

User Manual

OPERON PREDICTOR USING GENETIC ALGORITHM



Operons by GENETIC
ALGORITHM



**Biomedical Informatics Division,
Rajendra Memorial Research Institute for Medical
Sciences (I.C.M.R)
Patna, India.**

Table of Content

1. Introduction to the Tool:
 - a. About
 - b. Requirement
 - c. Installation
2. Using tool for operon prediction
 - a. Input files
 - b. Genetic Parameters
 - c. Fitness function
 - d. Start Prediction
 - e. Output Visualization
3. Algorithm
4. Evaluation
5. Reference

1. INTRODUCTION TO THE TOOL

1.1 What is GAOPP:

GAOPP is standalone GUI tool for operon prediction. It uses unsupervised method Genetic Algorithm for identifying promoters in annotated prokaryotic species. It uses biological features like intergenic distance, Cluster of Gene Ontology and pathway involvement of each gene pair and clusters them in to operons. There are several computational methods are available for this purpose but none of them are GUI based. They need heavy data preparation, also. To meet these requirements GAOPP has been created.

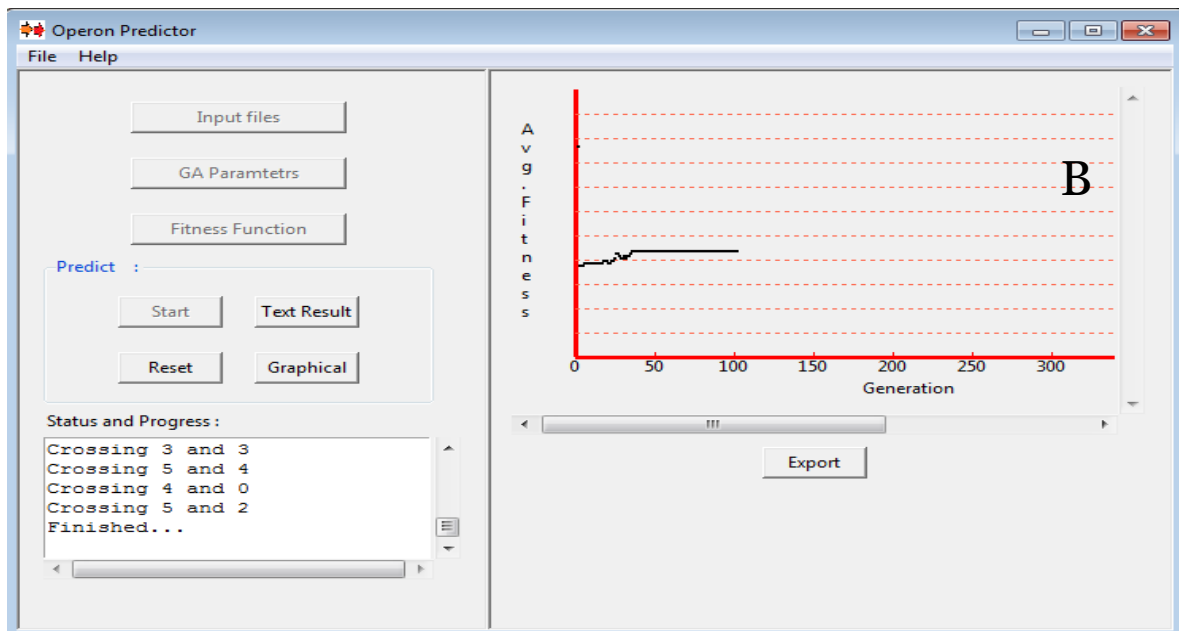
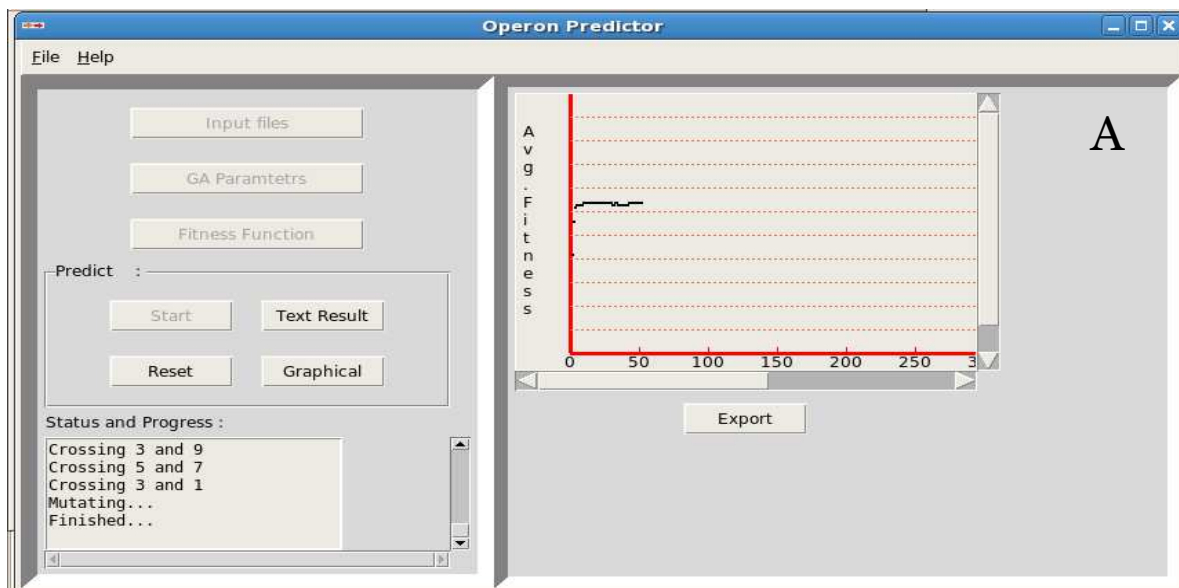
It has three different evaluating functions to evaluate the fitness of each putative operon structure, can be found in literatures. These functions use biological properties like intergenic distance, involvement in metabolic pathway, and functionality from Clusters of Gene ontology (COG) gene functional families. This need needs the protein table file found at *National Centre for Biotechnology Information (NCBI)* FTP (<ftp://ftp.ncbi.nih.gov/genomes/Bacteria/>). For Pathway information *Kyoto Encyclopedia of Genes and Genome (KEGG)* pathway database can be used. A track of experimental promoters in the target species can be used to predict promoters. Terminators can be predicted using *TranTerm* and the output file may be used to provide terminator coordinates in the genome.

1.2 Installation:

1. Download the executables.
2. Place it in a desired destination. Preferably in the C:\GAOPP\ in windows and make a Desktop shortcut.
3. Double click on the binary file or its shortcut to start the program
4. To run it from source code, it requires PERL5.8 above and Tkx module. Active perl can be used instead.

1.3 System Requirement:

1. Operating system Windows 2000/XP/Vista/7 , Linux
2. To run from source code it requires perl5.8 or above and Tkx installed.
3. To run larger genome sequences it may require higher configuration.
4. Additional software like PDF reader and Post Script Viewer may be required.



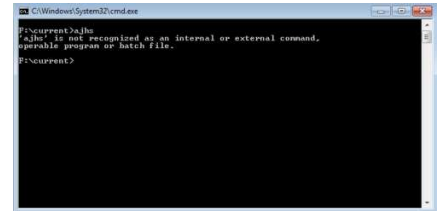
Different feel and look on Linux platform (A) and Windows Platform (B)

.list pathway file



KEGG - Table of Contents

Category	Entry Point	Release Info	Search & Compute	DBGET Search
Systems information	KEGG PATHWAY KEGG BRITE	New maps Update status New hierarchies Update status	Search objects in pathways Color objects in pathways Search objects in Brite view KEGG pathway modules KEGG Orthology (KO)	PATHWAY BRITE MODULE DISEASE
Genomic information	KEGG ORTHOLOGY KEGG GENES	New organisms Update status	SSDB search BLAST search FASTA search Sequence for ESTs KAAS automatic annotation	ORTHODOLOGY GENES GENES GENES / OGENES VIBRATIONS
Chemical information	KEGG LIGAND	Update status	SINCOMP compound search KCAH glycan search e-pyruvate reaction prediction PathComp computation	COMPOUND DRUG GLYCAN REACTION REACTOME ENZYME



TransTerm
Out put

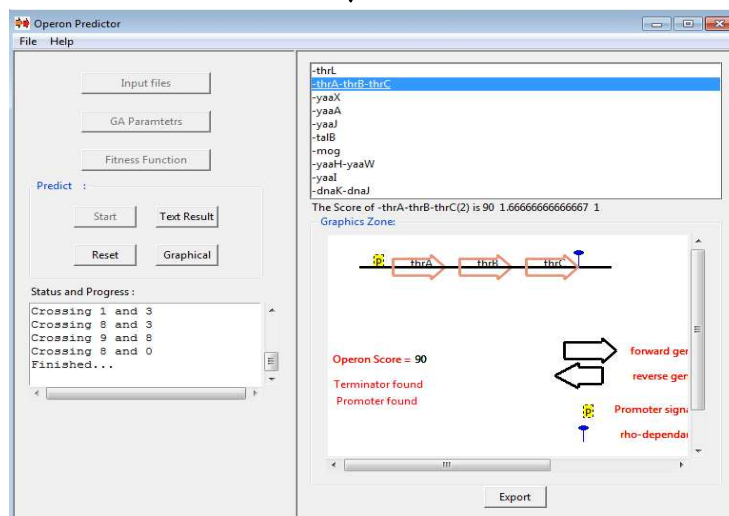
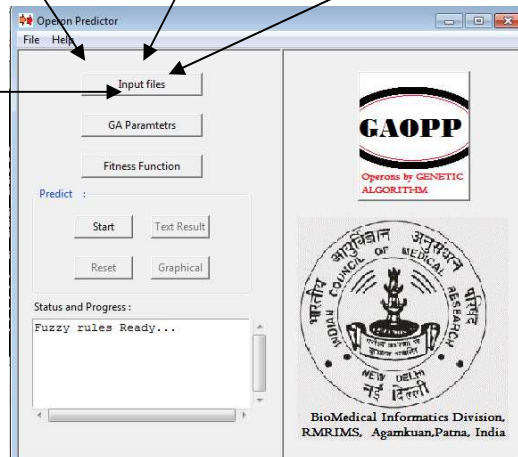
.ptt file from NCBI

Index on <http://ftp.ncbi.nlm.nih.gov/genomes/bacteria/>

Up to higher level directory

Name	Size	Last Modified
Acetobacter_morrisi_M8C11872_uid59157	12/6/2012	12:00:00 AM
Acetobacter_pasteurianus_PO_3281_01_42C_uid58377	4/3/2012	4:00:00 AM
Acetobacter_pasteurianus_PO_3281_01_uid59279	12/6/2012	12:00:00 AM
Acetobacter_pasteurianus_PO_3281_01_uid58373	4/12/2012	4:00:00 AM
Acetobacter_pasteurianus_PO_3281_07_uid58381	4/12/2012	4:14:00 AM
Acetobacter_pasteurianus_PO_3281_12_uid58379	4/12/2012	4:13:00 AM
Acetobacter_pasteurianus_PO_3281_12_uid58383	4/12/2012	4:21:00 AM
Acetobacter_pasteurianus_PO_3281_16_uid58381	4/12/2012	4:25:00 AM
Acetobacter_pasteurianus_PO_3281_12_uid58375	4/12/2012	4:28:00 AM
Acetobacterium_woodii_DSM_3030_uid58073	3/2/2012	5:00:00 AM
Acetobacterium_saccharum_DSM_5003_uid54233	12/6/2012	12:00:00 AM
Acetotrophium_indiani_PO_05_uid59801	12/6/2012	12:00:00 AM
Acetotrophium_yosoidensis_Ad_uid59809	12/6/2012	12:00:00 AM
Acidaminococcus_fermentans_DSM_20721_uid543471	1/11/2011	12:00:00 AM
Acidaminococcus_intestinalis_RyC_M895_uid54445	10/19/2011	12:00:00 AM
Acidimans_hospitalis_W1_uid56615	5/16/2011	12:00:00 AM
Acidobacterium_saccharovorans_345_15_uid51395	12/6/2012	12:00:00 AM
Acidomicrobium_ferredoxin_DSM_10331_uid59215	1/11/2011	12:00:00 AM
Acidobacterium_rubrum_IF_9_uid59447	1/11/2011	12:00:00 AM

Promoter
Training
set



GAOPP: Genetic Algorithm for Operon Prediction in Prokaryotes

2. Working with GAOPP:

2.1 Input files:

Download the required files like .ptt file and pathway file. Note down the KEGG organism code if you are planning to use pathway data, organism code has to be specified. Check that the .ptt file and pathway file are in the following format:

Escherichia coli str. K-12 substr. MG1655, complete genome - 1..4639675									
4132 proteins									
Location	Strand	Length	PID	Gene	Synonym	Code	COG	Product	
190..255	+	21	16127995	thrL	b0001 -	-		thr operon leader peptide	
337..2799	+	820	16127996	thrA	b0002 -	COG0460E,COG0527E		fused aspartokinase I and homoserine	
2801..3733	+	310	16127997	thrB	b0003 -	COG0083E		homoserine kinase	
3734..5020	+	428	16127998	thrC	b0004 -	COG0498E		threonine synthase	
5234..5530	+	98	16127999	yaaX	b0005 -	-		predicted protein	
5683..6459	-	258	16128000	yaaA	b0006 -	COG3022S		conserved protein	
6529..7959	-	476	16128001	yaaJ	b0007 -	COG1115E		predicted transporter	
8238..9191	+	317	16128002	talB	b0008 -	COG0176G		transaldolase B	
9306..9893	+	195	16128003	mog	b0009 -	COG0521H		predicted molybdochelatase	
9928..10494	-	188	16128004	yaaH	b0010 -	COG1584S		conserved inner membrane protein associated	
10643..11356	-	237	16128005	yaaW	b0011 -	COG4735S		conserved protein	

path:eco00010	eco:b0114	eco:aceE	ko:K00163	ec:1.2.4.1
path:eco00010	eco:b0115	eco:aceF	ko:K00627	ec:2.3.1.12
path:eco00010	eco:b0116	eco:lpd	ko:K00382	ec:1.8.1.4
path:eco00010	eco:b0356	eco:frmA	ko:K00121	ec:1.1.1.1 ec:1.1.1.284
path:eco00010	eco:b0688	eco:pgm	ko:K01835	ec:5.4.2.2
path:eco00010	eco:b0755	eco:gpmA	ko:K01834	ec:5.4.2.1
path:eco00010	eco:b0756	eco:galM	ko:K01785	ec:5.1.3.3
path:eco00010	eco:b1002	eco:agp	ko:K01085	ec:3.1.3.10

For promoter prediction, a promoter training set need to be specified. A Perl script provided with the program may be used to extract the promoter and non promoter training sets. Simply edit the script specifying your input file and sequence file. The input file is the same .ptt file. To generate the positive training set , edit the .ptt file keeping only those genes which contains upstream promoter signals, and delete others. Similarly, for negative training set keep only those genes not having upstream promoter sequence. Use these files one bay one to extract the upstream sequence for the genes present in respective files. Merge the positive and negative train sequences to form the final trained file.

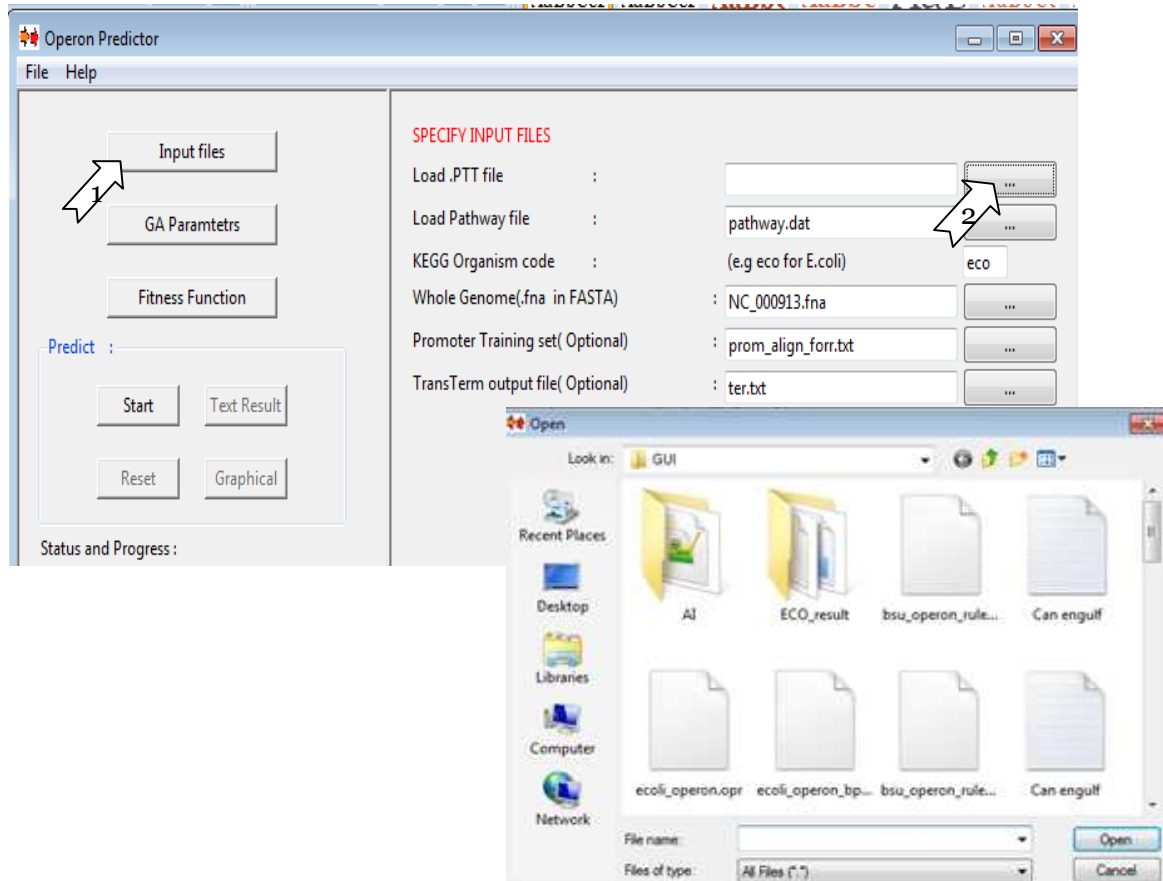
In order to generate the terminator coordinates, download the transterm binary executable. And *expterm.dat* file. Run the following command on windows command prompt:

```
transterm -p expterm.dat seq.fasta annotation.ptt > output.tt
```

remember to keep name of .ptt file and FASTA identifier in sequence file, exactly the same. And provide the sequence file earlier than .ptt file as the command line argument. The output file is written after '>'.

GAOPP: Genetic Algorithm for Operon Prediction in Prokaryotes

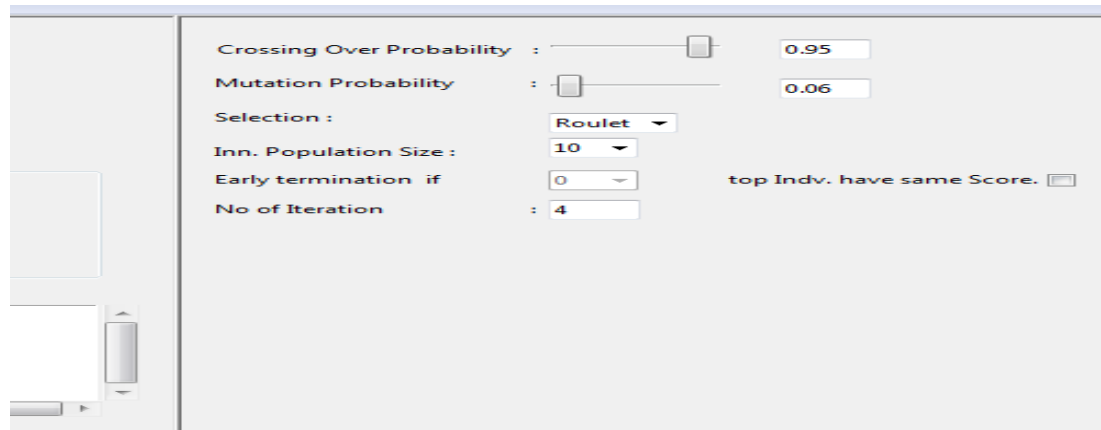
To load the input files click on the respective buttons and click on browse to load the files. Providing incorrect files causes anonymous error.



Load required files in correct format, otherwise result may be ambiguous.

2.2 Genetic Algorithm Parameters:

Clicking on GA parameters button opens the parameter panel:



2.2.a Operator Probability:

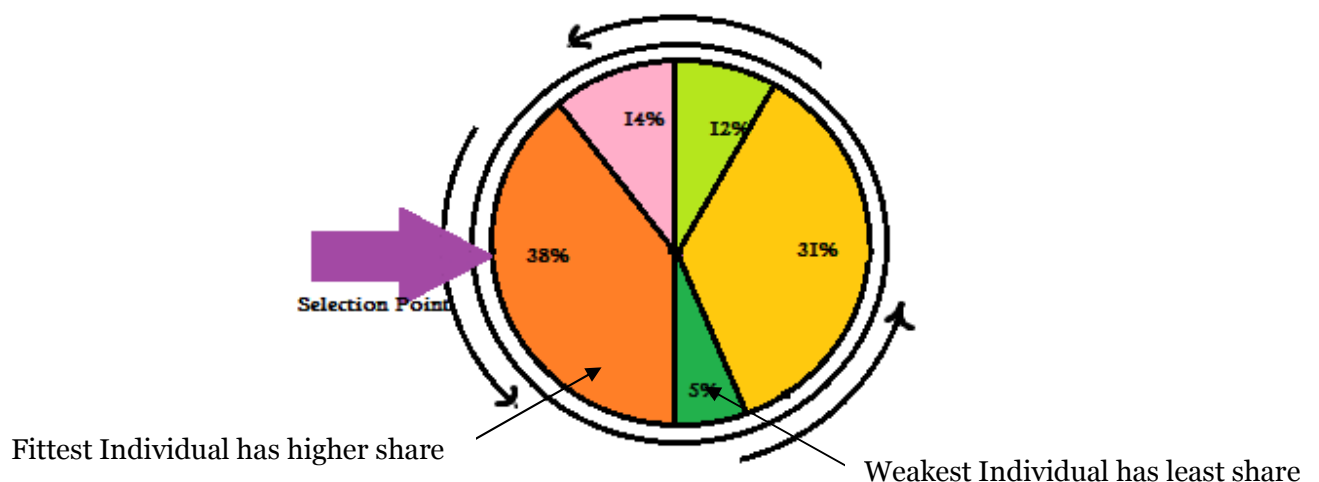
To implement genetic algorithm operators like Mutation and crossing over user need to set the probability. The probability indicates how often the operon has to be implemented. Generally a high cross over probability and low mutation probability combination gives optimized result. Use the sliders to adjust the probability.

2.2.b Selection:

A selection procedure selects an individual solution to be act as a parent for crossing over and generate offspring for next generation. There are two options for selecting the parents i. Roulette Wheel Selection ii. Best Individual selection.

i. Roulette Wheel Selection:

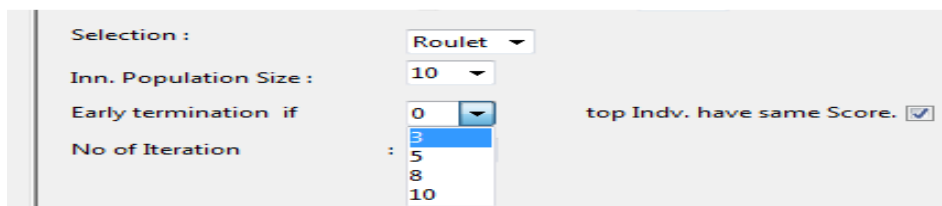
It selects an indivual stochastically form the current generation by simulating rotation of a wheel with an objective to select the fittest individual. During the process individuals having higher fitness score has higher probability to get selected in comparison to less fit individuals.



- ii. Best Individual: This method selects only the best individual from the generation. When user opts for this option, a higher mutation probability is advisable.

2.2.c Early Termination:

On attaining the best plausible solution, all the individuals will look much alike and mutation and crossing over does not make any change to the population. Hence continuing the process is worthless. Click on this the check box if user wants to terminate the evolution process when specified number of individuals in the current generation has same score.



The screenshot shows a configuration window with the following elements:

- Selection :** A dropdown menu set to "Roulet".
- Inn. Population Size :** A dropdown menu set to "10".
- Early termination if** and **No of Iteration**: A dropdown menu with options 0, 3, 5, 8, and 10. The value "3" is currently selected.
- top Indv. have same Score.**: A checkbox that is checked.

Initial Population Number must be higher than the number of individuals checked for early termination.

2.2. No. of Iterations:

This option explicitly specifies how many generations are to be evolved to find the best possible solution. Set this option as per your convenience. Until and unless early termination is not defined the program will run until the specified generation.

2.3 Fitness Function:

Click on Fitness Function Button to change the fitness function. Selecting a fitness function gives the literature reference used for calculating the score.

Fuzzy Fitness Finder (Jacob et.al) function takes a long run about 10-12 hrs for whole genome. Remember to set early termination option when FFF is used.

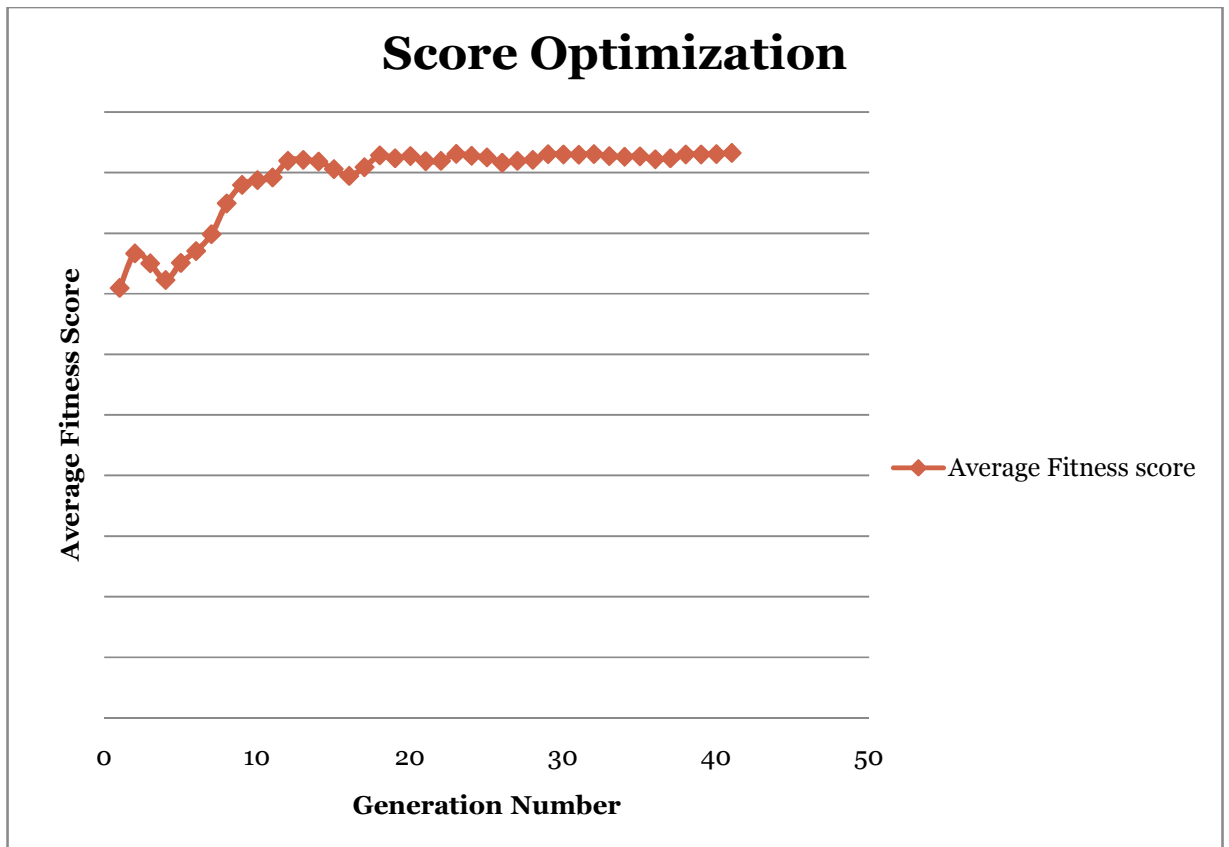
Rule based Fitness function is a heuristic one and can be used for quicker evaluation and doesn't guarantee better prediction.

2.4 Result Visualization:

Optimization process starts when start button is clicked. Like most standard GA software average fitness score in each generation plotted. This shows a uprising curve for successful optimization process. If the cure is not reliable (not uprising) user need to adjust the probabilities and run the program again.

Click on export button to save the plot in postscript format (.ps) to view it later in any post script virwer like ghostviwer. Otherwise the progress.xls file can be open after the run and select the two columns and plot using XY scetter.

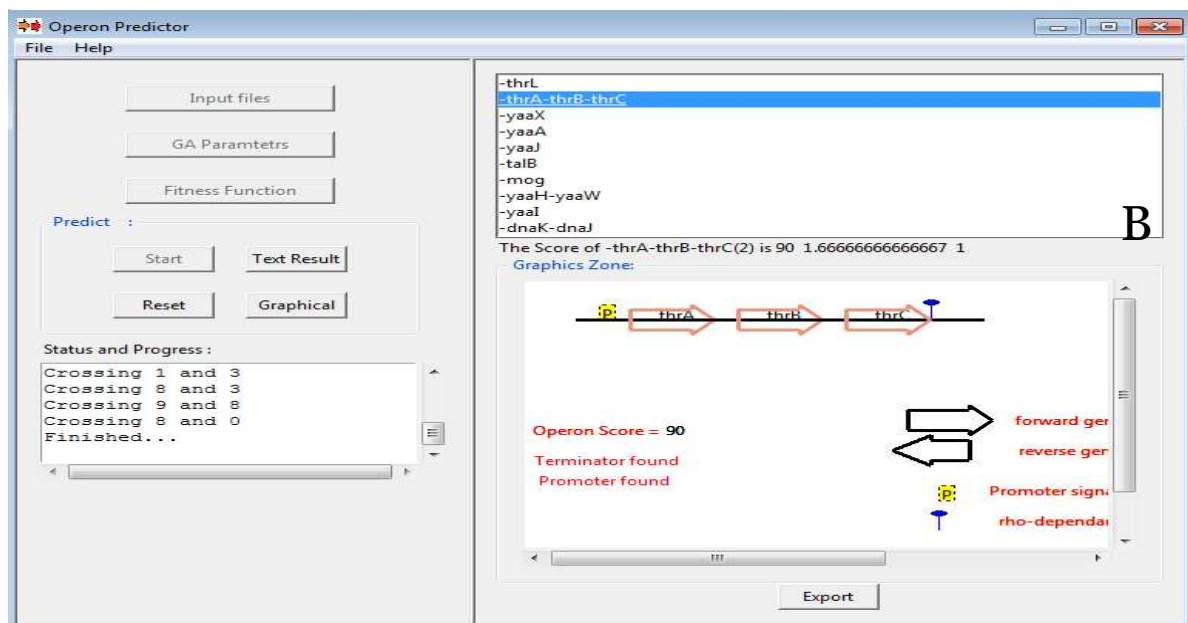
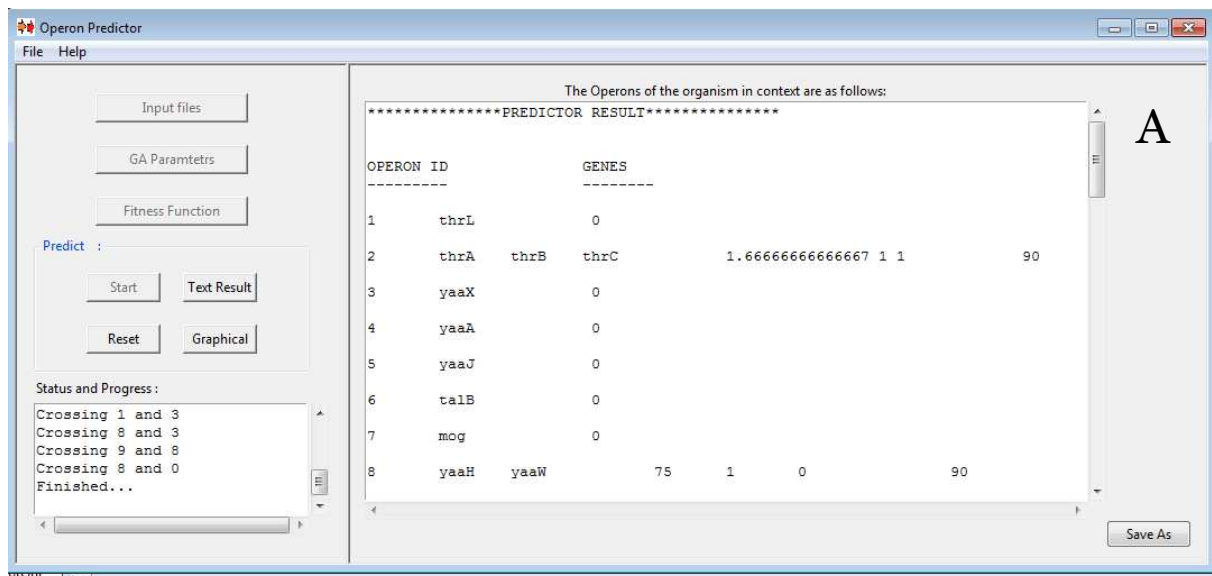




Operon clusters along with their corresponding scores are displayed in the result panel when Result in Text Button is clicked. Result exported to hard disc.

A Graphical viewer has been designed to represent individual operon clusters along with the promoter and terminator signals. The list of operons is displayed on the top. Selecting an cluster displays its total score at the bottom of list. Double clicking on a particular entry loads the entire operon map with terminator and promoter signals. Map in postscript format can be

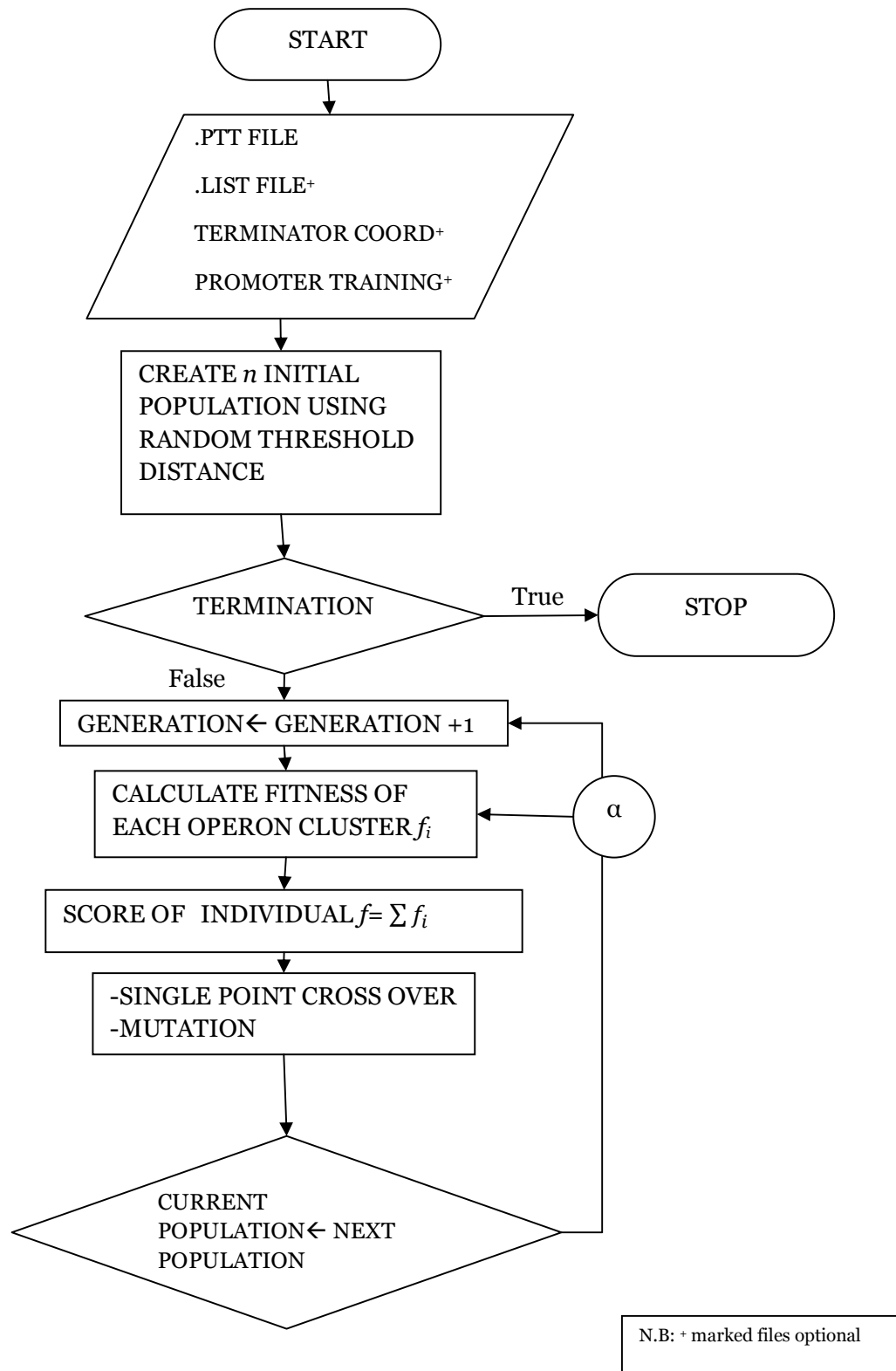
exported.

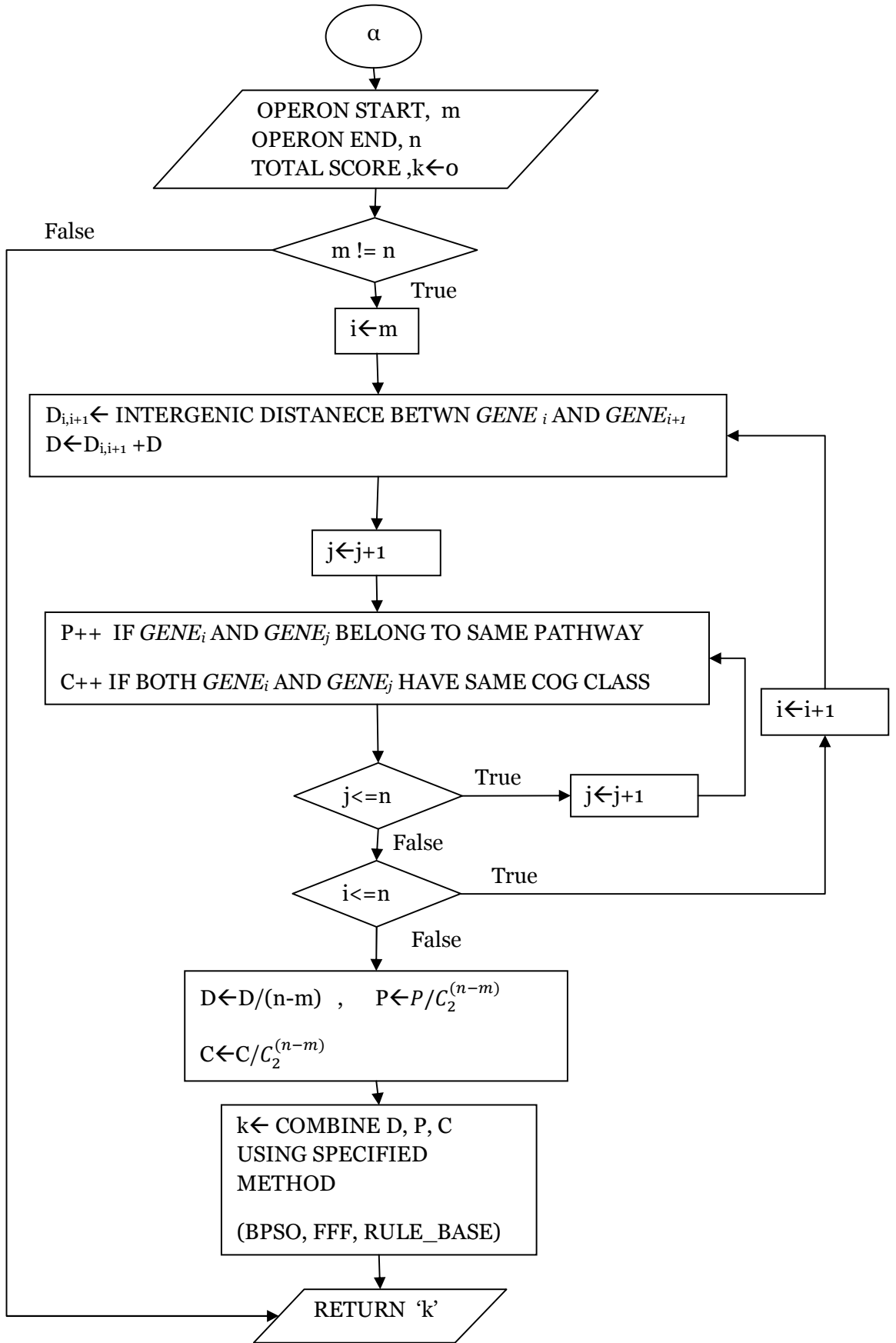


Output panel: Result in Text form (A) and Result in Graphical (B). GraphicalResult Shows visualizes regulatory signals.

GAOPP: Genetic Algorithm for Operon Prediction in Prokaryotes

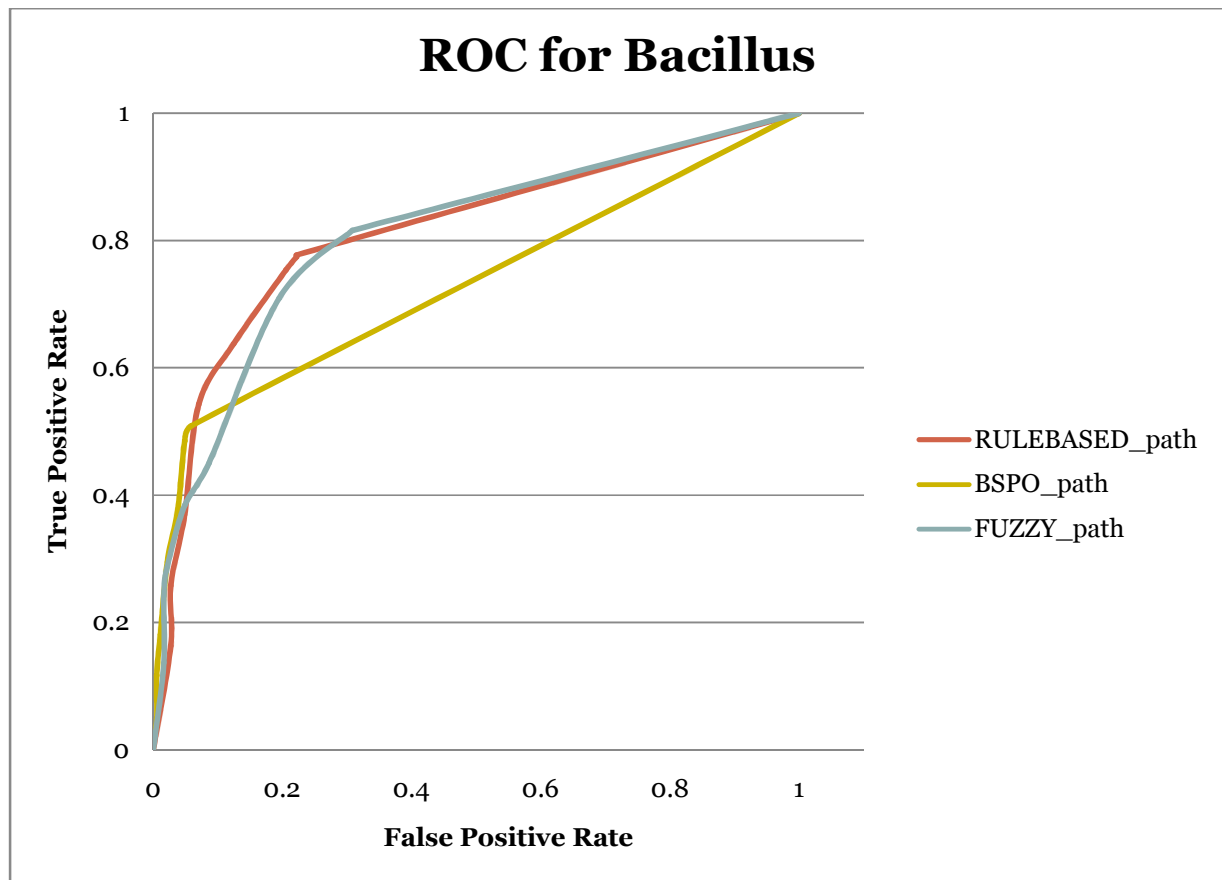
3. Algorithm:





Evaluation:

We used GAOPP for available test sets like *Escherichia coli* K-12 substr-MG1655 and *Bacillus subtilis*. We created positive and negative gene pairs from available experimental data. The predicted operons were compared with these available test set. From these observations we constructed Receiver operating curve.



Reference: