

# Estaciones satelitales

Sistemas Inteligentes

---

Aarón Arias Pérez

Carlos Gallardo Polanco

11 de diciembre de 2017

Escuela Superior de Ingeniería  
Universidad de Cádiz

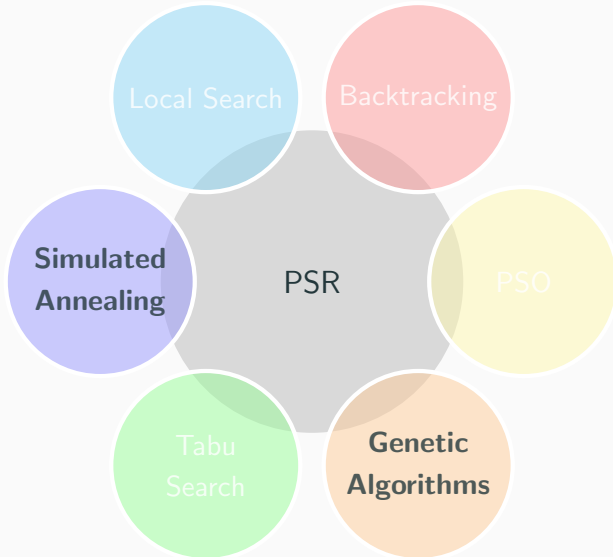
**¿Qué estrategias hemos elegido?**

---

# Estrategias elegidas



# Estrategias elegidas



# Simulated Annealing

---

# Simulated Annealing

```
1  function [best,value] =  
    ↪ simulatedAnnealing(N,M,coordinates,T,T_limit,pcool)  
2      % T: Temperatura inicial  
3      % T_limit: Temperatura mínima  
4      current=randperm(N,M) % Representantes iniciales  
5      best=current; % La mejor es la inicial  
6      while (T>T_limit)  
7          RepToChange=current(randi(M));  
8          % Elegimos un representante aleatorio a cambiar  
9          new =  
            ↪ random_sucessor(coordinates,current,RepToChange);  
10         % Nuevo conjunto de representantes aleatorio
```

```

11     deltaE = Fvalue(new,coordinates) -
        ↪ Fvalue(current,coordinates);
12     if(deltaE<0) % Si la solución es mejor, se selecciona
13         current = new;
14         best = current;
15     else
16         if(exp(-deltaE/T)>rand(0,1)) % Acepta la solución de
            ↪ forma aleatoria
17             current = new;
18         end
19     end
20     T=cool(T,pcool);
21 end
22 value=Fvalue(best,coordinates);
23 end

```

# Algoritmos Genéticos

---



# Algoritmos Genéticos

```
1  function [Pop,FitPop] = geneticAlgorithm(Pop, FitPop,  
    ↪ coordinates, MAX_itera, Pcross, Pmut, cross, sel, rep)  
2      itera=1;  
3      while itera<=MAX_itera  
4          % RouletteWheel or Tournament  
5          parents = selection(FitPop,sel); % Selecciona los  
    ↪ padres  
6          couples = match(parents,Pcross); % Empareja los padres
```

```

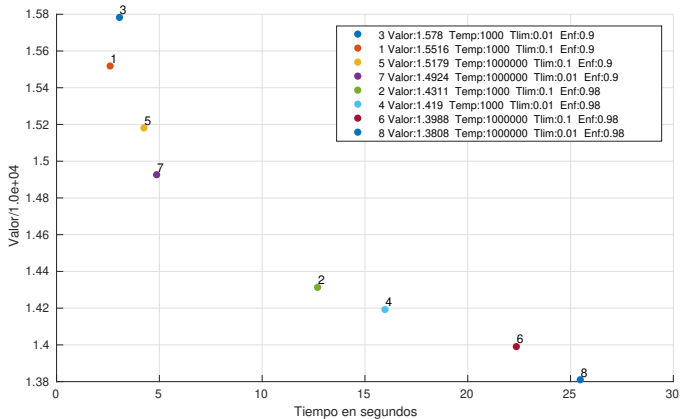
7      if ~isempty(couples)
8          % OX or PMX
9          newPop = crossover(couples,Pop,cross); % Poblacion
           ↪ cruzada
10     else
11         newPop = Pop;
12     end
13     newPop = mutation(newPop,Pmut,size(coordinates,1)); %
           ↪ Población con mutaciones
14     newFitPop = evaluatePopulation(newPop,coordinates);
15     % generational or elitist
16     [Pop,FitPop] =
           ↪ replace(Pop,newPop,FitPop,newFitPop,rep);
17     itera=itera+1;
18 end
19 end

```

# Comparativa SA y GA

---

# Simulated Annealing. Pruebas



**Figura 1:** Pruebas SA

# Algoritmos Genéticos. Pruebas.

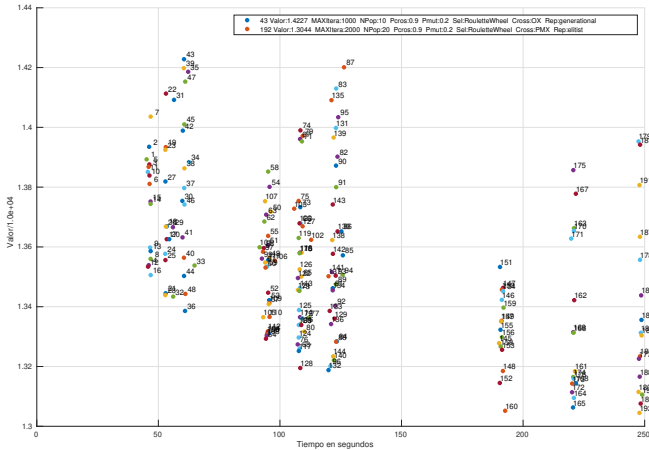
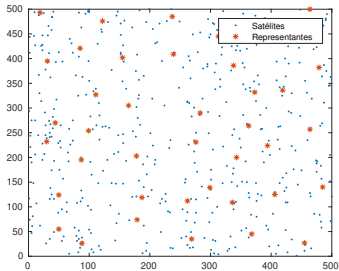
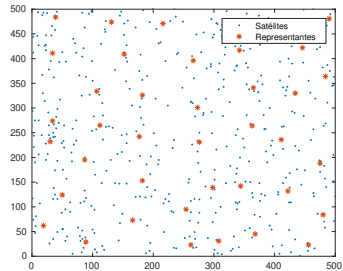


Figura 2: Pruebas GA

# Mapas



**(a)** SA. Valor:  $1.3808e+04$



**(b)** GA. Valor:  $1.3044e+04$

**Figura 3:** Mapas de los satélites

¿Preguntas?