Artificial Bee Colony

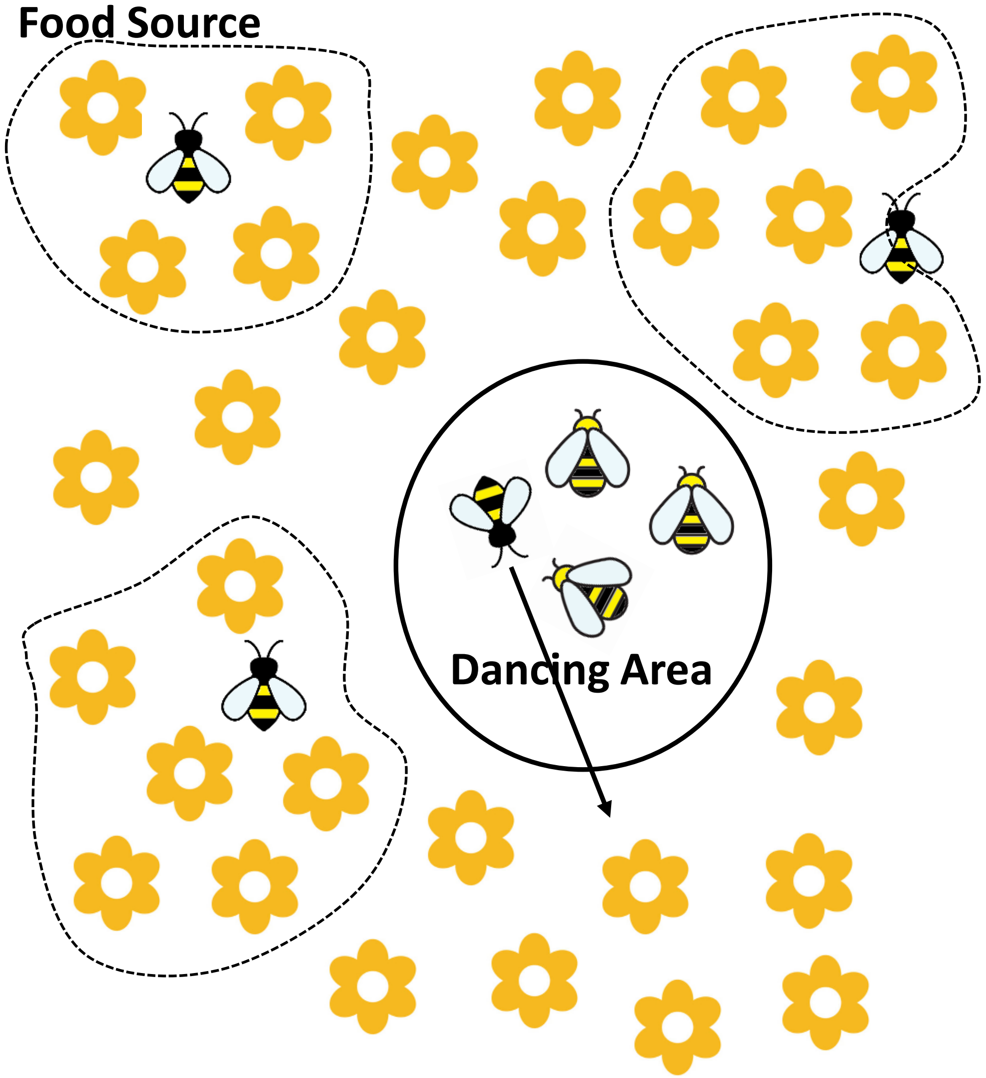
Nuno Antunes Ribeiro

Assistant Professor

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN | Engineering Systems and Design

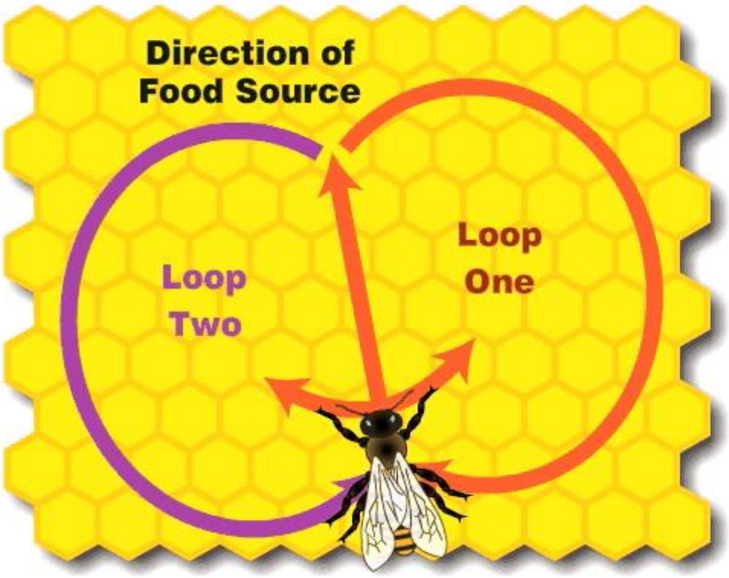
Behavior of Honey Bee Swarm

- In swarm behavior different tasks are performed simultaneously by specialized individuals – **division of labor**
- Honey bees are organized in 3 groups:
  - Employed Bees
  - Onlookers
  - Scouts
- **Employed Bees** search food around their assigned food sources.
- **Onlooker Bees** evaluate the nectar information taken from all employed bees and then choose a food source to further investigate
- **Scout bees** randomly search for new food sources to be in investigated – Scout bees are employed bees who's their assigned food source has been abandoned (due to low quantity of nectar).



Exchange of Information

- The exchange of information among bees is the most important occurrence in the formation of the collective knowledge
- Communication among bees related to quality of food sources (amount of nectar + distance) occurs in the **dancing area**
- The related dance is called **waggle dance**
  - Direction (angle of the dance)
  - Distance (duration of the dance)
  - Quality (frequency of the dance)



Artificial Honey Bee Algorithm – Search Steps

- **Step 1** – Generate initial population of honey bees (usually 50% of employed bees and 50% of onlooker bees) – each employed bee is assigned a **random food source** (random solution)
- **Step 2** – Employed bees produce **modifications** on the current food source location (**solution**). Provided that the nectar amount (**fitness**) of the new positions is higher than that of the previous one, the bee memorizes the new position (solution) and forgets the old one

$$X_i^{new} = X_i + \phi(X_i - X_p)$$

X_i^{new} - new location selected by bee i

X_i - old location of bee i

X_p - random location among all the bees

ϕ – random number U(-1,1)

Artificial Honey Bee Algorithm – Search Steps

- **Step 3** – After all employed bees complete the search process, they share the nectar information (fitness) of the food sources with the onlooker bees. **Onlooker bees evaluate the nectar information** taken from all employed bees and **chooses a food source with a probability** related to its nectar amount (fitness).
- **Step 4** – As in the case of the employed bees, it produces a modification on the selected food source location (**solution**).). Provided that the nectar amount (**fitness**) of the new positions are higher than that of the previous one, the bee memorizes the new position (solution) and forgets the old one.

$$p_i = \frac{f_i}{\sum f_i} \qquad if\ (rnd > p_i)\ , X_i^{new} = X_i + \phi(X_i - X_p)$$

p_i - probability of selecting food source from bee i

f_i - amount of nectar (fitness) of food source from bee i

rnd - random number U(0,1)

Artificial Honey Bee Algorithm – Search Steps

- **Step 5** – Food sources that a position cannot be improved further though a predetermined number of cycles, which is called **limit**, are abandoned. The corresponding employed bee becomes a scout bee. A new food source is randomly selected

Limit is typically set as $$limit = \frac{N}{2} \times D$$

N – number of bees in the populations

D – Dimension of the problem (i.e. number of decision variables for each bee)

- Iterate steps 2 to 5 until termination criterion satisfied

ABC - Example

- Objective: $\text{maximize } f(X) = x_1^2 - x_1x_2 + x_2^2 + 2x_1 + 4x_2 + 3$
- Where: $-5 \leq x_1, x_2 \leq 5$

- Population size $= 10$
- No. of employed bees = 5
- No. of onlooker bees = 5
- Limit = 1

ABC - Example

Step 1

- Iteration 1

$$f(X) = x_1^2 - x_1x_2 + x_2^2 + 2x_1 + 4x_2 + 3$$

$$f'(X) = \begin{cases} \dfrac{1}{(1+f)} \; ; f \geq 0 \\ 1 + |f| \; ; f < 0 \end{cases}$$

Food Source

x_1	x_2	$f(\mathrm{X})$	$f'(\mathrm{X})$	$trial$
3.1472	-4.0246	31.9645	0.0303	0
4.0579	-2.2150	32.6168	0.0297	0
-3.7301	0.4688	13.2971	0.0699	0
4.1338	**4.5751**	48.6753	**0.0201**	0
1.3236	4.6489	41.4537	0.0236	0

$f(\mathrm{X})$: maximize

$f'(\mathrm{X})$: minimize

ABC - Example

Step 1

- Iteration 1

$$f(X) = x_1^2 - x_1x_2 + x_2^2 + 2x_1 + 4x_2 + 3$$

$$f'(X) = \begin{cases} \frac{1}{(1+f)} & ; f \geq 0 \\ 1 + |f| & ; f < 0 \end{cases}$$

x_1	x_2	$f(\mathrm{X})$	$f'(\mathrm{X})$	$trial$
3.1472	-4.0246	31.9645	0.0303	0
4.0579	-2.2150	32.6168	0.0297	0
-3.7301	0.4688	13.2971	0.0699	0
4.1338	4.5751	48.6753	0.0201	0
1.3236	4.6489	41.4537	0.0236	0

- Iteration 2

x_1	x_2	$f(\mathrm{X})$	$f'(\mathrm{X})$	$trial$

Employed bee 1

Select random variable – let it be 1

Select random partner – let it be 4

Create new food location (solution)

$$X_i^{new} = X_i + \phi(X_i - X_p)$$

$$X_i^{new} = 3.1472 + 0.71(3.1472 - 4.1338) = 2.4467$$

$$X_i^{new} = [2.4467, -4.0246]$$

$$f(X_i) = 23.8259$$

$$f'(X_i) = 0.0403 > 0.0303$$

Worse location than before, thus preserve previous location

Increase trial to 1.

ABC - Example

Step 2

- Iteration 1

$$f(X) = x_1^2 - x_1x_2 + x_2^2 + 2x_1 + 4x_2 + 3$$

$$f'(X) = \begin{cases} \frac{1}{(1+f)} ; f \geq 0 \\ 1 + |f| ; f < 0 \end{cases}$$

x_1	x_2	$f(X)$	$f'(X)$	$trial$
3.1472	-4.0246	31.9645	0.0303	0
4.0579	-2.2150	32.6168	0.0297	0
-3.7301	0.4688	13.2971	0.0699	0
4.1338	4.5751	48.6753	0.0201	0
1.3236	4.6489	41.4537	0.0236	0

Employed bee 1

Select random variable – let it be 1

Select random partner – let it be 4

Create new food location (solution)

$$X_i^{new} = X_i + \phi(X_i - X_p)$$

$$X_i^{new} = 3.1472 + 0.71(3.1472 - 4.1338) = 2.4467$$

$$X_i^{new} = [2.4467, -4.0246]$$

$$f(X_i) = 23.8259$$

$$f'(X_i) = 0.0403 > 0.0303$$

Worse location than before, thus preserve previous location

Increase trial to 1.

- Iteration 2

x_1	x_2	$f(X)$	$f'(X)$	$trial$
3.1472	-4.0246	31.9645	0.0303	**1**

ABC - Example

Step 2

- Iteration 1

$$f(X) = x_1^2 - x_1x_2 + x_2^2 + 2x_1 + 4x_2 + 3$$

$$f'(X) = \begin{cases} \dfrac{1}{(1+f)} & ; f \geq 0 \\ 1 + |f| & ; f < 0 \end{cases}$$

x_1	x_2	$f(X)$	$f'(X)$	$trial$
3.1472	-4.0246	31.9645	0.0303	0
4.0579	**-2.2150**	32.6168	0.0297	0
-3.7301	**0.4688**	13.2971	0.0699	0
4.1338	4.5751	48.6753	0.0201	0
1.3236	4.6489	41.4537	0.0236	0

- Iteration 2

x_1	x_2	$f(X)$	$f'(X)$	$trial$
3.1472	-4.0246	31.9645	0.0303	1
4.0579	-3.0428	37.0428	0.0263	0

Employed bee 2

Select random variable – let it be 2

Select random partner – let it be 3

Create new food location (solution)

$$X_i^{new} = X_i + \phi(X_i - X_p)$$

$$X_i^{new} = -2.2150 + 0.31(-2.150 - 0.4688) = -3.0470$$

$$X_i^{new} = [4.0579, -3.0470]$$

$$f(X_i) = 37.0428$$

$$f'(X_i) = 0.0263 < 0.0297$$

Better location than before, thus update previous location

keep trial to 0.

ABC - Example

Step 2

- Iteration 1

$$f(X) = x_1^2 - x_1 x_2 + x_2^2 + 2x_1 + 4x_2 + 3$$

$$f'(X) = \begin{cases} \frac{1}{(1+f)} & ; f \geq 0 \\ 1 + |f| & ; f < 0 \end{cases}$$

x_1	x_2	$f(X)$	$f'(X)$	$trial$
3.1472	-4.0246	31.9645	0.0303	0
4.0579	-2.2150	32.6168	0.0297	0
-3.7301	0.4688	13.2971	0.0699	0
4.1338	4.5751	48.6753	0.0201	0
1.3236	4.6489	41.4537	0.0236	0

- Iteration 2

x_1	x_2	$f(X)$	$f'(X)$	$trial$
3.1472	-4.0246	31.9645	0.0303	1
4.0579	-3.0428	37.0428	0.0263	0
-3.7301	2.7604	38.4119	0.0254	0
4.1338	4.5751	48.6753	0.0201	1
1.6327	4.6489	41.5487	0.0235	0

Employed bees updates

Information is shared with the onlooker bees

ABC - Example

Step 3 & 4

- Iteration 2

Employed bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$	p_i
3.1472	-4.0246	31.9645	0.0303	1	0.2415
4.0579	-3.0428	37.0428	0.0263	0	0.2092
-3.7301	2.7604	38.4119	0.0254	0	0.2020
4.1338	4.5751	48.6753	0.0201	1	0.1602
1.6327	4.6489	41.5487	0.0235	0	0.1871

Onlooker bee 1
rnd= 0.26 >0.2415

Food source 1 is selected by the first onlooker bee

onlooker bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$

Select random variable – let it be 2
Select random partner – let it be 3
Create new food location (solution)

$$X_i^{new} = X_i + \phi(X_i - X_p)$$
$$X_i^{new} = [3.1472, 0.6571]$$
$$f(X_1) = 20.1914$$
$$f'(X_1) = 0.0472 > 0.0303$$

Worse location than before, thus increase trial to 2.

ABC - Example

Step 3 & 4

- Iteration 2

Employed bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$	p_i
3.1472	-4.0246	31.9645	0.0303	1	0.2415
4.0579	-3.0428	37.0428	0.0263	0	0.2092
-3.7301	2.7604	38.4119	0.0254	0	0.2020
4.1338	4.5751	48.6753	0.0201	1	0.1602
1.6327	4.6489	41.5487	0.0235	0	0.1871

Onlooker bee 1

rnd= 0.26 > 0.2415

Food source 1 is selected by the first onlooker bee

onlooker bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$
3.1472	-4.0246	31.9645	0.0303	2

Select random variable – let it be 2

Select random partner – let it be 3

Create new food location (solution)

$$X_i^{new} = X_i + \phi(X_i - X_p)$$

$$X_i^{new} = [3.1472, 0.6571]$$

$$f(X_1) = 20.1914$$

$$f'(X_1) = 0.0472 > 0.0303$$

Worse location than before, thus increase trial to 2.

ABC - Example

Step 3 & 4

- Iteration 2

Employed bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$	p_i
3.1472	-4.0246	31.9645	0.0303	1	0.2415
4.0579	-3.0428	37.0428	0.0263	0	0.2092
-3.7301	2.7604	38.4119	0.0254	0	0.2020
4.1338	4.5751	48.6753	0.0201	1	0.1602
1.6327	4.6489	41.5487	0.0235	0	0.1871

Onlooker bee 2

rnd= 0.10 < 0.2415

Food source 2 is not selected by the second onlooker bee

onlooker bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$
3.1472	-4.0246	31.9645	0.0303	2
4.0579	-3.0428	37.0428	0.0263	0

ABC - Example

Step 3 & 4

- Iteration 2

Employed bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$	p_i
3.1472	-4.0246	31.9645	0.0303	1	0.2415
4.0579	-3.0428	37.0428	0.0263	0	0.2092
-3.7301	2.7604	38.4119	0.0254	0	0.2020
4.1338	4.5751	48.6753	0.0201	1	0.1602
1.6327	4.6489	41.5487	0.0235	0	0.1871

Onlooker bee 2

rnd= 0.45 > 0.2020

Food source 3 is selected by the second onlooker bee

onlooker bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$
3.1472	-4.0246	31.9645	0.0303	2
4.0579	-3.0428	37.0428	0.0263	0
-5.0000	2.7604	50.4639	0.0194	0

Select random variable – let it be 1

Select random partner – let it be 2

Create new food location (solution)

$$X_i^{new} = X_i + \phi(X_i - X_p)$$

$$X_i^{new} = [-5.0000, 2.7604]$$

$$f(X_1) = 50.4639$$

$$f'(X_1) = 0.0194 < 0.0254$$

Better location than before, thus update location

ABC - Example

Step 3 & 4

- Iteration 2

Employed bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$	p_i
3.1472	-4.0246	31.9645	0.0303	1	0.2415
4.0579	-3.0428	37.0428	0.0263	0	0.2092
-3.7301	2.7604	38.4119	0.0254	0	0.2020
4.1338	4.5751	48.6753	0.0201	1	0.1602
1.6327	4.6489	41.5487	0.0235	0	0.1871

Onlooker bee 3
rnd= 0.07 < 0.1602

Food source 4 is not selected by the third onlooker bee

onlooker bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$
3.1472	-4.0246	31.9645	0.0303	2
4.0579	-3.0428	37.0428	0.0263	0
-5.0000	2.7604	50.4639	0.0194	0
4.1338	4.5751	48.6753	0.0201	1

ABC - Example

Step 3 & 4

- Iteration 2

Employed bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$	p_i
3.1472	-4.0246	31.9645	0.0303	1	0.2415
4.0579	-3.0428	37.0428	0.0263	0	0.2092
-3.7301	2.7604	38.4119	0.0254	0	0.2020
4.1338	4.5751	48.6753	0.0201	1	0.1602
1.6327	4.6489	41.5487	0.0235	0	0.1871

Onlooker bee 3
rnd= 0.14 < 0.1871

Food source 5 is not selected by the third onlooker bee

onlooker bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$
3.1472	-4.0246	31.9645	0.0303	2
4.0579	-3.0428	37.0428	0.0263	0
-5.0000	2.7604	50.4639	0.0194	0
4.1338	4.5751	48.6753	0.0201	1
1.6327	4.6489	41.5487	0.0235	0

ABC - Example

Step 3 & 4

- Iteration 2

Employed bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$	p_i
3.1472	-4.0246	31.9645	0.0303	1	0.2415
4.0579	-3.0428	37.0428	0.0263	0	0.2092
-3.7301	2.7604	38.4119	0.0254	0	0.2020
4.1338	4.5751	48.6753	0.0201	1	0.1602
1.6327	4.6489	41.5487	0.0235	0	0.1871

Onlooker bee 3

rnd= 0.65 > 0. 2415

Food source 1 is selected by the third onlooker bee

onlooker bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$
3.1472	-4.0246	31.9645	0.0303	**2**
4.0579	-3.0428	37.0428	0.0263	0
-5.0000	2.7604	50.4639	0.0194	0
4.1338	4.5751	48.6753	0.0201	1
1.6327	4.6489	41.5487	0.0235	0

Select random variable – let it be 2

Select random partner – let it be 2

Create new food location (solution)

$$X_i^{new} = X_i + \phi(X_i - X_p)$$

$$X_i^{new} = [3.1472, -3.5847]$$

$$f(X_1) = 28.9921$$

$$f'(X_1) = 0.0333 > 0.0303$$

Worse location than before, thus increase trial to 3

ABC - Example

Step 3 & 4

- Iteration 2

Employed bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$	p_i
3.1472	-4.0246	31.9645	0.0303	1	0.2415
4.0579	-3.0428	37.0428	0.0263	0	0.2092
-3.7301	2.7604	38.4119	0.0254	0	0.2020
4.1338	4.5751	48.6753	0.0201	1	0.1602
1.6327	4.6489	41.5487	0.0235	0	0.1871

Onlooker bee 4
rnd= 0.83 > 0. 2092

Food source 2 is selected by the fourth onlooker bee

onlooker bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$
3.1472	-4.0246	31.9645	0.0303	2
5.0000	**-3.0470**	**50.3311**	**0.0263**	0
-5.0000	2.7604	50.4639	0.0194	0
4.1338	4.5751	48.6753	0.0201	1
1.6327	4.6489	41.5487	0.0235	0

Select random variable – let it be 1
Select random partner – let it be 5
Create new food location (solution)

$$X_i^{new} = X_i + \phi(X_i - X_p)$$
$$X_i^{new} = [5.000, -3.0470]$$
$$f(X_1) = 50.3311$$
$$f'(X_1) = 0.0195 < 0.0263$$

Worse location than before, thus increase trial to 3

ABC - Example

Step 3 & 4

- Iteration 2

Employed bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$	p_i
3.1472	-4.0246	31.9645	0.0303	1	0.2415
4.0579	-3.0428	37.0428	0.0263	0	0.2092
-3.7301	2.7604	38.4119	0.0254	0	0.2020
4.1338	4.5751	48.6753	0.0201	1	0.1602
1.6327	4.6489	41.5487	0.0235	0	0.1871

Onlooker bee 5

rnd= 0.15 < 0. 2020

Food source 3 is not selected by the fifth onlooker bee

onlooker bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$
3.1472	-4.0246	31.9645	0.0303	2
5.0000	**-3.0470**	**50.3311**	**0.0263**	0
-5.0000	2.7604	50.4639	0.0194	0
4.1338	4.5751	48.6753	0.0201	1
1.6327	4.6489	41.5487	0.0235	0

ABC - Example

Step 3 & 4

- Iteration 2

Employed bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$	p_i
3.1472	-4.0246	31.9645	0.0303	1	0.2415
4.0579	-3.0428	37.0428	0.0263	0	0.2092
-3.7301	2.7604	38.4119	0.0254	0	0.2020
4.1338	4.5751	48.6753	0.0201	1	0.1602
1.6327	4.6489	41.5487	0.0235	0	0.1871

Onlooker bee 5

rnd= 0.01 < 0. 1602

Food source 4 is not selected by the fifth onlooker bee

onlooker bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$
3.1472	-4.0246	31.9645	0.0303	2
5.0000	**-3.0470**	**50.3311**	**0.0263**	0
-5.0000	2.7604	50.4639	0.0194	0
4.1338	4.5751	48.6753	0.0201	1
1.6327	4.6489	41.5487	0.0235	0

ABC - Example

Step 3 & 4

- Iteration 2

Employed bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$	p_i
3.1472	-4.0246	31.9645	0.0303	1	0.2415
4.0579	-3.0428	37.0428	0.0263	0	0.2092
-3.7301	2.7604	38.4119	0.0254	0	0.2020
4.1338	4.5751	48.6753	0.0201	1	0.1602
1.6327	4.6489	41.5487	0.0235	0	0.1871

Onlooker bee 5

rnd= 0.19 > 0.1871

Food source 5 is selected by the fifth onlooker bee

onlooker bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$
3.1472	-4.0246	31.9645	0.0303	2
5.0000	**-3.0470**	**50.3311**	**0.0263**	0
-5.0000	2.7604	50.4639	0.0194	0
4.1338	4.5751	48.6753	0.0201	1
1.6327	5.0000	45.7676	0.0214	0

Select random variable – let it be 2

Select random partner – let it be 1

Create new food location (solution)

$$X_i^{new} = X_i + \phi(X_i - X_p)$$

$$X_i^{new} = [1.6327, 5.0000]$$

$$f(X_1) = 45.7676$$

$$f'(X_1) = 0.0214 > 0.0235$$

Better location than before, thus update location

ABC - Example

Step 5

- Iteration 2

onlooker bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$	$trial > limit$
3.1472	-4.0246	31.9645	0.0303	2	true
5.0000	-3.0470	50.3311	0.0263	0	false
-5.0000	2.7604	50.4639	0.0194	0	false
4.1338	4.5751	48.6753	0.0201	1	false
1.6327	5.0000	45.7676	0.0214	0	false

scout bees phase

x_1	x_2	$f(X)$	$f'(X)$	$trial$
3.6045	**-1.7170**	**25.4710**	**0.0378**	**0**
5.0000	-3.0470	50.3311	0.0263	0
-5.0000	2.7604	50.4639	0.0194	0
4.1338	4.5751	48.6753	0.0201	1
1.6327	5.0000	45.7676	0.0214	0

Employed bee 1 becomes a scout bee
New solution is generated completely at random
We accept the new solution even if is worse

Cuckoo Search Algorithm

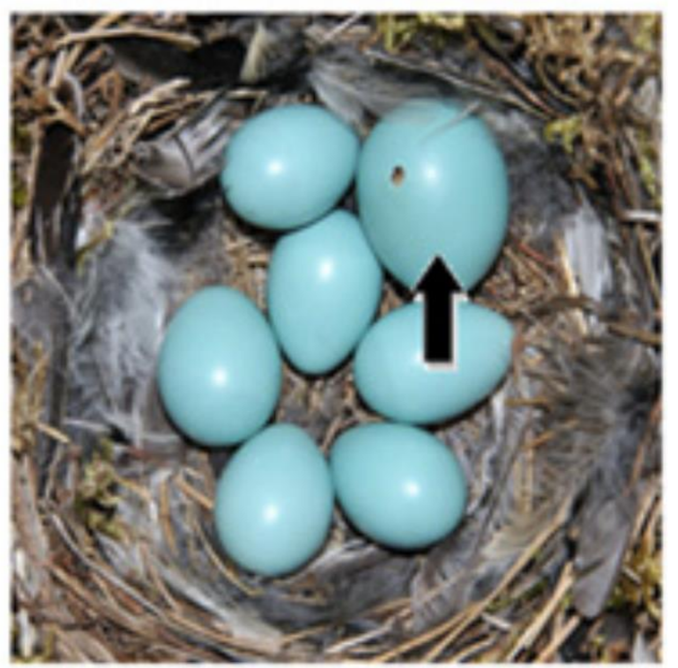
Nuno Antunes Ribeiro

Assistant Professor

Engineering Systems and Design

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

Behavior of Cuckoo breeding



- The Cuckoo Search Algorithm is inspired by the obligate **brood parasitism** of some cuckoo species by laying their eggs in the nests of host birds.
- Some cuckoos have evolved in such a way that female **parasitic cuckoos can imitate the colors and patterns of the eggs** of a few chosen host species.
- This reduces the **probability of the eggs being abandoned** and, therefore, increases their reproductivity .
- If host birds discover the eggs are not their own, they will either throw them away or simply abandon their nests and build new ones

Behavior of Cuckoo breeding

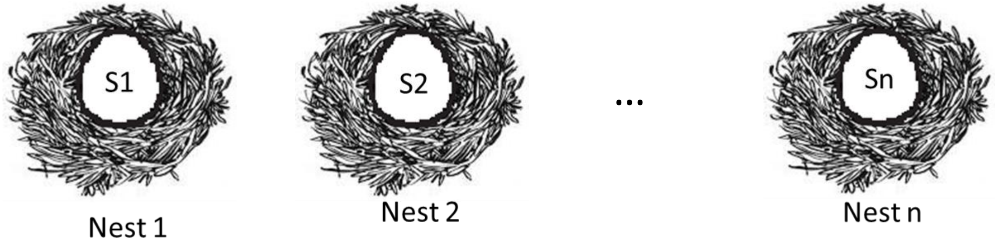
- Usually, the cuckoo eggs hatch slightly earlier than their host eggs.
- Once the first cuckoo chick is hatched, his first instinct action is to **evict the host eggs by blindly propelling the eggs** out of the nest.
- This action results in increasing the cuckoo chick's share of food provided by its host bird .
- Moreover, studies show that a cuckoo chick can imitate the call of host chicks to gain access to more feeding opportunity.

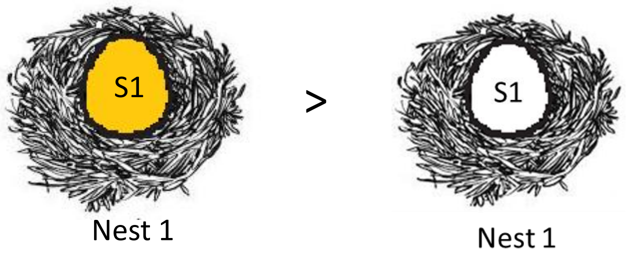
Cuckoo Search Algorithm

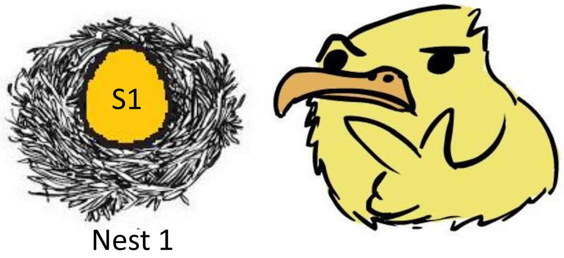
- Each egg in a nest represents a solution, and a cuckoo egg represents a new solution.
- In the **simplest form, each nest has only one egg**, but the algorithm can be extended to more complicated cases in which each nest has multiple eggs representing a set of solutions
- The CS algorithm is based on three rules:
  1. Each **cuckoo lays one egg** at a time in a randomly chosen nest
  2. The **best eggs** (solutions) in a nest **will carry over to the next generations**
  3. The number of nests is fixed, and a host can discover an alien egg with probability $p_a \epsilon [0,1]$. In this case, the host bird can either **throw the egg away or abandon the nest to build a completely new nest.**

Cuckoo Search Algorithm – Search Steps

- **Step 1** - Generate initial population of n host nests
- **Step 2** – Lay a new egg in the nest *n*
- **Step 3** – Compare the fitness of cuckoo's egg with the fitness of the host egg
- **Step 4** – If the fitness of cuckoo's egg is better than host egg, replace the egg in nest k by cuckoo's egg
- **Step 5** – If host bird notice it $p_a \epsilon [0,1]$. , the nest is abandoned and new random one is built.
- Iterate steps 2 to 5 until termination criterion satisfied







CS - Example

- Objective: $\text{minimize } f(X) = x_1^2 - x_1x_2 + x_2^2 + 2x_1 + 4x_2 + 3$
- Where: $-5 \leq x_1, x_2 \leq 5$

- Population size = 5
- Probability of abandoning the nest: $p_a = 0.9$

CS - Example

Step 1

$$f(X) = x_1^2 - x_1x_2 + x_2^2 + 2x_1 + 4x_2 + 3$$

- Iteration 1

Nest

x_1	x_2	$f(X)$
-4.5861	3.9930	65.0885
-0.0215	4.1747	37.1741
-0.8079	4.3295	41.5973
1.7389	2.6541	22.5470
1.0005	**-2.6027**	**4.9692**

Position randomly generated $U(0,1)$

G_1	G_2	$f(X)$
1.0005	**-2.6027**	**4.9692**

Levy Flights

- Original Cuckoo Search proposes a random perturbation relying on **levy flights** instead of random walks.

$$X_{new} = X + randn \times C$$

$$C = 0.01 \times S \times (X - gbest)$$

X – current solution

$N(0,1)$ – random number generated using a normal distribution

S – random step generated by a symmetric Levy distribution

$$s = \frac{u}{|v|^{1/\beta}}$$

$$\beta = 1.5$$

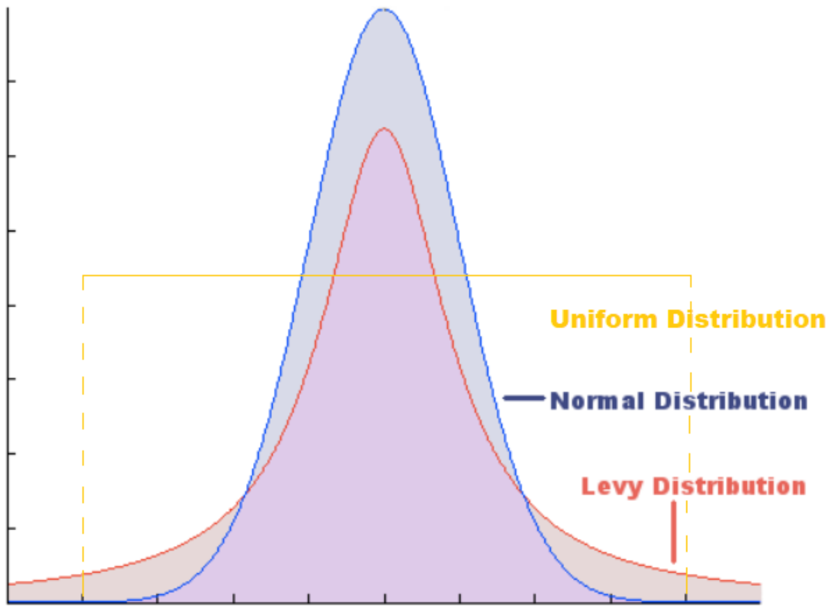
$$u = X * N(0,1) * \sigma_u$$

$$v = X * N(0,1)$$

Cuckoo Search via Lévy Flights

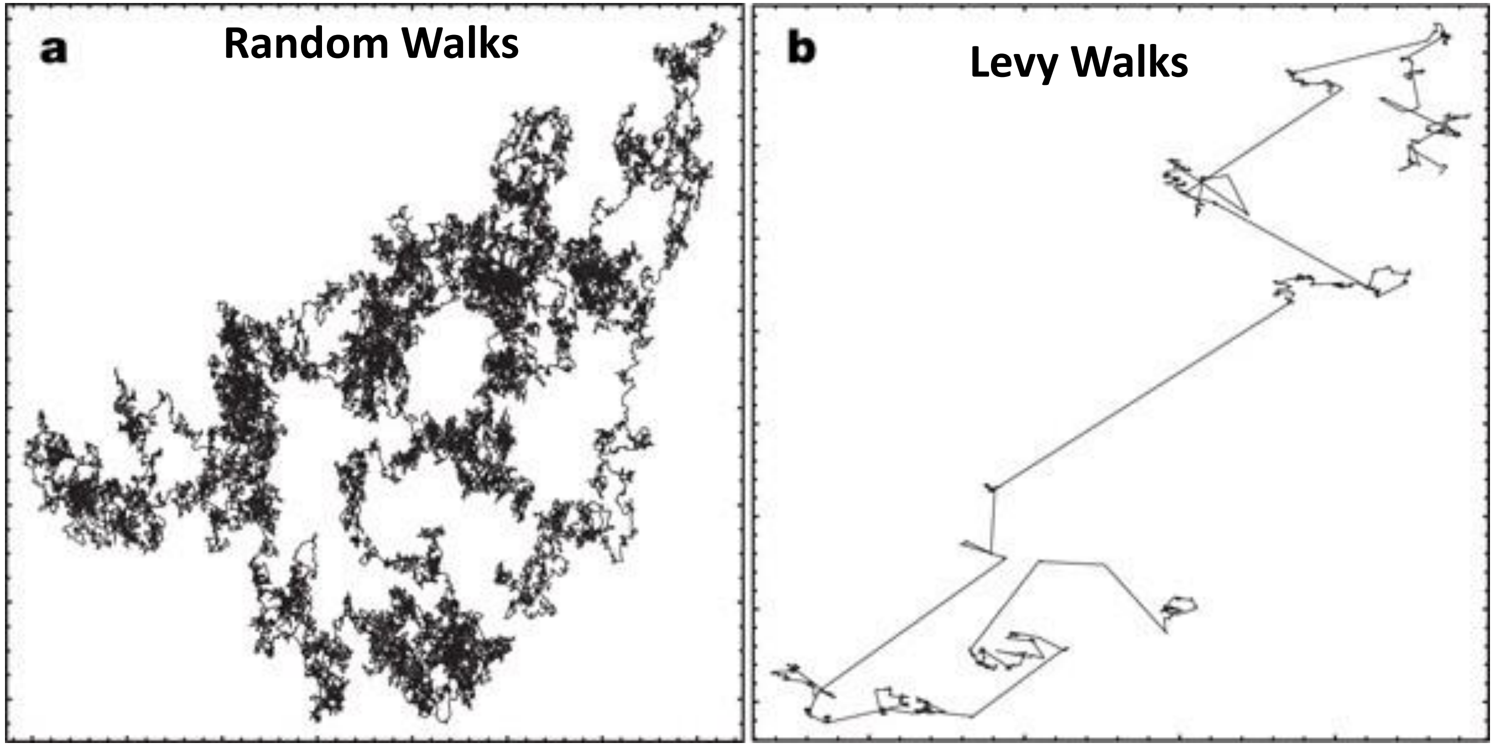
Xin-She Yang
Department of Engineering,
University of Cambridge
Trumpinton Street
Cambridge CB2 1PZ, UK
Email: xy227@cam.ac.uk

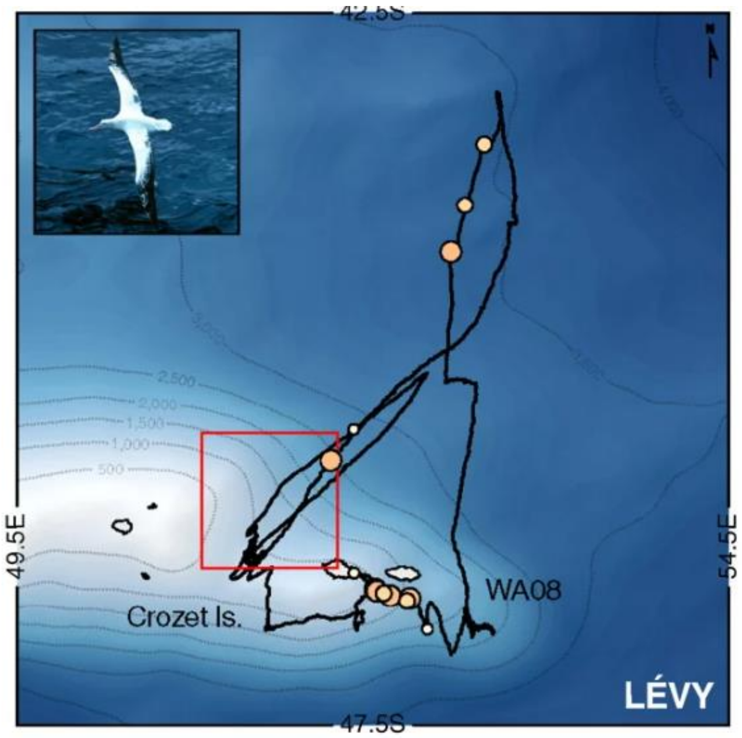
Suash Deb
Department of Computer Science & Engineering
C. V. Raman College of Engineering
Bidyanagar, Mahura, Janla
Bhubaneswar 752054, INDIA
Email: suashdeb@gmail.com



Levy Flights

- In a food-scarce environment, it is a waste of time and energy to always move a short distance from the previous position.
- It turns out that a better strategy is to occasionally move a long distance. That puts the animal in a new location, which it can explore by moving small distances again.
- This behavior has been observed in many animals that hunt at sea, including albatross, sharks, turtles, penguins, and tuna





CS - Example

- Objective: $\text{minimize } f(X) = x_1^2 - x_1 x_2 + x_2^2 + 2x_1 + 4x_2 + 3$
- Where: $-5 \leq x_1, x_2 \leq 5$

- Population size = 5
- Probability of abandoning the nest: $p_a = 0.9$
- $\beta = 1.5$
- $\sigma_u = 0.7$

CS - Example

Step 2 & 3

- Iteration 1

Nest

x_1	x_2
-4.5861	**3.9930**
-0.0215	4.1747
-0.8079	4.3295
1.7389	2.6541
1.0005	-2.6027

$f(\mathrm{X})$
65.0885
37.1741
41.5973
22.5470
4.9692

G_1	G_2
1.0005	-2.6027

$f(\mathrm{X})$
4.9692

- Iteration 2

Nest

x_1	x_2
-4.4946	3.9554

$f(\mathrm{X})$
63.4556

$$u = x_i * N(0,1) * \sigma_u = [-0.7180, -1.0874]$$

$$v = x_i * N(0,1) = [-0.2260, -0.5517]$$

$$S = \frac{u}{|v|^{1/\beta}} = [-1.9353, -1.6166]$$

$$x_i = x_i + N(0,1) \times 0.01 \times S \times (x_i - gbest)$$

$$x_1 = -4.5861 + 0.8468 \times 0.01 \times (-1.9353) \times (-4.5861 - 1.0005)$$

$$x_2 = 3.9930 + 0.3531 \times 0.01 \times (-1.6166) \times (3.9930 + 2.6027)$$

CS - Example

Step 2 & 3

- Iteration 1

Nest

x_1	x_2
-4.5861	3.9930
-0.0215	4.1747
-0.8079	4.3295
1.7389	2.6541
1.0005	-2.6027

$f(X)$
65.0885
37.1741
41.5973
22.5470
4.9692

G_1	G_2
1.0005	-2.6027

$f(X)$
4.9692

- Iteration 2

Nest

x_1	x_2
-4.4946	3.9554
-0.0326	4.2200
-0.8218	4.3220
1.7398	2.5688
1.0005	-2.6027

$f(X)$
63.4556
37.7618
41.5516
21.9115
4.9692

CS - Example

Step 4

- Iteration 1

Nest

x_1	x_2	$f(X)$
-4.5861	3.9930	65.0885
-0.0215	4.1747	37.1741
-0.8079	4.3295	41.5973
1.7389	2.6541	22.5470
1.0005	**-2.6027**	**4.9692**

G_1	G_2	$f(X)$
1.0005	**-2.6027**	**4.9692**

- Iteration 2

Nest

x_1	x_2	$f(X)$
-4.4946	3.9554	63.4556
-0.0326	4.2200	37.7618
-0.8218	4.3220	41.5516
1.7398	2.5688	21.9115
1.0005	-2.6027	4.9692

x_1	x_2	$f(X)$
-4.4946	3.9554	63.4556
-0.0215	4.1747	37.1741
-0.8218	4.3220	41.5516
1.7398	2.5688	21.9115
1.0005	-2.6027	4.9692

CS - Example

Step 5

- Iteration 1

Nest

x_1	x_2	$f(X)$
-4.5861	3.9930	65.0885
-0.0215	4.1747	37.1741
-0.8079	4.3295	41.5973
1.7389	2.6541	22.5470
1.0005	-2.6027	4.9692

G_1	G_2	$f(X)$
-2.5151	**-0.4597**	**1.5122**

$r < 0.25$?

- Iteration 2

Nest

random number	x_1	x_2	$f(X)$
0.97	-4.4946	3.9554	63.4556
	-0.0215	4.1747	37.1741
0.14	-0.8218	4.3220	41.5516
0.24	1.7398	2.5688	21.9115
	1.0005	-2.6027	4.9692

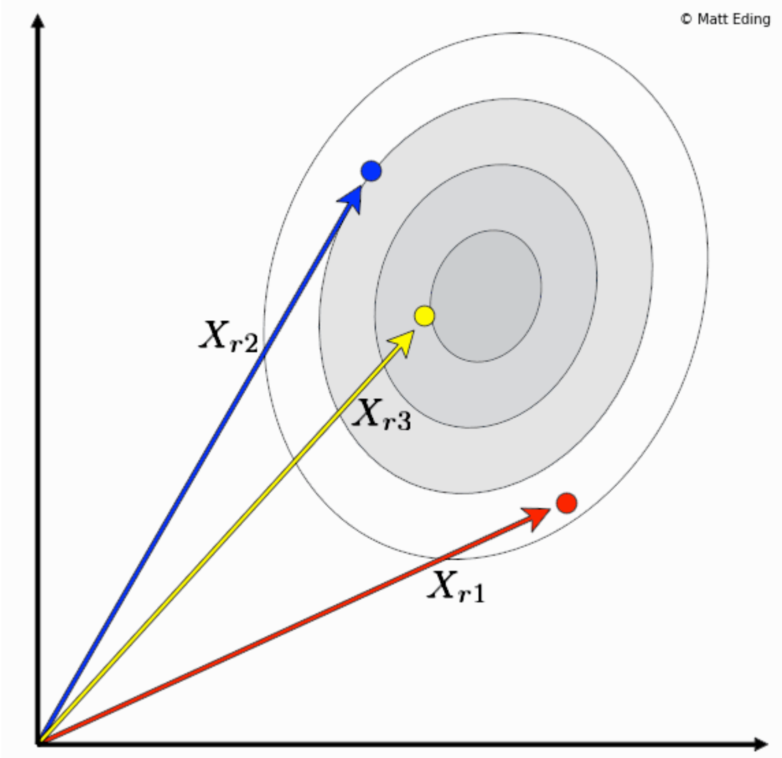
x_1	x_2	$f(X)$
-4.4946	3.9554	63.4556
-0.0215	4.1747	37.1741
-0.2781	1.0026	7.8155
-2.5151	**-0.4597**	**1.5122**
1.0005	-2.6027	4.9692

Recall: Mutation Operator in DE (Lec. 12)

- Original CS uses the mutation operator from differential evolution (see lecture 12 – slide 60) to randomly generate a new solution when a cuckoo rejects a solution. This ensures that the solution generated is not completely random.

$$V_{iG} = X_{r1_iG} + \lambda\left(X_{r2_iG} - X_{r3_iG}\right)$$

λ is a factor from 0 to 2



https://matteding.github.io/2019/04/17/differential-evolution/

Grey Wolf Optimizer

Nuno Antunes Ribeiro

Assistant Professor

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

Engineering Systems and Design

Social Hierarchy of Grey Wolf

- **Grey wolfs** are organized in 4 groups:
  - **Alpha**, or the **leaders**, are responsible for making decisions about hunting, sleeping place, time to wake, etc. Interestingly the alpha is not necessarily the strongest member of the pack but the best in terms of managing the pack.
  - **Beta**, or the **advisors**, are subordinate wolves that help the alpha in decision-making or other pack activities. The beta wolf should respect the alpha, but commands the other lower-level wolves. Beta wolves are often the best candidates to be the alpha when this passes away or becomes old.
  - **Delta**, or the **subordinate**, often comprise wolves from different categories – scouts, sentinels, hunters, caretakers, elders. Scouts are responsible for watching the boundaries of the territory and warning the pack in case of any danger. Sentinels protect and guarantee safety of the pack. Elders are the experienced wolves who used to be alpha or beta. Hunters help the hunters and betas hunting prey. Caretakers are responsible for caring for the weak, ill, and wounded wolves
  - **Omega**, or the **scapegoat**, the lowest ranking member of the pack. The omega lives on the outskirts of the pack, usually eating last. The omega serves as both a stress-reliever and instigator of play. They may seem not important, but the whole pack fights when there is no omega.\

Hunting Mechanism of Grey Wolves

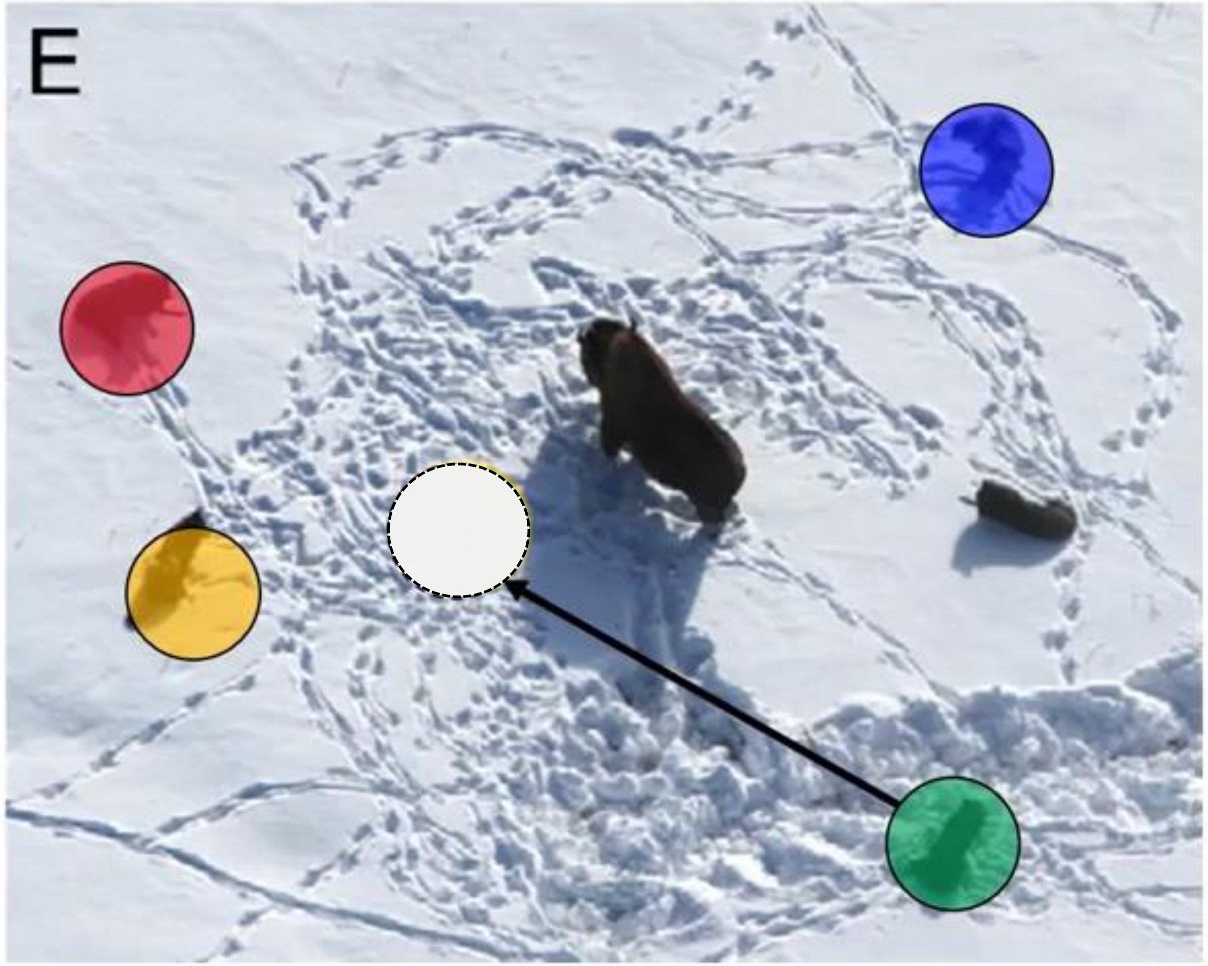
- The hunting operation is usually guided by the alpha.
- The beta and delta might participate in hunting occasionally
- The main phases of grey wolf hunting are as follows:
  - Tracking, chasing and approaching the prey
  - Pursuing, **encircling**, and harassing the prey until it stops moving
  - Attack towards the prey

Hunting behaviour of grey wolves:

(A) chasing, approaching, and tracking prey
(B-D) pursuing, harassing, and encircling
(E) stationary situation and attack

Grey Wolf Optimizer

- In the grey wolf optimizer (GWO), we consider the **fittest solution** as the alpha α, and the **second and the third fittest** solutions are named beta β and delta δ, respectively.
- The rest of the solutions are considered omega ω.
- The **ω solutions are guided by the α, β and δ**



Grey Wolf Optimizer

- During the hunting, the grey wolves encircle the prey.
- The mathematical model of the **encircling behaviour** is presented as follows

$$X(t+1) = X_p - AD \qquad D = \left|CX_p - X_i\right|$$

- Where t is the current iteration, X_p is the position vector of the “prey”, and X is the position vector of a omega grey wolf. A and C are coefficient vectors calculated as follows

$$A = 2a \times r_1 - a \qquad C = 2 \times r_2$$

- a is a parameter that decreases linearly from 2 to 0 over the course of the iterations, and r_1 and r_2 are random vectors in [0,1]

Grey Wolf Optimizer

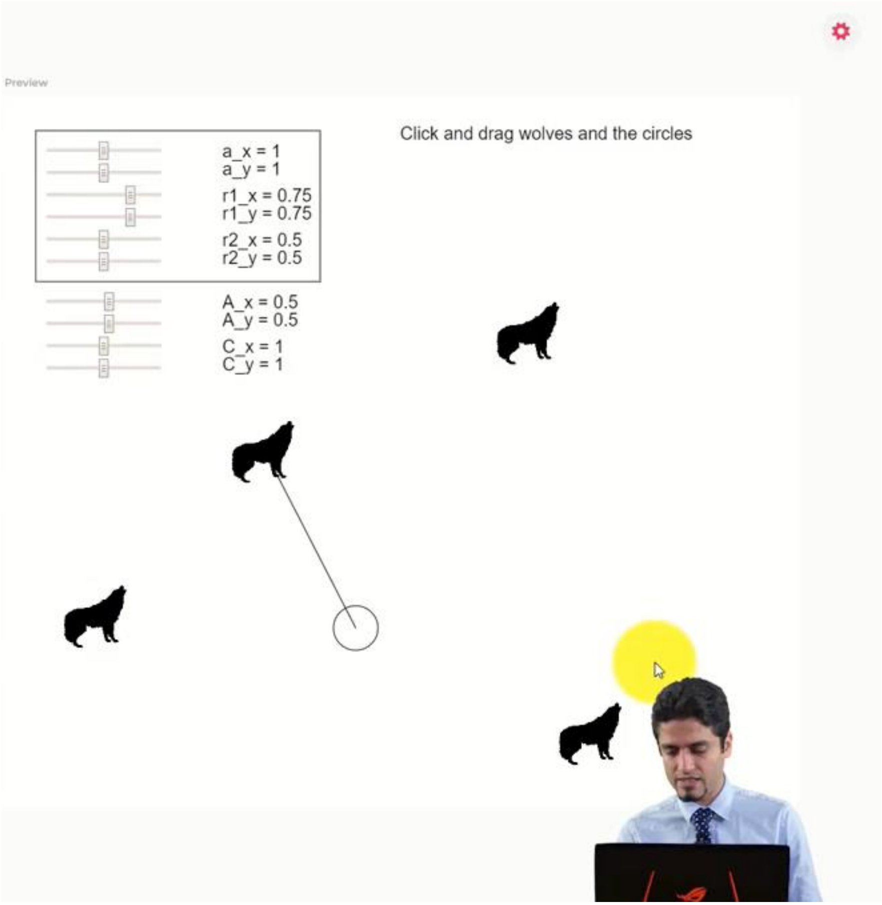
- In the optimization problem, we do not know where the "prey" (optimal solution is located).
- We assume the alpha, beta and delta have better knowledge about the potential location of prey.

$$X_1(t+1) = X_\alpha - A_\alpha D_\alpha$$

$$X_2(t+1) = X_\beta - A_\beta D_\beta \qquad X(t+1) = \frac{X_1 + X_2 + X_3}{3}$$

$$X_3(t+1) = X_\delta - A_\delta D_\delta$$

- If r_1 and r_2 are equal to zero, then the new position of the omega wolf is in the centre of the three wolves (alpha, beta and gamma).
- r_1 and r_2 are used to stimulate exploration (randomness). As, the number of iterations increase, a decreases linearly to 0, and exploration is reduced



Source: https://www.youtube.com/watch?v=uzcOcXI2C_0

GWO - Example

$$a = 2 - 2\left(\frac{1}{100}\right) = 1.99$$

- Iteration 1

wolves

	x_1	x_2	$f(X)$
	-4.5861	3.9930	65.0885
δ	-0.0215	4.1747	37.1741
	-0.8079	4.3295	41.5973
β	1.7389	2.6541	22.5470
α	1.0005	-2.6027	4.9692

1st wolf

$$A = 2a \times r_1 - a$$

$$C = 2 \times r_2$$

$$D_\alpha = |CX_p - X_i|$$

$$X_1 = X_p - AD$$

$$X = \frac{X_1 + X_2 + X_3}{3}$$

$$A_\alpha = 2 \times 1.99 \times 0.54 - 1.99 = 0.1592$$

$$C_\alpha = 2 \times 0.65 = 1.30$$

$$D_\alpha = \left|1.30\begin{pmatrix}1.0005\\-2.6027\end{pmatrix} - \begin{pmatrix}-4.5861\\3.9930\end{pmatrix}\right| = \begin{pmatrix}5.88675\\7.3765\end{pmatrix}$$

$$X_1 = \begin{pmatrix}1.0005\\-2.6027\end{pmatrix} - 0.1592\begin{pmatrix}5.88675\\7.3765\end{pmatrix} = \begin{pmatrix}0.06333\\-3.77704\end{pmatrix}$$

$$A_\beta = 2 \times 1.99 \times 0.34 - 1.99 = -0.6368$$

$$C_\beta = 2 \times 0.75 = 1.50$$

$$D_\beta = \left|1.50\begin{pmatrix}1.7389\\2.6541\end{pmatrix} - \begin{pmatrix}-4.5861\\3.9930\end{pmatrix}\right| = \begin{pmatrix}7.1945\\0.0118\end{pmatrix}$$

$$X_2 = \begin{pmatrix}1.7389\\2.6541\end{pmatrix} + 0.6368\begin{pmatrix}7.1945\\0.0118\end{pmatrix} = \begin{pmatrix}6.32036\\2.66161\end{pmatrix}$$

$$A_\delta = 2 \times 1.99 \times 0.74 - 1.99 = 0.9552$$

$$C_\delta = 2 \times 0.34 = 0.68$$

$$D_\delta = \left|0.68\begin{pmatrix}-0.0215\\4.1747\end{pmatrix} - \begin{pmatrix}-4.5861\\3.9930\end{pmatrix}\right| = \begin{pmatrix}4.57148\\1.1542\end{pmatrix}$$

$$X_3 = \begin{pmatrix}-0.0215\\4.1747\end{pmatrix} - 0.9552\begin{pmatrix}4.57148\\1.1542\end{pmatrix} = \begin{pmatrix}-4.38818\\3.07221\end{pmatrix}$$

$$x_1 = \frac{0.06333 + 6.32036 + 4.38818}{3} = 0.66517$$

$$x_2 = \frac{-3.77704 + 2.66161 + 3.07221}{3} = 0.65226$$

$f(x) = 7.3734 < 65.0885$ **(accept)**

wolves

x_1	x_2
0.66517	0.65226

Repeat for the other wolves

GWO - Example

- Iteration 1

wolves

	x_1	x_2	$f(X)$
	-4.5861	3.9930	65.0885
δ	-0.0215	4.1747	37.1741
	-0.8079	4.3295	41.5973
β	1.7389	2.6541	22.5470
α	1.0005	-2.6027	4.9692

wolves

	x_1	x_2	$f(X)$
δ	0.66517	0.65226	7.3734
	2.39678	3.07265	27.90547
β	0.48362	-0.25494	3.36963
	1.7389	2.6541	22.5470
α	0.41150	-0.91718	1.54224

Same solution because a better solution was not found

Repeat until termination criteria

Swarm Optimization & Hybridization with Evolutionary Algorithms

Nuno Antunes Ribeiro

Assistant Professor

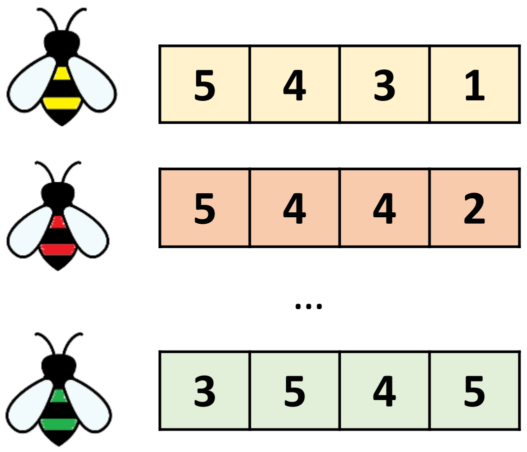
SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

Engineering Systems and Design

Artificial Honey Bee Algorithm

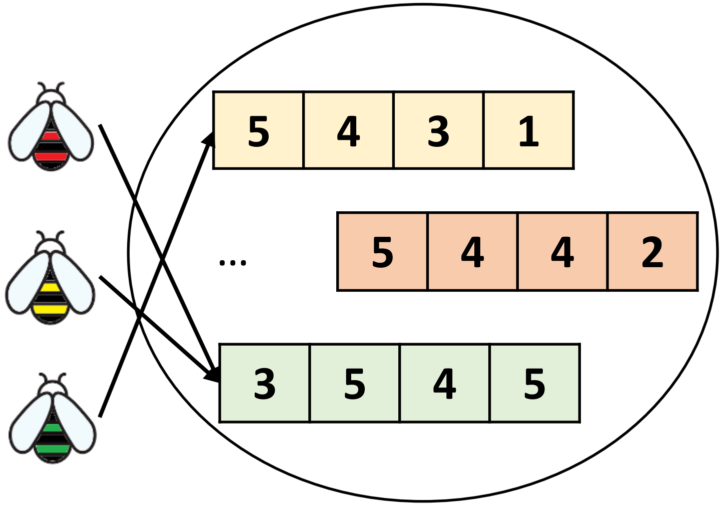
- The Artificial Honey Bee Colony Algorithms is aimed at dealing with continuous optimization problems. What about discrete optimization problems?

Step 1 – Employed Bees



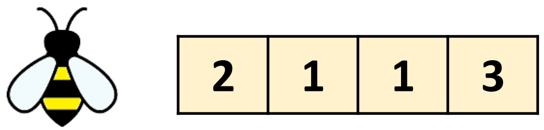
Alternative 1: Loop through all the bees and apply crossover with a random solution
Alternative 2: Loop through all the bees apply local search algorithm

Step 2 – Onlooker Bees



Loop through all the bees by randomly picking 1 solution with a probability proportional to the fitness.
Alternative 1: apply crossover with a random solution
Alternative 2: apply local search algorithm
If solution obtained is not improved, increase trial.

Step 3 – Scout Bees

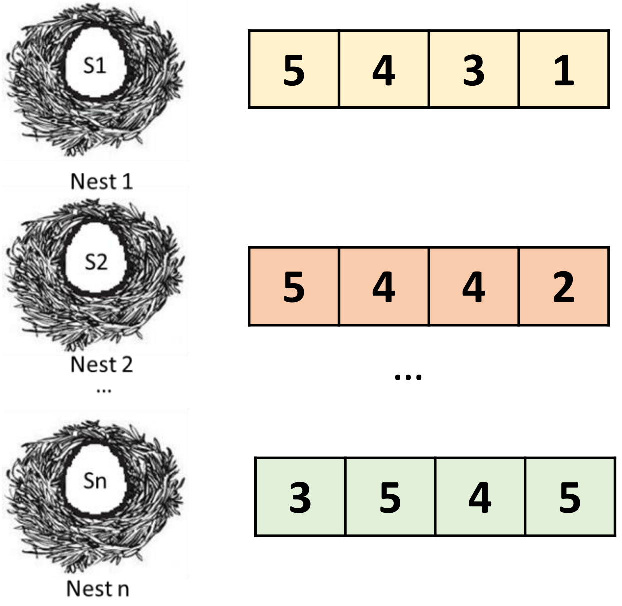


For all the bees assigned to solutions that exceed the trial limit specified, generate a new random solution – by applying crossover between 2 random solutions + mutation

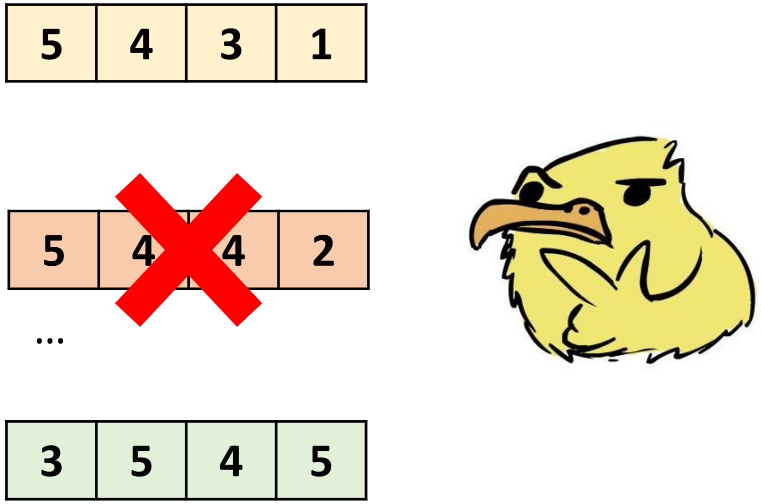
Cuckoo Search Algorithm

- The Cuckoo Search Algorithm is aimed at dealing with continuous optimization problems. What about discrete optimization problems?

Step 1 – Crossover with the best solution + mutation; if better replace current solution in the nest



Step 2 – Eliminate new solutions given a random probability, and generate new solutions by selecting two random solutions + mutation



Grey Wolf Optimizer

- The Grey Wolf Optimizer Algorithm is aimed at dealing with continuous optimization problems. What about discrete optimization problems?

ω	3	3	3	2	5	3	3	1	5	3

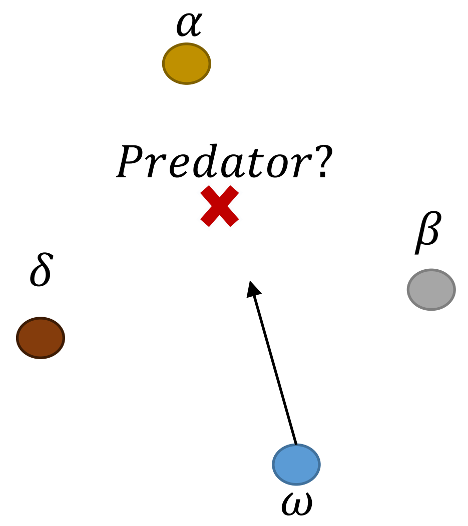
α	5	4	3	1	1	5	3	4	1	5
β	5	4	4	2	5	2	1	1	4	5
δ	3	5	4	5	3	1	4	1	2	2

Step 1 - crossover with the δ wolf

Step 2 - crossover with the β wolf

Step 3 - crossover with the α wolf

Step 4 - mutation (mutation rate decreases with the number of iterations)



Concluding Remarks

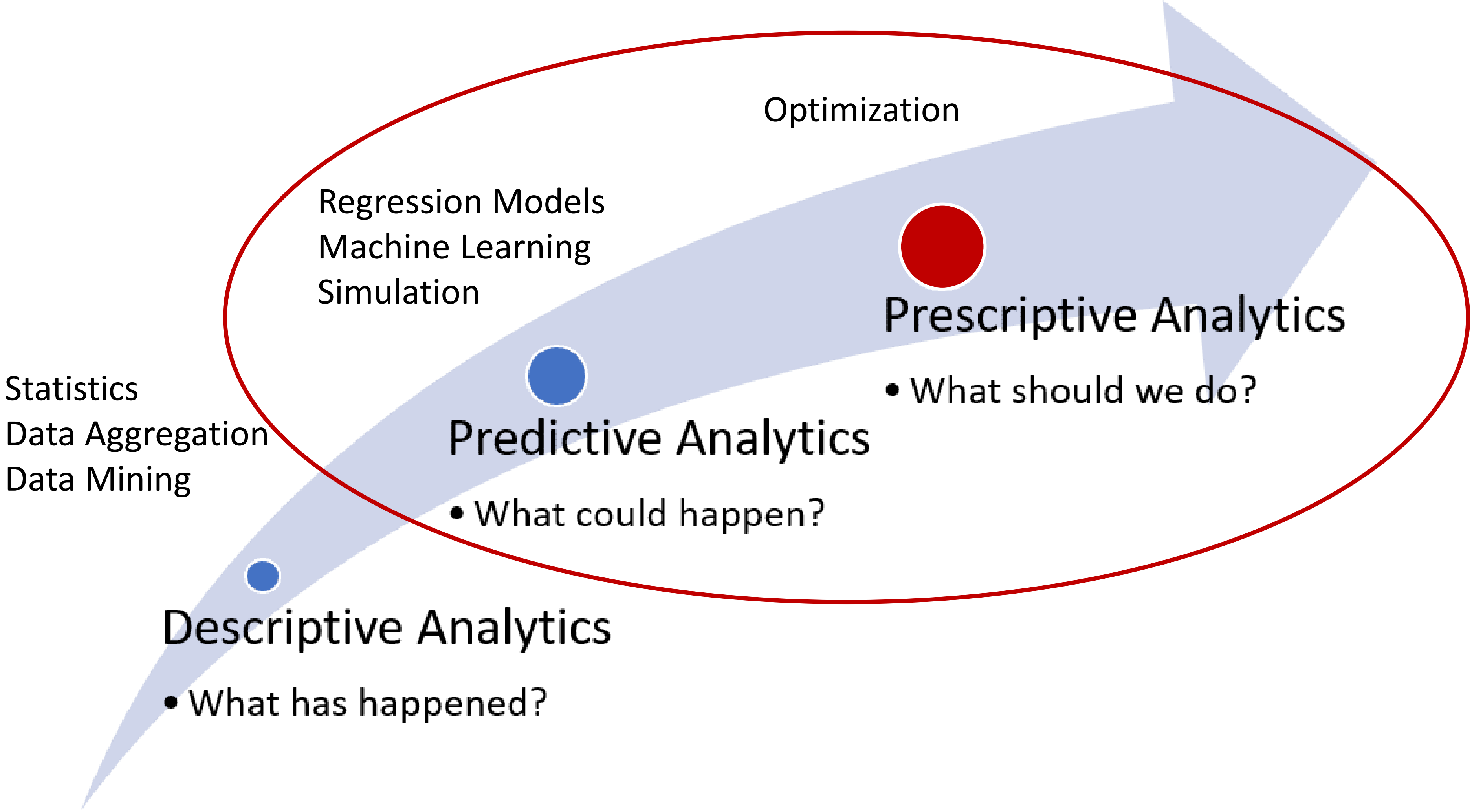
Nuno Antunes Ribeiro

Assistant Professor

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

Engineering Systems and Design

Metaheuristics for Prescriptive and Predictive Analytics



Metaheuristics for Prescriptive and Predictive Analytics

Prescriptive Analytics (optimize)

Types of Models

- Linear Optimization
- **Non-linear Optimization**
- **Discrete Optimization**
- **Multi-Objective Optimization**
- **Black-box Optimization**

Solution Methods

- Exact Methods
- **Metaheuristics**
- Specialized Heuristics

Applications

- Scheduling
- Routing
- Assignment
- Investment
- Design
- …

Predictive Analytics (predict)

Types of Models

- Supervised Learning
- Unsupervised Learning
- **Reinforcement Learning**
- **Simulation**

Solution Methods

- Regression Methods
- Trial and Error
- Specialized Heuristics
- **Metaheuristics**

Feature Selection
Parameter Tunning
Model Selection
Model Architecture
Neuroevolution
Genetic Programming
…

Metaheuristics for Prescriptive and Predictive Analytics

Prescriptive Analytics (optimize)

Types of Models
- Linear Optimization
- **Non-linear Optimization**
- **Discrete Optimization**
- **Multi-Objective Optimization**
- **Black-box Optimization**

Solution Methods
- Exact Methods
- **Metaheuristics**
- Specialized Heuristics

Predictive Analytics (predict)

Types of Models
- Supervised Learning
- Unsupervised Learning
- **Reinforcement Learning**
- **Simulation**

Solution Methods
- Regression Methods
- Trial and Error
- Specialized Heuristics
- **Metaheuristics**

Applications

- Transportation
- Supply Chain & Logistics
- Distribution Systems
- Storage Systems
- Healthcare
- Energy Systems
- Water Resources
- Airports and Airlines
- Structural Optimization
- Product development
- Sports
- Project Management
- Finances
- Marketing
- Robotics
- Image Recognition
- Text Analytics
- Etc.

From Prescriptive and Predictive Analytics

Example: Airline Planning

Predict

Passenger Demand

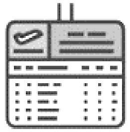
Passenger Choices

How passenger choose among flights?
How passengers are influenced by the schedules, prices, airlines, airports, etc.

Formulated as a discrete-choice model, or machine learning

Optimize

Route Development — **Where to fly the aircraft profitably?**



Timetable Development — **At what times should flight departures be schedule?**

Fleet Assignment — **What type of aircraft should be used for each flight?**

Pricing and Revenue Management — **Which fares to offer on each flight and how many seats to allocate across each fare?**

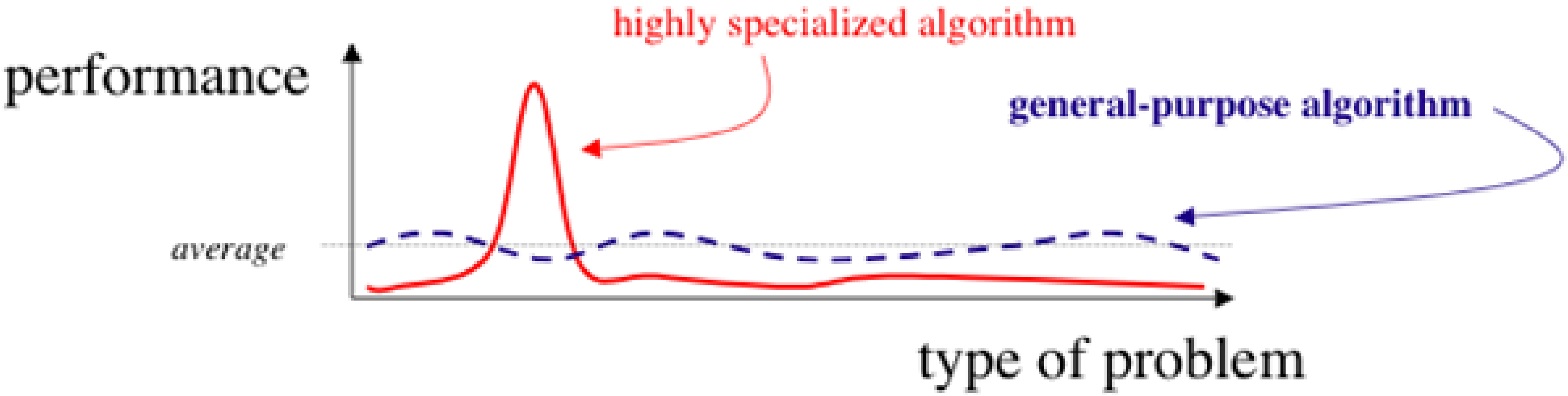
Formulated as a sequence of Discrete Optimization Problems

No Free Lunch" Metaphor

- “All optimization algorithms perform equally well when their performance is averaged across all possible problems”

https://www.cartoonstock.com/directory/r/reataurant.asp

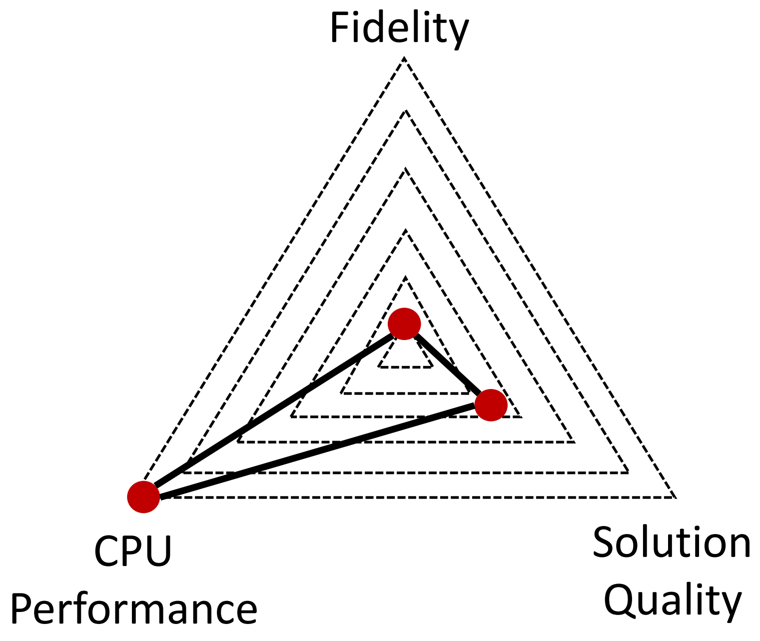
No Free Lunch" Metaphor

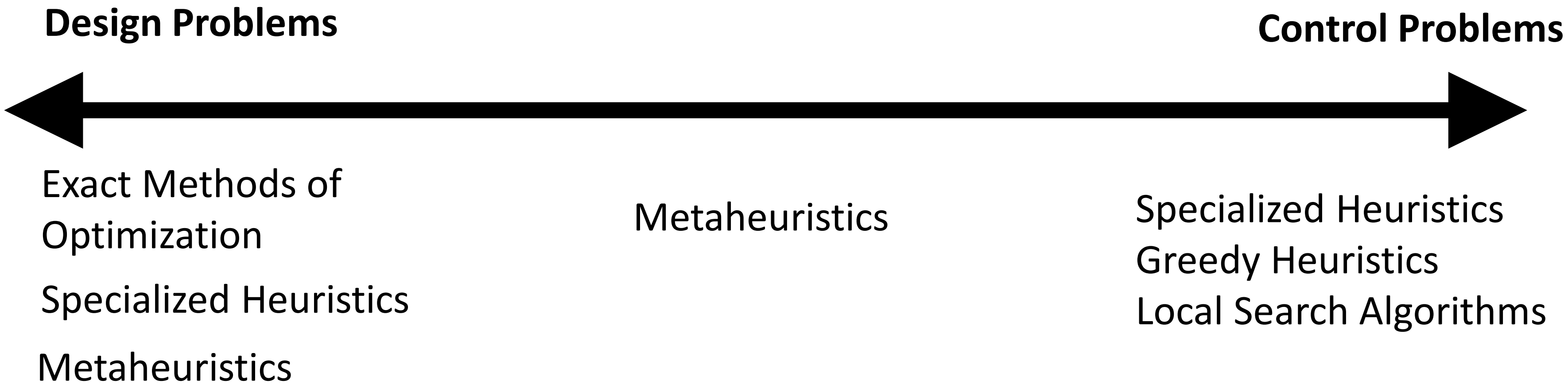


The original NFL theorems, were derived for ML and only later generalized to optimization

Selection of the Optimization Algorithm

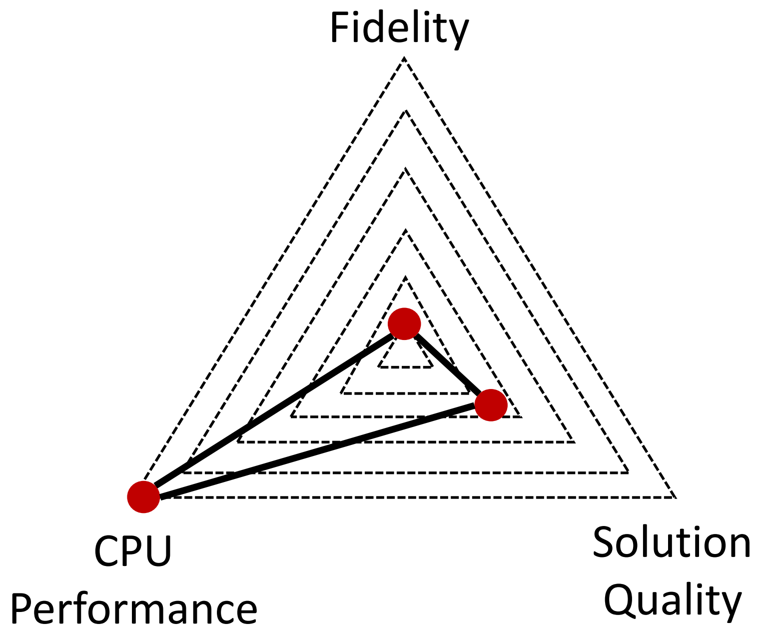
- **Design problems:** Design problems are generally solved once and involve investments (e.g. telecommunication network design and processor design, etc.)
- **Control problems:** Require very fast heuristics; the quality of the solutions is less critical (e.g. routing messages in a computer network; traffic management in a city; ride-sharing operations; text analytics, etc.) .
- **Planning problems:** In this class of problems, a trade-off between the quality of solution and the search time must be optimized; (e.g. scheduling of operations ; task assignment, etc.)

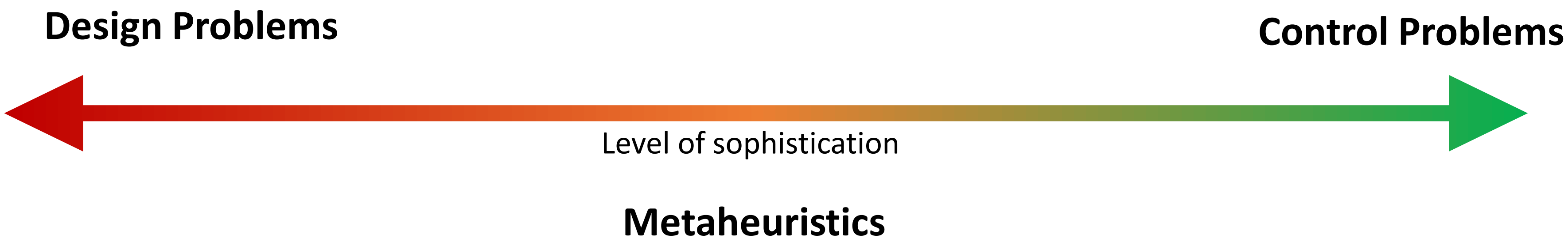




Selection of the Optimization Algorithm

- **Design problems:** Design problems are generally solved once and involve investments (e.g. telecommunication network design and processor design, etc.)
- **Control problems:** Require very fast heuristics; the quality of the solutions is less critical (e.g. routing messages in a computer network; traffic management in a city; ride-sharing operations; text analytics, etc.) .
- **Planning problems:** In this class of problems, a trade-off between the quality of solution and the search time must be optimized; (e.g. scheduling of operations ; task assignment, etc.)





Last Slide :(