Applied Mathematics & Information Sciences
An International Journal

A Thread Pool Scheduling Optimization Method of Real-time System

Yi Liu

Information College, Zhongkai University of Agriculture and Engineering, Guangzhou 510225, China

Received: 7 Aug. 2012; Revised 21 Nov. 2012; Accepted 5 Dec. 2012
Published online: 1 Mar. 2013

Abstract: To deal with concurrent requests, thread pool technology is wildly used in multi-task real-time systems, which chief demand is timeout avoidance. In order to reduce request timeout ratio, least square support vector machine(LS-SVM) algorithm which parameters optimization based on hybrid quantum-behaved particle swarm optimization(HQPSO) was applied to estimate the execution time of requests. Also, a scheduling priority algorithm of thread pool was designed based on the estimation results. In performance test, a node of wireless sensor network is implemented to test the request timeout ratios of HQPSO LS-SVM based thread pool and other ones in different states. Test result indicates that HQPSO LS-SVM based thread pool has remarkable superiority of timeout avoidance while the thread amount of thread pool is set properly.

Keywords: HQPSO, LS-SVM, real-time, thread pool, scheduling

1. Introduction

Application of multithreading technology based on thread pool in real-time system can improve parallel processing ability of request, manage operations for threads, and solute coupling problems between threads. But the researches of thread pool are mainly focus on optimum performance of system, lack of researches on finishing request in due time, which is the chief demand of real-time systems.

According to [1], Least square support vector machine (LS-SVM) which was proposed by Suykens is a learning method based on the structural risk minimization principle. It converts the solving process of support vector machine(SVM) shown in [2] from quadratic programming to linear equations, having specialties of fast learning and easy to use. In [3], execution time of requests can be estimated effectively by LS-SVM, giving thread pool accordance to determine priorities of requests dynamically, and avoiding request timeout furthest. As the learning precision and generalization of LS-SVM model is determined by its parameters, algorithms for LS-SVM parameter choosing were put forward, such as genetic algorithm, particle swarm optimization (PSO) algorithm, quantum-behaved particle swarm optimization (HQPSO) algorithm, see e.g. [4−−6]. QPSO is widely used as it can search for the global optimal solution with great convergence speed. In the paper, a QPSO algorithm cooperated with Powell algorithm (HQPSO) was proposed to optimize the LS-SVM parameters according to [7]. Based on HQPSO LS-SVM algorithm, a thread pool model of real-time system for wireless sensor network's node was established for testing. The result indicates that the prediction of HQPSO LS-SVM for execution time of request decreased request timeout ratio effectively.

2. LS-SVM Algorithm

Input samples and their corresponding output samples are $x_i : x_1, x_2, \cdots, x_l \in R^n$ and $y_j : y_1, y_2, \cdots, y_l \in R$. Input samples are mapped to feature space R^m by nonlinear mapping $\Psi(x) = [\varphi(x_1), \varphi(x_2), \cdots, \varphi(x_l)]$, which converts estimation from nonlinear original space to linear feature space ($f(x) = w\varphi(x_i) + b$, w is weight vector of feature space). The regression of LS-SVM is:

$$\min_{w,\xi} J(w,\xi) = \frac{1}{2}w^T w + \frac{1}{2}\gamma\sum_{i=1}^{l}\xi_i^2 \qquad (1)$$

where γ is normalization parameter, ξ is error, constraint condition is $y_i = w\varphi(x_i) + b + \xi_i$. The corresponding La-

* Corresponding author: e-mail: baddielao@126.com

grange function is:

$$L(w,\xi,a,b) = -\sum_{i=1}^{l} a_i[w^T\varphi(x_i)+b+\xi_i-y_i] + J(w,\xi) \tag{2}$$

Defines $K(x_i,y_j)=\varphi^T(x_i)\varphi(x_j)$. Linear matrix equation (3) is obtained based on Karush-Kuhn-Tucher optimal condition.

$$\begin{bmatrix} 0 & 1 & \cdots & 1 \\ 1 & K(x_1,x_1)+\frac{1}{c} & \cdots & K(x_1,x_l) \\ \vdots & \vdots & \ddots & \vdots \\ 1 & K(x_l,x_1) & \cdots & K(x_l,x_l)+\frac{1}{c} \end{bmatrix} \begin{bmatrix} b \\ a_1 \\ \vdots \\ a_l \end{bmatrix} = \begin{bmatrix} 0 \\ y_1 \\ \vdots \\ y_l \end{bmatrix} \tag{3}$$

The regression result of LS-SVM algorithm is:

$$\hat{f}(x) = \sum_{i=1}^{l} a_i K(x_i,x)+b \tag{4}$$

Radial basis function(RBF) is the most widely used in practical applications, that is:

$$K(x_i,x_j) = \exp[-(x_i-x_j)^2/(2\sigma^2)] \tag{5}$$

In order to increase the estimate precision of thread pool model based on LS-SVM, HQPSO algorithm is applied to optimize γ and σ, which are the most important parameters of LS-SVM.

3. HQPSO algorithm and its Application on LS-SVM

According to [8], QPSO algorithm bases on potential well, it can find out the global optimal solution in its solution space. The particle renewing method is:

$$m_{best} = \frac{1}{M}\sum_{i=1}^{M} P_i \tag{6}$$

$$P_{C_{ij}} = \varphi P_{ij} + (1-\varphi)P_{gj} \tag{7}$$

$$x_{ij} = P_{C_{ij}} \pm \alpha|m_{best_j} - x_{ij}|\ln(\frac{1}{\mu}) \tag{8}$$

where m_{best} is the center of all particles' best positions; M is particle's amount; P_{ij} stands for the No. j dimension's optimal position of No. i particle; P_{gj} is the No. j dimension's optimal position in all particles; $P_{C_{ij}}$ is a random position; φ and μ are random number between 0 and 1. α is contraction-expansion coefficient; x_{ij} stands for the No. j dimension's position of No. i particle.

Powell algorithm is a pattern search method as well as conjugate direction method, it has a fast speed of convergence rate, but easily falls into local optimum. Supposed parameter $x \in R^n$ should be optimized, the method is:

Step 1: initialize x_0 and n linear independent directions $(Q_1,Q_2,\cdots,Q_n)$;

Step 2: search for the minimal point x_1 in the direction of Q_1 started from x_0, repeat till x_n is found in the direction of Q_n started from x_{n-1};

Step 3: calculate $\Delta m = \max(\varphi(x_{i-1})-\varphi(x_i))$, $i=1,2,\cdots,n$, $\varphi(x_i)$ is objective function value of x_i, m is No. i when maximum occur;

Step 4: update Q_m to the direction of (x_n-x_0) and renew x_n by minimal point in direction Q_m;

Step 5: start from x_n, repeat step 2-4 until accuracy requirement of objective function or iteration limit is met.

HQPSO algorithm searches for global optimal solution based on QPSO, and searches local optimal solution for every position by Powell algorithm during HQPSO optimizing to improve solution accuracy. HQPSO optimizing process is:

Step 1: initialize particle position x_{ij} randomly in possible range, search for optimal position of every particle by Powell algorithm and renew x_{ij};

Step 2: calculate objective function value of every particle, renew local optimal position P_{ij} of every particle, renew P_{ij} again by Powell;

Step 3: renew global optimal position P_{gj} and x_{ij};

Step 4: repeat step 2 and 3 until accuracy requirement of objective function or iteration limit is met;

Step 5: locally optimize P_{gj} by Powell algorithm, output the HQPSO optimization result.

The optimizing space of HQPSO algorithm applied for LS-SVM is a $\gamma-\sigma$ space, as the optimizing targets are LS-SVM parameters: γ and σ. Mean square error(MSE), mean absolute error(MAE) et al can be applied to be the objective function.

4. Thread pool modeling in real-time system

As shown in Fig. 1, thread pool of real-time system has two layers, asynchronous layer monitors the interface of thread pool, it will send request's ID, request occurring time, time limit of request accomplishment into request list while request comes; synchronous layer calculates priorities of requests inside request list, assigns idle thread to the request with top priority, and feedbacks to request list after request is handled.

It's hard to statically determine request's priority, but the time limit of each request accomplishment is certain,

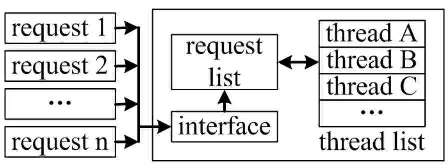


Figure 1 Thread pool framework of real-time system.

so based on the agency of request accomplishment, request's priority can be determined dynamically. T_E is defined as execution time of request; T_L is time limit of request accomplishment; T_P stands for present time; T_O stands for request occurring time; T_U stands for urgency of request. The relation between them is:

$$T_U = T_L - (T_P - T_O) - T_E \tag{9}$$

$(T_P - T_O)$ represents request's latency time. As T_L is certain, T_U could be figured out by (9) after T_E is estimated. The less of T_U, the more urgent request is. Timeout request will occur when T_U is negative. Therefore, T_U is request's priority. The input/output of thread pool model is shown in Table 1.

5. Results and discussion

A wireless sensor network(WSN) node N composed by a S3C2440A CPU, a Netgear WN111 WLAN card, an OV9640 image sensor, 768MB SDRAM and 256MB FLASH ROM, was used for testing, besides computer C1 and C2. System of node N was embedded and real-time; it would handle 5 types of request through thread pool. Node N would receive and retransmit data from computer C1 to C2, acquire image data (5 frames per second, 320*240 pixels, YUV422 format) and transmit them to C2. Time limit of request accomplishment T_L in thread pool was set as Table 2.

In order to simulate node N's concurrent requests from adjacent nodes in WSN, computer C1 ran in multithread, each thread sent 100 images(320*240 pixels, YUV422 format) which should be retransmitted to C2. The adjacent node's amount of concurrent requesting was set as 5, 10, 20 and 30. Thread amount in thread pool of node N was set as 8, 16 and 24. 160 groups of data were randomly selected, and set as training samples after normalization. HQPSO algorithm's particle amount M was set as 30, iteration limit was set as 100, parameter S and D in Table 1 were both set as 10, initialization ranges of LS-SVM parameters were: γ=[0,1000] and σ=[0,10]. MSE was used as objective function.

After optimized by HQPSO algorithm, the parameters of LS-SVM were γ=121, σ=1.67. Request timeout ratios of thread pool which scheduling priority based on HQPSO LS-SVM, PSO LS-SVM, and first request first handle (FRFH) are shown in Fig. 2-4. In most situations, HQPSO LS-SVM based thread pool has a lower request timeout ratio than others'. System handling ability is close to limit when the amount of concurrent requests from adjacent nodes is about 30. Furthermore, request timeout ratio is affected by the thread amount of thread pool. While thread amount is too small to meet the demand, most requests can't reach their deadlines without enough threads. Superiority of HQPSO LS-SVM algorithm isn't obvious, as Fig. 2. While thread amount is too large, scheduling priority algorithm will be meaningless as almost all requests acquire thread handling immediately. Because of overmuch active thread, system performance decreases, and request timeout ratio can not be dropped down. Moreover, as HQPSO LS-SVM and PSO LS-SVM based thread pools need more system resources for estimation, their corresponding request timeout ratios will be higher than FRFH based thread pool's when system handling ability is close to limit, as shown in Fig. 4. HQPSO LS-SVM based thread pool has significant superiority of timeout avoidance when thread amount of thread pool was set properly, as shown in Fig. 3.

Table 1 Input/output of thread pool model.

Input(x_i)	Output(y_i)
Non-idle thread amount	T_E
Average elapsed time of latest S threads	
Average latency time of topmost D requests in request list	
Blocking amount of Non-idle threads	
Blocking time of Non-idle threads	
Same type amount of new request and blocked threads'	

Table 2 Time limit of request accomplishment.

Request Type	Time Limit (Millisecond)
WSN input request	30
image acquiring request	20
data encryption request	10
routing request	10
WSN output request	30

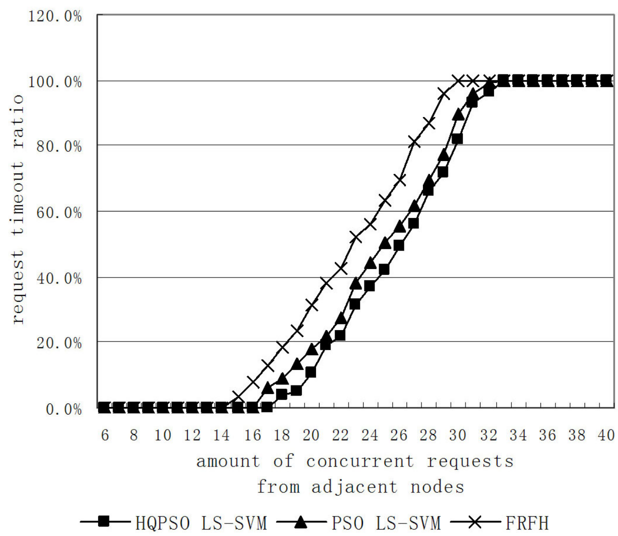


Figure 2 Request timeout ratios with 8 threads.

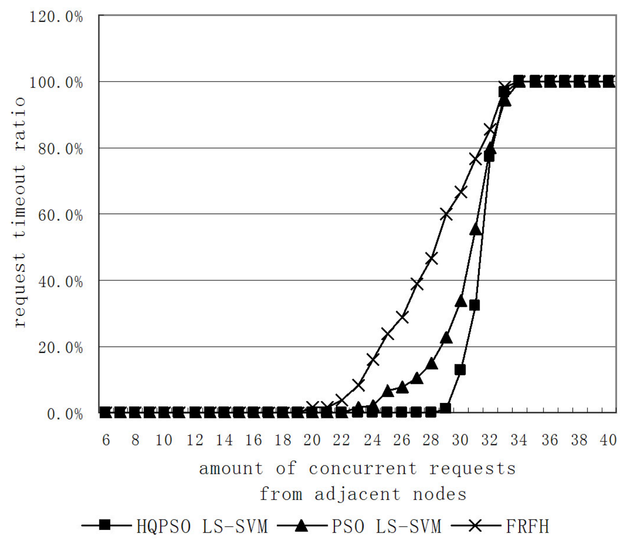


Figure 3 Request timeout ratios with 16 threads.

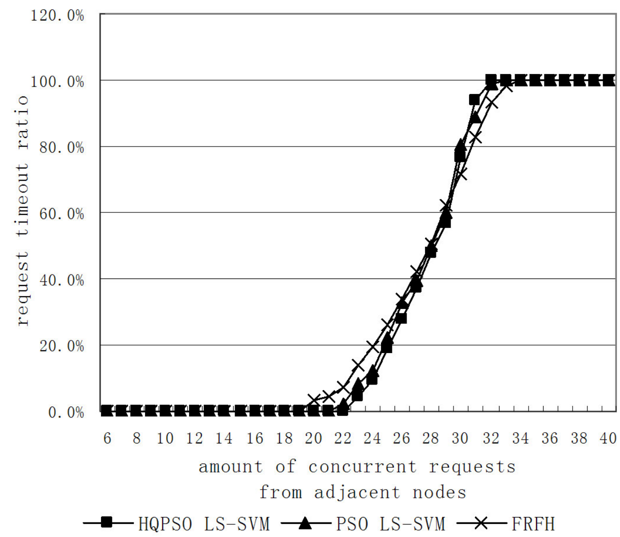


Figure 4 Request timeout ratios with 24 threads.

6. Conclusion

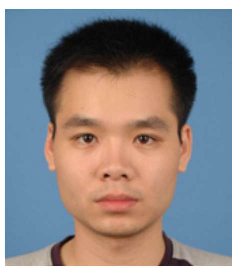
HQPSO LS-SVM algorithm has good effects on training and prediction. Although resource consumption is increased while the algorithm is applied to thread pool, request timeout ratio will dropped down obviously when thread amount is set appropriately. Compared with the other scheduling priority algorithm, HQPSO LS-SVM algorithm is more powerful and more suitable for thread pool optimization of real-time system.

Acknowledgement

The author is financially supported by the Science and Technology Planning Project of Guangdong Province from Science and Technology Department of Guangdong Province, project No. 2010B0609001. The author is grateful to Associate Professor Mingming Xiao for a careful checking of the details and for helpful comments that improved this paper.

References

[1] J. A. K. Suykens and J. Vandewalle. Least squares support vector machine classifiers, Neural Processing Letters. **9**(3):293-300 (1999).
[2] V. N. Vapnik, Editor, The Nature of Statistical Learning Theory, Springer-Verlag, New York (1998).
[3] Y. Liu and C. J. Liu. Scheduling strategy design of thread pool for real-time control system, Computer Engineering and Applications. **46**(32):71-73,104 (2010).
[4] S. Osowski, R. Siroic, T. Markiewicz. Application of support vector machine and genetic algorithm for improved blood cell recognition, Instrumentation and Measurement. **58**(7):2159-2168 (2009).
[5] C. L. Huang and J. F. Dun. A distributed PSO?SVM hybrid system with feature selection and parameter optimization, Applied Soft Computing. **8**(4):1381-1391 (2008).
[6] L. C. Zhou, H. Z. Yang, C. B. Liu. QPSO-based hyper-parameters selection for LS-SVM regression. Proceedings of Fourth International Conference on Natural Computation, IEEE, Jinan, 130-133 (2008).
[7] H. Q. Zhu, C. H. Yang, J. Wang, W. H. Gui. Hybrid-QPSO-based parameters optimization of LS-SVM and its application, Journal of Central South University: Science and Technology. **42**(4):1000-1004 (2011).
[8] J. Sun, B. Feng, W. B. Xu. Particle swarm optimization with particles having quantum behavior. Proceedings of Congress on Evolutionary Computation, IEEE, Portland, 325-331 (2004).



Yi Liu is presently employed as a Lecturer at Zhongkai University of Agriculture and Engineering, China. He obtained his master degree from South China University of Technology(China). His main research fields are embedded computer system and auto control system.