Solving Substitution Ciphers with Genetic Algorithms

Ladislav Šulák (laco.sulak@gmail.com)
Krisztian Benko (kristianbnk@gmail.com)

Home university: Brno University of Technology, Faculty of Information Technology (Czech Republic, but we both are from Slovak Republic)

Key Lengths, Initialization

* Key Lengths

- possible lengths are chosen

– based on repeating part of substrings in encrypted text

* Initialization

- number of members in population is defined at the start

- population is created according to algorithm’s properties

Roulette wheel

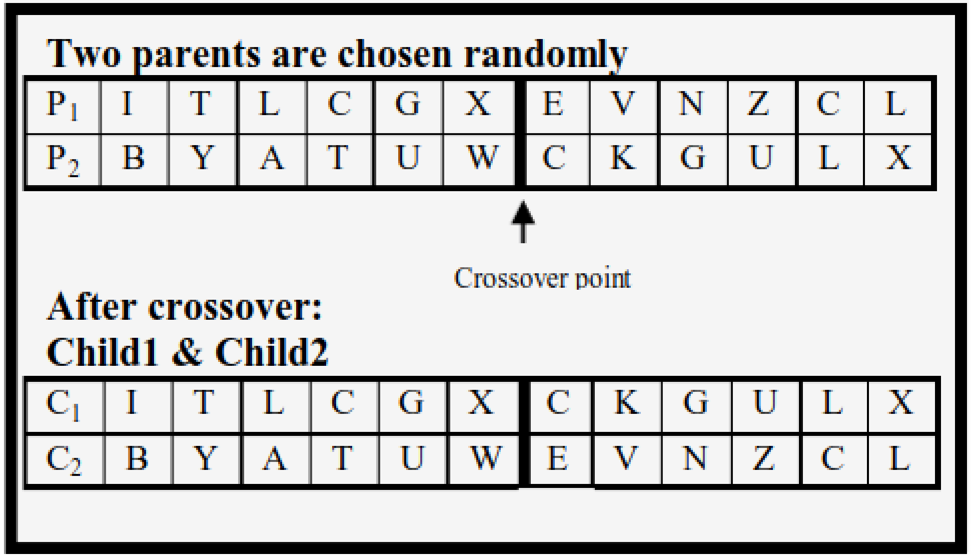
* After the fitness value is calculated for every key, parent pairs are made by roulette wheel, which decide according to fitness value, which keys will be chosen

* Some parents are used to be chosen more than once

Crossover

* Two mating chromosomes are being cut by one-point crossover

* Strings were converted to binary representation for better results



Two parents are chosen randomly

P_1	I	T	L	C	G	X	E	V	N	Z	C	L
P_2	B	Y	A	T	U	W	C	K	G	U	L	X

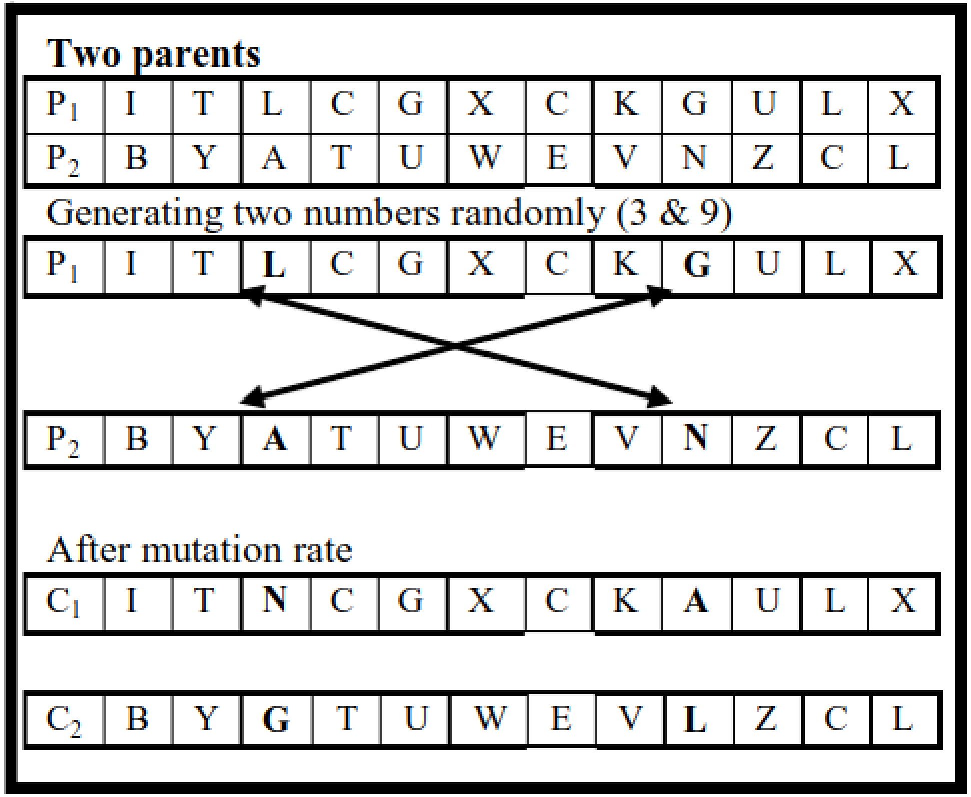
Crossover point

After crossover:
Child1 & Child2

C_1	I	T	L	C	G	X	C	K	G	U	L	X
C_2	B	Y	A	T	U	W	E	V	N	Z	C	L

Mutation

* **Prevention of being trapped in local minimum**



Two parents

P_1	I	T	L	C	G	X	C	K	G	U	L	X
P_2	B	Y	A	T	U	W	E	V	N	Z	C	L

Generating two numbers randomly (3 & 9)

P_1	I	T	**L**	C	G	X	C	K	**G**	U	L	X

P_2	B	Y	**A**	T	U	W	E	V	**N**	Z	C	L

After mutation rate

C_1	I	T	**N**	C	G	X	C	K	**A**	U	L	X

C_2	B	Y	**G**	T	U	W	E	V	**L**	Z	C	L

Fitness function

* **Fitness value of each member (key) is given by:**

- PyEnchant library - check if word belongs to English language (can be extended)

- Markov chain model classifier - if PyEnchant is not successful, this model is used, which is based on using trigrams (can be extended to digrams)

https://github.com/exp0se/dga_detector

Results

Time	6 min	22 min	33 min	51 min
Size of Key	3 bytes	60 bytes	7 bytes	10 bytes
Size of Encrypted Text	40 bytes	170 bytes	60 bytes	183 bytes
Size of population	40	40	40	40
Num of generations	160	160	300	300

Improvements

* Roulette wheel – would be maybe better to create less pairs or do not use roulette at all, but tournament selection

* Crossover – could be over more points in string (two or three pointed), so we could get better varations

- or try to use different types of crossover, e.g.: uniform

* Fitness measurement

References

A Cryptanalytic Attack on Vigenère Cipher Using Genetic Algorithm

* www.researchgate.net/publication/261451438_A_cryptanalytic_attack_on_Vigenere_cipher_using_genetic_algorithm

Genetic algorithm implementation is based on:

* materials from Computational Intelligence course from TEI

* http://www.obitko.com/tutorials/genetic-algorithms/ga-basic-description.php

* https://github.com/rodhilton/Geneticrypt

Markov Model classifier (gibberish_detector) is based on:

* https://github.com/exp0se/dga_detector

Other materials:

* [1996] The Applications of Genetic Algorithms in Cryptanalysis by A. J. Bagnall

* [2003] Solving Substitution Ciphers with Genetics Algorithm by Joe Gester

* [2008] Cryptoanalysis using genetic algorithms by P. Bergmann, Karel & Scheidler, Renate & Jacob, Christian

* [2008] Applying Genetic Algorithms for Searching KeySpace of Polyalphabetic Substitution Ciphers
by Ragheb Toemeh and Subbanagounder Arumugam

* [2011] A cryptanalytic attack on Vigenère cipher using genetic algorithm
by Omran, Safaa & Al-Khalid, A.S. & Alsaady, Dalal