Complementary Binary Particle Swarm Optimization for Operon Prediction

Li-Yeh Chuang, Jui-Hung Tsai, and Cheng-Hong Yang, *Member, IAENG*

***Abstract*—An operon is a fundamental unit of transcription and contains specific functional genes for the construction and regulation of networks at the whole genome level. The prediction of operons is critical for understanding gene regulation and functions in newly sequenced genomes. As experimental methods for operon detection tend to be non-trivial and time-consuming, various methods for operon prediction have been proposed in the literature. In this study, a complementary binary particle swarm optimization (CBPSO) is used for operon prediction in bacterial genomes. We used complementary operation to improve the initialization procedure, and then used the intergenic distance, the metabolic pathway and the cluster of orthologous groups (COG) to design a fitness function. The proper values were trained on the *Escherichia coli* genome. Experimental results show that the prediction accuracy of this method reached 92.6%, 93.6%, 95.8% and 96.3% on *Bacillus subtilis, Pseudomonas aeruginosa PA01, Staphylococcus aureus* and *Mycobacterium tuberculosis* genomes, respectively. The proposed method predicted operons with high accuracy for the four test genomes.**

***Index Terms*—operon prediction, CBPSO, intergenic distance, metabolic pathway, COG.**

I. Introduction

In prokaryotic organisms, operons contain one or more consecutive genes on the same strand. A few eukaryotic organisms also have operon-like structures, e.g., *Caenorhabditis elegans* [1]. Co-transcribed genes have the same biological function and directly affect each other. Operon prediction can therefore be used to infer the function of putative proteins if the functions of other genes in the same operon are known. A well-known example is the lactose operon in *Escherichia coli*. This operon contains three consecutive structural genes, namely *lac*Z, *lac*Y and *lac*A, which all share the same promoter and terminator.

Operons of bacterial genomes contain valuable information for drug design and determining protein functions [2]. The gram-positive *Staphylococcus* bacterium, for example, is a human pathogen that is responsible for nosocomial infections [3]. Operon prediction on this bacterium can facilitate drug target identification and the development of antibiotic drugs. However, knowledge of operons is scarce, and experimental methods for operon prediction are generally difficult to implement [4]. To gain better insight, the number and organization of operons in bacterial genomes should be studied in greater detail.

In recent years, a number of scientists have proposed certain properties to accurately predict operons. These properties can be divided into the following five categories [5]: intergenic distance, conserved gene clusters, functional relations, genome sequence, and experimental evidence. In each of the aforementioned categories, it is pivotal to detect the promoter and terminator at the operon boundaries and to identify the biologically most representative properties [4]. The simplest and most important prediction property is to observe whether the distance between gene pairs within an operon (WO pairs) is shorter than the distance between gene pairs at the borders of the transcription units (TUB pairs) [3]. This distance property generally provides good results for operon prediction.

Many computational algorithms have been proposed to properly balance the sensitivity and specificity of operon prediction. Jacob *et al.* proposed a fuzzy guided algorithm for operon prediction [4]. This method does not rely on complicated mathematical formulas to calculate fitness values of chromosomes. Genetic algorithms (GA) [2] use four biological properties, the intergenic distance, the metabolic pathway, the cluster of orthologous (COG) gene function and the microarray expression data, to predict operons. Zhang *et al.* presented a support vector machine algorithm (SVM) to predict operons (6). This method uses the above four biological properties as SVM input vectors and divides gene pairs into operon pairs (OP) and non-operon pairs (NOP). In this study, a comparision of the following predictors is presented: FGA [4], GA [2], SVM [6], genome-specific [7], FGENESB, ODB [8], JPOP [9], UNIPOP [1] and Genome-wide operon prediction in *Staphylococcus aureus* [3].

In this paper, we propose an effective complementary binary particle swarm optimization (CBPSO) for operon prediction. To validate the method, we calculated the logarithmic likelihood of each property in the *Escherichia coli* (NC_000913) genome as a fitness value of each gene in the particle. Four bacterial genomes, *Bacillus subtilis* (NC_000964), *Pseudomonas aeruginosa PA01* (NC_002516), *Staphylococcus aureus* (NC_002952) and *Mycobacterium tuberculosis* (NC_000962), were selected as benchmark genomes of known operon structure. In a first step, half of the particles in the swarm are randomly generated, and

L. Y. Chuang is with the Chemical Engineering Department, I-Shou University, 84001, Kaohsiung, Taiwan. (e-mail: chuang@isu.edu.tw).

J. H. Tsai is with the Computer Science and Information Engineering Department, National Kaohsiung University of Applied Sciences, 80778, Kaohsiung, Taiwan. (e-mail: 109730812@cc.kuas.edu.tw).

C. H. Yang is with the Network Systems Department, Toko University, 61363, Chiayi, Taiwan.

C. H. Yang is also with the Electronic Engineering Department, National Kaohsiung University of Applied Sciences, 80778, Kaohsiung, Taiwan. (corresponding author to provide phone: 886-7-3814526#5639; e-mail: chyang@cc.kuas.edu.tw).

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

the other half of the particles is determined by a complementary operation. The particles are subsequently updated by an update formula at each generation. The detailed updating process is described in the next section. The experimental results indicate that the proposed method obtained higher accuracy, sensitivity, and specificity on the test data sets compared with other methods from the literature.

II. Related Methods

A. Data set preparation

The entire microbial genome data was downloaded from the GenBank database (http://www.ncbi.nlm.nih.gov/). The data contains a total of 4225, 5651, 2845 and 4047 genes in the *B. subtilis* genome, *P. aeruginosa PA01* genome, *S. aureus* genome and *M. tuberculosis* genome, respectively. The related genomic information consists of the gene name, the gene ID, the position, strand and product. The operon databases of *E. coli* and *B. subtilis* were obtained from RegulonDB (http://regulondb .ccg.unam.mx/) [10] and DBTBS (http://dbtbs.hgc.jp/) [11], respectively. The operon databases of *P. aeruginosa PA01* genome, *S. aureus* genome and *M. tuberculosis* were obtained from ODB (http://odb.kuicr.kyoto-u.ac.jp/) [8]. The genomes' metabolic pathway data and COG data was obtained from KEGG (http://www.genome.ad.jp/kegg/pathway.html) and NCBI (http://www.ncbi.nlm.nih.gov/COG/), respectively.

B. Definition of a potential operon pair

In order to gain valuable information pertaining to drug and protein functions, operons need to be predicted based on an organism's genomic sequence. The entire genome is scanned for adjacent gene pairs, and each gene pair is classified into one of three types: (i) adjacent; (ii) OP pair; or (iii) NOP pair. The WO pair and TUB pair are defined base on biological experiments, and the gene pairs are labelled 'positive' and 'negative', respectively. In Fig. 1, the white arrows represent genes as yet unclassified by experiments, and the gray arrow represents an operon containing only a single gene. In addition, the black arrows represent operons composed of several genes. As shown in Figure 1, adjacent genes in the same operon are called WO pairs. If the operon contains a single gene, and the downstream gene is of unknown status, the gene pair is called a TUB pair. However, if the upstream gene is the last gene of an operon, and the downstream gene is of uncertain status, the gene pair can not be labelled a TUB pair [12]. In addition, the first gene of an operon and the upstream gene are TUB pairs by default.

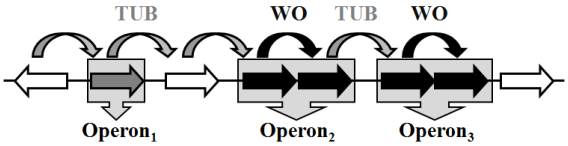

Figure 1. WO and TUB pairs

C. Operon properties

As stated above, many powerful properties can be used to predict operons. In this study, we use three properties, namely the intergenic distance, the metabolic pathway and the COG gene function, to identify operons. Each of these properties is individually described in the following three sections.

(1) Intergenic distance: This property can predict operons in the sequence of completely mapped genomes. To prevent mRNA degradation, the distance of adjacent genes in the same operon is shorter than the distance of TUB pairs [13]. As shown in Fig. 2, $gene_2$, $gene_3$ and $gene_4$ all share the same promoter and terminator. These genes are on the same operon. Therefore, the intergenic distance of $gene_2$ and $gene_3$, or $gene_3$ and $gene_4$, is shorter than the intergenic distance of $gene_1$ and $gene_2$, or $gene_4$ and $gene_5$. As shown in Eq.1, the distance of adjacent genes is calculated using base pairs of adjacent genes. However, adjacent genes may sometimes overlap as shown in Fig. 3. The chart displays the frequency of the distance of WO pairs and TUB pairs. Adjacent genes with shorter intergenic distances are more likely located within an operon [2]. The maximum frequency of the WO pair distance is -4 [14]. However, the distance distribution frequency of TUB pairs is increased with the distance, and becomes gradually higher than the frequency of WO pairs. Thus, this property can be used to identify operons in the bacterial genomes.

$$\text{Distance} = \text{Gene}_2\text{\_start} - (\text{Gene}_1\text{\_finish}+1) \quad (1)$$

Figure 2. Operon diagram

(2) Metabolic pathway: Gene ontology contains three levels of biological functions, namely a biological process, a molecular function and a cellular component [15]. However, genes within an operon often participate in the same biological process [6]. Therefore, if adjacent genes have the same metabolic pathway, we assume that the gene pair is located in the same operon.

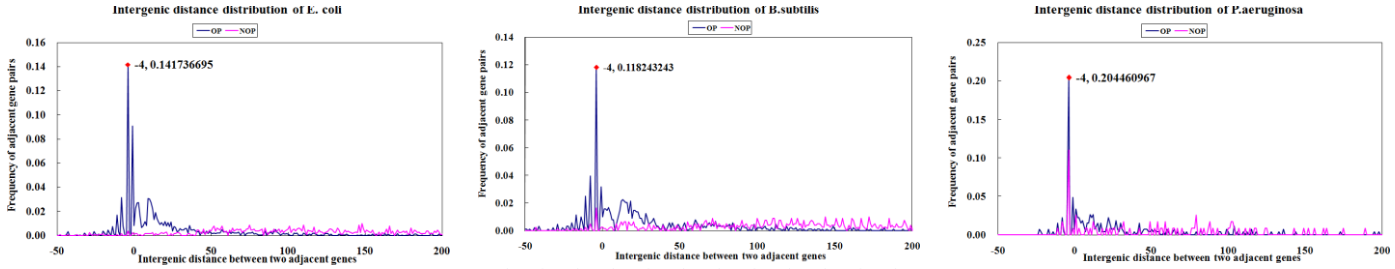

Figure 3. Intergenic distance distributions of WO and TUB pairs

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

(3) COG gene function: COGs consist of three main levels. The first level contains the following four classes: information storage and processing, cellular processing and signaling, metabolism, and different COG categories. Each class is divided into multiple functional categories. Adjacent genes are often in the same class, so we assume that the gene pair is located in the same operon.

III. Math Experiment Framework

A. Binary particle swarm optimization (BPSO)

Particle swarm optimization (PSO) is a population-based stochastic optimization technique developed by Kennedy and Eberhart in 1995 [16]. PSO has been developed through simulation of the social behavior of organisms, such as the social behavior observed of birds in a flock or fish in a school; it describes an automatically evolving system. In PSO, each single candidate solution (called particle) in the search space can be considered "an individual bird of the flock". Each particle uses their memory and knowledge gained by the swarm as a whole to find the optimal solution. The fitness value of each particle is evaluated by an optimized fitness function, and the particle velocity directs the movement of the particles. Each particle adjusts its position according to its own experience during movement. In addition, each particle also searches for the optimal solution in a search space based on the experience of a neighboring particle, thus making use of the best position encountered by itself and its neighbor. The particles move through the problem space by following a current of optimum particles. The entire process is reiterated a predefined number of times or until a minimum error is achieved. PSO has been successfully employed to many application areas; it obtains better results quickly and has a lower cost compared to other methods. However, PSO is not suitable for optimization problems in a discrete feature space. Hence, Kenney and Eberhart developed binary PSO (BPSO) to overcome this problem [17]. The basic elements of BPSO are briefly introduced below:

1) Population: A swarm (population) consists of N particles.

2) Particle position, x_i: Each candidate solution can be represented by a D-dimensional vector; the i^{th} particle can be described as $x_i = (x_{i1}, x_{i2}, \ldots, x_{iD})$, where x_{iD} is the position of the i^{th} particle with respect to the D^{th} dimension.

3) Particle velocity, v_i: The velocity of the i^{th} particle is represented by $v_i = (v_{i1}, v_{i2}, \ldots, v_{iD})$, where v_{iD} is the velocity of the i^{th} particle with respect to the D^{th} dimension. In addition, the velocity of a particle is limited within $[V_{\min}, V_{\max}]^D$.

4) Inertia weight, w: The inertia weight is used to control the impact of the previous velocity of a particle on the current velocity. This control parameter affects the trade-off between the exploration and exploitation abilities of the particles.

5) Individual best, $pbest_i$: $pbest_i$ is the position of the i^{th} particle with the highest fitness value at a given iteration.

6) Global best, *gbest*: The best position of all *pbest* particles is called global best.

7) Stopping criterion: The process is stopped after the maximum allowed number of iterations is reached.

B. Complementary Operation

The initialization is very important for operon prediction, and therefore we use complementary operation to improve the prediction ability of BPSO. In this study, half of the particle swarm is initialized with a random threshold value, and the other half of the particle swarm is then initialized by a complementary operation. Figure 4 illustrates these criteria.

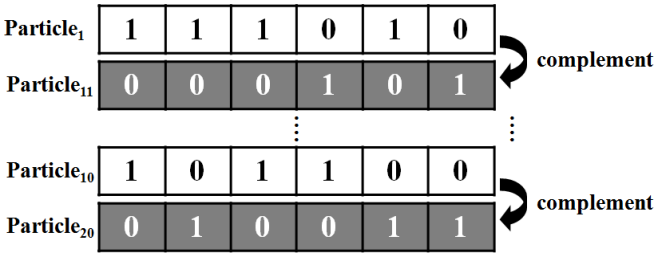

Figure 4. Complementary Operation

C. Encoding and Initialization

If the gene pair is a considered non-operon pair (NOP), the upstream gene is encoded to 0. If the gene pair is an operon pair (OP), the upstream gene is encoded to 1. As shown in Figure 5, if $gene_3$, $gene_4$ and $gene_6$ are the last genes of $operon_1$, $operon_2$ and $operon_3$, respectively, the elements of the array are 110010. In addition, the proposed method uses the intergenic distance and strands to create P binary particles. Each particle is initialized with a random threshold value of between 0 and 600 bps [4]. For adjacent genes to be considered in the same operon, they must conform to the following two conditions: the distance of adjacent genes must be smaller than the random threshold, and adjacent genes must be on the same strand. If the distance between adjacent genes is greater than the random threshold, we assume that the two adjacent genes are within a different operon. Adjacent genes on different strands are considered NOP. Figure 6 illustrates these criteria.

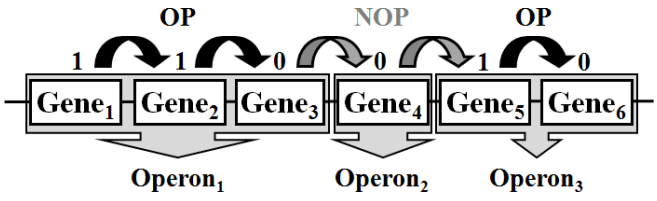

Figure 5. Encoding

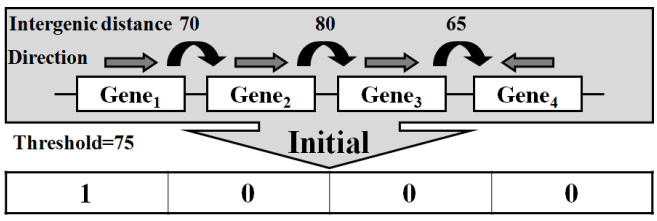

Figure 6. Initial population

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

D. Particle update

In BPSO, each particle is updated according to the following equations:

$$v_{id}^{new} = w \times v_{id}^{old} + c_1 \times r_1 \times \left(pbest_{id} - x_{id}^{old}\right) + c_2 \times r_2 \times \left(gbest_{id} - x_{id}^{old}\right) \quad (1)$$

$$\text{if } v_{id}^{new} \notin (V_{\min}, V_{\max}) \text{ then } v_{id}^{new} = \max\left(\min\left(V_{\max}, v_{id}^{new}\right), V_{\min}\right) \quad (2)$$

$$S\left(v_{id}^{new}\right) = \frac{1}{1 + e^{-v_{id}^{new}}} \quad (3)$$

$$\text{if } \left(r_3 < S\left(v_{id}^{new}\right)\right) \text{ then } x_{id}^{new} = 1 \text{ else } \quad (4)$$

where w is the inertia weight that controls the impact of the previous velocity of a particle. c_1 and c_2 are acceleration constants that control the distance a particle moves at each generation; r_1, r_2 and r_3 are three random numbers between [0, 1]. v_{id}^{new} and v_{id}^{old} represent the velocity of the new and old particles, respectively. Particles x_{id}^{old} and x_{id}^{new} denote the position of the current particle and the updated particle, respectively. The velocity of a dimension in Eq. 2 is limited within $[V_{\min}, V_{\max}]^D$. The positions of the updated particles are calculated by Eq. 3 [18]. If the function $S\left(v_{id}^{new}\right)$ is greater than r_3, the position of the particle is updated to {1} (meaning this gene is part of the operon). If $S\left(v_{id}^{new}\right)$ is smaller than r_3, the position is updated to {0} (i.e., this gene is the final gene of the operon).

E. Fitness function

As stated previously, many properties can be used to predict operons. In this study, five properties are used and described individually in the following section. The pair-scores of the intergenic distance, the metabolic pathway, the COG gene function, and the gene length ratio are calculated by the logarithmic likelihood ratio test. The pair-score of the operon length is calculated by the Bernoulli process.

(1) Intergenic distance: the score of each separated interval in 10bp bins [19] is calculated based on an intergenic distance from -100 bps to 300bps using the following equation:

$$\mathrm{LL}_{dist}\left(gene_i, gene_j\right) = \ln\left(\frac{N_{WO}(dist)/TN_{WO}}{N_{TUB}(dist)/TN_{TUB}}\right) \quad (5)$$

where $N_{WO}(dist)$ and $N_{TUB}(dist)$ correspond to the number of WO and TUB pairs in the interval distance *dist* (10, 20, 30…). TN_{WO} and TN_{TUB} are the total pair numbers within WO and TUB, respectively.

(2) Metabolic pathways: The pair-score of the metabolic pathway is also calculated by the log-likelihood method. The pathway pair-score is only taken into account when the two adjacent genes have the same pathway. Otherwise the pathway pair-score is 0 [2]. The following Eq. 6 is used to calculate the pathway pair-score.

$$\mathrm{LL}_{path}\left(gene_i, gene_j\right) = \ln\left(\frac{N_{WO}(path)/TN_{WO}}{N_{TUB}(path)/TN_{TUB}}\right) \quad (6)$$

where N_{WO}(*path*) and N_{TUB}(*path*) correspond to the total number of WO and TUB pairs in the same metabolic pathway. TN_{WO} and TN_{TUB} are the total pair numbers within WO and TUB, respectively.

(3) COG gene function: We use the log-likelihood method to calculate the pair-score of the COG gene function based on three main levels. The following equations are used [9]:

$$\mathrm{LL}_{COG}\left(gene_i, gene_j\right) = \ln\left(\frac{N_{WO}(COG)/TN_{WO}}{N_{TUB}(COG)/TN_{TUB}}\right) \quad (7)$$

$$\mathrm{LL}_{COGd}\left(gene_i, gene_j\right) = \ln\left(\frac{1 - N_{WO}(COG)/TN_{WO}}{1 - N_{TUB}(COG)/TN_{TUB}}\right) \quad (8)$$

where TN_{WO} and TN_{TUB} are again the total pair numbers within WO and TUB, respectively. $N_{WO}(COG)$ and $N_{TUB}(COG)$ stand for the total number of WO and TUB pairs in the same COG gene function. LL_{COGd} ($gene_i$, $gene_j$) represents the pair-score of adjacent genes with a different COG gene function.

While the individual pair-scores are obtained by the calculations above, the overall pair-score of adjacent genes is calculated as the sum of the individual pair-scores from the three properties mentioned above..

The fitness value of the c^{th} putative operon is thus calculated by the following equation:

$$fitness_c = \sum_{i=1}^{m-1}(d_i) + \left(\frac{\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}(\mathrm{LL}_{path}(gene_i, gene_j) + \mathrm{LL}_{COG}(gene_i, gene_j))}{n} + \right) \times m \quad (9)$$

where d_i is the pair-score of the intergenic distance of the i^{th} gene in the c^{th} operon, and m and n are the total number of genes and gene pairs in the c^{th} operon, respectively.

Finally, the fitness value of a particle is calculated as the sum of the fitness values from all putative operons in the particle and thus given by the following equation:

$$fitness = \sum_{i=1}^{c} fitness_i \quad (10)$$

where c is the number of operons in a particle.

F. Parameter Settings

In the present study, the population number P was set to 20, the iteration number G was 100, the initial inertia weight w was 1, c_1 and c_2 were 2 [20], and $V_{\max}$ and $V_{\min}$ were 6 and -6, respectively [17].

IV. Results and Discussion

A. Performance measurement

In this study, the *E. coli* genome was used to estimate the fitness value, and then accuracy tests were conducted on other genomes. To do this, the training data set was further divided to estimate the prediction accuracy during the search. For a large data set like the *E. coli* genome, it is easy to build a predictor

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Table 1. Evaluation method for operon prediction

Value to be estimated	Equation for estimation
Sensitivity (SN)	SN=TP/(TP+FN)
Specificity (SP)	SP=TN/(FP+TN)
Positive Prediction Rate (PPR)	PPR=TP/(TP+FP)
Negative Prediction Rate (NPR)	NPR=TN/(FN+TN)
Accuracy (ACC)	ACC=(TP+TN)/(TP+FP+TN+FN)

Table 2. The positive and negative evaluation

True data / Prediction result	Positive	Negative
Positive	TP	FP
Negative	FN	TN

that clearly identifies WO and TUB pairs. Most previous efforts have focused on the operon prediction of *E. coli* genome. This has lead to an extensive database of experimentally identified transcripts for this genome. For these reasons, the *E. coli* genome was chosen as the training data set. We used the entire data set to estimate the fitness values since dividing the data set into subgroups does not provide a clear advantage over using the entire data set [7]. In order to verify the generalization ability of our method, the test data sets do not contain the *E. coli* genome which has genome-specific properties. The predictive performance [7] was evaluated based on the sensitivity and specificity shown in Table 1. As show in Table 2, true positive (TP) and false negative (FN) are the numbers of correct and incorrect prediction of gene pairs among the WO gene pairs, respectively, whereas true negative (TN) and false positive (FP) are the numbers of correct and incorrect prediction of gene pairs among the TUB gene pairs. The sensitivity, specificity and accuracy were determined based on TP, FN, TN and FP; results are shown in Table 3. The experimental operon encoding of the genome is 111010, and the predicted operon encoding is 110110. The third and fourth genes are FN and FP, respectively. The first, second and fifth genes are TP, and the sixth gene is TN. The accuracy obtained by the proposed method was compared to other methods. It should be noted that a good balance between sensitivity and specificity was achieved.

B. Comparison to other methods

CBPSO was applied to search for the best putative operon at each generation. The best putative operon identified by the search was then compared to experimentally verified operons. As shown in Table 3, the prediction accuracy of the proposed method obtained the highest accuracy values on the *B. subtilis* (0.926), *P. aeruginosa PA01* (0.936), *S. aureus* (0.959), and *M. tuberculosis* (0.963) data sets. The proposed method also showed the best performance in terms of prediction sensitivity and specificity on most of the tested bacterial genomes. In addition, even through BPSO obtained a higher specificity than CBPSO on the *B. subtilis* and *P. aeruginosa PA01* genome, CBPSO obtained a good balance between sensitivity and specificity. Hence, the prediction results of CBPSO are not only superior to BPSO, but are also better in terms of accuracy, sensitivity, and specificity when compared to other methods from the literature.

C. Discussion

Most methods use the properties of adjacent genes to identify OP or NOP for operon prediction. However, this process does not take the properties of near genes into account, and thus generally results in lower accuracies for operon prediction. The CBPSO used in this study evaluates the properties of near genes, and thereby increases the probability of finding an optimal solution. In order to raise the CBPSO

Table 3. Accuracy, sensitivity, and specificity of operon prediction on four genomes

Genome	Methodology	Accuracy	Sensitivity	Specificity
B. subtilis (NC_000964)	CBPSO	**0.926**	**0.919**	0.932
	BPSO	0.883	0.742	**0.996**
	UNIPOP [1]	0.792	0.782	0.821
	GA [2]	0.883	0.873	0.897
	Using both genome-specific and general genomic information [7]	0.902	N/A	N/A
	SVM [6]	0.889	0.900	0.860
	ODB [8]	0.632	0.499	0.992
	FGA [4]	0.882	N/A	N/A
	JPOP [9]	0.746	0.720	0.900
P. aeruginosa PA01 (NC_002516)	CBPSO	**0.936**	**0.933**	0.941
	BPSO	0.911	0.885	**0.953**
	GA [2]	0.813	0.870	0.763
S. aureus (NC_002952)	CBPSO	**0.959**	**0.959**	**0.959**
	BPSO	0.927	0.911	**0.959**
	Genome-wide operon prediction in *Staphylococcus aureus* [3]	0.920	N/A	N/A
M. tuberculosis (NC_000962)	CBPSO	**0.963**	**0.963**	**0.963**
	BPSO	0.951	0.944	**0.963**
	A Predicted Operon map for Mycobacterium tuberculosis [21]	0.908	N/A	N/A

Legend: N/A: Data not available. Highest values in bold type.

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

prediction performance, we limit the velocity of CBPSO to between V_{min} and V_{max}. If the velocity is close to 0, the probability of a state changing is increased, and vice versa. Hence, CBPSO has global and local search capabilities. The probability of obtaining the best solution is thus increased.

We used the complementary operator to initialize half of the particle swarm in the initiation step. As shown in Table 3, CBPSO obtained a better prediction performance than BPSO and other methods form the literature. BPSO without adding the complementary operator obtained lower prediction accuracy than some literature methods. BPSO does not achieve a good balance between the sensitivity and specificity; for the balance to be considered acceptable, both sensitivity and specificity need to be higher than 0.8 [6]. By boosting the quality of particles at the initiation, the best particles can be obtained by successive progression through the generations.

Generally, the prediction accuracy is proportional to the fitness value of a particle. Although adjacent genes have related properties, they still have a different probability of being in different operons. This necessitates the implementation of a fitness function in the proposed method. In this study, we calculated the fitness value of each particle based on the log-likelihood that is designed on the basis of statistics. Therefore, the fitness value of a putative operon is directly proportional to the prediction accuracy. The prediction accuracy of CBPSO and BPSO prove that this fitness function can identify better particles.

Experimental data on the *E. coli* genome can be downloaded from the RegulonDB database, but for other genomes extensive experimental data is not readily available. In order to apply the proposed method to other genomes with fewer attributes, three common properties for operon prediction were used. Theoretically, methods using more properties for operon prediction achieve a higher accuracy. Ever though many methods in the literature use numerous properties, our method only uses three properties. Yet CBPSO achieves better results. ODB uses four properties for operon prediction but obtained a lower prediction sensitivity [1]. The results reveal that the pathway and COG properties are more suitable for identification of WO and TUB pairs. Since adjacent gene share a common pathway, the probability of a gene pair to be a WO pair is very high [4]. The probability of gene pairs with the same first-level categories is 83.5% [9]. Our method achieved the highest accuracy for operon prediction even though only three properties were used on all bacterial genomes. The contributions to operon prediction are thus self-evident.

V. Conclusion

We propose a novel method, called CBPSO, for operon prediction in bacterial genomes. This study uses a complementary operator to generate half of the particle swarm in the initiation step, and CBPSO thus uses superior particles at the initialization of a population. We used the intergenic distance, the metabolic pathway and the COG gene functions of the *E. coli* genome to design a fitness function. The experimental results show that the proposed method increases the accuracy of operon prediction on four test genome data sets. In the future, we intend to investigate other algorithms and different properties on the problems of operon prediction in order to increase the prediction performance further.

References

[1] G. Li, D. Che, and Y. Xu, "A universal operon predictor for prokaryotic genomes," *J Bioinform Comput Biol*, vol. 7, Feb 2009, pp. 19-38.

[2] S. Wang, Y. Wang, W. Du, F. Sun, X. Wang, C. Zhou, and Y. Liang, "A multi-approaches-guided genetic algorithm with application to operon prediction," *Artif Intell Med*, vol. 41, Oct 2007, pp. 151-9.

[3] L. Wang, J. D. Trawick, R. Yamamoto, and C. Zamudio, "Genome-wide operon prediction in Staphylococcus aureus," *Nucleic Acids Res.*, vol. 32, 2004, pp. 3689-702.

[4] E. Jacob, R. Sasikumar, and K. N. Nair, "A fuzzy guided genetic algorithm for operon prediction," *Bioinformatics*, vol. 21, Apr 15 2005, pp. 1403-7.

[5] R. W. Brouwer, O. P. Kuipers, and S. A. van Hijum, "The relative value of operon predictions," *Brief Bioinform*, vol. 9, Sep 2008, pp. 367-75.

[6] G. Q. Zhang, Z. W. Cao, Q. M. Luo, Y. D. Cai, and Y. X. Li, "Operon prediction based on SVM," *Comput Biol Chem*, vol. 30, Jun 2006, pp. 233-40.

[7] P. Dam, V. Olman, K. Harris, Z. Su, and Y. Xu, "Operon prediction using both genome-specific and general genomic information," *Nucleic Acids Res.*, vol. 35, 2007, pp. 288-98.

[8] S. Okuda, T. Katayama, S. Kawashima, S. Goto, and M. Kanehisa, "ODB: a database of operons accumulating known operons across multiple genomes," *Nucleic Acids Res.*, vol. 34, Jan 1 2006, pp. D358-D362.

[9] X. Chen, Z. Su, P. Dam, B. Palenik, Y. Xu, and T. Jiang, "Operon prediction by comparative genomics: an application to the Synechococcus sp. WH8102 genome," *Nucleic Acids Res.*, vol. 32, 2004, pp. 2147-57.

[10] S. Gama-Castro, V. Jimenez-Jacinto, M. Peralta-Gil, A. Santos-Zavaleta, M. Penaloza-Spinola, B. Contreras-Moreira, J. Segura-Salazar, L. Muniz-Rascado, I. Martinez-Flores, and H. Salgado, "RegulonDB (version 6.0): gene regulation model of Escherichia coli K-12 beyond transcription, active (experimental) annotated promoters and Textpresso navigation," *Nucleic Acids Res.*, vol. 36, 2007, pp. D120-D124.

[11] N. Sierro, Y. Makita, M. de Hoon, and K. Nakai, "DBTBS: a database of transcriptional regulation in Bacillus subtilis containing upstream intergenic conservation information," *Nucleic Acids Res.*, vol. 36, Jan 2008, pp. D93-D96.

[12] C. Sabatti, L. Rohlin, M. K. Oh, and J. C. Liao, "Co-expression pattern from DNA microarray experiments as a tool for operon prediction," *Nucleic Acids Res.*, vol. 30, Jul 1 2002, pp. 2886-93.

[13] H. Salgado, G. Moreno-Hagelsieb, T. F. Smith, and J. Collado-Vides, "Operons in Escherichia coli: genomic analyses and predictions," *Proc. Natl Acad. Sci. USA*, vol. 97, pp. 6652-7, Jun 6 2000.

[14] Y. Yan and J. Moult, "Detection of operons," *Proteins*, vol. 64, Aug 15 2006, pp. 615-28.

[15] T. T. Tran, P. Dam, Z. Su, F. L. Poole, 2nd, M. W. Adams, G. T. Zhou, and Y. Xu, "Operon prediction in Pyrococcus furiosus," *Nucleic Acids Res.*, vol. 35, 2007, pp. 11-20.

[16] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings., IEEE International Conference on Neural Networks*, 1995, pp. 1942-1948.

[17] J. Kennedy and R. Eberhart, "A discrete binary version of the particle swarm algorithm," in *System, Man, and Cybernetics, Computational Cybernetics and Simulation, 1997. Proceedings., IEEE International Conference*, 1997, pp. 4104-4108.

[18] K. Crammer and Y. Singer, "On the learnability and design of output codes for multiclass problems," *Machine Learning*, vol. 47, 2002., pp. 201-233

[19] P. R. Romero and P. D. Karp, "Using functional and organizational information to improve genome-wide computational prediction of transcription units on pathway-genome databases," *Bioinformatics*, vol. 20, Mar 22 2004, pp. 709-17.

[20] J. Kennedy, R. Eberhart, and Y. Shi, *Swarm intelligence*: Springer, 2001.

[21] P. Roback, J. Beard, D. Baumann, C. Gille, K. Henry, S. Krohn, H. Wiste, M. I. Voskuil, C. Rainville, and R. Rutherford, "A predicted operon map for Mycobacterium tuberculosis," *Nucleic Acids Res.*, vol. 35, 2007, pp. 5085-95.

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)