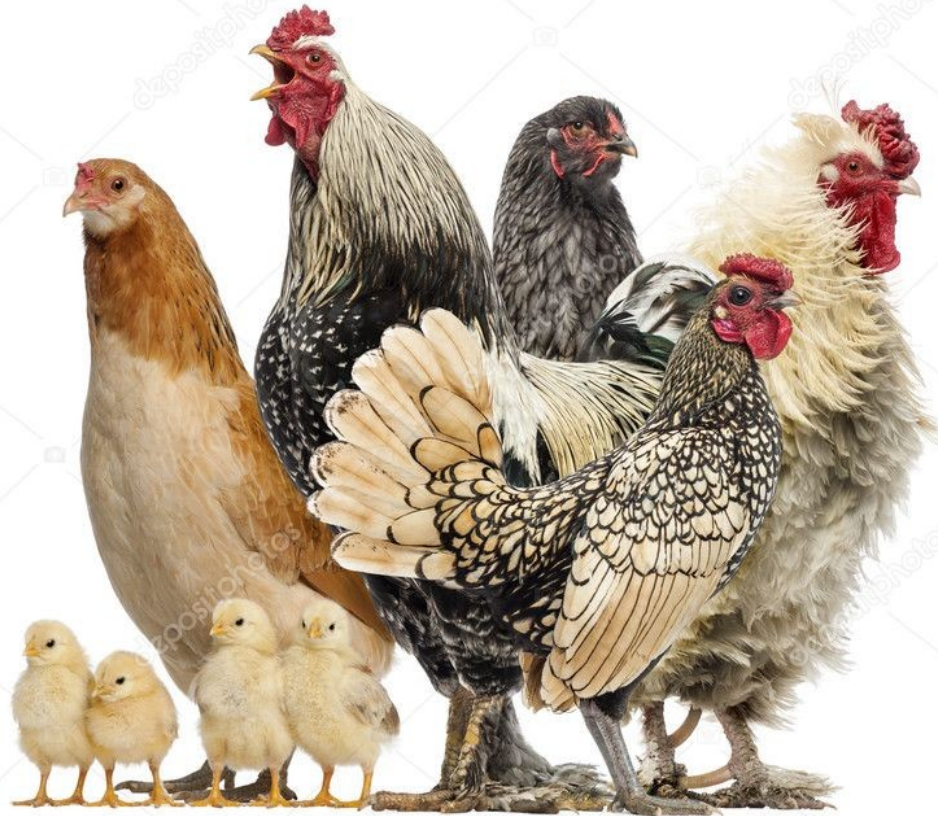




SWARM-BASED ALGORITHMS





ALGORITMO Optimización de enjambre de pollos Chicken Swarm Optimization (CHSO)

INTRODUCCIÓN

- Autores: Meng *et al.* (2014).
- Objetivo: optimización.
- Inspiración: orden jerárquico de los grupos de pollos y comportamiento de gallinas, gallos y pollitos dentro del grupo.

INSPIRACIÓN NATURAL

- Se imita la relación jerárquica que se establece entre gallos, gallinas y pollitos de un grupo.
- Se simula la búsqueda de alimento.
- Se imita el comportamiento de cada tipo de individuo.

INSPIRACIÓN NATURAL

Los individuos siguen al gallo de su grupo para buscar comida, mientras que pueden evitar que los demás coman su propia comida.

Los individuos roban al azar la buena comida que otros ya encontraron.

Los pollitos buscan comida alrededor de su madre.

Los individuos dominantes tienen ventaja en la competencia por la comida.

LOS ELEMENTOS DE LA SOLUCIÓN

- Se considera una población de P individuos.
- Cada individuo está representado por su posición, que varía con el tiempo:

individuo i en el instante t :

posición: $x_i(t) = (x_{i1}, \dots, x_{ir})$

calidad: fitness_i

LOS ELEMENTOS DE LA SOLUCIÓN

- Solución al problema: mejor posición global.

Posición con mejor fitness de las ocupadas por el grupo durante la ejecución del algoritmo

$g(t) \rightarrow$ mejor solución encontrada hasta el instante t

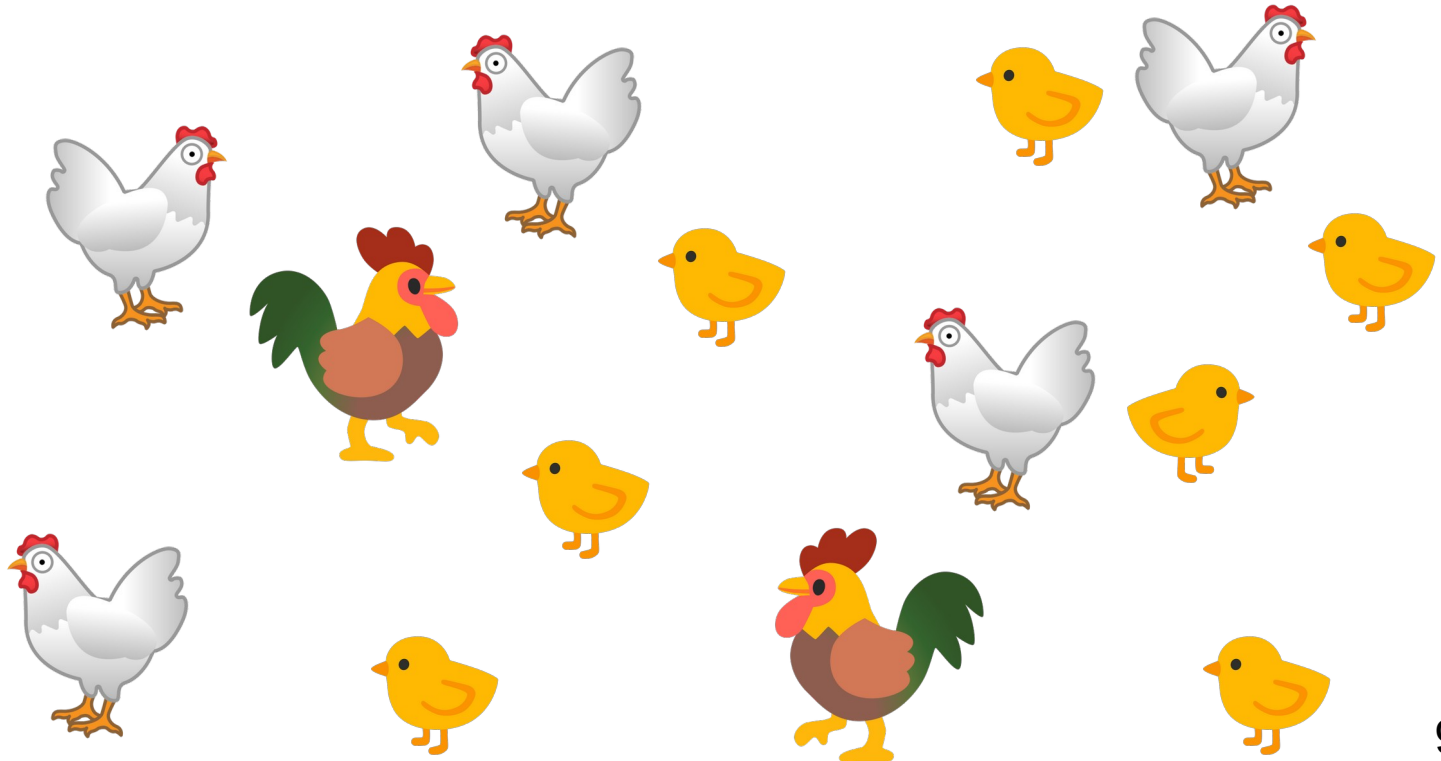
LOS ELEMENTOS DE LA SOLUCIÓN

Todos los individuos buscan comida en un espacio r -dimensional (espacio de solución del problema a resolver).

Los mejores individuos son los que ocupan posiciones con mejor fitness.

LOS ELEMENTOS DE LA SOLUCIÓN

- La población está formada por 3 tipos de individuos: gallos, gallinas y pollitos.



LOS ELEMENTOS DE LA SOLUCIÓN

- **gallos:** los individuos con mejor fitness. Cada uno será el líder de un grupo.



- **pollitos:** los individuos con peor fitness.



- **gallinas:** el resto de individuos.



LOS ELEMENTOS DE LA SOLUCIÓN

N
individuos

RN: número de gallos



CN: número de pollitos



HN: número de gallinas

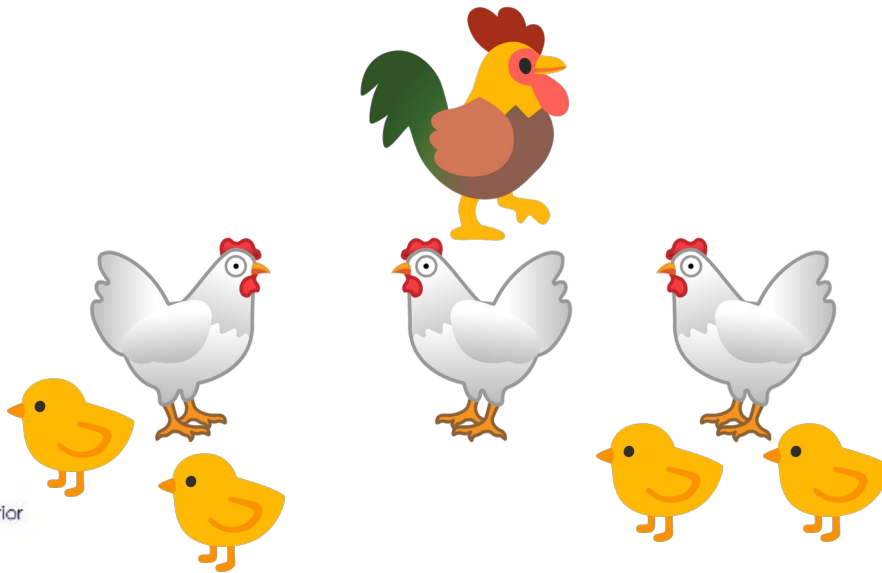


MN: número de gallinas madre

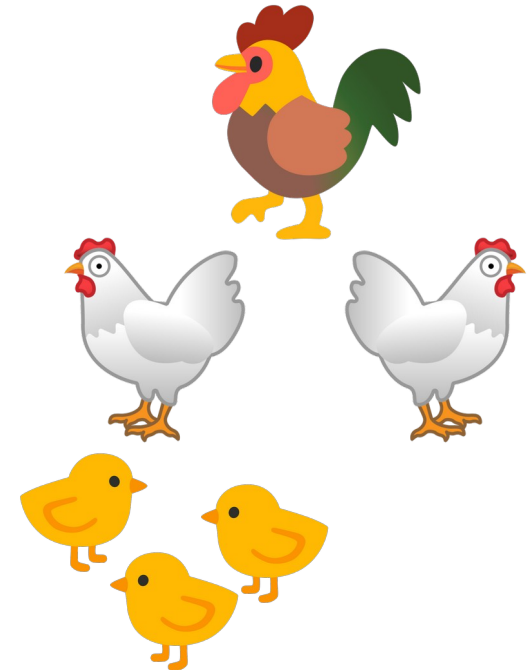
- Debe haber más gallinas que gallos ($HN > RN$).
- Debe haber más gallinas que madres ($HN > MN$).
- La población de animales adultos superará a la de los pollitos ($RN + HN > CN$).

LOS ELEMENTOS DE LA SOLUCIÓN

- Se identifican varios grupos, cada uno de los cuales está compuesto por un gallo (*rooster*), varias gallinas (*hens*) y varios pollitos (*chicks*).



GRUPO 1



GRUPO 2

LOS ELEMENTOS DE LA SOLUCIÓN

GALLO



Lidera un grupo de individuos

GALLINAS



Deciden al azar en qué grupo vivir

POLLITOS



La relación madre-hijo entre los pollitos y las gallinas se establece al azar.

LOS ELEMENTOS DE LA SOLUCIÓN

El orden jerárquico, la relación de dominio y la relación madre-hijo en un grupo permanecerán sin cambios. Estos estados sólo se actualizan cada varios pasos de tiempo, determinados por el parámetro **G** del algoritmo.

ALGORITMO CHSO

Inicializar la población de P individuos
Calcular el fitness de cada individuo
Evaluar relación actual

REPETIR para t desde 1 a TMAX

SI ($t \% G = 0$)

 Evaluar relación actual

FIN-SI

 Actualizar la posición de cada individuo

 Actualizar la mejor solución global

FIN-REPETIR

ALGORITMO CSO

A- Inicializar la población de individuos

Se establece la posición inicial de todos los individuos, $x_i(0)$, para $i=1, \dots, P$.

Posición inicial:
valores aleatorios del
espacio de búsqueda

ALGORITMO CHSO

B- Calcular el fitness de cada individuo

Para calcularlo se aplica la función objetivo del problema a la posición de la partícula:

$$\text{fitn}_i \approx f(x_i)$$

(se puede tomar directamente este resultado o usarlo para calcular un valor final).

El fitness asociado a la posición del individuo i determina la calidad de la solución que representa.

ALGORITMO CHSO

C- Evaluar relación actual

Ordenar los individuos por fitness
Determinar el tipo de cada individuo
Determinar la relación entre gallinas y gallos
Determinar la relación entre los pollitos y las madres

ALGORITMO CSO

C- Evaluar relación actual

Determinar el tipo de cada individuo



Si aplico porcentajes {

- 10% son gallos $\Rightarrow 15 \times 0,1 = 1,5 \Rightarrow 2$ gallos
- 60% son gallinas $\Rightarrow 15 \times 0,6 = 9$
- El resto son pollitos $\Rightarrow 15 - (2 + 9) = 4$

ALGORITMO CHSO

C- Evaluar relación actual

Determinar el tipo de cada individuo



Si tengo valores
asociados al número de
individuos de cada tipo

$$\begin{cases}
 RN = 3 \\
 HN = 7 \\
 CN = 15 - (3 + 7) = 5
 \end{cases}$$

ALGORITMO CHSO

C- Evaluar relación actual
Determinar la relación entre gallinas y gallos

Para cada gallina se elige aleatoriamente uno de los gallos.

La gallina se asocia al grupo que encabeza ese gallo.

ALGORITMO CHSO

C- Evaluar relación actual

Determinar la relación entre los pollitos y las madres

Se eligen aleatoriamente **MN** gallinas como madres.

Cada pollito elige aleatoriamente entre dichas gallinas.

Cada pollito queda asociado al grupo al que pertenece su madre

ALGORITMO CHSO

C- Evaluar relación actual

La relación existente entre los individuos del grupo se actualiza periódicamente, en función del parámetro **G**.

G debe fijarse en función del problema a resolver.

- muy grande, puede que el algoritmo no converja al óptimo global rápidamente.
- muy pequeño, el algoritmo puede quedar atrapado en el óptimo local.

Un valor en $[2, 20]$ genera buenos resultados para muchos problemas.

ALGORITMO CSO

D- Actualizar la posición de cada individuo

PARA cada individuo i

SI es un gallo

Calcular posición x_i' para gallo i

SINO SI es gallina

Calcular posición x_i' gallina i

SINO

Calcular posición x_i' pollito i

FIN-SI

Actualizar la posición del individuo i

FIN-REPETIR

ALGORITMO CSO



D-1- Calcular posición de gallo

Los gallos con mejores valores de fitness tienen más prioridad para el acceso al alimento que los que tienen peores valores de fitness.

Para simplificar, este caso se puede simular considerando que los gallos con mejor fitness pueden buscar alimento en una zona más amplia del espacio de solución



ALGORITMO CSO

D1- Calcular posición de gallo i

$$x_i' = x_i(t) (1 + GAUSS(0, \sigma^2)) \quad (1)$$

$$\sigma^2 = \begin{cases} 1 & \text{si } fit_i < fit_k \\ \exp\left(\frac{fit_k - fit_i}{|fit_i| + \varepsilon}\right) & \text{en otro caso} \end{cases} \quad (2)$$

Parámetros:

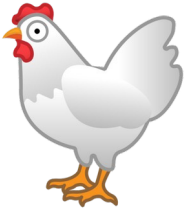
GAUSS(): valor de una distribución gaussiana de media 0 y desviación típica σ^2

épsilon: constante muy pequeña (para evitar un error de división por cero)

fit_j: fitness del gallo j

k: un gallo aleatorio del grupo de gallos ($k \neq i$)

ALGORITMO CSO



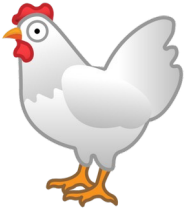
D-2- Calcular posición de gallina

Pueden seguir al gallo de su grupo para buscar comida.

Además, también robarían al azar la comida que encontraran otros pollos, aunque serían reprimidas por los otros pollos.

Las gallinas más dominantes tendrían ventaja en competir por la comida que las más sumisas.

ALGORITMO CSO



D-2- Calcular posición de gallina i

$$x_i' = x_i(t) + s1 \text{Rand}(x_{r1}(t) - x_i(t)) + s2 \text{Rand}(x_{r2}(t) - x_i(t))$$

(3)

$$s1 = \exp\left(\frac{fit_i - fit_{r1}}{|fit_i| + \varepsilon}\right) \quad (4)$$

$$s2 = \exp(fit_{r2} - fit_i) \quad (5)$$

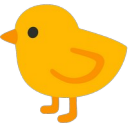
Parámetros:

Rand: valor de una distribución uniforme en el intervalo [0, 1]

épsilon: constante muy pequeña (para evitar un error de división por cero)

r1: índice del gallo del grupo al que pertenece la gallina

r2: índice de un gallo o una gallina (distinto de r1) tomado al azar



ALGORITMO CSO

D-3- Calcular posición de pollito i

Se mueven alrededor de su madre para buscar comida.

$$x_i' = x_i(t) + FL(x_m(t) - x_i(t)) \quad (6)$$

Parámetros:

x_m : posición de la madre del pollito i

FL: parámetro en (0, 0.2)

m: índice de la gallina madre del pollito

ALGORITMO PSO

D-4- Actualizar la posición del individuo i

Una vez calculada una nueva posición x' para un individuo, se calcula su fitness, $\text{fitness}(x')$.

Si la nueva posición calculada para un individuo es mejor que la actual, se toma como nueva posición actual.

Calcular fitness de la nueva solución x' , $\text{fitness}(x')$

SI x' es mejor que $x(t)$

$$x(t+1) = x'$$

SINO

$$x(t+1) = x(t)$$

FIN-SI

ALGORITMO PSO

E- Actualizar la mejor solución global

Si se ha encontrado una solución que mejora a la considerada mejor hasta el momento, guardarla como nueva mejor solución global.

Dar valor a $g(t+1)$

Referencias

- Meng, X., Liu, Y., Gao, X., Zhang, H. (2015). *A new bio-inspired algorithm: chicken swarm optimization*. In International Conference in Swarm Intelligence (pp. 86-94), Springer, Cham