Journal of King Saud University – Computer and Information Sciences (2011) **23**, 1–6

King Saud University

Journal of King Saud University – Computer and Information Sciences

www.ksu.edu.sa
www.sciencedirect.com

ORIGINAL ARTICLE

A Quantum Swarm Evolutionary Algorithm for mining association rules in large databases

Mourad Ykhlef

King Saud University, College of Computer and Information Sciences, Saudi Arabia

Received 4 April 2009; accepted 22 March 2010
Available online 8 December 2010

KEYWORDS

Quantum Evolutionary Algorithm;
Swarm intelligence;
Association rule mining;
Fitness

Abstract Association rule mining aims to extract the correlation or causal structure existing between a set of frequent items or attributes in a database. These associations are represented by mean of rules. Association rule mining methods provide a robust but non-linear approach to find associations. The search for association rules is an NP-complete problem. The complexities mainly arise in exploiting huge number of database transactions and items. In this article we propose a new algorithm to extract the best rules in a reasonable time of execution but without assuring always the optimal solutions. The new derived algorithm is based on Quantum Swarm Evolutionary approach; it gives better results compared to genetic algorithms.

1. Introduction

Data mining methods such as association rule mining (Agrawal et al., 1993a,b) are gaining popularity for their power and ease of use. Association rule learning methods provide a robust and non-linear approach to find associations (correlations) and causal structures among sets of frequent items or attributes in a database. Association rule algorithms, such as Apriori (Agrawal et al., 1993a,b), examine a long list of transactions in order to determine which items are most frequently purchased together. The challenge of extracting association patterns from data draws upon research in databases, machine learning and optimization to deliver advanced intelligent solutions. The algorithms for performing association rule mining are NP-complete as they were proved in Angiulli et al. (2001), the authors of Angiulli et al. (2001) have shown that association rule mining can be reduced to finding a CLIQUE in a graph which is NP-complete. The complexities mainly arise in exploiting huge number of items and database transactions.

Many algorithms have been proposed for mining association rules; we can categorize these algorithms into two branches: (1) Exact algorithms such as Apriori (Agrawal et al., 1993a,b) and FP-Growth (Pei et al., 2000). These algorithms guaranty the optimal solution despite the time required to obtain that solution. (2) Evolutionary algorithms (Lopes et al., 1999; Melab and El-Ghazali, 2000), which give good solution and may be non-optimal ones but in a reasonable time (polynomial) of execution.

E-mail address: ykhlef@ksu.edu.sa

1319-1578

Peer review under responsibility of King Saud University.
doi:10.1016/j.jksuci.2010.03.001

Production and hosting by Elsevier

Association rule mining in large databases is a very complex process and exact algorithms are very expensive to use. We think that evolutionary computing provides much help in this arena. In this article, we address the issue of using a Quantum Swarm Evolutionary Algorithm (QSE) (Wang et al., 2006) for mining association rules. QSE is a hybridization of Quantum Evolutionary Algorithm (QEA) (Han and Kim, 2002) and particle swarm optimization (PSO) (Kennedy and Eberhart, 1995).

QEA approach is better than classical evolutionary algorithms like genetic algorithm, instead of using binary, numeric or symbolic representation; QEA uses a Q-bit as a probabilistic representation, defined as the smallest unit of information. A Q-bit individual is defined by a string of Q-bits called multiple Q-bits. The Q-bit individual has the advantage that it can represent a linear superposition of states (binary solutions) in search space probabilistically. Thus, the Q-bit representation has a better characteristic of population diversity than chromosome representation used in genetic algorithm. A Q-gate is also defined as a variation operator of QEA to drive the individuals toward better solutions and eventually toward a single state.

QSE (Wang et al., 2006) employs a novel quantum bit expression mechanism called quantum angle and adopted the improved PSO to update Q-bit of QEA automatically. The authors of Wang et al. (2006) prove that QSE is better than QEA.

The remainder of this article is organized as follows: Section 2 presents basics of association rule mining. In Section 3, we give a general description of quantum computing and particle swarm optimization. In Section 4, we present a new approach to mine association rules. Section 5 illustrates our experimental results.

2. Association rule mining

2.1. Problem definition

Association rule mining is formally defined as follows. Let $I = \{i_1, i_2, \ldots, i_m\}$ be a set of Boolean attributes called items and $S = \{s_1, s_2, \ldots, s_n\}$ be a multi-set of records representing data instances or transactions, where each record or data instance $s_i \in S$ is constituted from the non-repeatable attributes from I. The presence of a Boolean attribute in a data instance s_i means that its value is 1, if it is absent, its value is set to 0. For example, let $I = \{A, B, C\}$ be a set of Boolean attributes and let $S = \{\langle A, B\rangle, \langle C\rangle, \langle C\rangle\}$ be a multi-set of data instances, the multi-set S can be rewritten as follows:

$$S = \{\langle A = 1, B = 1, C = 0\rangle, \langle A = 0, B = 0, C = 1\rangle, \langle A = 0, B = 0, C = 1\rangle\}$$

For categorical attribute, instead of having one attribute in I, we have as many attributes as the number of attribute values. For example, the more general multi-set of data instances S given by:

{⟨height-166 = 1, height-170 = 0, height-174 = 0, gender-male = 0, gender-female = 1⟩,
⟨height-166 = 0, height-170 = 1, height-174 = 0, gender-male = 1, gender-female = 0⟩,
⟨height-166 = 0, height-170 = 0, height-174 = 1, gender-male = 0, gender-female = 1⟩}

is intended to abstract a multi-set of three data instances having two categorical attributes: height and gender. The values of (height, gender) are {(166, female), (170, male), (174, female)}, respectively.

An association rule is denoted by IF C THEN P when C states for Condition(s) and P for Prediction(s) where C, $P \subset I$ and $C \cap P = \emptyset$.

In this article we are particularly interested by the conjunctive association rules where C is a conjunction of one or more condition(s) and P is also a conjunction of one or more prediction(s). The following notations are used in the remainder of the article:

- $|C|$: The number of data instances which are covered by (i.e. satisfying) the C part of the rule.
- $|P|$: The number of data instances which are covered by the P part of the rule.
- $|C\&P|$: The number of data instances which are covered by both the C part and the P part of the rule.
- N: The total number of data instances being mined.

The confidence b of a rule is the probability of the occurrence of P knowing that C is observed; b is equal to $\frac{|C\&P|}{|C|}$. The prediction frequency a is equal to $\frac{|P|}{N}$. Note that the support is equal to the fraction $\frac{|C\&P|}{|N|}$.

2.2. Fitness function

The quality of a candidate rule is evaluated by means of a fitness function. Several fitness functions have been defined in the literature (Agrawal et al., 1993a,b; Lopes et al., 1999). They can be basic or complex. An example of a basic function is the support of a rule (the percentage of data instances satisfying the C part of the rule) and the confidence factor (the percentage of data instances satisfying the implication IF C THEN P). It is claimed that such basic fitness function is not sufficient. In this article we adopt the complex fitness function of Lopes et al. (1999). This function is derived from information theory and it is based on *J-measure* J_m given by:

$$J_m = \frac{|C|}{N} * \left(b * \log\left(\frac{b}{a}\right)\right)$$

The fitness function F is the following:

$$F = \frac{w_1 * (J_m) + w_2 * \left(\frac{n_{pu}}{n_T}\right)}{w_1 + w_2}$$

where n_{pu} is the number of potentially useful attributes. A given attribute A is said to be potentially useful if there is at least one data instance having both the A's value specified in the part C and the prediction attribute(s). The term n_T is the total number of attributes in the part C of the rule; w_1, w_2 are user defined weights set to 0.6 and 0.4, respectively.

3. Quantum computing and particle swarm optimization

Quantum computing (QC) is an emergent field calling upon several specialties: physics, engineering, chemistry, computer science and mathematics. QC uses the specificities of quantum mechanics for processing and transformation of data stored in two-state quantum bits or Q-bit(s) for short. A Q-bit can take state value 0, 1 or a superposition of the two states at the same time. The state of a Q-bit can be represented as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ where α and β are the amplitudes of $|0\rangle$ and $|1\rangle$, respectively, in this state. When we measure this Q-bit, we see $|0\rangle$ with probability $|\alpha|^2$, and $|1\rangle$ with probability $|\beta|^2$ such that $|\alpha|^2 + |\beta|^2 = 1$.

The idea of superposition makes it possible to represent an exponential whole of states with a small number of Q-bits. According to the quantum laws like interference, the linearity of quantum operations makes the quantum computing more powerful than the classical machines.

In order to exploit effectively the power of quantum computing, it is necessary to create efficient quantum algorithms. A quantum algorithm consists in applying a succession of quantum operations on quantum systems. Shor (1994) demonstrated that QC could solve efficiently NP-complete problems by describing a polynomial time quantum algorithm for factoring numbers.

One of the most known algorithms is Quantum-inspired Evolutionary Algorithm (QEA) (Han and Kim, 2002), which is inspired by the concept of quantum computing. This algorithm has been first used to solve knapsack problem (Han and Kim, 2002) and then it has first used to solve different NP-complete problems like Traveling Salesman Problem (Talbi et al., 2004) and Multiple Sequence Alignment (Layeb et al., 2006, 2008).

Meanwhile, particle swarm optimization (PSO) has demonstrated a good performance in many functions and parameter optimization problems. PSO is a population-based optimization strategy. It is initialized with a group of random particles and then updates their velocities and positions with the following formula:

$$\begin{aligned} v(t+1) &= v(t) + c1 * rand() * (pbest(t) - present(t)) \\ &\quad + c2 * rand() * (gbest(t) - present(t)) \\ present(t+1) &= present(t) + v(t+1) \end{aligned}$$

where $v(t)$ is the particle velocity, $present(t)$ is the current particle. $pbest(t)$ and $gbest(t)$ are defined as individual best and global best. $rand()$ is a random number between $[0, 1]$. $c1$, $c2$ are learning factors; usually $c1 = c2 = 2$ (Wang et al., 2006).

In the next section we will tailor the hybrid Quantum Swarm Evolutionary Algorithm (QSE) (Wang et al., 2006) to the problem of mining association rules.

4. The QSE-RM approach

In this section we first present QEA-RM for association rule mining and then we give a PSO version of QEA-RM named QSE-RM.

In order to show how QEA concepts have been tailored to the problem of association rule mining, a formulation of the problem in terms of quantum representation is presented and a Quantum Swarm Evolutionary Algorithm for association rules mining QSE-RM is derived.

4.1. Quantum representation

QEA-RM uses the novel representation based on the concept of string of Q-bits called multiple Q-bit defined as below:

$$Q = \left[\begin{array}{c|c|c|c|} \alpha_1 & \alpha_2 & \cdots & \alpha_m \\ \beta_1 & \beta_2 & & \beta_m \end{array} \right]$$

where $|\alpha_t|^2 + |\beta_t|^2 = 1$, $t = 1, \ldots, m$, m is the number of Q-bits. Quantum Evolutionary Algorithm with the multiple Q-bit representation has a better diversity than classical genetic algorithm since it can represent superposition of states. Only one multiple Q-bit with three Q-bits such as:

$$\left[\begin{array}{c|c|c|} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{2} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & \frac{\sqrt{3}}{2} \end{array} \right]$$

is enough to represent the following system with eight states:

$$\frac{1}{4}|000\rangle + \frac{\sqrt{3}}{4}|001\rangle - \frac{1}{4}|010\rangle - \frac{\sqrt{3}}{4}|011\rangle + \frac{1}{4}|100\rangle + \frac{\sqrt{3}}{4}|101\rangle - \frac{1}{4}|110\rangle - \frac{\sqrt{3}}{4}|111\rangle$$

This means that the probabilities to represent the states $|000\rangle$, $|001\rangle, |010\rangle, |011\rangle, |100\rangle, |101\rangle, |110\rangle, |111\rangle$ are 1/16, 3/16, 1/16, 3/16, 1/16, 3/16, 1/16, 1/16 respectively. However in genetic algorithm one needs eight chromosomes for encoding.

For the data instances S of Section 2.1 given by $S = \{\langle A=1, B=1, C=0\rangle, \langle A=0, B=0, C=1\rangle, \langle A=0, B=0, C=1\rangle\}$ one would have a multiple Q-bits representation constituted from 3 Q-bits.

4.2. Measurement

The measurement of single Q-bit projects the quantum state onto one of the basis states associated with the measuring device. The process of measurement changes the state to that measured. The multiple Q-bit measurement can be treated as a series of single Q-bit measurements to yield a binary solution P. In association rules, the occurrence of 1 in P means that the corresponding item or the attribute value is present in P however 0 means that the corresponding item or attribute value is absent from P.

4.3. Structure of QEA-RM

The Quantum-inspired Evolutionary Algorithm for association rules mining (QEA-RM) is described as follows:

Procedure QEA-RM
begin
  $t \leftarrow 0$
  initialize population of Q-bit individuals $Q(t)$
  project $Q(t)$ into binary solutions $P(t)$
  compute fitness of $P(t)$
  generate association rule from each $P(t)$ if there is any
  store the best solutions among $P(t)$
  while (not end-condition) **do**
    $t \leftarrow t + 1$
    project $Q(t-1)$ into binary solutions $P(t)$
    compute fitness from $P(t)$
    generate association rule from each $P(t)$ if there is any
    update $Q(t)$ using Q-gate
    store the best solutions among $P(t)$
  end
end

Table 1 Lookup table.

xi	bi	$f(x) \geqslant f(b)$	$\Delta\theta_i$	$s(\alpha_i, \beta i)$			
				$\alpha i\beta_i > 0$	$\alpha_i\beta_i < 0$	$\alpha_i = 0$	$\beta_i = 0$
0	0	False	0	0	0	0	0
0	0	True	0	0	0	0	0
0	1	False	0	0	0	0	0
0	1	True	Delta	−1	+1	±1	0
1	0	False	Delta	−1	+1	±1	0
1	0	True	Delta	+1	−1	0	±1
1	1	False	Delta	+1	−1	0	±1
1	1	True	Delta	+1	−1	0	±1

In the step "initialize population of Q-bit individuals $Q(t)$" the values of α_i and β_i are initialized with $1/\sqrt{2}$. The step "project $Q(t)$ into binary solutions $P(t)$" generates binary solutions by observing the states of population $Q(t)$; for each bit in multiple Q-bit we generate a random variable between 0 and 1; if random$(0,1) < |\beta_i|^2$ then we generate 1 else 0 is generated. In the step "compute fitness of $P(t)$", each binary solution $P(t)$ is evaluated for the fitness value computed by the formula F of Section 2.2. The step "update $Q(t)$ using Q-gate" is introduced as follows (Han and Kim, 2002):

Procedure update $Q(t)$
begin
  $i \leftarrow 0$
  while $(i < m)$ **do**
    $i \leftarrow i + 1$
    determine $\Delta\theta_i$ with the lookup table
    $[\alpha'_i \quad \beta'_i]^T = U(\Delta\theta_i)[\alpha_i \quad \beta_i]^T$
  end
end

Quantum gate $U(\Delta\theta 1pt_i)$ is a variable operator, it can be chosen according to the problem. We use the quantum gate defined in Han and Kim (2002) as follows:

$$U(\Delta\theta_i) = \begin{vmatrix} \cos(\xi(\Delta\theta_i)) & -\sin(\xi(\Delta\theta_i)) \\ \sin(\xi(\Delta\theta_i)) & \cos(\xi(\Delta\theta_i)) \end{vmatrix}$$

where $\xi(\Delta\theta_i) = s(\alpha_i, \beta_i) * \Delta\theta_i$; $s(\alpha_i, \beta_i)$ and $\Delta\theta_i$ represents the rotation direction and angle, respectively. The lookup table is presented in Table 1, Delta is the step size and should be designed in compliance with the application problem. However, it has not had the theoretical basis till now, even though it usually is set as small value. Many applications set Delta $= 0.01\pi$. The function $f(x)$ (resp. $f(b)$) is the profit of the binary solution x (resp. best solution b). For example, if the condition $f(x) \geqslant f(b)$ is satisfied and xi, bi are 1 and 0, respectively, we can set the value of $\Delta\theta_i$ as 0.01π and $s(\alpha_i, \beta_i)$ as $+1$, -1, or 0 according to the condition of α_i, β_i; so as to increase the probability of the state $|1\rangle$.

4.4. Structure of QSE-RM

In order to introduce QSE-RM we present quantum angle. A quantum angle (Wang et al., 2006) is defined as an arbitrary angle θ and a Q-bit is presented as $[\theta]$. Then $[\theta]$ is equivalent to the original Q-bit as $\begin{bmatrix} \frac{\sin(\theta)}{\cos(\theta)} \end{bmatrix}$. It satisfies the condition:

$|\sin(\theta)|^2 + |\cos(\theta)|^2 = 1$.

Then a multiple Q-bit $\left[\begin{matrix} \alpha_1 \\ \beta_1 \end{matrix}\middle| \begin{matrix} \alpha_2 \\ \beta_2 \end{matrix}\middle| \begin{matrix} \ldots \end{matrix} \begin{matrix} \alpha_m \\ \beta_m \end{matrix}\right|\Big]$could be replaced by: $[\theta_1| \ \theta_2| \ \ldots| \ \theta_m]$.

The common rotation gate

$[\alpha'_i \quad \beta'_i]^T = U(\Delta\theta_i)[\alpha_i \quad \beta_i]^T$

where $U(\Delta\theta_i) = \begin{vmatrix} \cos(\xi(\Delta\theta_i)) & -\sin(\xi(\Delta\theta_i)) \\ \sin(\xi(\Delta\theta_i)) & \cos(\xi(\Delta\theta_i)) \end{vmatrix}$, is replaced by $[\theta'_i] = [\theta_i + \xi(\Delta\theta_i)]$.

QSE-RM uses the concept of swarm intelligence of the PSO and regards all multiple Q-bit in the population as an intelligent group, which is named quantum swarm. First QSE-RM finds the local best quantum angle and the global best value from the local ones. Then according to these values, quantum angles are updated by quantum gate. The QSE-RM based on QEA-RM is given as follows:

1. Use quantum angle to encode Q-bit $Q(t)$ using $Q(t) = \{q^t_1, q^t_2, \ldots, q^t_m\}$ and $q^t_i = [\theta^t_{j1}|\theta^t_{j2}|\ldots|\theta^t_{jm}]$
2. Project $Q(t)$ into binary solutions $P(t)$ by observing the state of $Q(t)$ through $|\cos(\theta)|^2$ as follows: for quantum angle, we generate a random variable between 0 and 1; if random$(0,1) > |\cos(\theta)|^2$ then we generate 1 else 0 is generated.
3. The "update $Q(t)$ using Q-gate" is modified with the following PSO formula (Wang et al., 2006):

$$v^{t+1}_{ji} = \chi * (\omega * v^t_{ji} + c1 * rand() * (\theta^t_{ji}(pbest) - \theta^t_{ji}) + c2 * rand() * (\theta^t_i(gbest) - \theta^t_{ji}))$$
$$\theta^{t+1}_{ji} = \theta^t_{ji} + v^{t+1}_{ji}$$

where v^t_{ji}, θ^t_{ji}, $\theta^t_{ji}(pbest)$ and $\theta^t_i(gbest)$ are the velocity, current position, individual best and global best of the ith Q-bit of the jth multiple Q-bit. The parameters χ, ω, $c1$, $c2$ are, respectively, set to 0.99, 0.7298, 1.42, 1.57.

5. Test and evaluation

In this section we compare Quantum Swarm Evolutionary Algorithm (QSE-RM) to the non-parallel version of Genetic Algorithm (GA-PVMINER) (Lopes et al., 1999). Since the parameters of QSE-RM are different from the parameters of GA-PVMINER, the comparison between QSE-RM and GA-PVMINER is done by fixing a threshold of time

Table 2 Structure of the Nursery School database.

	Attribute name	Attribute values
1	Parents	Usual, pretentious, great_pret
2	Has_nurs	Proper, less_proper, improper, critical, very_crit
3	Form	Complete, completed,incomplete, foster
4	Children	1, 2, 3, more
5	Housing	Convenient, less_conv, critical
6	Finance	Convenient, inconv
7	Social	Non-prob, slightly_prob, problematic
8	Health	Recommended, priority, not_recom
9	Recommendation	Not_recom, recommend, very_recom, priority, spec_prior

execution. In the remainder of this section, we will see that for the same goal and for the same time of execution, QSE-RM has generated rules with fitness better than the fitness of rules given by GA-PVMINER. Recall that QSE-RM and GA-PVMINER algorithms belong to the class of evolutionary algorithms. Evolutionary algorithms give good solution and may be non-optimal ones but in a reasonable time (polynomial) of execution. All the tests were performed on 1.86 GHz Intel® Centrino™ PC machine with 1.00 GB RAM, running on Windows XP platform. QSE-RM algorithm is written with MATLAB programming language. The dataset used for testing, namely the nursery school dataset, is a public domain and available from UCI repository (http://www.archive.ics.uci.edu/ml/) of machine learning. Nursery database was derived from a hierarchical decision model originally developed to rank applications for nursery schools (Bohanec and Rajkovic, 1990).

The Nursery database contains 12,960 instances and 9 attributes, all of them categorical. The structure of Nursery database is given in Table 2.

As it is done in Lopes et al. (1999) we have specified three goal attributes, namely Recommendation, Social and Finance. A threshold of execution time is fixed. In all cases, our results are better than those found by GA-PVMINER.

Table 3 Results for goal Recommendation = not_recom.

	Rule	$\lvert C\&P\rvert$	b	Fitness	J-measure
1	IF Housing = convenient AND Finance = inconv THEN Recommendation = not_recom	720	0.33	0.40003	0.00005144
2	IF Parents = great_pret AND Has_nurs = proper AND Children = 2 AND Housing = less_conv AND Finance = inconv AND Social = nonprob AND Health = not_recom THEN Recommendation = not_recom	4	1	0.40036	0.00059339
3	IF Parents = great_pret AND Health = not_recom THEN Recommendation = not_recom	1440	1	0.40010	0.00016954
4	IF Health = not_recom THEN Recommendation = not_recom	4320	1	0.40005	0.00008476

Table 4 Results for goal Recommendation = spec_prior.

	Rule	$\lvert C\&P\rvert$	b	Fitness	J-measure
1	IF Has_nurs = very_crit AND Health = priority THEN Recommendation = spec_prior	855	0.98	0.40011	0.00017626
2	IF Parents = pretentious AND Has_nurs = very_crit AND Children = 1 AND Housing = critical AND Finance = convenient AND Social = slightly_prob AND Health = priority THEN Recommendation = spec_prior	4	1	0.40038	0.00062905

For the goal "*Recommendation = not_recom*", the best rule found by GA-PVMINER is given in the first row of Table 3. In addition to this rule, our algorithm QSE-RM has discovered other more interesting rules, which are given in rows 2, 3 and 4 of Table 3. For example, the following rule is very important than the best rule given by GA-PVMINER:

$$\text{“}\textit{IF Health} = \textit{not\_recom THEN Recommendation} = \textit{not\_recom}\text{”}$$

with support $|C\&P| = 4320$, confidence $b = 1$ and fitness = 0.40005.

For the goal "*Recommendation = spec_prior*", the best rule found by GA-PVMINER is given in the first row of Table 4. In addition to this rule, our algorithm QSE-RM has discovered other more interesting rule with fitness = 0.40038 (see row 2 of Table 4).

The authors of Lopes et al. (1999) stated that the best rule found by their GA-PVMINER algorithm is:

$$\text{“}\textit{IF Has\_nurs} = \textit{very\_crit AND Health} = \textit{priority THEN Recommendation} = \textit{spec\_prior}\text{”}$$

with confidence $b = 0.9$ and fitness = 0.4. The following rule is more important than the previous rule for the support reason:

$$\text{“}\textit{IF Finance} = \textit{inconv AND Health} = \textit{not\_recom THEN Recommendation} = \textit{not\_recom}\text{”}$$

with support $|C\&P| = 2160$, confidence $b = 1$ and fitness = 0.400.

Concerning the goals Social and Finance our results are also better than those found by GA-PVMINER.

6. Conclusion

In this article, we discussed the use of Quantum Swarm Evolutionary approach (Wang et al., 2006) to improve the process of mining association rules. A derived algorithm QSE-RM is proposed. The experimental studies prove the effectiveness QSE-RM algorithm comparing with PVMINER (Lopes et al., 1999). As ongoing work we study the effect of parallelization of QSE-RM in the same spirit of PGA-RM (Melab and El-Ghazali, 2000) and we plan to add more hybridization to QSE-RM.

References

Agrawal, R., Imielinski, T., Swami, S., 1993a. Mining association rules between sets of items in large databases. In: Buneman, P., Jajodia, S. (Eds.), Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, Washington, DC, May 26–28, pp. 207–216.

Agrawal, R., Imielinski, T., Swami, S., 1993b. Mining association rules between sets of items in large databases. SIGMOD Record 22 (2), 207–216 (ACM Special Interest Group on Management of Data).

Angiulli, F., Ianni, G., Palopoli, L., 2001. On the complexity of mining association rules. In: Proc. Nono Convegno Nazionale su Sistemi Evoluti di Basi di Dati (SEBD), pp. 177–184.

Bohanec, M., Rajkovic, V., 1990. Expert system for decision making. Sistemica 1 (1), 145–157.

Han, K.H., Kim, J.H., 2002. Quantum-inspired Evolutionary Algorithm for a class of combinatorial optimization. IEEE Transaction on Evolutionary Computation 6 (6), 580–593.

Kennedy, J., Eberhart, R.C., 1995. Particle swarm optimization. In: Proceedings of the IEEE International Conference on Neural Networks, vol. 9, Australia, pp. 2147–2156.

Layeb, A., Meshoul, S., Batouche, M., 2006. Multiple sequence alignment by quantum genetic algorithm. In: Proceedings of the IEEE Conference of the International Parallel and Distributed Processing Symposium (IPDPS'2006), Rhodes Island, Greece, April 25–29.

Layeb, A., Meshoul, S., Batouche, M., 2008. Quantum genetic algorithm for multiple RNA structural alignment. In: IEEE Proceedings of the Second Asia International Conference on Modelling and Simulation (AMS 2008), Kuala Lumpur, Malaysia, May 13–15.

Lopes, H.S., Araujo, D.L.A., Freitas, A.A., 1999. A parallel genetic algorithm for rule discovery in large databases. In: IEEE Systems, Man and Cybernetics Conf., pp. 940–945.

Melab, M., El-Ghazali, T., 2000. A parallel genetic algorithm for rule mining. In: IPDPS, IEEE Computer Society.

Pei, J., Han, J., Yin, Y., 2000. Mining frequent patterns without candidate generation. In: ACM SIGMOD Int. Conference on Management of Data.

Shor, P.W., 1994. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. In: Proceedings of the 35th Annual Symposium on Foundations of Computer Science, Santa Fe, NM, November 20–22.

Talbi, T., Draa, A., Batouche, M., 2004. A quantum inspired genetic algorithm for solving the traveling salesman problem. In: Proceedings of the IEEE ICIT 04, Tunisia, December 8–10.

Wang, Y., Feng, X., Huang, Y., Pu, D., Zhou, W., Liang, Y., Zhou, C., 2006. A novel quantum swarm evolutionary algorithm and its applications. Neurocomputing 70 (4–6), 633–640.