CUCKOO SEARCH ALGORITHM

- **Step 1: Initialize Population**

  We have, Total Population of Nest (n) = 5.
- **Step 2: Initialize Worse Case Parameter. [i.e., Probability of discovery of Cuckoo Egg]**

  Pa = 0.25.
- **Step 3: Set Parameter for Total number of Maximum Iteration.**

  Maxt = 300.

- NOTE: We can not make any difference between an Egg / a Nest / or a Cuckoo.
- **AIM:** Use New and Better solution to replace bad solution in the current nest population.

SEARCH PATHS OF NEST USING CUCKOO SEARCH

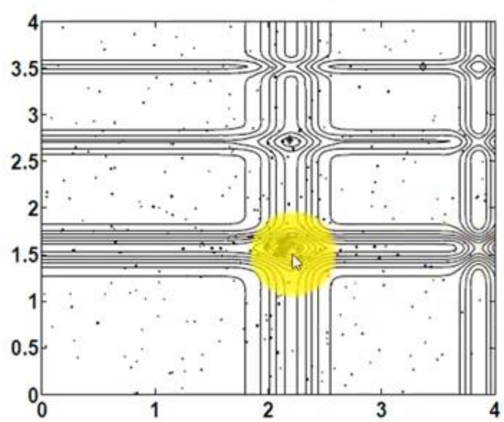
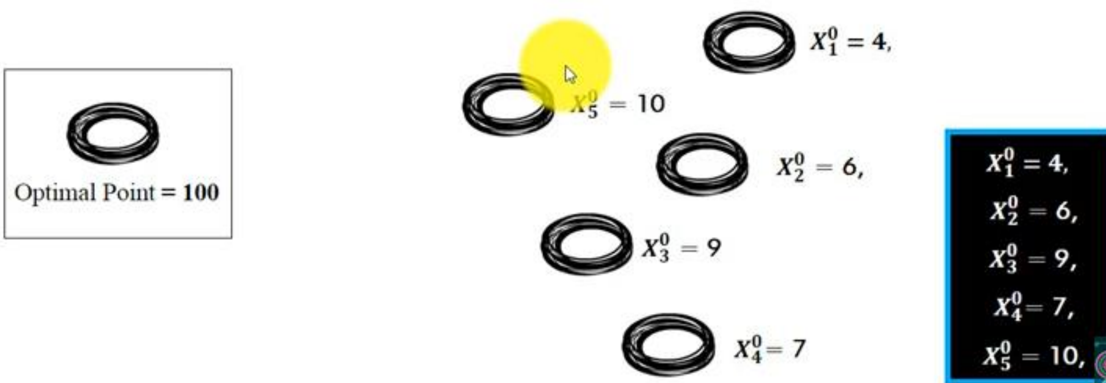


Figure 3: Search paths of nests using Cuckoo Search. The final locations of the nests are marked with ⋄ in the figure.

CUCKOO SEARCH ALGORITHM

- **Step 1:** Initialize Population of total Nest : [n =5]
- Randomly Generated Position of each Host Nest.
- $X_1^0, X_2^0, X_3^0, X_4^0, X_5^0$.



CUCKOO SEARCH ALGORITHM

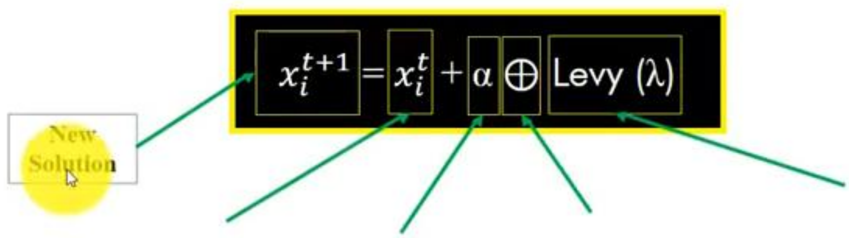
- **Step 1:** Initialize Population of total Nest : [n =5]
- **Step 2:** Probability of Cuckoo egg discovery [$P_a = 0.25$]
- **Step 3:** Maximum number of iteration: [Maxt = 300]
- **Step 4:** Obtain a new position of ith cuckoo randomly by Levy's Flight.
- Get a cuckoo (**say i = 1**) randomly by Levy's Flight.
- Levy Flight is Performed as:

$$x_i^{t+1} = x_i^t + \alpha \oplus \text{Levy}(\lambda)$$

$X_1^0 = 4,$
$X_2^0 = 6,$
$X_3^0 = 9,$
$X_4^0 = 7,$
$X_5^0 = 10,$

CUCKOO SEARCH ALGORITHM

- Calculation for Levy's Flight
- One important Feature of CS algorithm is use of Levy's Flight to generate New Solution.
- $x_i^{t+1} = x_i^t + \alpha \oplus \text{Levy}(\lambda)$



CUCKOO SEARCH ALGORITHM

- *Calculation for Levy's Flight* - **Levy's Flight** Provide a Random Walk.
- $x_i^{t+1} = x_i^t + \alpha \oplus \text{Levy}(\lambda)$
- **Random Steps** can be drawn from a Levy's Distribution for large steps:

  $$\text{Levy} \sim u = t^{-\lambda} \ (1 < \lambda \leq 3)$$

- Which has an infinite variance with an infinite mean. Here, the consecutive jumps/steps of a cuckoo essentially from a random walk process which obeys a power-law step length distribution. **The Step Size can be Expressed as:**

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

- Here, v is normal stochastic variable.

CUCKOO SEARCH ALGORITHM

- The Step Size can be Expressed as:

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

- Here, u and v obey normal distribution and calculated by considering normal distribution:

If (s = Too Large)
Than – New Solution generated will be too far from old solution.
If (s = small)
Than – Change in position will be too small.

NOTE: PROPER STEP SIZE (S) IS IMPORTANT THE SEARCH SPACE.

CUCKOO SEARCH ALGORITHM

- The Step Size can be Expressed as:

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

- Here, u and v obey normal distribution and calculated by considering normal distribution:
- $u \sim N(0, \sigma_u^2)$
- $v \sim N(0, \sigma_v^2)$ and $\sigma_v = 1$.

$$\sigma_u = \left(\frac{\Gamma(1+\beta) \cdot \sin(\pi \cdot \beta/2)}{\Gamma((1+\beta)/2) \cdot \beta \cdot 2^{(\beta-1)/2}} \right)^{1/\beta},$$

Put the value of β as:
$\beta = \frac{3}{2}$

- $\boxed{\sigma_u = 0.6966}$

CUCKOO SEARCH ALGORITHM

- Put the values in the above equation and calculate the step size:
- The Step Size can be Expressed as:

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

$\sigma_u = 0.6966$
$\sigma_v = 1$
$\beta = \frac{3}{2}$

- Step size Calculated: **s = 0.338026236696**

- Here, s is the step size determine how far a random walker can go for fixed number of iteration.
- In general, Random Walk is a Markov Chain whose next location depend on current location [the first term in above equation].

Step Size $(s) = \mathbf{0.338026236696}$

CUCKOO SEARCH ALGORITHM

- *STEP 4: Levy's Flight generally use Random walk strategy as:*
- $x_i^{t+1} = x_i^t + \alpha * s \oplus (x_i^t - x_{best})$
- z_i = Change of Position
- And $z_i = \alpha * s \oplus (x_i^t - x_{best})$

x_{gbest}^t = Current Global Best position. $x_{gbest}^t = 0$

CUCKOO SEARCH ALGORITHM

- *STEP 4: Levy's Flight generally use Random walk strategy as:*
- $x_i^{t+1} = x_i^t + \alpha * s \oplus (x_i^t - x_{best})$
- z_i = Change of Position
- And $z_i = \alpha * s \oplus (x_i^t - x_{best})$
- Finally, the candidate solution is Calculated as:

$$\boxed{x_i^{t+1} = x_i^t + z_i}$$

$Step\ Size\ (s) = 0.338026236696$

$\alpha = 1$

CUCKOO SEARCH ALGORITHM

- *STEP 4:* Generate new solution for cuckoo (i = 1) using Levy's Flight
- $X_1^0 = 4$,
- Levy Flight is Performed as:
- $x_i^{t+1} = x_i^t + \alpha * s \oplus (x_i^t - x_{best})$
- *Set iteration counter (t = 0)*
- $x_i^{t+1} = x_i^t + \alpha * s \oplus (x_i^t - x_{best})$
- $x_1^{0+1} = x_1^0 + 1*0.338026236696 \oplus (x_1^0 - 0)$
- $x_1^1 = 4 + 0.338026236696 \oplus (4 - 0)$
- $x_1^1 = 5.35208$

$X_1^0 = 4$,
$X_2^0 = 6$,
$X_3^0 = 9$,
$X_4^0 = 7$,
$X_5^0 = 10$,

CUCKOO SEARCH ALGORITHM

- *STEP 5:* Choose a nest n (say j) Randomly.
- Check $f(x_i) \geq f(x_j)$
- Randomly Selected Nest (n= 2) $X_2^0 = 6$
- 5.352 ≥ 6 [Condition false]
- Cuckoo Egg is not similar to Host Egg.

$X_1^0 = 4,$
$X_2^0 = 6,$
$X_3^0 = 9,$
$X_4^0 = 7,$
$X_5^0 = 10,$

CUCKOO SEARCH ALGORITHM

- *STEP 5:* Choose a nest n (say j) Randomly. Check $f(x_i) \geq f(x_j)$
- Randomly Selected Nest (n= 2) $X_2^0 = 6$
- 5.352 ≥ 6 [Condition false] – i.e., Worse Case
- Cuckoo Egg is not similar to Host Egg.
- Destroy lowest rank egg with Pa and initialize new egg of the nest.
- Worse nest are replaced by new one.
- $x_i^{t+1} = x_i^t + \alpha * s \oplus (x_i^t - x_{best})$
- $x_1^1 = 5.352 + 1*0.338026236696 \oplus (5.352 - 0)$
- $x_1^1 = 7.1610$

$X_1^0 = 4,$
$X_2^0 = 6,$
$X_3^0 = 9,$
$X_4^0 = 7,$
$X_5^0 = 10,$

CUCKOO SEARCH ALGORITHM

- *STEP 6:* Keep the Best Solution and Increment the counter ($t = t+1$) and repeat until condition met.

Counter (t = 0)	Counter (t = 1)
$X_1^0 = 4,$	$X_1^1 = 7.1610$
$X_2^0 = 6,$	$X_2^0 =$
$X_3^0 = 9,$	$X_3^0 =$
$X_4^0 = 7,$	$X_4^0 =$
$X_5^0 = 10,$	$X_5^0 =$

CUCKOO SEARCH ALGORITHM

- *STEP 4:* Generate new solution for cuckoo (i = 2) using Levy's Flight
- $X_2^0 = 6,$
- Levy Flight is Performed as:
- $\boldsymbol{x_i^{t+1} = x_i^t + \alpha * s \oplus (x_i^t - x_{best})}$
- *Set iteration counter (t = 0)*
- $\boldsymbol{x_i^{t+1} = x_i^t + \alpha * s \oplus (x_i^t - x_{best})}$
- $x_2^{0+1} = x_2^0 + 1*0.338026236696 \oplus (x_2^0 - 0)$
- $x_2^1 = 6 + 0.338026236696 \oplus (6 - 0)$
- $x_2^1 = 8.028$

$X_1^0 = 4,$
$X_2^0 = 6,$
$X_3^0 = 9,$
$X_4^0 = 7,$
$X_5^0 = 10,$

CUCKOO SEARCH ALGORITHM

- ***STEP 5:*** Choose a nest n (say j) Randomly.
- Check $f(x_i) \geq f(x_j)$
- Randomly Selected Nest (n= 4) $X_4^0 = 7$
- 8.028 ≥ 7 [Condition True]
- Cuckoo Eggs are Similar to Host Bird Eggs.
- Replace j [X_4^0] by new solution [x_2^1].
- Destroy lowest rank egg with Pa and initialize new egg of the nest.

$X_1^0 = 4,$
$X_2^0 = 6,$
$X_3^0 = 9,$
$X_4^0 = 7,$
$X_5^0 = 10,$

$X_1^0 = 4,$
$X_2^0 = 6,$
$X_3^0 = 8.028,$
$X_4^0 = 7,$
$X_5^0 = 10,$

CUCKOO SEARCH ALGORITHM

- ***STEP 6: Build New Solution***
- $x_i^{t+1} = x_i^t + \alpha * s \oplus (x_i^t - x_{best})$

New solution = Old Position + Levy's flight

- $x_2^1 = 8.028 + 0.338026236696 \oplus (8.028 - 0)$
- $x_2^1 = 10.741$

$X_1^0 = 4,$
$X_2^0 = 6,$
$X_3^0 = 9,$
$X_4^0 = 7,$
$X_5^0 = 10,$

$X_1^0 = 4,$
$X_2^0 = 6,$
$X_3^0 = 8.028,$
$X_4^0 = 7,$
$X_5^0 = 10,$

CUCKOO SEARCH ALGORITHM

- *STEP 6:* Keep the Best Solution and Increment the counter (t = t+1) and repeat until condition met.

Counter (t = 0)	Values Updated	Counter (t = 1)
$X_1^0 = 4,$	$X_1^0 = 4,$	$X_1^1 = 7.1610$
$X_2^0 = 6,$	$X_2^0 = 6,$	$X_2^1 = 10.741$
$X_3^0 = 9,$	$X_3^0 = 8.028,$	$X_3^1 =$
$X_4^0 = 7,$	$X_4^0 = 7,$	$X_4^1 =$
$X_5^0 = 10,$	$X_5^0 = 10,$	$X_5^1 =$

CUCKOO SEARCH ALGORITHM

- *STEP 4:* Generate new solution for cuckoo (i = 3) using Levy's Flight
- $X_3^0 = 8.028,$
- Levy Flight is Performed as:
- $x_i^{t+1} = x_i^t + \alpha * s \oplus (x_i^t - x_{best})$
- *Set iteration counter (t = 0)*
- $x_i^{t+1} = x_i^t + \alpha * s \oplus (x_i^t - x_{best})$
- $x_3^{0+1} = x_3^0 + 1*0.338026236696 \oplus (x_3^0 - 0)$
- $x_3^1 = 8.028 + 0.338026236696 \oplus (8.028 - 0)$
- $x_3^1 = 10.741$

Values Updated
$X_1^0 = 4,$
$X_2^0 = 6,$
$X_3^0 = 8.028,$
$X_4^0 = 7,$
$X_5^0 = 10,$

CUCKOO SEARCH ALGORITHM

- Cuckoo Search via Levy's Flight

begin
  Objective function $f(\mathbf{x}),\ \mathbf{x} = (x_1, ..., x_d)^T$
  Generate initial population of
    n host nests $\mathbf{x}_i\ (i = 1, 2, ..., n)$
  while *(t <MaxGeneration) or (stop criterion)*
    Get a cuckoo randomly by Lévy flights
      evaluate its quality/fitness F_i
    Choose a nest among n (say, j) randomly
    if $(F_i > F_j)$,
      replace j by the new solution;
    end
    A fraction (p_a) *of worse nests*
      are abandoned and new ones are built;
    Keep the best solutions
      (or nests with quality solutions);
    Rank the solutions and find the current best
  end while
  Postprocess results and visualization
end

CUCKOO SEARCH ALGORITHM

- Current Best Solution.