Natural Computing

ffgt86

February 4, 2020

1 Relevant background material

The Artificial Bee Colony (ABC) algorithm is an optimisation algorithm based on the foraging behaviour of the honeybee swarm.

2 Pseudocode

Algorithm 1 My algorithm

```
procedure ABC
    S_n: number of food sources
    limit: number of trials before abandoning a food source
    generations: number of generations
    C_max: number of cycles
begin:
    //Initialisation
    num_eval ← 0
    for s ← 1 to S_N do
        X(s) ← random solution
        f_s ← f(X(s))
        trial(s) ← 0
        num_eval + +
    end for
    //Employed bees phase
    while c < C_max do
        for s ← 1 to S_N do
            x' ← a new solution
        end for
    end while
end procedure
```

3 Natural language description

4 Details of experiments

The paper used 5 classic optimisation benchmark functions, sourced from [**?**] to test the performance of ABC against PSO, PS-EA, and GA:

- Griewank: the Griewank function has a product term that introduce interdependence between variables. Algorithms that seek to optimise each variable independently will therefore do poorly.

- Rastrigin: the Rastrigin function produces many local, regularly distributed minima, so that an optimisation algorithm can be easily trapped.

- Rosenbrock: the Rosenbrock function produces a global minimum inside a long, narrow, parabolic valley. Converging is difficult. It is used to test the performance of an algorithm.

- Ackley: the Ackley function tests whether an algorithm effectively combines exploration and exploitation

- Schwefel: the Schwefel function produce a complex map, with a second-best minimum far from the global minimum, which itself is close to the bounds of the domain.

5 Overview of results