Hybrid intelligent gearshift control of technical vehicles based on AGA-NN

Yongjing Huang*
School of Manufacturing Science and Engineering, Sichuan University, Chengdu, China
Department of Electronic Information and Electric Engineering, Chengdu Textile College, Chengdu, China
huangyongjingscu@gmail.com

Jin Yao
School of Manufacturing Science and Engineering, Sichuan University, Chengdu, China
jinyao163@163.com

Jiahua Han
School of Manufacturing Science and Engineering, Sichuan University, Chengdu, China
hanjiahua139@139.com

Hua Li
School of Manufacturing Science and Engineering, Sichuan University, Chengdu, China
sculihua2005@sina.com

Abstract

This paper introduces the shift schedule of saving energy to keep the torque converter working high-efficiency. The author carries on the further study of saving energy shift theory, and presents the control method of saving energy shift schedule. At the same time, this paper expounds how to use the shift theory and puts forward the control method of saving energy shift schedule. At last, the author expounds how to use the delayed shift to solve the problem of shift cycle, so as to ensure the accuracy of the automatic shift control.

Keywords:*technical vehicles, Adaptive Genetic Algorithms, neural network, Intelligent Control*

1 Introduction

It is very important to realize automatic shift for engineering vehicles, because its poor working conditions and complicated operation condition. Realizing automatic shift can improve the performance and fuel economy of engineering vehicles, it can also improve efficiency and quality of projects, lighten the working strength of drivers, etc. At present, there is not a perfect automatic shift technology for engineering vehicles, they can only use the automatic shift technology of ordinary cars. Engineering vehicles mainly work on engineering work, so they will consume lots of power of the engine, and the change range is very large. This is very different from the cars used for driving. We cannot greatly improve the engineering vehicle‘s performance and fuel economy if we applicate the existing gear shift technology. Therefore, studying automatic shift technology of the engineering vehicle has important theoretical and practical significance.

Parmee [1] predicts that automatic transmission vehicle chooses the best shift rule based on traffic conditions of the vehicle, road conditions, and the driver's information of manipulation. The method of gear position decision is the core of developing technology for the vehicle automatic transmission system. It affects the vehicle's power performance, economy and comfort. Therefore the shift decision of taking automatic transmission technology will directly affect the vehicle. Yoshimura and Hirako [2] argued that the rule of shift decision changed from the traditional shift decision to intelligent shift decision. Traditional shift decision divided into the shift decision of single parameter, two parameters and three parameters according to the different control parameters selected. The shift decision of single parameter selected the vehicle‘s speed which can reflect the state of the vehicle synthetically as the control parameter. The shift decision of three parameters selected accelerator , speed and acceleration as the control parameters. Its performance is superior to the shift decision of single parameter and two parameters that we just mentioned according to the test [3]. The limitation of the three methods of shift decision above is that it is only suitable for balanced driving condition. When drivers encounter ramp, corners, brake and other complicated conditions, drivers have to shift the gear very frequently. And the shift decision based on traditional shift rules is different from the shift decision that drivers want. So there is a method of gear position decision based on fuzzy logic and knowledge of experts. Mangan et al. [4] thinks the shift decision of fuzzy logic utilizes the information of the road‘s condition, the operation habits of drivers and the operation condition of vehicles. The working principle is collecting the information of the vehicle‘s running status and the driver's operation information. We can estimate the driver's operation purpose by the fuzzy inference 1, such as acceleration, overtaking, braking and deceleration, etc. The shift decision of fuzzy logic can fully reflect the driver's intention. It fits the driver's actual operation process in the same condition. It also can effectively solve the problems that shift gear frequently and unexpectedly. But we must also know that the shift decision given by the decision of fuzzy logic is not the best when on fine road. Because the shift decision is based on statistics and analysis. The establishment of the rule base and repository of fuzzy logic‘s shift decision need consulting lots of experienced drivers and experts. The shift rules formed by it cannot cover all the condition of roads in the process of driving.

The author thinks that we can resolve the problem of the shift decision of automatic transmission vehicle. An important aspect of vehicle‘s development is intelligent control. And the key to intelligent vehicle is the intelligent gear shifting. Panos [5] thinks there are two methods for intelligent shift decision: one is based on the theory of neural network , the other one is based on fuzzy logic comprehensive ability. Kevin [6] considers neural network has the adaptability, it can be trained. Neural network can learn and remember experience in the process of training. Yamakawa [7] classified the intelligent shift decision can use the data obtained by the driver on gear shifting to train the neural network offline. Kosko [8] thinks it also can make the network learning the driver's best shift points online, and make the automatic transmission system has the ability of self learning.

The method of gear position decision develops gradually from traditional shift decision to the intelligent gear position decision. The author thinks that the intelligent gear position decision is the development direction of the automatic transmission vehicle. Because it considers the operating conditions, such as traffic environment and operational intention of the drivers and other information , it can satisfy the vehicle performance and fuel economy and drivability.

2 Problem statement

Gear position decision is one of the key technologies of the automatic transmission technology, namely determines the current vehicle gear according to the situation of vehicle running, road conditions and driver's intention. The gear position decision must obey some objective optimal principles. At present, the automatic transmission technology of the vehicle has entered the era of intelligence, and people began to pursue higher intelligent vehicles.

The shift schedule is the basis for automatic transmission system to achieve the best dynamic and economic performance, and the shift schedule is also the core to control automatic transmission system. Shift schedule refers to the law that two automatic shift changes when the control parameters change. According to the control parameters, shift schedule could be classified into single parameter, two-parameter and three-parameter shift schedule. Single parameter most involves vehicle speed; two-parameter involves vehicle speed and accelerator opening degree; three-parameter involves vehicle speed, accelerator opening degree and acceleration. More control parameters used, better automatic shift effect will achieve. However, the shift system becomes more complex, and costs will be higher. Therefore, this system uses a two-parameter control.

(1) Best power shift. Best power shift schedule lead to the car's traction fullest utilization and engine power' maximum effectiveness. In order to keep no loss of momentum, the same speed and acceleration speed should be guaranteed at the shift points. In the relationship curve between different throttle opening speed and acceleration speed, adjacent curve intersection point under certain throttle opening is the shift point to ensure the optimum power. If no intersection point of two curves at high throttle or lower throttle, the maximum speed for the low-end throttle could be recognized as shift points. Linking the best shift points under each throttle opening could form the best shift power characteristic curve. As shown in Fig. 1, point A is the power shift point under a throttle opening. (2) Most economic shift schedule. Economy shift schedule strive to meet the minimum 100 km fuel consumption requirements, to ensure that, the car should travel with smallest stall the engine's fuel consumption rate. Best vehicle economy shift schedule obtained by the vehicle's fuel economy curve. Assuming shift times is very short which means the speed is the same during the shift time; in order to ensure the car is good power and comfort ride, traction shifting conditions should keep unchanged before or after the shift. Resulting from the equal traction shifting conditions, the throttle opening should automatically adjust and adapt to the shift. Assuming the shift traction is a constant, it is represented as a horizontal line at certain traction characteristics. According to curve intersection formed by the horizontal line and the neighboring two different traction characteristic lines of throttle opening degree, the speed corresponding to an accelerator opening degree obtained. According to curve Q1=f (Va), fuel consumption of the speed range and throttle opening degree could be computed. With the vehicle speed and fuel consumption, the fuel consumption curve of two tranches can be obtained, and its intersection is the best fuel consump-

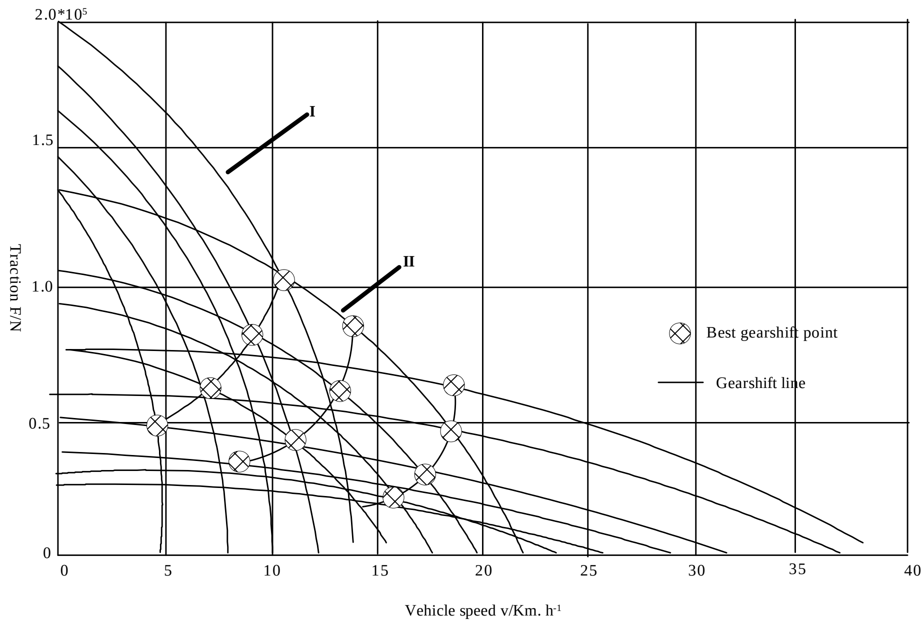

Fig. 1. Advance-speed traction characteristics

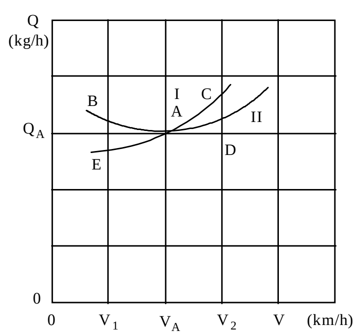

Fig. 2. Economy shift point method

tion shift points. Reason by analogy, shift points of the adjacent two tranches could be computed with a given adjacent, and the connection line of the shift points is the lowest fuel consumption of two adjacent tranche. The shift profiles is based on the shift points corresponding to a vehicle speed and traction, and it can analyze the impact on the vehicle traction characteristics to obtain the economy shift schedule in accordance with the corresponding accelerator opening degree and the vehicle speed. Fig. 2 shows fuel consumption characteristic curve of two adjacent stalls, and the point A is the intersection of the two fuel consumption characteristic curve. Point A has the same fuel consumption of two tranches, which is the optimum economy shift point. Both schedule are ideal shift schedule, the system uses a neural network to learn the two shift schedule. When use neural to simulate the shift schedule in the actual running of the vehicle, we should consider the vehicle's power, economy, intent road environment conditions and so on, and by setting the switch to select the shift mode.

3 Optimal control of automatic gearshift operations by AGA-NN

3.1 Structure optimization of RBF neural network

RBF neural network, as one of the most popular neural networks, has been proved that as long as there are enough nodes in the hidden layer, mapping at arbitrary precision could be realized. However, it does mean that more nodes will better. Increasing the complexity of the network can improve mapping accuracy, at the same time, the cost will increase. First, excessive hidden nodes will cause over fitting of the training data, which has a bad effect on unknown data. Furthermore, factors, such as initial values of the network parameters, training sample characteristics and outside interference, have an influential impact on the network connection weights, which make little changes exist between the input mode and the training sample, and then correct results cannot be obtained. Second, large number of parameters also means an increase in the probability of parameter estimation errors. Third, it increases the possibility of portion minimum value.

In current researches, training RBF networks usually use K-means clustering algorithm to determine the hidden node centers. Considering the same input sample mode may correspond to more than one cluster, the number of hidden nodes obtained with clustering algorithm is still excessive and the excessive number of hidden nodes will cause the over-learning problems. To solve this problem, reducing the network size and optimizing the learning network should be done. After unsupervised clustering, this paper will use genetic algorithm to optimize the RBF network, choose the most significant cluster center and finally complete the network optimization.

3.2 Adaptive Genetic optimization algorithm

Since evolutionary computation was proposed, ingrowing researchers has been interested in simulating evolution to solve complex optimization problems. Among them, the genetic algorithm introduced by Holland [9] is paid more and more attention to. As a kind of meta-heuristics, it could search the optimal solution without regard to the specific inner connections of the problem. Especially, the application of GA to multiobjective optimization problems has caused a theoretical and practical challenge to the mathematical community. In the past two decades, there are many approaches on GA developed by the scholars in all kinds of field. Globerg [10] firstly suggested the Pareto ranking based fitness assignment method to find the next set of nondominated individuals. Then the multiobjective genetic algorithm in which the rank of individual corresponds to the number of current parent population was proposed by Fonseca and Fleming [11]. There are still two weighted sum genetic algorithms to solve multiobjective optimization problems. One is the random-weight genetic algorithm proposed by Ishibuchi et al.[12], the other is adaptive-weight genetic algorithm proposed by Gen and Chen [13]. In this section, the GA based on adaptive weight approach introduced by Xu and Yao [14] is combined with NN algorithm to optimize the RBF network.

3.2.1 Adaptive weight approach

The adaptive weight approach proposed by Gen and Cheng [13] makes use of the useful information from current generation to readjust the weights of every objectives,

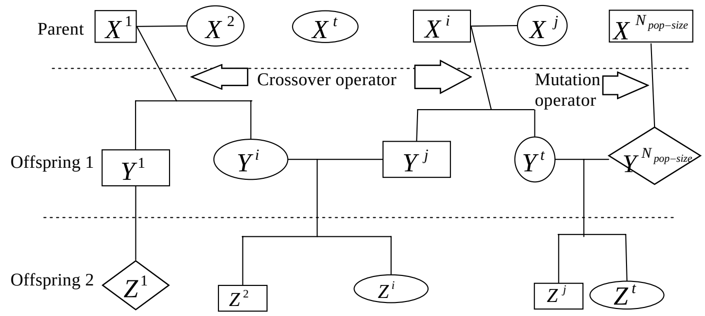

Fig. 3. Genetic operators

then obtains a search pressure towards to positive ideal points. Let P denote the set of the current population, we can define the maximal and minimal values as follows for each objective, respectively.

$$z_k^{max} = \max\{E[f_k(\boldsymbol{x}, \boldsymbol{\xi})] | \boldsymbol{x} \in P\}, k = 1, 2, \cdots, m.$$
$$z_k^{min} = \min\{E[f_k(\boldsymbol{x}, \boldsymbol{\xi})] | \boldsymbol{x} \in P\}, k = 1, 2, \cdots, m.$$

Then the adaptive weight for objective k is calculated by the following equation:

$$w_k = \frac{1}{z_k^{max} - z_k^{min}}, k = 1, 2, \cdots, m.$$

For a given individual x, the weighted-sum objective function is given by the following equation:

$$z(\boldsymbol{x}) = \sum_{k=1}^{m} \frac{E[f_k(\boldsymbol{x}, \boldsymbol{\xi})] - z_k^{min}}{z_k^{max} - z_k^{min}}.$$

3.2.2 Genetic operators

When GA proceeds, some important factors should be considered, such as, the search direction to optimal solution and the search speed and so on. In general, the exploitation of the accumulated information resulting from GA search is done by the selection mechanism, while the exploitation to new regions of the search space is accounted by genetic operators. Readers could to refer to Fig. 3 to get the intuitive understanding.

(1) Selection process. Selection provides the driving force in a GA. With too much force, genetic search will be slower than necessary. The selection directs the genetic search toward promising regions in the search space. Roulette wheel selection, proposed by Holland [9] is the best known selection type. The basic idea is to determine selection probability or survival probability for each chromosome proportional to the fitness value. We can apply the roulette wheel method to develop the selection process. Each time a single chromosome for a new population is selected in the following way: Compute the total probability q,

$$q = \sum_{j=1}^{N_{pop-size}} eval(\boldsymbol{x}^j).$$

Then compute the probability of the ith chromosome q_i, $q_i = \frac{eval(\boldsymbol{x}^i)}{q}$. Generate a random number r in $[0, 1]$ and select the ith chromosome $\boldsymbol{x}_i$ such that $q_{i-1} < r \leq q_i, 1 \leq i \leq$

$N_{pop-size}$. Repeat the above process $N_{pop-size}$ times and we obtain $N_{pop-size}$ copies of chromosomes. The selection probability can be computed by the following function,

$$p_i = \frac{eval(\boldsymbol{x}^i) - eval(\boldsymbol{x})_{min}}{\sum_{j=1}^{pop-size} eval(\boldsymbol{x}^i) - eval(\boldsymbol{x})_{min}}$$

where $eval(\boldsymbol{x})_{min}$ is the minimum fitness value of current population.

(2) Crossover operation. Crossover is the main genetic operator. It operates on two chromosomes at a time and generates offspring by combing both chromosomes' features. The crossover probability (denoted by P_c) is defined as the probability of the number of offspring produced in each generation to the population size. This probability controls the expected number $P_c \cdot N_{pop-size}$ of chromosomes to undergo the crossover operation. The detailed step is as follows. Generate a random number c from the open interval (0, 1) and the chromosome $\boldsymbol{x}^i$ is selected as a parent provided that $c < P_c$, where parameter P_c is the probability of crossover operation. Repeat this process $N_{pop-size}$ times and $P_c \cdot N_{pop-size}$ chromosomes are expected to be selected to undergo the crossover operation. The crossover operator on x^1 and x^2 will produce two children y^1 and y^2 as follows:

$$\boldsymbol{y}^1 = c\boldsymbol{x}^1 + (1-c)\boldsymbol{x}^2, \qquad \boldsymbol{y}^2 = c\boldsymbol{x}^2 + (1-c)\boldsymbol{x}^1.$$

If both children are feasible, then we replace the parents with them, or else we keep the feasible one if it exists. Repeat the above operation until two feasible children are obtained or a given number of cycles is finished.

(3) Mutation operation. Mutation is a background operator which produces spontaneous random changes in various chromosomes. In GA, mutation serves the crucial role of either replacing the genes lost from the population during the selection process so that they can be tried in a new context, or providing the genes that were not present in the initial population. The mutation probability (denoted with P_m) is defined as the percentage of the total number genes in the population. The mutation probability controls the probability at which new genes are introduced into the population for trial. The detailed process is as follows. Similar to the crossover process, the chromosome $\boldsymbol{x}^i$ is selected as a parent to undergo the mutation operation provided that random number $m < P_m$, where parameter P_m as the probability of mutation operation. $P_m \cdot N_{pop-size}$ are expected to be selected after repeating the process $N_{pop-size}$ times. Suppose that $\boldsymbol{x}$ is chosen as a parent. Choose a mutation direction $\mathbf{d} \in \mathbf{R^n}$ randomly. Replace $\boldsymbol{x}$ with $\boldsymbol{x} + M \cdot \mathbf{d}$ if $\boldsymbol{x} + M \cdot \mathbf{d}$ is feasible, otherwise we set M as a random between 0 and M until it is feasible or a given number of cycle is finished. Here, M is a sufficiently large positive number.

3.2.3 Procedure for GA

We illustrate the Ra-Fu simulation-based genetic algorithm procedure as follows:

Procedure `The procedure for GA`

Input: The parameters $N_{pop-size}, P_c$ and P_m
Output: The optimal chromosomes
Step 1. Initialize $N_{pop-size}$ chromosomes whose feasibility may be checked by Ra-Fu simulation;
Step 2. Update the chromosomes by crossover and mutation operations and Ra-Fu simulation is used to check the feasibility of offspring. Compute the fitness of each chromosome based on weight-sum objective;
Step 3. Select the chromosomes by spinning the roulette wheel;
Step 4. Make the crossover operation;
Step 5. Make the mutation operation for the chromosomes generated by crossover operation;
Step 6. Repeat the second to fourth steps for a given number of cycles;
Step 7. Report the best chromosome as the optimal solution.

3.3 AGA-NN: adaptive genetic algorithm-based neural network

To use genetic algorithms to optimize the central values in RBF network hidden layer, first, data should be encoded. Set the number of central values at s obtained by learning as a population of vector, and take ID of the s data as population (a chromosome) for encoding. Each individual's adaptive function can be selected as a reciprocal which is the sum of the absolute value between the expected output and actual output in RBF network and this selection directly reflect the good or bad quality of each chromosome. According to the chromosome adaption, choose some chromosome in current population to constitute a new population. Then produce offspring through putting individuals in new population into genetic operators (crossover and mutation), compare the offspring to parent and put the offspring into the parent to keep the population size. Finally, check GA termination condition. If it meets the termination condition, GA end, on the contrary, repeat the above steps. By the genetic optimization algorithm, the optimal number of hidden layer nodes could be determined. The concrete steps are as follows.

(1) Coding. Coding is to translate candidate solution of the problem space into a chromosome or individual which is composed of a certain structure in the genetic space. This article uses K-means clustering algorithm to obtain the numbers of RBF center (the number of RBF center is s) and take those numbers as chromosome to be encoded, specially set C_i with pointer $i(i = 1, 2...s)$ identity. RBF network structure encoded the following chromosomes: the value of chromosome genes is a natural number from $1 \sim s$ and genetic value actually represents the radial basis function pointer obtained by the selected radial basis function center in K-means clustering algorithm. The length of the number of genes in RBF neural network hidden layer is unknown in advance, so that we use chromosome variable length encoding.

(2)Initial population. Population initialization, it includes selections in cross-scale, crossover probability pc and mutation probability pm. Population size affects the final result of genetic optimization and efficiency in the implementation of the GA. When the population size is too small, the efficiency of GA optimization is low. However, larger population size can reduce possibility in portion optimal, but it implies high computational complexity. Set the population size at n, which is limited in the range [10,160].

(3) Crossover operation design. First, cross the two parent individual (p_1, p_2) at the probability p_c, and randomly generated the cross-initial position (b_1, b_2) and the cross-end position (E_1, E_2). Then make the offspring retain gene cluster of parent individual before the start position and after the cross position. Replace the genes which are different from $p_1 - p_2$ genes and existing in $p_1 - p_2$ two intersecting position, with gene cluster in $p_1 - p_2$ two intersecting position, and then pass it to the offspring.

(4)Variation. Set the new chromosome at p_1, which were mutation produced at the probability p_m. In the complement element sets (i.e., Number sets of all the genes that different from individual), randomly choose one or several elements to replace genes in a random position of the individual. Then, insert the new chromosome into the population, and calculate the adaptation degree of new chromosome.

(5)Add&delete operators design. We could add or delete operations in algorithm to increase the flexibility of the structure optimization, and specific design is as follows.

Add operator: randomly select the integer La in the range $(0, s - L)$ as the added length, where L is the length of chromosomes, and randomly add the data (the data number was set at L_a) into an individual tail to produce new individuals.

Delete operators: randomly select the integer L_d in the range $(0, L)$ L_d gene as delete length, and deleted the genes (the gene number is L_d) in individuals tail to produce new individuals.

(6) Select the objective function and adaptive function. The objective function provides a method to evaluate the pros and cons of chromosome performance in the problem space. In this article, the objective function can select the sum (the sum is set at $E(i)$ of the absolute value between expected output and the actual output in the RBF network, and the function $f(i)$ can be selected as the countdown of $E(i)$ as Eq 1.

$$E(i) = \sum_{j=0}^{n} |e_j(i)| \tag{1}$$

$$f_i = \frac{1}{E(i)} \tag{2}$$

where, the weight adjustment use recursive least squares algorithm (RLS).

(7) Design the algorithm terminated guidelines. The algorithm terminates guidelines should take into account of the learning accuracy and algorithm speed, and the criterion is that the optimal index remains unchanged in S-generation consecutive learning. That is, when the algorithm initialize, set the optimal indicators as deviation of optimal chromosome in population, and setting the terminating counting variable at m is 0. If optimal index changes during the generation evolution, update the optimal index, and set $m = 0$; otherwise, let $m = m + 1$ and once $m = s$, the algorithm ends.

3.4 Design the control system

To solve the application problems of engineering vehicle automatic transmission control, the neural network is applied to the control system design for the nonlinear, uncertain and complex system, and genetic optimization algorithm optimizes the structural parameters of the neural network according to the performance of the control system, in order to achieve optimal control of the controlled object. In neural network control system, information processing is usually divided into adaptive learning phases and control phases. During the control phases, the network connection modes and the weights

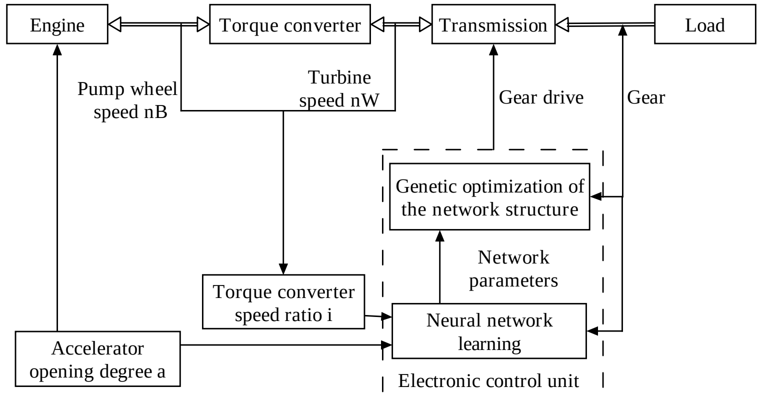

Fig. 4. Genetic-RBF neural network control system program

remain known and unchanged, and the neurons producing output values are mainly based on the input information and status information. In learning phases, the network adjusts its internal weights according rules and optimizes the given performance indicators. The two phases are independent, and can also be alternately performed. The learning principle is shown in Fig. 4.

4 Simulation and result discussion

GA-NN was used to control the automatic transmission shift of engineering vehicles. According to the driver's experience and expert knowledge, the shift schedule formed. Striving to make the automatic transmission gear selection and manipulation process similar, the shift schedule should consider more factors and indicators in its decision-making process. During the process, as long as the input and output data trained, neural network model which has the input-output data characteristic simulation equivalent to the actual process could be obtained. For there is no necessity to make deeper analysis in the process or object, it is very convenient to set up a process-unknown system model like neural network. Furthermore, neural network has a strong advantage in real-time learning and automatic pattern recognition for controlling automatic shift. However, sample data is a prerequisite before using neural network and the control effect depends entirely on the quality of the sample data.

This paper used the GA-NN to simulate the automatic transmission shift. Based on the simulation, clutch control was achieved and an automatic transmission shift control system was designed.

4.1 The automatic transmission shift model

4.1.1 Automatic transmission shifts decision

Main parameters to reflect engine and vehicle operating condition include the engine speed, the accelerator opening degree, vehicle speed and so on. Although it is easy to detect engine speed, the value is the flux in the shift transition process. Therefore it is not suitable as a shift control parameter. However, the vehicle speed in the shift process remains substantially unchanged due to the inertia of the vehicle itself, so that it can be

Table 1. Experimental data for neural network shift

Sample number	Input data		Output data				
	a	v	G1	G2	G3	G4	G5
1	0	0	1	0	0	0	0
2	20	0	1	0	0	0	0
...	...	...	...	...	...	...	...
18	20	18	0	1	0	0	0
...	...	...	...	...	...	...	...
63	20	40	0	0	1	0	0
...	...	...	...	...	...	...	...
104	20	60	0	0	0	1	0
...	...	...	...	...	...	...	...
149	100	195	0	0	0	0	1

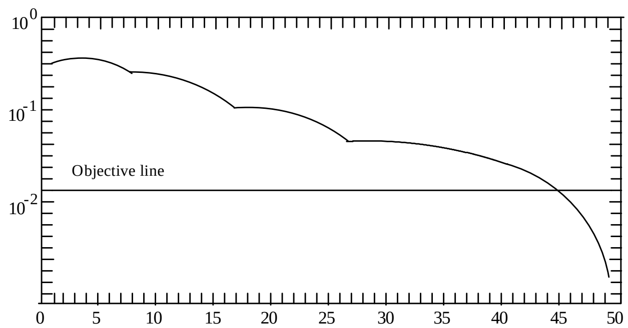

Fig. 5. Shift neural network training chart error range of results

used as the shift control parameters to reflect the vehicle state. The accelerator opening degree reflects the vehicle driving power requirements of the driver. Theoretically, the vehicle speed and the throttle opening has a relationship (segmented nonlinear function relationship) with degree optimal gear.

BP network use offline training, and set throttle opening at a, vehicle speed at v and the target gear at i as input / output samples. Calculate and train the network by Matlab / Toolbox function, and then store the trained network weights value and threshold into the appropriate controller. Two hidden layers must be used to solve the non-convex domain problem in neural network. Compare the experiment data and finalize the number of neurons in every layer is 2-3-5-5, the selected iteration convergence target is 0.01, and the maximum number of convergence step is 500. With randomly initialize weights and threshold, 67 times of training will meet the requirements. Set L_M as the training algorithm and part of the sample data shown in the Tab. 1. Where k is a sample number; a, v represent units of centigrade and km/h of the throttle opening degree and the vehicle speed value, respectively; G1-G5 respectively are corresponding stall ideal output value. Train those normalized data samples through the network. After training, the deviation should be checked and the results show a significant impact on the shift determine. The training results are shown in Fig. 5. Back to training weights and deviation values in neural network, the network training process the error curve is shown in Fig. 5.

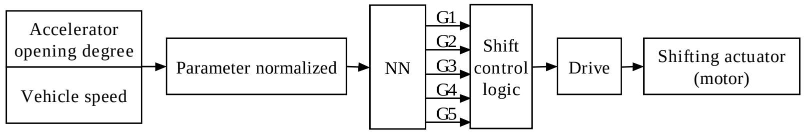

Fig. 6. The best stalls do and shift control logic diagram

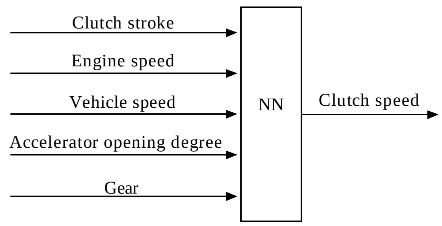

Fig. 7. Clutch neural network input and output

Fig. 6 shows a control logic diagram of an optimal gear and shift. The largest values corresponding to the shift is the best shift when get G1 as an output. If G2 is the largest, then the best is the shift 2. At the same time, the gearbox should set for this gear. Shift control logic part integrated the output G1, G2... G5, changes in throne dealing and give control series signals. Those signals will promote the operation part finish the shift process after power amplification. To avoid the cycle shift resulting from shift power off, the shift control has a delay in shift change of G1, G2 ... G5. That means no longer response to network standard shift change after one shift in a period. When there is one or more disappear in G1, G2 ... G5, the current gear will not change if one is the same as the current gear; otherwise, choose the closet one.

4.2 Clutch control

The combining process of the clutch is influenced by a plurality of parameters. In a controlled process, all the parameters cannot be taken as a control parameter. From the simple use perspective, set the accelerator opening degree at , gearshift at ig, the engine rotational speed at NE and the input shaft speed at ni as control parameters. Those signals have a nonlinear relationship to clutch speed. It is difficult to express it with a function relationship or the linear approximation. The neural network could achieve the nonlinear relationship between them, as shown in Fig. 7.

From the simulation results, the automatic transmission system can learn by itself according to the experimental sample data and modify those parameters of the neural network, then obtain the shift principle for saving energy. As the operational states of vehicles and the revolving speed of the engine change, the automatic transmission system will make some changes according to the principle of the energy saving shift. After the vehicles stably operate, the proposed algorithm can guarantee that the efficiency of the automatic transmission system always stay an efficient range. Therefore, it shows that the proposed model and algorithm for realizing automatic shift of engineering vehicles to keep the torque converter working high-efficiency is feasible, accurate and reliable.

5 Conclusion

Optimal control of automatic gearshift operations by AGA-NN is considered in this paper. It introduces intelligent neural network control technique and gene algorithm to deal with the automatic transmission shift control system. The AGA-NN assesses the current state and uses each layer's approaches individually to realize the automatic gearshift operations. A numerical example is conducted to apply the proposed measure. The result is satisfying. Further, the combination of the two methods have balanced the time and cost. It does not only make up the disadvantages of each method, but also improves the whole efficiency.

6 Acknowledgements

This work was financially supported by the Natural Science Foundation of China (No.51077096) and the Science and Technology Support Program of Science and Technology Department of Sichuan Province(2010GZ0173).

References

[1] K. Parmee. Challenges for transmission engineering (past and future), SAE transactions. 1057-1069 **(1989)**

[2] H. Yashimura and A. Hirako. Automated mechanical transmission control, Proceeding of the International Congress on Transportation Electronics. 179-189 **(1986)**

[3] K. Nordgard and H. Hoonhrst. Developments in automated clutch management systems, Society of Automotive Engineers. 137-144 **(1995)**

[4] S. Mangan and J. Wang and Q. Wu. Longitudinal road gradient estimation using vehicle CAN bus data, IEEE International Conference on Systems, Man and Cybernetics. 2336-2341 **(2003)**

[5] A. J. Panos. Neural networks for control systems, IEEE Transactions on Neural Networks. 242-244 **(1990)**

[6] K. M. Passino. Neural computing for numeric-to-symbolic conversion in control systems, IEEE Control Systems Magazine. 44-52 **(1989)**

[7] T. Yamakawa. High-speed fuzzy controller hardware system: The mega-FIPS machine, Information Sciences. 45, 113-128 **(1988)**

[8] B. Kosko. Neural networks and fuzzy systems: a dynamical systems approach to machine intelligence, Prentice-Hall **(1991)**

[9] J. H. Holland. Adaptation in natural and artificial systems, MIT Press Cambridge, USA **(1992)**

[10] D. E. Goldberg. Genetic algorithms in search, optimization, and machine learning, Addison-wesley **(1989)**

[11] C. M. Fonseca and P.J. Fleming. An overview of evolutionary algorithms in multiobjective optimization, Evolutionary computation. 3, 1-16 **(1995)**

[12] H. Ishibuchi and T. Murata. A multi-objective genetic local search algorithm and its application to flowshop scheduling, IEEE Transactions on Systems, Man, and Cyberneticsart C: Applications and Reviews. 28, 392-403 **(1998)**

[13] M. Gen and R. Cheng, Genetic algorithms and engineering optimization, Wiley-interscience **(2000)**

[14] J. Xu and L. Yao, Random-like Multiple Objective Decision Making, Springer **(2011)**