2010 International Conference on Technologies and Applications of Artificial Intelligence

Operon Prediction using Particle Swarm Optimization and Reinforcement Learning

Li-Yeh Chuang
Department of Chemical Engineering & Institute of Biotechnology and Chemical Engineering, I-Shou University, Kaohsiung, Taiwan.
chuang@isu.edu.tw

Jui-Hung Tsai
Department of Computer Science and Information Engineering, National Kaohsiung University of Applied Sciences, Kaohsiung, Taiwan.
bigblack918@hotmail.com

Cheng-Hong Yang
Department of Network Systems, Toko University, Chiayi, Taiwan.
Department of Electronic Engineering, National Kaohsiung University of Applied Sciences, Kaohsiung, Taiwan.
chyang@cc.kuas.edu.tw

***Abstract*—An operon is a fundamental unit of transcription contains a specific function of genes for the construction and regulation of networks at the whole genome level. The operon prediction is critical for the understanding of gene regulation and functions in newly sequenced genomes. Various methods for operon prediction have been proposed in the literature shows that the experimental methods for operon detection are tend to be non-trivial and time-consuming.. In this study, a binary particle swarm optimization (BPSO) and reinforcement learning (RL) are used for operon prediction in bacterial genomes. The intergenic distance, participation in the same metabolic pathway and the gene length ratio property of the *Escherichia coli* genome are used to design a fitness function based on the conception of RL. Then the three genomes are used to test the prediction performance of BPSO with RL. Experimental results show that the prediction accuracy of this method reached to 92.8%, 94.3% and 95.9% on *Bacillus subtilis*, *Pseudomonas aeruginosa* PA01 and *Staphylococcus aureus* genomes respectively. The proposed method for the predicted operons with the highest accuracy contains the three test genomes.**

Keywords- operon prediction; property; BPSO; reinforcement learning.

I. INTRODUCTION

Operons in prokaryotic organisms contain one or more consecutive genes on the same strand. However, a few eukaryotic organisms also have operon-like structures, e.g., *Caenorhabditis elegans* [1]. Due to co-transcribed genes have the same biological function that is directly affecting each other, thus the operon prediction can 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 are all sharing the same promoter and terminator.

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

At present, a number of scientists have proposed various properties to predict operons accurately. These properties are 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 identify the most biological representative properties [4]. Nevertheless, the simplest and most important prediction property is to observe whether the distance between the 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 superior results for operon predictions.

In recent years, many computational algorithms have been proposed to balance out the sensitivity and specificity of operon prediction properly. Jacob *et al.* proposed a fuzzy guided algorithm for operon prediction [4]. This method does not depend on the complicated mathematical formulas for calculating the 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 four biological properties as SVM input vectors and divides gene pairs into operon pairs (OP) and non-operon pairs (NOP).

In this paper, we proposed a binary particle swarm optimization (BPSO) with reinforcement learning (RL) for operon prediction. To validate this 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. Three bacterial genomes, *Bacillus subtilis* (NC_000964), *Pseudomonas aeruginosa* PA01 (NC_002516) and *Staphylococcus aureus* (NC_002952), were selected as benchmark genomes of the known operon structure. In BPSO with RL, the particles in the swarm are generated and then evaluated the fitness value of each particle based on the concept of RL. The particles are subsequently updated by an update formula at each generation. The experimental results indicate that the proposed method obtains higher accuracy,

978-0-7695-4253-9/10 $26.00 © 2010 IEEE
DOI 10.1109/TAAI.2010.65

sensitivity and specificity on the target genomes that are compared with other methods from the literature.

II. Related Methods

A. Data set preparation

The entire microbial genome data downloaded from the GenBank database (http://www.ncbi.nlm.nih.gov/). The data contain a total of 4488, 4225, 5651 and 2845 genes in *E. coli* genome (NC_000913), *B. subtilis* genome (NC_000964), *P. aeruginosa* PA01 genome (NC_002516) and *S. aureus* genome (NC_002952) respectively. The related genomic information consists of the gene names, the gene IDs, the positions, strands and products. The experimental operons of *E. coli* and *B. subtilis* obtained from RegulonDB (http://regulondb.ccg.unam.mx/) [7] and DBTBS (http://dbtbs.hgc.jp/) [8] database, respectively. The experimental operons of *P. aeruginosa* PA01 genome and *S. aureus* genome were extracted from ODB (http://odb.kuicr.kyoto-u.ac.jp/) [9] database. The metabolic pathway data and COG data 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 receive the valuable information pertaining to drug and protein functions, operons need to be predicted based on the genomic sequence of organism. The entire genome is scanned for adjacent gene pairs, and each gene pair is classified into one of the 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 unclassified by experiments, and the gray arrow represents an operon containing only a single gene. In addition, the black arrows represent operons composed by several genes. As shown in Fig. 1, adjacent genes in the same operon are called WO pairs. If the operon contains a single gene, and the downstream gene is in unknown status, then 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 in uncertain status, the gene pair can not be labelled as TUB pair [10]. In addition, the first gene of an operon and the upstream gene are TUB pairs by default.

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 gene length ratio to identify operons. Each of these properties is individually described in the following three sections.

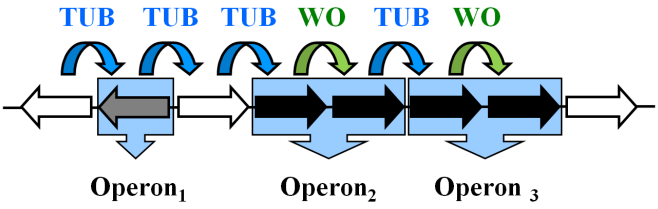

Figure 1. WO and TUB pairs.

1) Intergenic distance: This property predicts operons in the sequence of the complete mapped genomes. To prevent mRNA degradation, the distance of adjacent genes in the same operon is shorter than the distance of TUB pairs [11]. 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 $gene_4$, are 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 the base pairs of adjacent genes. However, adjacent genes sometimes may 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 [12]. 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 is used to identify operons in the bacterial genomes.

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

2) Metabolic pathway: Gene ontology contains three levels of biological functions, namely a biological process, a molecular function and a cellular component [13]. However, genes within an operon often participate in the same biological process [6]. Therefore, if adjacent genes have the same metabolic pathway, we assumed that the gene pair is located in the same operon.

3) Gene length ratio: TUB pairs are often associated with small values of the natural log of the length ratio when natural log of the length ratio is examined. Hence, the gene length ratio is used to calculate the ratio of the upstream-gene length to the downstream-gene length. In other words, this ratio influences the probability of the gene pair that is being located within an operon [14]. Dam *et al.* used their experimental results to verify the gene length ratio which is a powerful tool for discerning operons [14].

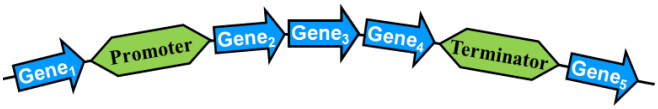

Figure 2. Operon diagram.

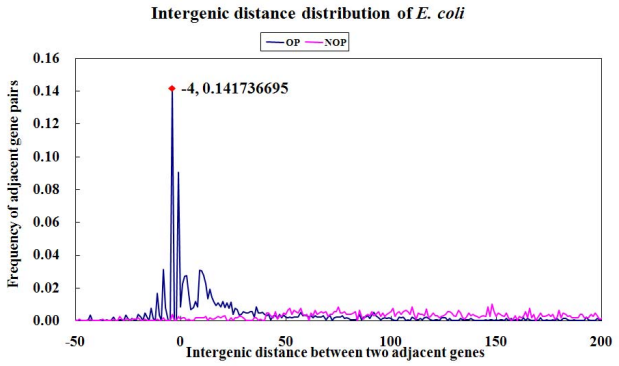

Figure 3. Intergenic distance distributions of WO and TUB pairs.

D. Reinforcement learning

Reinforcement learning was initially used in artificial intelligence (AI). RL is an approach for machine intelligence that combines the fields of dynamic programming and supervised learning, and able to produce powerful machine-learning systems. The generality of RL appeals to many researchers and thus RL has been successfully employed to solve problems that other disciplines could not address individually [15]. A related factor that limits the influence of RL principles in AI is the belief of that these principles are computationally too weak to be much of use [16]. In RL, a computer system learns how to accomplish a goal by trial-and-error interactions with its search space.

III. BINARY PARTICLE SWARM OPTIMIZATION

Particle swarm optimization (PSO) is a population-based stochastic optimization technique developed by Kennedy and Eberhart in 1995 [17]. PSO has been developed through simulation of the social behavior from 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 as "an individual bird of the flock". Each particle uses their memories 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 numbers 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, Kennedy and Eberhart developed binary PSO (BPSO) to overcome this problem [18]. The basic elements of BPSO are briefly introduced below:

- *Population*: A swarm (population) consists of N particles.
- *Particle position,* x_i : Each candidate solution is represented by a D-dimensional vector; the i^{th} particle is 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.
- *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 $[V_{\min}, V_{\max}]^D$.
- *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.
- *Individual best*, $pbest_i$: $pbest_i$ is the position of the i^{th} particle with the highest fitness value at a given iteration.
- *Global best, gbest*: The best position of all *pbest* particles is called global best.
- *Stopping criterion*: The process is stopped after the maximum allowed number of iterations is reached.

A. 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 Fig. 4, if $gene_3$, $gene_5$ and $gene_6$ are the last genes of $operon_1$, $operon_2$ and $operon_3$ respectively, the elements of the array are 110100. 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 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 the adjacent genes must be on the same strand. If the distance between adjacent genes is greater than the random threshold, we assumed that the two adjacent genes are within a different operon. Adjacent genes on different strands are considered NOP. Fig. 5 illustrates these criteria.

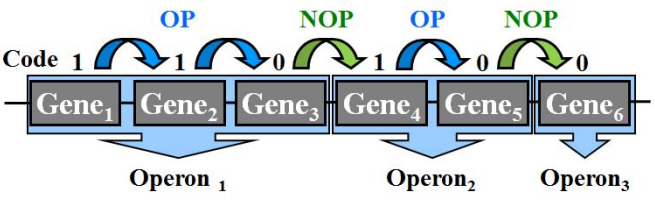

Figure 4. Encoding.

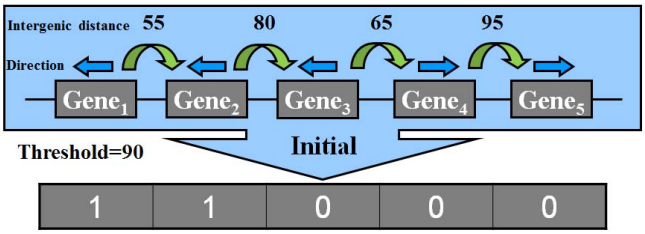

Figure 5. Initial population.

B. Particle update

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

$$v_{id}^{new} = w \times v_{id}^{old} + c_1 \times r_1 \times \left(pbest_{id} - x_{id}^{old}\right) + c_2 \times r_2 \times \left(gbest_{id} - x_{id}^{old}\right) \quad (2)$$

$$\text{if } v_{id}^{new} \notin \left(V_{\min}, V_{\max}\right) \text{ then } v_{id}^{new} = \max\left(\min\left(V_{\max}, v_{id}^{new}\right), V_{\min}\right) \quad (3)$$

$$S\left(v_{id}^{new}\right) = \frac{1}{1 + e^{-v_{id}^{new}}} \quad (4)$$

$$\text{if } \left(r_3 < S\left(v_{id}^{new}\right)\right) \text{ then } x_{id}^{new} = 1 \text{ else } x_{id}^{new} = 0 \quad (5)$$

where w is the inertia weight that controls the impact of the previous velocity of a particle. c_1 and c_2 are the acceleration constants that control the distance of a particle that moves at each generation; r_1, r_2 and r_3 are the three random numbers between [0, 1]. v_{id}^{new} and v_{id}^{old} represent the velocity of the new and old particles respectively. Particles of 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. 3 is limited within $[V_{\min}, V_{\max}]^D$. The positions of the updated particles are calculated by Eq. 4-5 [19]. 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).

C. Fitness function

As stated previously, many properties are used to predict operons. In this study, three properties are used and described individually in the following section. The pair-scores of the intergenic distance, the metabolic pathway, and the gene length ratio are calculated by the logarithmic likelihood ratio test.

1) Intergenic distance: the score of each separated interval in 10bps bins [20] is calculated based on an intergenic distance from -100 bps to 300bps using the following equation:

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

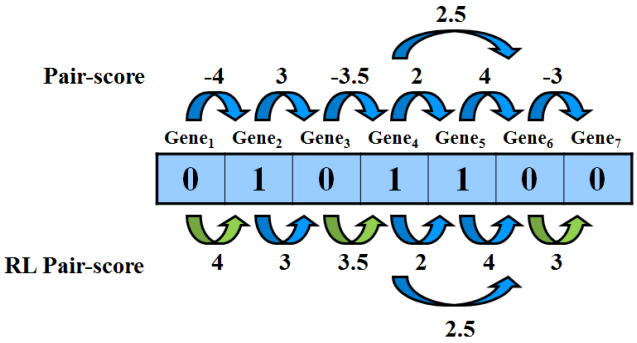

Figure 6. RL pair-score switch.

where $N_{OW}(property)$ and $N_{TUB}(property)$ correspond to the number of WO and TUB pairs in the interval distance (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 Eq. 6 is used to calculate the pathway pair-score.

3) The gene length ratio: The pair-score of the gene length ratio is calculated as the natural logarithm of the length ratio of upstream genes and downstream genes [14]. It is defined by the following equation:

$$\mathrm{LL}_{glr}\left(gene_i, gene_j\right) = \ln\left(\frac{length_i}{length_j}\right) \quad (7)$$

where $length_i$ and $length_j$ are the length of the upstream and downstream gene respectively.

In this study, if the gene pair is a NOP, the obtained lower pair-score must be rewarded by multiplying it by -1. As shown in Fig. 6, $Gene_1$, $Gene_3$ and $Gene_6$ are NOPs, and the obtained RL pair-score of the three genes were 4, 3.5 and 3 after multiplication with -1 respectively. Thus the fitness values of three genes obtain rewarded, therefore they evaluate each particle accurately in the swarm.

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:

$$fitness_c = \sum_{i=1}^{m-1}\left(d_i\right) + \left(\frac{\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}\left(\mathrm{LL}_{path}\left(gene_i, gene_j\right) + \mathrm{LL}_{COG}\left(gene_i, gene_j\right)\right)}{n} + \right) \times m \quad (8)$$

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:

$$fitness = \sum_{i=1}^{c} fitness_i \quad (9)$$

where c is the number of operons in a particle.

D. 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 [21], and V_{max} and V_{min} were 6 and -6, respectively [18].

IV. Results and Discussion

A. Performance measurement

In this study, the *E. coli* genome was used to estimate the fitness value, and then the accuracy tests were conducted on the 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 *E. coli* genome, it is easy to build a predictor that clearly identifies WO and TUB pairs. Most previous efforts focused on the operon prediction of *E. coli* genome. This led to an extensive database of the experimental 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 [14]. In order to verify the generalization ability of our method, the test of data sets does not contain the *E. coli* genome which has genome-specific properties. The predictive performance [14] was evaluated based on the sensitivity and specificity shown in Table I. As in Table II, true positive (TP) and false negative (FN) are the numbers of both 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 both 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. 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 was noted that a good balance between the sensitivity and specificity was achieved.

TABLE I. 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 II. The positive and negative evaluation

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

B. Comparison to other methods

BPSO with RL was applied to search for the best putative operon at each generation. The best putative operon identified by the search was then compared to experimental verified operons. As shown in Table III, the prediction accuracy of the proposed method obtained the highest accuracy values on the *B. subtilis* (92.8%), *P. aeruginosa* PA01 (94.3%), and *S. aureus* (95.9%) 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, ODB and BPSO obtained a higher specificity than BPSO with RL on the *B. subtilis* and *P. aeruginosa* PA01 genome, respectively. However, BPSO with RL obtained a good balance between sensitivity and specificity. Hence, the prediction results of BPSO with RL are not only superior to ODB and BPSO, but also better in terms of accuracy, sensitivity, and specificity when compared to other literature methods.

C. Discussion

Most methods use the properties of adjacent genes to identify OP or NOP for operon prediction. However, these prediction methods do not take the properties of near genes into account, and thus generally resulting in lower accuracies for operon prediction. The BPSO 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 BPSO prediction performance, this study limits the velocity of BPSO between V_{min} and V_{max}. If the velocity is close to 0, the probability of a state changing is increased *vice versa*. Hence, BPSO has global and local search capabilities. The probability of obtaining the best solution is increased.

Operon prediction accuracy is increased if better particles are selected in the initial step since the benefits of the initially superior particle are multiplied through the repeating updated process at each generation. In our study, the intergenic distance and the gene strand condition were evaluated in the initiation step. As shown in Table III, BPSO obtained a higher specificity and lower sensitivity when the initiation threshold was set to 300 bps. When the threshold was adjusted to 600 bps, the sensitivity rose up, but the specificity reduced. A sensitivity and specificity value higher than 80% represents a good balance between the two parameters [6]. In order to obtain a good balance

TABLE III. ACCURACY, SENSITIVITY, AND SPECIFICITY OF OPERON PREDICTION ON THREE GENOMES

Genome	Methodology	Accuracy (%)	Sensitivity (%)	Specificity (%)
B. subtilis (NC_000964)	BPSO with RL	**92.8**	90.3	94.8
	BPSO (initiation threshold = 600bps) [22]	92.1	**93.0**	89.9
	BPSO (initiation threshold = 300bps) [22]	90.5	88.7	94.5
	UNIPOP [1]	79.2	78.2	82.1
	GA [2]	88.3	87.3	89.7
	Classification [14]	90.2	N/A	N/A
	SVM [6]	88.9	90.0	86.0
	ODB [9]	63.2	49.9	**99.2**
	DVDA [23]	48.5	31.9	93.2
	OFS [24]	68.3	76.5	43.9
	VIMSS [25]	78.0	76.4	87.1
	FGA [4]	88.2	N/A	N/A
	JPOP [26]	74.6	72.0	90.0
	OPERON [27]	62.9	53.1	89.2
	FGENESB (http://www.softberry.com)	77.1	72.1	90.4
P. aeruginosa PA01 (NC_002516)	BPSO with RL	**94.3**	**94.4**	94.1
	BPSO (initiation threshold = 600bps) [22]	93.3	93.0	**95.1**
	BPSO (initiation threshold = 300bps) [22]	91.0	88.5	**95.1**
	GA [2]	81.3	87.0	76.3
S. aureus (NC_002952)	BPSO with RL	**95.9**	**95.9**	**95.9**
	BPSO (initiation threshold = 600bps) [22]	**95.9**	**95.9**	95.8
	BPSO (initiation threshold = 300bps) [22]	93.6	92.4	95.8
	Genome-wide operon prediction in *Staphylococcus aureus* [3]	92.0	N/A	N/A

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

between sensitivity and specificity and increasing the accuracy of operon prediction, proper settings at the initiation step are the critical importance. By boosting the quality of particles from the initiation, the best particles are obtained by successive progression through the generations.

Generally, the prediction accuracy is proportional to the fitness value of a particle. Although adjacent genes have the 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. Therefore, the fitness function used in this study was designed based on RL. Essentially, if an adjacent gene has a higher pair-score, then this pair-score will be multiplied by -1 in order to change the fitness value of gene into a lower pair-score. As shown in Table III, BPSO with RL obtained a better prediction performance than BPSO and the other methods from the literature. In addition, both BPSO without RL and BPSO 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 80% [6]. By boosting the quality of particles at the initiation, the best particles are obtained by successful progression through the generations. In addition, 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 BPSO with RL and BPSO prove that this fitness function identify better particles.

Experimental data on the *E. coli* genome can be downloaded from the RegulonDB database, but for other genomes extensive experimental data are readily not available. In order to apply the proposed method to the 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. Even though many methods in the literature used numerous properties, our method only used three properties. BPSO with RL achieves better results. ODB uses four properties for operon prediction but obtained lower prediction sensitivity [1]. The results reveal that the pathway and the gene length ratio property are more suitable for identification of WO and TUB pairs. Since adjacent gene shared a common pathway, the probability of a gene pair to be a WO pair is very high [4]. The powerful prediction ability of gene length ratio is also verified [14]. Our method achieved the highest accuracy for operon prediction even though only three properties were used on all bacterial genomes. The operon prediction contributions are self-evident.

V. CONCLUSION

In this study, BPSO with RL was proposed to predict operons in bacterial genomes. First of all, we used the three properties of the *E. coli* genome to design a fitness function. The concept of RL is employed to improve the fitness function, and thus each particle can be evaluated accurately. The experimental results show that the proposed method increases the accuracy of operon prediction in three test genome data sets. We intend to investigate the other algorithms and different properties for the problems of operon prediction to increase the performance of prediction further.

REFERENCES

[1] G. Li, D. Che, and Y. Xu, "A universal operon predictor for prokaryotic genomes," Bioinformatics and Computational Biology, vol. 7, pp. 19-38, Feb 2009.

[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, pp. 151-9, Oct 2007.

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

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

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

[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, pp. 233-40, Jun 2006.

[7] 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, pp. D120-D124, 2007.

[8] 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, pp. D93-D96, Jan 2008.

[9] 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, pp. D358-D362, Jan 1 2006.

[10] 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, pp. 2886-93, Jul 1 2002.

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

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

[13] 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, pp. 11-20, 2007.

[14] 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, pp. 288-98, 2007.

[15] M. Harmon and S. Harmon, "Reinforcement learning: a tutorial," Citeseer, 1997.

[16] A. Barto and R. Sutton, "Reinforcement learning in artificial intelligence," Neural-Networks Models of Cognition, vol. 121, pp. 358-386, 1997.

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

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

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

[20] 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, pp. 709-17, Mar 22 2004.

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

[22] L. Chuang, J. Tsai, and C. Yang, "Binary particle swarm optimization for operon prediction," Nucleic acids research, doi:10.1093/nar/gkq204, 2010.

[23] M. T. Edwards, S. C. Rison, N. G. Stoker, and L. Wernisch, "A universally applicable method of operon map prediction on minimally annotated genomes using conserved genomic context," Nucleic Acids Res., vol. 33, pp. 3253-62, 2005.

[24] B. P. Westover, J. D. Buhler, J. L. Sonnenburg, and J. I. Gordon, "Operon prediction without a training set," Bioinformatics, vol. 21, pp. 880-8, Apr 1 2005.

[25] M. N. Price, K. H. Huang, E. J. Alm, and A. P. Arkin, "A novel method for accurate operon predictions in all sequenced prokaryotes," Nucleic Acids Res., vol. 33, pp. 880-92, 2005.

[26] 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, pp. 2147-57, 2004.

[27] M. D. Ermolaeva, O. White, and S. L. Salzberg, "Prediction of operons in microbial genomes," Nucleic Acids Res., vol. 29, pp. 1216-21, Mar 1 2001.