Genetic Algorithms

by Thomas Christensen & Emil Frost

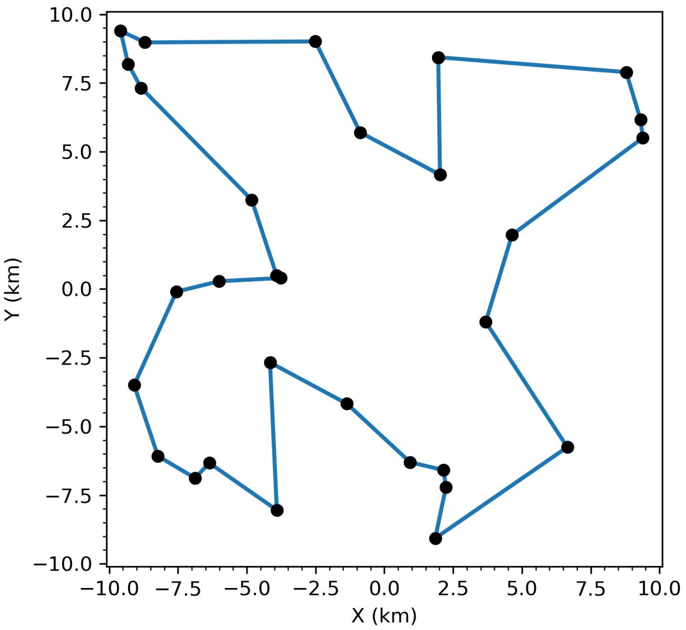
P. Larrañaga, C. Kuijpers, R. Murga, I. Inza, and S. Dizdarevic, *Genetic algorithms for the travelling salesman problem: A review of representations and operators*, Artificial Intelligence Review **13**, 129–170 (1999)

Abstract genetic algorithm

BEGIN AGA
    Make initial population at random.
    WHILE NOT stop **DO**
        BEGIN
            Select parents from the population.
            Produce children from the selected parents.
            Mutate the individuals.
            Extend the population adding the children to it.
            Reduce the extend population.
        END
    Output the best individual found.
END AGA

The Travelling Salesman Problem (TSP)

Given a list of *N* cities on a 2-dimensional grid, find the connected route that minimizes the traversed distance and visits each city exactly once and returns to the starting point.



NB: (N-1)!/2 individual routes

Representations and Operators

Path representation is the most common

- Label cities by numbers: 1, 2, 3, …, N
- List cities in order of visitation, e.g. (1, 3, 2, 4) for the tour 1 -> 3 -> 2 -> 4 -> 1.
- NB: valid tours contain all digits 1, 2, …, N exactly once.

Crossover & mutation operators define the breeding and mutation in the population.

- Crossover: parent + parent -> child
- Mutation: individual -> individual

Edge Recombination Crossover (ER)

Consider parent tours: (1, 3, 5, 6, 4, 2, 8, 7)
(1, 4, 2, 3, 6, 5, 7, 8)

city 1 has edges to : 3 4 7 8
city 2 has edges to : 3 4 8
city 3 has edges to : 1 2 5 6
city 4 has edges to : 1 2 6
city 5 has edges to : 3 6 7
city 6 has edges to : 3 4 5
city 7 has edges to : 1 5 8
city 8 has edges to : 1 2 7

Figure 16. The edge map.

City 1 is selected

(a)
city 2 has edges to : 3 4 8
city 3 has edges to : 2 5 6
city 4 has edges to : 2 6
city 5 has edges to : 3 6 7
city 6 has edges to : 3 4 5
city 7 has edges to : 5 8
city 8 has edges to : 2 7

City 8 is selected

(b)
city 2 has edges to : 3 4
city 3 has edges to : 2 5 6
city 4 has edges to : 2 6
city 5 has edges to : 3 6 7
city 6 has edges to : 3 4 5
city 7 has edges to : 5

City 7 is selected

(c)
city 2 has edges to : 3 4
city 3 has edges to : 2 5 6
city 4 has edges to : 2 6
city 5 has edges to : 3 6
city 6 has edges to : 3 4 5

City 5 is selected

(d)
city 2 has edges to : 3 4
city 3 has edges to : 2 6
city 4 has edges to : 2 6
city 6 has edges to : 3 4

City 6 is selected

(e)
city 2 has edges to : 3 4
city 3 has edges to : 2
city 4 has edges to : 2

City 4 is selected

(f)
city 2 has edges to : 3
city 3 has edges to : 2

City 2 is selected

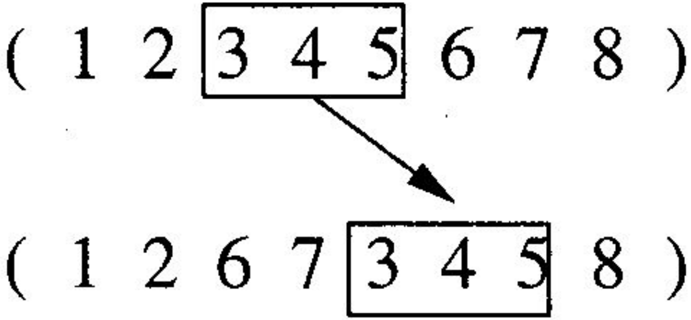
(g)
city 3 has edges to :

City 3 is selected

Potvin, JY. *Genetic algorithms for the traveling salesman problem*. *Ann Oper Res* **63**, 337–370 (1996).

Displacement Mutation (DM)

Consider following tour: (1, 2, 3, 4, 5, 6, 7, 8)



Select random subtour: (3, 4, 5)

Remaining tour: (1, 2, 6, 7, 8)

Insert subtour (3, 4, 5) at random spot in remaining tour: (1, 2, 6, 7, 3, 4, 5, 8)

Our Algorithm

Generate map (*N* random x,y-coordinates as cities).

Generate *n* random tours between cities (Initial population).

Determine fitness value i.e. quality of route (Inverse distance)

Select all routes to mating pool and breed parent routes pairwise using ER crossover

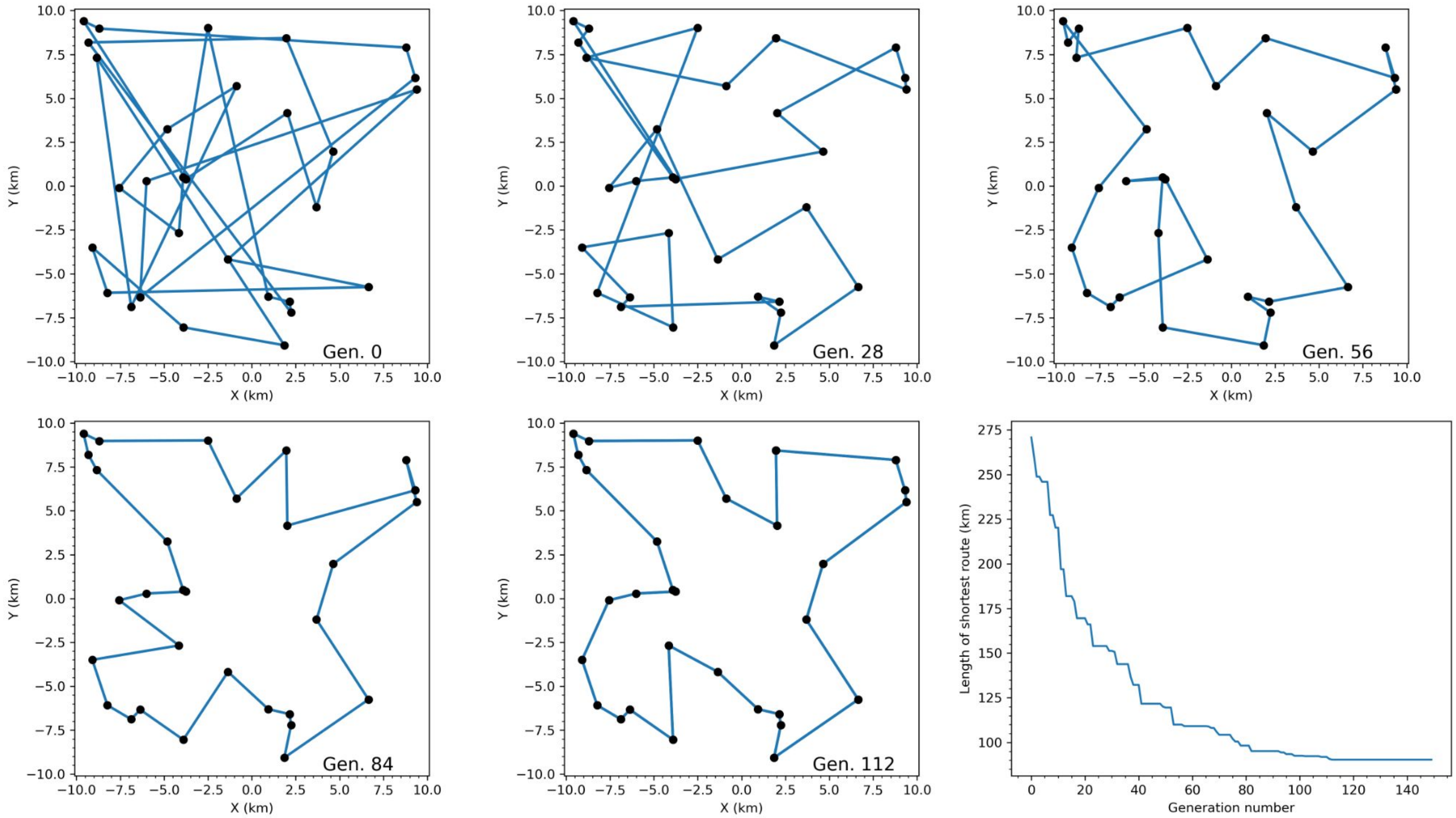
Children are added to the initial population and *n* shortest routes are kept.

Individuals mutated using DM mutation with probability *m*

Repeat

Terminate after fixed number of generations

N = 30,   n = 100,   m = 0.01



Some applications of GAs

Train neural networks (instead of backpropagation or as supplement)

Design neural networks (number of hidden layers, type, sequence, etc.)

Aerodynamic optimization of wing design

…

https://en.wikipedia.org/wiki/List_of_genetic_algorithm_applications

Natural Sciences, Mathematics and Computer Science [edit]

- Bayesian inference links to particle methods in Bayesian statistics and hidden Markov chain models[1][2]
- Artificial creativity
- Chemical kinetics (gas and solid phases)
- Calculation of bound states and local-density approximations
- Code-breaking, using the GA to search large solution spaces of ciphers for the one correct decryption.[3]
- Computer architecture: using GA to find out weak links in approximate computing such as lookahead.
- Configuration applications, particularly physics applications of optimal molecule configurations for particular systems like C_{60} (buckyballs)
- Construction of facial composites of suspects by eyewitnesses in forensic science.[4]
- Data Center/Server Farm.[5]
- Distributed computer network topologies
- Electronic circuit design, known as evolvable hardware
- Feature selection for Machine Learning[6]
- Feynman-Kac models [7][8][9]
- File allocation for a distributed system
- Filtering and signal processing [10][11]
- Finding hardware bugs.[12][13]
- Game theory equilibrium resolution
- Genetic Algorithm for Rule Set Production
- Scheduling applications, including job-shop scheduling and scheduling in printed circuit board assembly.[14] The objective being to schedule jobs in a sequence-dependent or non-sequence-dependent setup environment in order to maximize the volume of production while minimizing penalties