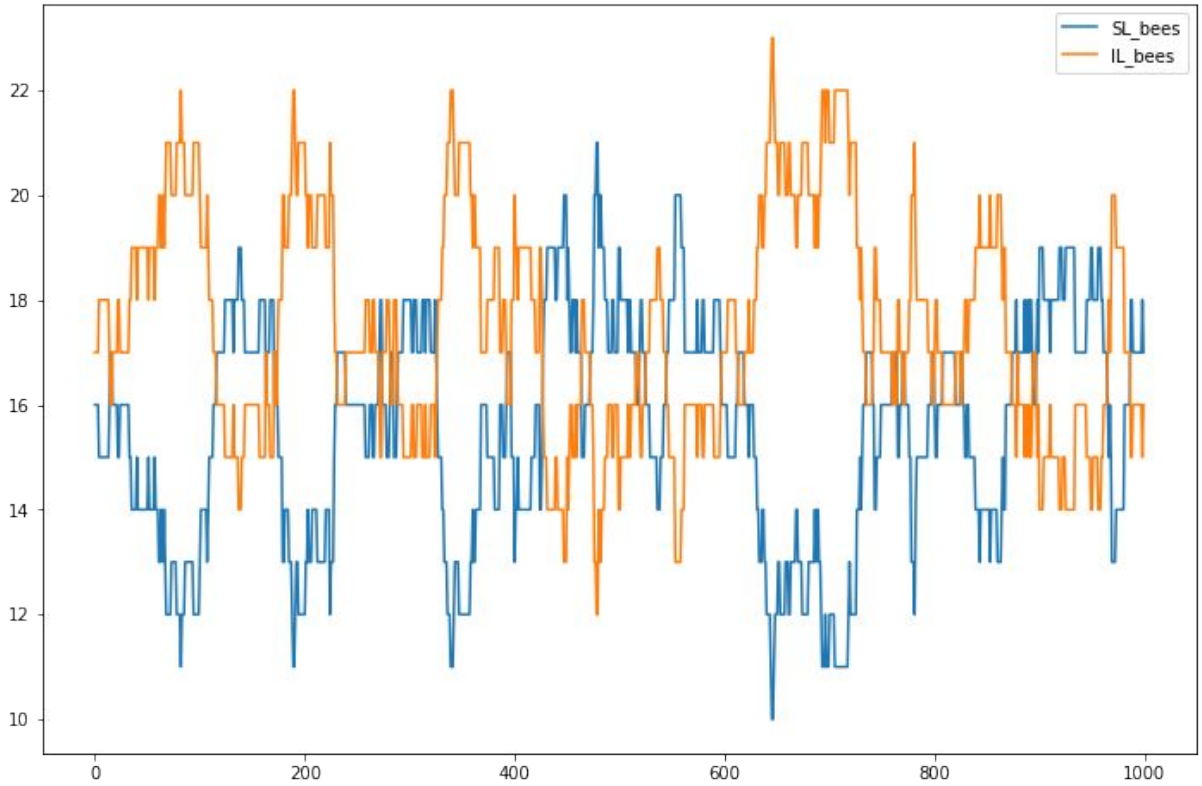
Social bees (BLUE)

vs

Individual bees (ORANGE)

Social bees dominate after initial iterations

Results - No Variance



Population of

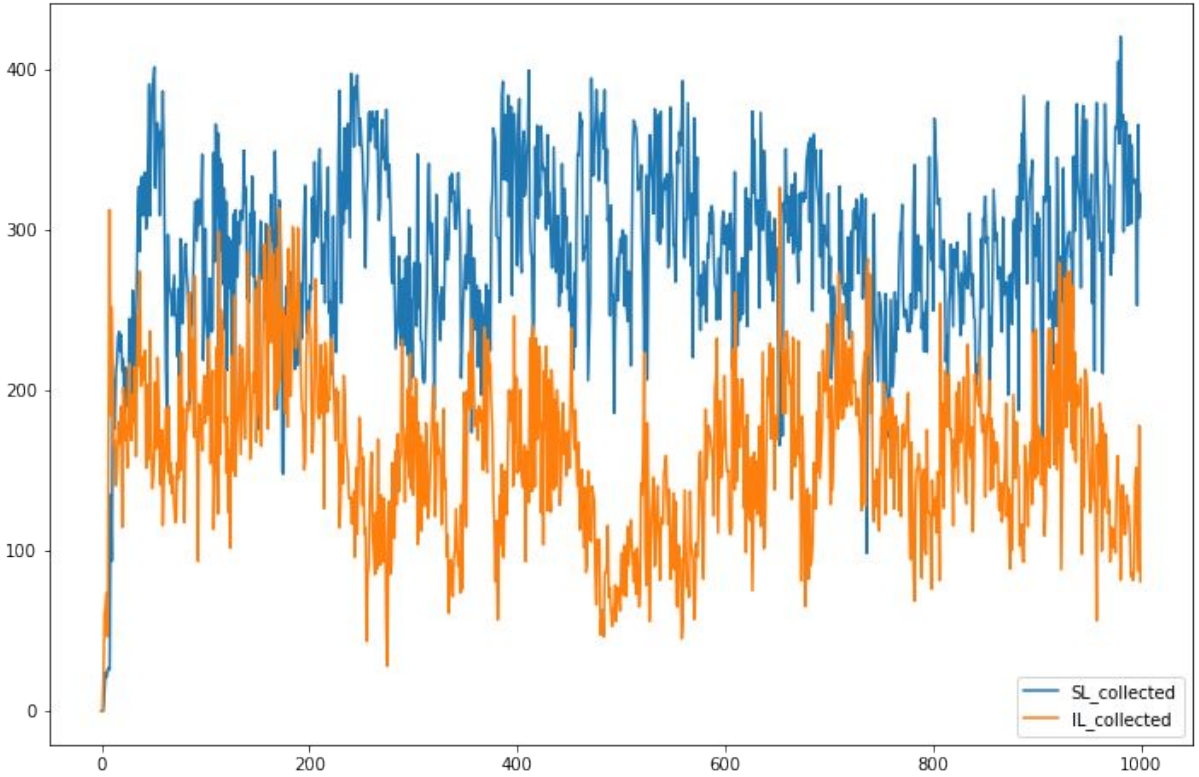
Social bees (BLUE)

vs

Individual bees (ORANGE)

Individual bees dominate from
The beginning

Results - High Variance



Resources collected by

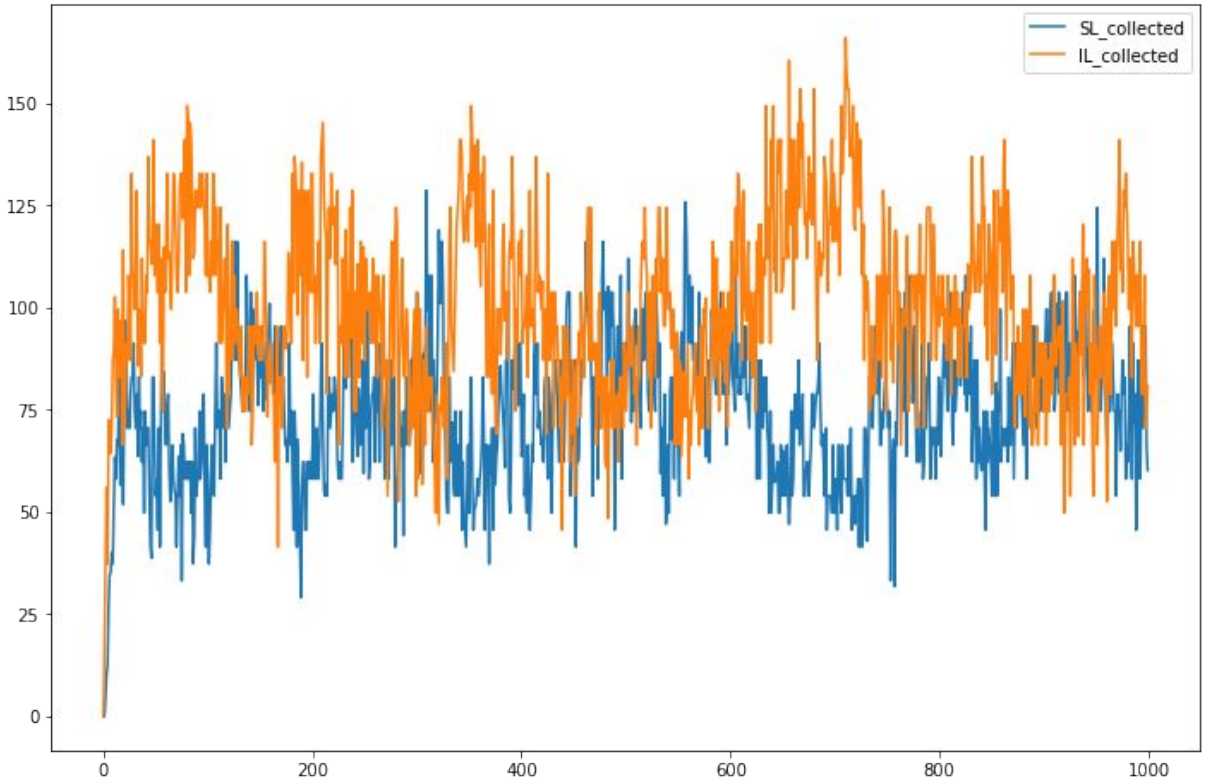
Social bees (BLUE)

vs

Individual bees (ORANGE)

Social bees dominate after initial iterations

Results - No Variance



Resources collected by

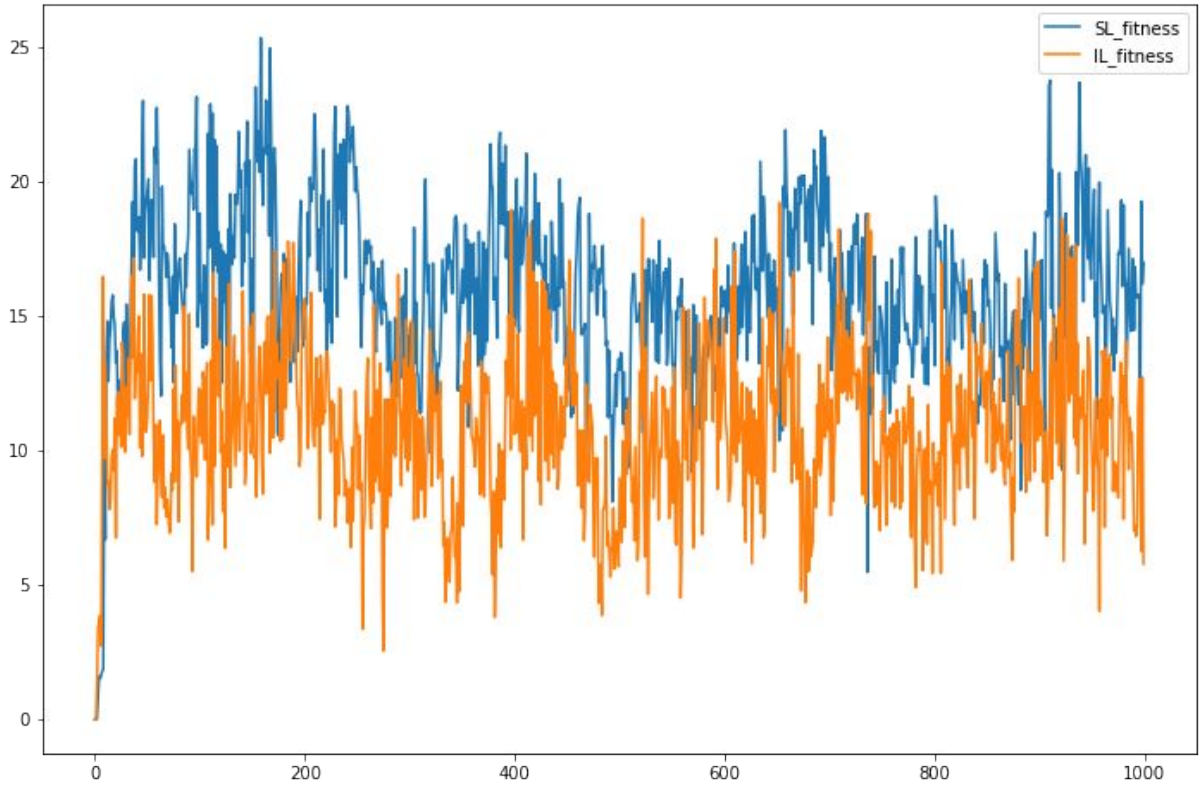
Social bees (BLUE)

vs

Individual bees (ORANGE)

Individual bees dominate from The beginning

Results - High Variance



Mean Fitness of groups:

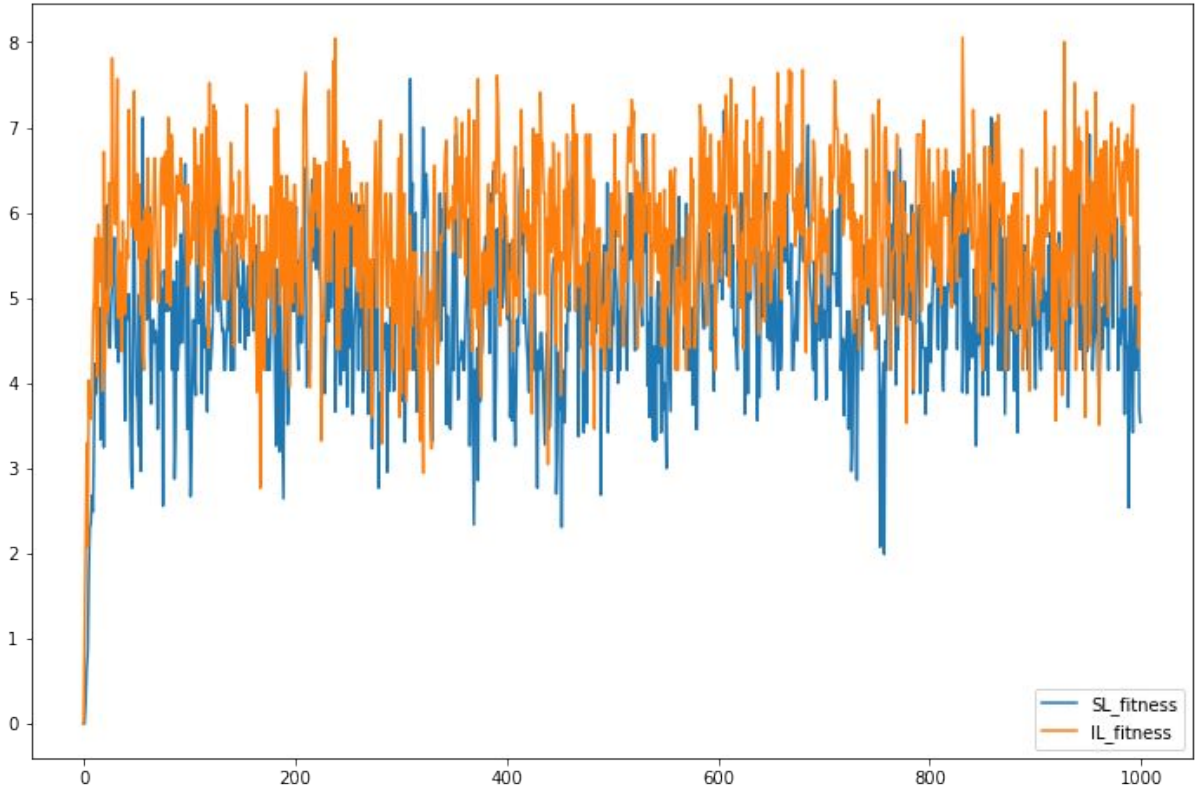
Social bees (BLUE)

vs

Individual bees (ORANGE)

Social bees dominate after initial iterations

Results - No Variance



Mean Fitness of groups:

Social bees (BLUE)

vs

Individual bees (ORANGE)

Individual bees dominate from The beginning