International Journal of Innovative
Computing, Information and Control
Volume 8, Number 10(B), October 2012
 ISSN 1349-4198
pp. 7381–7391

BUILDING AN INTEGRATED HYBRID MODEL FOR SHORT-TERM AND MID-TERM LOAD FORECASTING WITH GENETIC OPTIMIZATION

Jiliang Xue[1,2], Zhenyuan Xu[2] and Junzo Watada[2]

[1]Department of Electrical Engineering
Shanghai Jiaotong University
No. 800, Dongchuan Road, Minhang District, Shanghai 200240, P. R. China
xrayseed@hotmail.com

[2]Graduate School of Information, Production and Systems
Waseda University
2-7 Hibikino, Wakamatsu, Kitakyushu 808-0135, Japan
xzyzx05@moegi.waseda.jp; junzow@osb.att.ne.jp

Received April 2011; revised August 2011

Abstract. *The enhancement of load forecasting has become one of the core research topics in the energy field. Because power load has both time-variant and nonlinear characteristics, different types of methods, neural networks (NN) in particular, have been applied to power load forecasting. This study proposes a real-valued genetic algorithm (RGA)-based neural network with support vector machine (NN-SVM) model to predict the power load in both short-term and mid-term forecasting by using a radial-basis-function neural network (RBFNN), SVM and RGA. The model consists of two stages. In short-term load forecasting (STLF), the first stage applies the RBFNN to predict monthly variations, and the second stage trains the SVM through hourly data to obtain the final forecast. Similar operations are used in mid-term load forecasting (MTLF). In the process of SVM training and NN learning, RGA is used to find the optimal parameters. The results of several experiments show that this new model performs more accurately and stably than some conventional models including RBFNN, RGA-SVM, Karman filter in STLF. Also it is able to function well in MTLF.*
Keywords: Radial-basis-function neural network (RBFNN), Support vector machine (SVM), Real-valued genetic algorithm (RGA), Short-term load forecasting (STLF), Mid-term load forecasting (MTLF)

1. **Introduction.** Load forecasting, such as short-term load forecasting (STLF) and mid-term load forecasting (MTLF), plays a pivotal role in running power systems in the power industry. A high precision is required in STLF and MTLF [8] because many strategies, such as unit commitment, fuel scheduling economic dispatch and unit maintenance are based on predicted values. Additionally, with the increased demand for energy, it is crucial to comprehend how power load changes due to outer features and factors such as social and economic structures, weather conditions and seasonal fluctuation trends.

Many algorithms have been proposed to deal with STLF and MTLF. For STLF, Palalexopoulos and Heserberg proposed a regression-based approach [19] twenty years ago. Soon, other techniques were developed including the autoregressive model (AR) [16], Kalman filter [25] and neural networks (NN) [14]. Though the AR algorithms give good performances, which adapt seasonal factors, the method loses reliability in variable model. Kalman filter (Brown, 1983) was another technique which requires long-period historical data to analyze periodic and independent components of the power load model. In order to deal with load forecasting under different conditions, Barbosa and Sadownik (1999) offered a dynamic multivariate time series model with Bayesian sequential estimation [3]. Research focus began to change from refining and adding the factors of model to the elaborating of meta-heuristic, especially NN. Comparisons show that fuzzy logic (FL) and artificial neural networks (ANN) can be good candidates for STLF [15]. Recent research studies are limited in providing flexible approaches for time-series data and influential factors. Additionally, computation methods are mainly focused on ANN [13, 20]. For MTLF, the general goal is to predict the daily load demand in a month. The general concern is how to adjust predictions based on the error propagation of the models. Bhattacharya and Basu (1993) have also applied the Kalman filter to this problem by combining it with the Walsh transform [4]. Ghiassi et al. [10] and Misasgedis et al. [17] proposed prediction of monthly energy consumption using dynamic NN model and dynamic regression model, respectively. Also Amjady and Keynia [2] proposed a hybrid prediction model consisting of pre-forecast mechanism, neural algorithm and evolutionary algorithm.

Support vector machine (SVM) theory is based on the statistic learning theory raised by Vapnik [7]. This theory employs the criterion of minimizing the structure risk while lowering the global error of the model. Recently SVMs have been introduced into load forecasting problem [23]. It is believed that SVMs can perform well in some prediction and classification problems, but the outcome shows reluctance or chaotic character if the data exceeds its historical scope. It is promising to reinforce most widely used NNs by combining SVM approaches [1, 26].

The optimization theory has been flourishing thanks to the dedication toward seeking suitable methods for unique optimization problems. Meta-heuristic solutions such as tabu search (TS) (1989), particle swarm optimization (PSO) (1995) and genetic algorithms (GA) (1997) have shown abundant results for optimal searching problems. The simplest genetic algorithm represents each chromosome as a bit string. There are also many variants of GA, among which real-valued genetic algorithm (RGA) is effective, which was proved by Su and Chang [22].

Through the above review, this paper presents a combination of the radial basis function neural network (RBFNN) and SVM with the RGA optimization to mitigate the limitations of the existing method while enhancing the accuracy and reliability of the forecasting. It is named the RGA-based NN-SVM model. Here, in the process of RBFNN and SVM, RGA supplies a framework to find the optimal parameters for the final prediction. Compared with some conventional algorithms that can only deal with specific problems, RGA-based NN-SVM is capable of functioning well for both STLF and MTLF problems.

The rest of this paper is organized as follows. Section 2 introduces the basic knowledge of RBFNN and SVM. Section 3 describes the structure of the problem. Section 4 explains how the RGA-based NN-SVM model forecasts power loads. Section 5 analyzes and discusses three experiments using some data from France, and finally, Section 6 presents the conclusions.

2. Background Knowledge.

2.1. Radial basis functionneural network (RBFNN).

As one particular type of feed forward neural networks with supervised learning [9], RBFNN consists of input layer, hidden layer and output layer. The input layer is used as a sensing unit containing r neurons. The hidden layer has p radial basis function (RBF) type hidden neurons and the output layer contains k neurons. Input information is assigned to a node in the input layer, which links to the hidden layer directly. RBF is defined by a center position and a spread parameter. This function gives larger value output when input variables are closer to the center, and smaller value output monotonically as their distances from the center increase. After the process in the hidden layer, it is delivered to the output layer, forming an analytical mapping. The performance of the RBFNN depends critically on the choice of the nonlinear activation function, center c_i and spread parameter σ_i. If we observe an r-dimension input vector with components: $x_1, x_2, \ldots, x_r$, the output h_i can be obtained as the following formulation:

$$h_i = e^{-\frac{\|x_i - c_i\|^2}{2\sigma_i^2}} \tag{1}$$

where c_i is the neuron's center and σ_i the spread parameter.

Finally, a linear mapping is executed from the hidden layer to the output layer as follows:

$$y_k = \sum_{j=1}^{p} w_{jk} h_j \tag{2}$$

where w_{jk} is the synaptic weight connecting hidden neuron j with output neuron k.

2.2. **Support vector machine.** Vapnik first proposed a version of SVM for regression (SVR) in 1997 [21]. SVR algorithms have been proved to be able to solve large-scale regression problems efficiently [5] after various fields were studied, such as smoothing technique, hyper plane thinking, and considerations concerned with kernels.

Here in the SVM computation, the input X is first mapped onto an m-dimensional feature space using some fixed (nonlinear) mapping $g_i(\mathbf{x})$, and then a linear model is constructed in this feature space. The linear model (in the feature space) $f(\mathbf{x}, \varpi)$ is given by

$$f(\mathbf{x}, \varpi) = \sum_{i=1}^{m} \varpi_i g_i(\mathbf{x}) + b \tag{3}$$

where $g_i(\mathbf{x})$, $i = 1, \ldots, m$ denotes a set of nonlinear transformations, and b is the bias term.

The SVM computation here uses the following loss function ε-insensitive loss proposed by Vapnik:

$$L(y, f(\mathbf{x}, \varpi)) = \begin{cases} 0, & \text{if } |y - f(\mathbf{x}, \varpi)| \le \varepsilon \\ |y - f(\mathbf{x}, \varpi)| - \varepsilon, & \text{otherwise} \end{cases} \tag{4}$$

It performs linear regression in the high dimension feature space using ε-insensitive loss and at the same time, reduces model's complexity by minimizing $||\varpi||^2$. This can be described by introducing non-negative slack variables ξ_i, ξ_j, $i = 1, \cdots, n$, to measure the deviation of training samples outside ε-insensitive zone. Thus, the SVM computation here is formulated in the following:

$$\begin{aligned} &\min \tfrac{1}{2}||\varpi||^2 + \sum_{i=1}^{n} (\xi_i + \xi_i^*) \\ &\text{s.t.} \begin{cases} y_i - f(x_i, \varpi) \le \varepsilon + \xi_i^* \\ f(x_i, \varpi) - y_i \le \varepsilon + \xi_i \end{cases} \quad \xi_i, \xi_i^* \ge 0, \quad i = 1, \cdots, n \end{aligned} \tag{5}$$

This optimization problem can be transformed into the dual problem and its solution is given by:

$$\begin{aligned} &f(x) = \sum_{i=1}^{n_{SV}} (a_i - a_i^*) K(x_i, x) \\ &\text{s.t. } 0 \le a_i^* \le C, \quad 0 \le a_i \le C \end{aligned} \tag{6}$$

where n_{SV} is the number of Support Vectors and the kernel function $K(x_i, x) = e^{-\frac{\|x_i - x\|^2}{2\sigma_i^2}}$.

In lib-SVM, the value of ε is not necessary to find; the key point is to find the optimal values for C and σ.

3. **Problem Formulation.** Before extending the forecasting problem, the variables and functions should be first defined for the problem as shown in Table 1.

TABLE 1. Notation of data

Attributes	Attribure meaning
t	Time, which counts half hourly in a day as $1, 2, \cdots, 48$.
y	Year
m	Monbth$(1, 2, \cdots, 12)$
day	Day$(1, 2, \cdots, 31)$
$E(m, y)$	Economic factor of year y
$da(m, y)$	Returns the count of days $(= 28, 29, 30, 31)$ in month m, year y
$T(m, y)$	Average templerature in year y
$T(day, m, y)$	Average emperature on day day, month m in year y
$P_c(m, y)$	Power consumption in month m, year y
$P_c(day, m, y)$	Power consumption in day day, month m, year y
$L(t, day, m, y)$	Actual power load at time t, on day day, month m in year y
$L_f(t, day, m, y)$	Forecating power load at time t, on day day, month m in year y
$L_p(day, m, y)$	Peak load on day day, month m in year y
$Daytype(day, m, y)$	Day type on day day, month m, in year y, later explained in Table 2

TABLE 2. Definition of *Daytype*

$(day, m.y)$	Sun.	Mon.	Tue.	Wed.	Thu.	Fri.	Sat.	Holiday
$Daytype(day, m, y)$	1	2	3	4	5	6	7	8

Remark 3.1. *Monthly and daily consumption can be approximately computed by:*

$$\begin{aligned} P_c(day, m, y) &= 0.5 \sum_{i=1}^{48} L(t, day, m, y) \\ P_c(m, y) &= \sum_{day(m,y)} P_c(day, m, y) \end{aligned} \tag{7}$$

The problem is to find the load condition $L_f(t, day, m_f, y_f)$, $L_p(day, m_f, y_f)$, $P_c(day, m, y)$ and $P_c(m, y)$ in future target time (month m_f, year y_f) using the given historical data including $E(m, y)$, $T(m, y)$, $T(day, m, y)$, $L_c(m, y)$, $L(t, day, m, y)$ and $Daytype(day, m, y)$.

4. **RGA Based NN-SVM.** Aiming at an adaptive integrative model with high reliability, the algorithm consists of two parallel procedures called NN thread and SVM thread, where both are optimized by RGA. NN thread deals with the general yearly variation of power load concerning with basic influential factors. The output offers a standard medium value for target month called growth index here. Meantime, SVM thread obtains the moment-specified forecast with an output as a monthly deviation value from monthly standard explained in thread two. Both the parameter selection procedures will be solved by RGA, searching the optimal values of center C, width σ in RBFNN and SVM. Through output combination, we are able to arrive at the final forecast for the target.

Thread I:

Step 1: Assume the objective month is m_f, y_f, for $y < y_f$, train RBFNN using the historical data. The network contains two dimensional input $T(m, y)$ and $E(m, y)$ as fundamental influential factor while $P_c(m, y)$ is output growth index.

Step 2: Train RBFNN model by RGA optimization. Table 3 shows the type of parameters (chromosome) coding for RBFNN.

Table 3. Definition of parameters for RGA-NN

No.	C_1	$\cdots$	C_p	σ_1	$\cdots$	σ_p

Step 3: Predict $p_c(m, y)$ through the constructed model. Figure 1 shows the flow chart of Thread I.

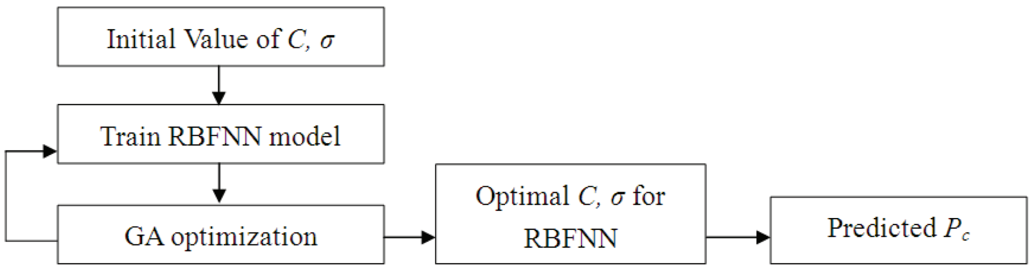

Figure 1. Flow chart of thread one

Thread II:

Step 1: Scale the data referring to monthly average $L_{avg}(t, m, y)$ into $L'_f(t, day, m, y)$ according to the periodic characteristic of power load and prevent the overflow.

$$L_{avg}(t,m,y) = \sum_{day(m,y)} \frac{L(t,day,m,y)}{day(m,y)}$$
$$L'_f(t,day,m,y) = \frac{L(t,day,m,y) - L_{avg}(t,m,y)}{L_{avg}(t,m,y)} \tag{8}$$

The new values illustrate the variance from standard amount. This makes the SVM more effective because the load condition has resemblances in the same duration of different years.

Step 2: For $y < y_f$ construct SVM model with four dimension input $T(day, m, y)$, $Daytype(day, m, y)$, m and t as subtle factors and one dimension output $L'_f(t, day, m, y)$.

Step 3: Train SVM model by RGA optimization. Table 4 shows the type of parameters (chromosome) coding for SVM.

Table 4. Definition of parameters for RGA-SVM

No.	C	σ

Step 4: Predict $L'_f(t, day, m_f, y_f)$ through the constructed model. Figure 2 shows the flow chart of Thread II.

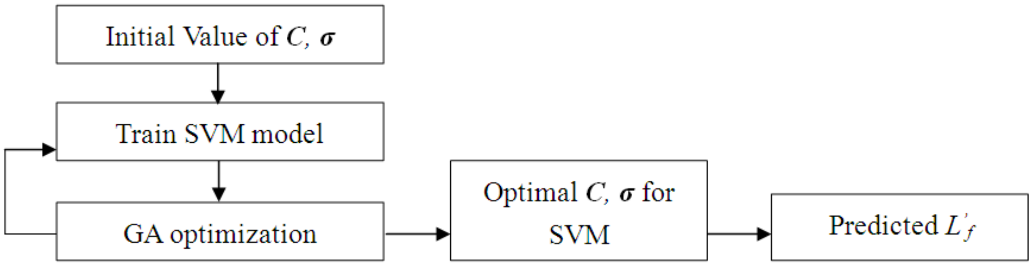

FIGURE 2. Flow chart of thread two

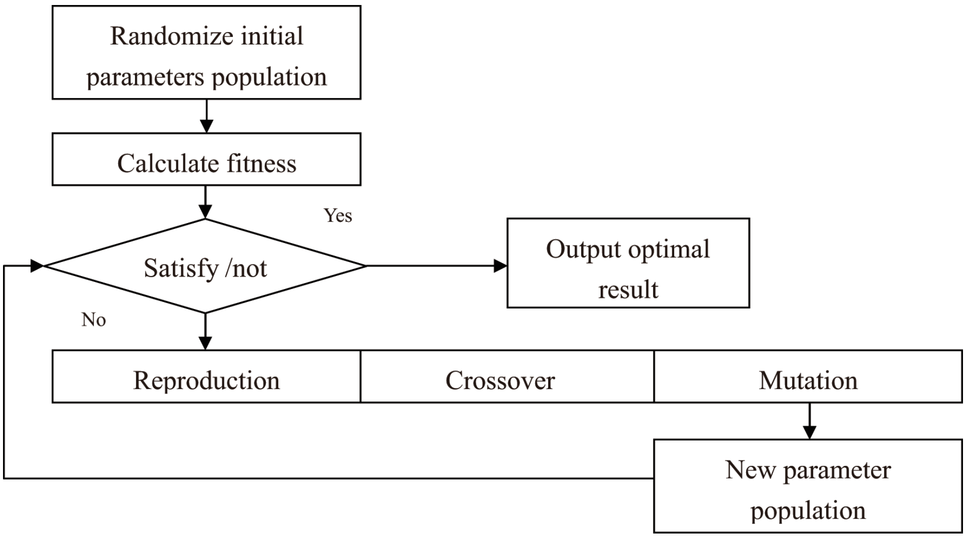

FIGURE 3. Flow chart of RGA optimization

RGA Optimization:

Genetic algorithms (GAs) are widely applied to various optimization problems since they can search non-linear solution spaces without requiring gradient information or a priori knowledge about the characteristics of the model [1, 11]. Among GAs, the real-valued genetic algorithm (RGA) uses only real values for parameters of the chromosome in populations without endless encoding (Y. P. Huang and C. H. Huang, 1997) [12]. RGA is faster and more efficient than binary-coding genetic algorithm (BGA) in our optimization case. Therefore, RGA is adopted to perform the optimization process to search for unknown parameters, shown in Figure 3. RGA uses selection, crossover and mutation to generate new offsprings. Tournament selection and non-uniform mutation [5, 18] are used for genetic operators. After that, MAPE (mean absolute percentage error) is examined in the fitness function

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{A_t - F_t}{A_t}\right| \tag{9}$$

where A_t is the actual value at time t, while F_t is the forecast value at time t.

After these two parallel threads, we come to an integration process. The final output of forecast is computed through combining two output $p_c(m_f, y_f)$ and $L_f^{'}(t, day, m_f, y_f)$. The final result: $L_f(t, day, m_f, y_f = p_c(m_f, y_f) \times (L_f^{'}(t, day, m_f, y_f) + 1)$. The whole flow chart is shown in Figure 4.

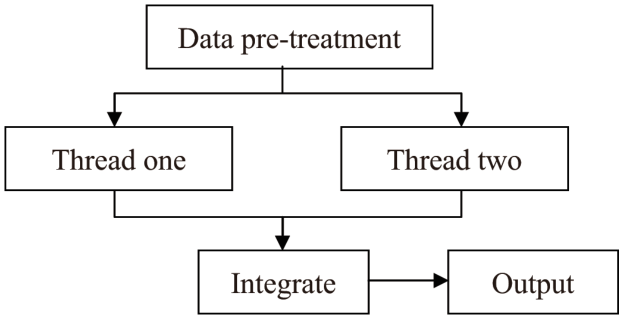

FIGURE 4. Whole flow chart of RGA based NN-SVM

5. **Experiment Results.** We performed several experiments including a one-week daily curve forecast, one-month peak load and consumption forecast to test the new model. In first scenario, three other methods were compared. RBFNN was chosen as one counterpart to stand for hybrid ANNs which are mostly widely researched for power system forecasting as was stated in the introduction. Also RGA-SVM, one type of hybrid SVM, is applied into comparison due to the good results from SVMs recently piloted for load forecasting problems. The second and third scenarios give a good example for MTLF. The experiments used historical load data from year 1996 to 2009 given half-hourly by RTE institution in France. Since the data was amply given with comparatively long time series, Kalman Filter is also a good candidate in this case. The data from year 1996 to 2008 was used for training, and year 2009 was for testing. The scenario was assumed on the end of June of 2009 to obtain forecast result for later days.

5.1. **First week daily curve analysis.** The load curves are depicted by a minimum timescale of 30 minutes. The evaluation criteria for the correctness of this experiment are MAPE and RMSE (root mean square error) which is the main result evaluating index:

$$RMSE = \sqrt{1/n \sum_{t=1}^{n} (A_t - F_t)^2} \tag{10}$$

where A_t is the actual value at time t, while F_t is the forecast value at time t.

The experiment result and the comparison with RGA-SVM, Kalman Filter and RBFNN are listed in Table 5. Also a brief comparison is showed in Figure 5 which gives a clear interpretation that the RGA based NN-SVM model has more satisfactory forecasting in each day than the other three methods. Solely using RGA-SVM is not suitable for this hourly forecasting due to the high MAPE. The reason is that SVM shows hesitation

TABLE 5. Result of four algorithms

		July 1st	July 2nd	July 3rd	July 4th	July 5th	July 6th	July 7th	week
		Tue.	Wed.	Thu.	Fri.	Sat.	Sun.	Mon.	
MAPE (%)	RGA-SVM	8.83	10.88	17.20	16.00	22.55	16.93	7.44	14.26
	KF	4.82	6.73	6.02	5.35	8.31	9.57	5.60	6.63
	RBFNN	5.39	7.57	7.88	2.96	8.89	10.45	1.60	6.39
	RGA based NN-SVM	5.01	4.88	4.06	1.99	3.51	2.22	0.75	3.20
RMSE (%)	RGA-SVM	10.80	11.78	19.63	17.82	25.23	20.47	8.59	17.27
	KF	4.89	7.02	6.53	5.61	9.72	10.73	5.93	7.20
	RBFNN	5.76	7.95	8.36	3.22	10.56	11.68	1.89	7.82
	RGA based NN-SVM	5.22	4.96	4.21	2.12	3.64	2.28	0.89	3.65

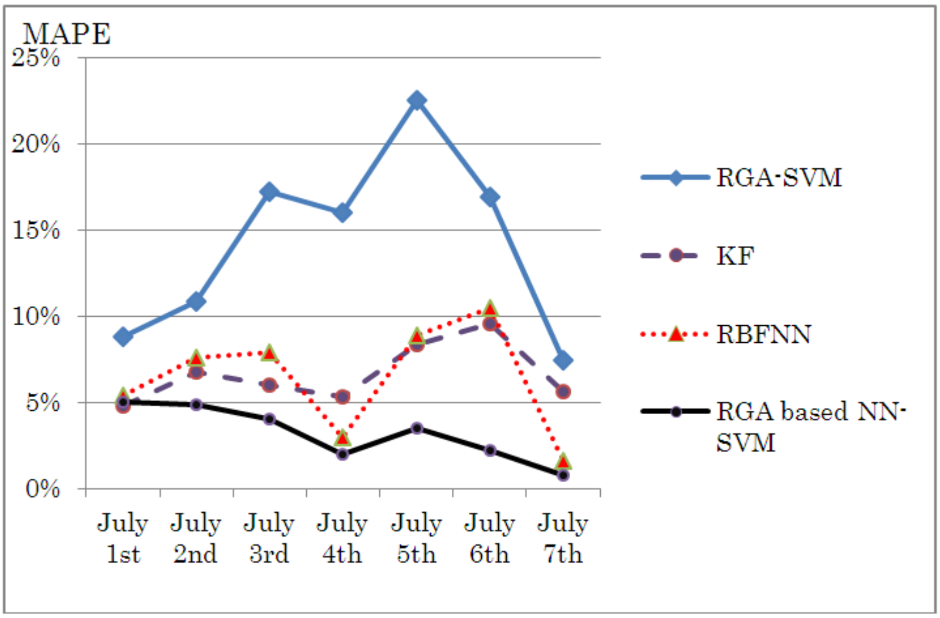

FIGURE 5. Forecasting and actual daily load curve

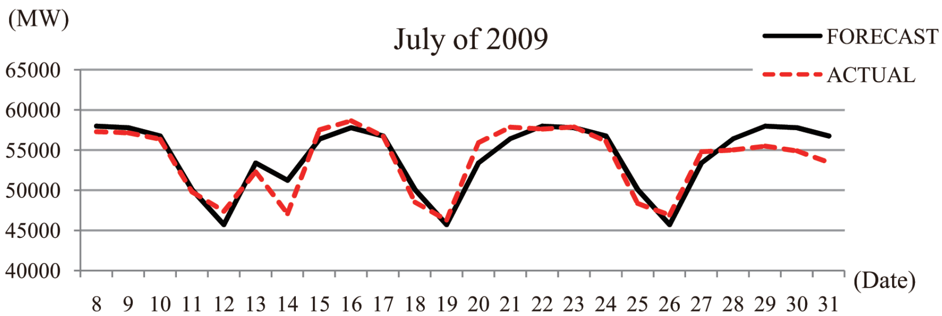

FIGURE 6. Forecasting and actual daily peak load curve

when dealing with both the growth of time series load and the different influential labels at the same time. RBFNN and Kalman filter obtain close results with MAPE around 6.5. Both methods show weak points on July 5th and 6th which is Saturday and Sunday, separately. This means both methods are not sensitive to the change of date, but focus on the smoothness of the time-series curve. On the other hand, the new algorithm shows significant improvement on July 5th and 6th which proves the correctness of labelling step in thread two against the difficulty of forecasting weekends and holidays in RBFNN and RGA-SVM. RGA based NN-SVM's comparatively high stability through whole week makes the forecasting model reliable for different period of a year.

5.2. **Peak load analysis.** This scenario gives an overlook of July's peak load. From July 8th, it is beyond possibility to get the comparatively accurate weather information like first week. The results in Figure 6 shows that it is still available to forecast the daily peak using same RGA based NN-SVM model through training historical peak load among the whole dataset and substituting correct weather information of future day with default values such as the average temperature of first week. Forecasting errors are in terms of MAPE and PAPE (peak absolute percentage error).

TABLE 6. Peak moment estimation

Date	TP Forecast	TP Actual	Date	TP Forecast	TP Actual	Date	TP Forecast	TP Actual
July 8th	12:00	12:00	July 16th	12:00	14:30	July 24th	13:00	13:00
July 9th	12:00	12:00	July 17th	13:00	13:00	July 25th	13:00	13:00
July 10th	13:00	13:00	July 18th	13:00	12:00	July 26th	23:00	23:00
July 11th	13:00	13:00	July 19th	23:00	23:00	July 27th	13:00	13:00
July 12th	23:00	23:00	July 20th	13:00	13:00	July 28th	12:00	12:00
July 13th	13:00	13:00	July 21st	12:00	14:30	July 29th	12:00	13:00
July 14th	23:00	23:00	July 22nd	12:00	12:00	July 30th	12:00	13:00
July 15th	13:00	13:00	July 23rd	12:00	12:00	July 31st	13:00	13:00

T_P: Moment when peak load occurs

The MAPE and PAPE of this experiment is 2.59% and 6.31% separately, which explains that accuracy remains high in peak load estimation. We can find out that except July 14th (MAPE: 8.8) and last three days of July (MAPE: 4.5, 5.3, 6.1), the peak load estimation is quite accurate for power industry. We believe that the difficulty on July 14th is due to the uncertainty of the demand on National Day. It is beyond pure statistical model to fix this special result though it may be possible through efforts on social study. As to the last days of the months, our estimation is higher than actual data. This is because of the weather condition is far beyond estimation after almost one month. In the real case, the temperatures surprisingly dropped on these days which may be a big affect to the error. In the meantime, we watch the moment peak load occurs and compare the actual moment in Table 6. The hit rate of peak time is 91.7% according to one-hour time deviation tolerance. This illustrates the success of catching peak load label with peak time towards different day type.

5.3. **One month consumption analysis.** The last scenario tests the daily consumption in July as shown in Figure 7. Instead of L, P_c is applied to RGA based NN-SVM to forecast daily consumption. The MAPE is 2.54% and the RMSE is 3.11% in the test.

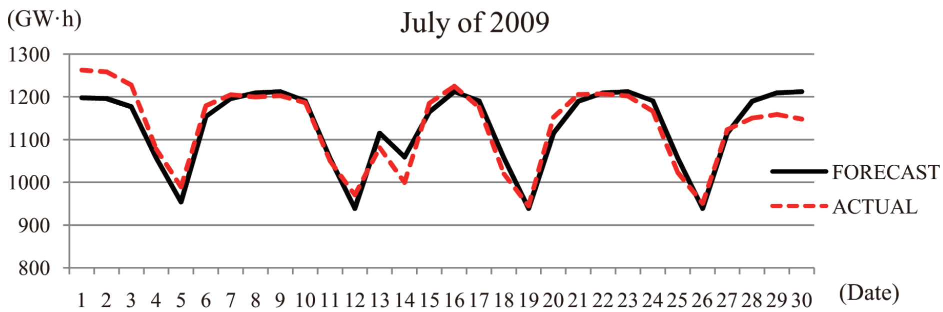

FIGURE 7. Forecasting and actual curve for daily consumption

6. **Conclusions.** STLF and MTLF are two important aspects of the power system that have been studied recently. Various methods have been proposed to obtain good solutions for these two types of load forecasting. In particular, ANN was widely investigated and gives reasonable results for STLF. This paper presents an integrative algorithm for forecasting hourly load change in a short period and for forecasting the daily load peak and the daily consumption in coming months with high accuracy and reliability using the NN-SVM with RGA optimization. This algorithm not only benefits from the neural network

structure for tendency prediction, but also uses the successful data labelling characteristic of the SVM, which arranges the time series data in an efficient way. The experiment using the load data from the RTE institution in France gives several application examples. The results show that the new method gives a detailed illustration for daily and weekly load change and offers a reliable prediction for next month consumption with high accuracy.

Research is under way to test the feasibility of the algorithm with information loss or small sample data because historical information is not always complete. Additionally, the application of forecasting results on real power systems is also being considered.

REFERENCES

[1] A. M. Al-Kandari and K. M. El-Naggar, A genetic-based algorithm for optimal estimation of input-output curve parameters of thermal power plants, *Electrical Engineering*, vol.89, no.8, pp.585-590, 2007.
[2] N. Amjady and F. Keynia, Mid-term load forecasting of power systems by a new prediction method, *Energy Conversion and Management*, vol.49, no.10, pp.2678-2687, 2008.
[3] E. P. Barbosa and R. Sdafadownik, A multiplicative seasonal growth model for multivariate time series analysis and forecasting, *Communications in Statistics-Simulation and Computation*, vol.28, no.2, pp.291-308, 1999.
[4] T. K. Bhattacharya and T. K. Basu, Medium range forecasting of power system load using modified Kalman filter and Walsh transform, *International Journal of Electrical Power and Energy Systems*, vol.15, no.2, pp.109-115, 1993.
[5] C. A. C. Coello and E. M. Montes, Constraint-handling in genetic algorithms through the use of dominance based tournament selection, *Advanced Engineering Informatics*, vol.16, no.3, pp.193-203, 2002.
[6] R. Collobert and S. Bengio, SVMTorch: Support vector machines for large-scale regression problems, *Journal of Machine Learning Research*, vol.1, no.2, pp.143-160, 2001.
[7] C. Cortes and V. Vapnik, Support-vector networks, *Machine Learning*, vol.20, no.3, pp.273-297, 1995.
[8] A. P. Douglas and A. M. Breipohl, Risk due to load forecast uncertainty in short term power system planning, *IEEE Transactions on Power Systems*, vol.13, no.4, pp.1493-1499, 1998.
[9] S. Elanayar and Y. C. Shin, Radial basis function neural-network for approximation and estimation of nonlinear stochastic dynamic-systems, *IEEE Transactions on Neural Networks*, vol.5, no.4, pp.594-603, 1994.
[10] M. Ghiassi, D. K. Zimbra and H. Saidane, Medium term system load forecasting with a dynamic artificial neural network model, *Electric Power System Research*, vol.76, no.5, pp.302316, 2006.
[11] C. L. Huang and C. J. Wang, A GA-based feature selection and parameters optimization for support vector machines, *Expert Systems with Applications*, vol.31, no.2, pp.231-240, 2006.
[12] Y. P. Huang and C. H. Huang, Real-valued genetic algorithms for fuzzy grey prediction system, *Fuzzy Sets and Systems*, vol.87, no.3, pp.265-276, 1997.
[13] M. Kurban and U. B. Filik, Next day load forecasting using artificial neural network models with autoregression and weighted frequency bin blocks, *International Journal of Innovative Computing, Information and Control*, vol.5, no.4, pp.889-898, 2009.
[14] K. Y. Lee, Y. T. Cha and J. H. Park, Short-term load forecasting using an artificial neural network, *Transactions on Power Systems*, vol.7, no.1, pp.124-132, 1992.
[15] K. Liu, S. Subbarayan and R. R. Shoults, Comparison of very short-term load forecasting techniques, *IEEE Transactions on Power Systems*, vol.11, no.2, pp.877-882, 1996.
[16] G. Mbamalu and M. Elhawary, Load forecasting via suboptimal seasonal autoregressive models and iteratively reweighted least-squares estimation, *IEEE Transactions on Power Systems*, vol.8, no.1, pp.343-348, 1993.
[17] S. Mirasgedis, Y. Sarafidis and E. Georgopoulou, Models for mid-term electricity demand forecasting incorporating weather influences, *Energy*, vol.31, no.2-3, pp.208-227, 2006.
[18] A. Neubauer, Theoretical analysis of the non-uniform mutation operator for the modified genetic algorithm, *Proc. of the IEEE Conference on Evolutionary Computation*, pp.93-96, 1997.
[19] A. D. Papalexopoulos and T. C. Hesterberg, A regression-based approach to short-term system load forecasting, *IEEE Transactions on Power System*, vol.5, no.4, pp.1335-1947, 1990.

[20] P. J. Santos and A. G. Martins, Designing the input vector to ANN-based models for short-term load forecast in electricity distribution systems, *Electrical Power and Energy Systems*, vol.29, no.4, pp.338-347, 2007.

[21] A. Smola and V. Vapnik, Support vector regression machines, *Advances in Neural Information Processing Systems*, vol.9, pp.155-161, 1997.

[22] M. C. Su and M. C. Chang, Application of neural networks incorporated with real-valued genetic algorithms in knowledge acquisition, *Fuzzy Sets and Systems*, vol.112, no.1, pp.85-97, 2000.

[23] J. Wang and G. Ren, An improved SVM and its using in electric power system load forecasting, *Neural Information Processing, Lecture Notes in Computer Science*, vol.4232, pp.847-855, 2006.

[24] H.-Y. Wu, C.-Y. Hsu, T.-F. Lee and F.-M. Fang, Improved SVM and ANN in incipient fault diagnosis of power transformers using clonal selection algorithms, *International Journal of Innovative Computing, Information and Control*, vol.5, no.7, pp.1959-1974, 2009.

[25] K. Xie and F. Wang, Very short-term load forecasting by Kalman filter algorithm, *Proc. of the Chinese Society of Electrical Engineering*, vol.16, no.4, pp.245-249, 1996.

[26] J. Xue and J. Watada, Short-term power load forecasting method by radial-basis function neural network with support vector machine model, *ICIC Express Letters*, vol.5, no.5, pp.1523-1528, 2011.