10th World Congress on Structural and Multidisciplinary Optimization
May 19 -24, 2013, Orlando, Florida, USA

Scheme for positions of radial basis functions and radius considering supports for accuracy of approximation in convolute RBF

Masao ARAKAWA[1], Satoshi KITAYAMA[2]

[1] Kagawa University, Takamatsu, Kagawa, JAPAN, arakawa@eng.kagawa-u.ac.jp
[2] Kanazawa University,Kanazawa, Ishikawa, JAPAN, kitayama-s@se.kanazawa-u.ac.jp

1. Abstract

There are a lot of studies on surrogate optimization, and most of them use Gaussian distribution type of kernels as basis function. In these cases, there are a few keys to make success. No matter which type of unification, such as Kriking, SVR and/or RBF, this parameter optimization is the most important parts for better approximation. “Better” means not only giving accurate approximation for given data, but also for good approximation for less density area of given data. Without any doubt, if you give kernels on the given data, and give a small radius, local accuracy will rise, but there are no generality. Thus, it is very questionable to use approximate function without giving additional data around it. As for recommendation of teaching data, which is the second important key in surrogate optimization, important points are giving data for approximate optimum point and some additional points in less density area. These ideas are used in so called EGO, recommendation function by authors, and so on.

In this study, we propose a scheme to give kernel position together with setting radius for convolute RBF. In convolute RBF, we gave radius of each convolution priori to learning. And we give radius from large radius to small radius in a few steps. We assumed generality of each kernel can be determined from the number of given data with in radius of the kernel. We called the number of data for that kernel as supports. And we automatically adjust the given radius, so that the smallest radius kernel must have at least two supports to keep generality. For that purpose, we calculate distance between given data and find maximum of minimum distance and calculate ratio between given smallest radius, and adjust all radius before we give kernels. Next, we calculate average of function values of given data, and shift all data so that average of given data will come as 0. In each convolution, we give kernels position as follows. First, we calculate the maximum error position. Then calculate gravity position that is given by the data included with in the given radius from maximum error position. Next, we do the same thing for the minimum error position. And we learn from RBF and calculate error again. We do the same scheme up to the given number of kernels for each convolution. After learning, we optimize the approximate functions. We give additional data, not only to de optimum of full convolution, but also to some lower number of convolutions. In this study, we have examined the effectiveness of the proposed method with some benchmarking tests, and showed the effectiveness of the method.

2. Keywords: Surrogate Optimization, RBF, Data distribution

3. Introduction

There are a lot of optimization softwares these days and they tend to prepare approximate optimization, so called surrogate optimization. And, good approximation becomes more and more important these days. Kriging[1], SVM[2] and RBF[3,4,5,6,7,8] are the good candidates. There are definitely some differences but all of them have the same character that they do not care so much on basis function, they can update partially, and they are so good at both local and global approximation. We have been developing sequential approximation toolbox with RBF. Among them we had some industrial examples like optimization of synchrotron[9], and they show the effectiveness of function approximation. Since then, we add recommendation of new set of design variables just like EGO in Kriging[1] that use information of approximate function and existence of sets of design variables with the information of constraints. Next, we have applied this toolbox to multi-objective optimization using satisficing method[10]. Shirakawa[11] showed industrial example in designing control method of power plant. Kitayama[12] showed optimization of the problem how to deal with spring back problem. Probably, more and more applications were reported and also are going to be.

If we focus our attention to RBF, key issues are considered to determine proper radius for each basis function. Nakayama and Kitayama proposed equation to give radius, they consider width of range and neighbor of each basis function and according to those information, they give some formula. They work very well, but they tend to have large radius for the basis function close to their edge of range. Arakawa and Sugimoto optimized them. Arakawa give radius for each design variable to reduce the number of radius to be optimized. Again, they worked very well, but it was time consuming. To overcome these situations, we have proposed convolution technique. As RBF remain some error in each teaching data, we use them as new teaching data in convolution. It is easy to give radius and raise accuracy in teaching data. However, there are some limitation in generality of approximation, when we use teaching data as the center of basis function.

In this study, we are going to propose scheme to give position of the center of basis function and also to give supports for accuracy, and examined its effectiveness through numerical example.

4. RBF Approximation

RBF is a kind of neural network and it approximates the objective function by weighted sum of radial basis functions. Output of RBF is given by

$$f(x) = \sum_{j=1}^{m} w_j h_j(x) \tag{1}$$

We use Gaussian-like functions as basis function given by

$$h(x) = Exp\left(-\sum_{k=1}^{p} \frac{(x_k - c_k)^2}{{r_k}^2} \right) \tag{2}$$

where c_k and r_k is the center and the radius of the basis. In this study, we optimize radius for each design variable by minimizing error. Learning in RBF is to obtain appropriate weights for each basis, and it is made by minimizing the network energy that is calculated form sum of the square error and the term introduced for purpose regularization. Given the training data (x_i, y_i), it is usually made by solving

$$E = \sum_{i=1}^{n} (y_i - f(x_i))^2 + \sum_{j=1}^{m} \lambda_j {w_j}^2(x) \to \min \tag{3}$$

Letting $A = (H^T H + \Lambda)$, we have necessary condition for the above minimization.

$$A\mathrm{w} = H^T \mathrm{y} \tag{4}$$

Where

$$H = \begin{bmatrix} h(\mathrm{x}_1, \mathrm{c}_1, r_1) & h(\mathrm{x}_1, \mathrm{c}_2, r_2) & \cdots & h(\mathrm{x}_1, \mathrm{c}_m, r_m) \\ h(\mathrm{x}_2, \mathrm{c}_1, r_1) & h(\mathrm{x}_2, \mathrm{c}_2, r_2) & \cdots & h(\mathrm{x}_2, \mathrm{c}_m, r_m) \\ \vdots & \vdots & \ddots & \vdots \\ h(\mathrm{x}_p, \mathrm{c}_1, r_1) & h(\mathrm{x}_p, \mathrm{c}_2, r_2) & \cdots & h(\mathrm{x}_p, \mathrm{c}_m, r_m) \end{bmatrix} \tag{5}$$

$$\Lambda = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_m \end{bmatrix} \tag{6}$$

Therefore, learning in RBF network comes down to calculate inverse matrix A^{-1}, and the weights of basis are calculated as follows

$$\mathrm{w} = A^{-1} H^T \mathrm{y} \tag{7}$$

We combine RBF and polynomials in this method. RBF appropriate for approximation of nonlinear and complex problems, but it is inappropriate for global approximation of simple problems or those problems that with low density of data. We use quadratics as base of RBF to cover the weakness of RBF. Therefore, proposed method can have both global approximation and accurate local approximation. The approximation is made by following procedure. As complexity can be followed by RBF, we neglect interaction and make it easier as followings.

$$\bar{y}(\mathbf{x}) = \Phi(\mathbf{x}) + RBF(\mathbf{x}) \tag{8}$$

In this way, we can follow trend of function by polynomial, and error between polynomial has average of zero. That is suitable for RBF especially using Gaussian function.

5. Convolute RBF

As a nature of RBF, it remains error on teaching data. And most likely, their average is close to zero. Therefore, they are quite suitable data sets for RBF approximation. Convolute RBF is a simple concept that convolute RBF approximation again and again till designed and/or satisfied.

$$\bar{y}(\mathbf{x}) = \Phi(\mathbf{x}) + \sum_{i=1}^{N} RBF_i(\mathbf{x}) \tag{9}$$

When we optimize radius for each RBF, it tends to cover from big waves with larger radius. Gradually, response surface of error become small waves, and RBF tends to cover them and radius becomes smaller. Indeed, if we know these tendencies, we do not need to optimize radius. We can give them from larger radius to smaller ones by schedule. Even then, we do not have any difference when we use the same center of basis function.

When we convolute RBF for a few times, we can make error almost zero. However, it does not mean we have reached to generality. As it is good for reproducing teaching data, but there are no guarantee for those without teaching data. In a sense of neural network, it means over learning.

6. Position of basis function

In RBF, if we place basis function on teaching data, we can easily reduce error by giving small radius. Indeed, if we give them small enough so that they lost interconnection, error might become close to zero. But, which make response surface like combination of needles. In order to guarantee generality, one of choices is to place basis function other than teaching data. It might be easy to place it randomly to its surrounding area. However, when its position is close to opposite sign area, weight for basis function becomes lower than its original position. So, we would like to position the center of basis function a little bit shifts to the same sign range as followings.

Scheme

(1) Find maximum absolute error of teaching data. (input data as X)

(2) Count the number of data within radius of R(user's setting) that has the same sign of output data, and calculate average for each input data. (average as A)

(3) Position basis function as follow

$$\mu_i = \alpha X_i + (1-\alpha) A_i \tag{10}$$

(4) Give self-weight λ_t (user's setting) and radius $\mathbf{r}_t$ (user's setting), and make RBF.

(5) Calculate new error of teaching data as

$$O_{new}(\mathbf{x}_j) = O_{old}(\mathbf{x}_j) - RBF(\mathbf{x}_j) \tag{11}$$

(6) If the number of basis functions becomes presetting number (user's setting), then if the number of convolution becomes presetting number (user's setting) then finish, otherwise exchange O_{old} to Onew, give all basis functions to archived data and refresh all basis functions and go to (1), otherwise go to (1).

Final RBF is composed with all archived ones.

7. Numerical Example

We use a famous pressure vessel problem as an example. Table 1 shows comparison of the results in some methods. As this problem is mixed variable problem, so it is very complicated problem. In GRGAs, it needs more than 10,000 function calls to obtain final results, and also it needs more than 6000 function calls to get close to global optima. Figure 2 shows the results of the proposed method.

Table 1 Comparison of the results of Pressure Vessel Problem

	Sandgren Penalty 1990	Qian GA 1993	Kannan ALM 1994	Lin SA 1992	Hsu GA 1995	Lewis RS+NLP 1996	Arakawa ARGA 1997
R m	1.212	1.481	1.481	N/A	1.316	0.985	0.986
L m	2.990	1.132	1.198	N/A	2.587	5.672	5.626
Ts cm	2.858	2.858	2.858	N/A	2.540	1.905	1.905
Th cm	1.588	1.588	1.588	N/A	1.270	0.953	0.953
g1	0.840	1.000	1.000	N/A	1.000	0.997	1.000
g2	0.747	0.890	0.890	N/A	0.989	0.986	1.000
g3	0.445	0.186	0.182	N/A	0.424	0.938	0.922
g4	1.000	1.000	1.000	N/A	0.831	0.930	1.000
f $	8129.80	7238.83	7198.20	7197.70	7021.67	5980.95	5850.38

Table 2 was the results that we have started from the same three different data sets that have 50 data at the beginning in the conventional method (including optimization of radius). Correlation means that we have added 10 new data after “Cut 5900”, and it is the average of correlation of actual value and RBF outputs for all functions. Actually, for the teaching data, it is almost always 1 for teaching data, therefore it is close to 1. We use the same strategy to put new data in this example. Table 3 shows scheduling of self-weight and radius in each convolution, and we use the number of radial basis functions to the size of 60% of teaching data, where in Table 2 we have used 75% of them. That means we have more difficulty to catch global solution in this example. Table 4 shows the results of the proposed method.

Table 2 Results of the proposed method

Case	1	2	3
Close to Global	140	70	130
Inside Constraints	140	100	130
Cut 5900	140	100	140
Its cost	5872.44	5899.39	5894.09
Correlation	0.987	0.985	0.991

Table 3 Schedule of Radius and Self-weight

Convolution	Raidus	Self-Weight
1	12	1
2	8	0.1
3	4	0.05
4	2	0.01
5	1	0.001

Table 4 Results of the proposed method

Case	1	2	3
Close to Global	90	60	80
Inside Constraints	110	100	120
Cut 5900	120	100	120
Its cost	5885.24	5888.68	5875.98
Correlation	0.994	0.992	0.993

As we can see in Table 4, although we have less number of basis functions, we have reached to global solution with less number of function calls, and also there correlation between actual data and RBF outputs are better than the conventional ones. We think this tendency shows the effectiveness of the proposed method that we have shifted the place of center of basis a little bit, and we have raised generality of the approximation.

8. Conclusions

In this paper, we have proposed the scheme to place position of basis function for convolute RBF. With pre-scheduling of radius and self-weights, we do not have to optimize radius, therefore the speed of learning becomes so fast. And we have begun from larger radius, which means we make sure that we have a number of supports of teaching data for basis function at the beginning of convolution. With those supports, especially teaching data with the same sign, we have shifted the center of basis function a little bit for that direction. As a results, in the numerical example, we have reached to global solution faster than the conventional method, and also we have had a better correlation for unknown data. Which means that we can raise generality of approximation.

6. References

[1]Michael J. Sasena, Panos Y. Papalambros, Pierre Goovaerts, Metamodeling Sampling Criteria in a Global Optimization Framework, AIAA-2000-4921.

[2]Nakayama H., Yun, Y., 2006, "MOP/GP Approaches to Support Vector Regression and Their Applications to Engineering Design", Proc. 7th int. Conf. on Mulit-objective Programming and Goal Programming.

[3]Arakawa,M., Nakayama, H., Ishikawa,H., Optimum Design Using Radial Basis Function Network and Adaptive Range Genetic Algorithms (1st Report: Consideration in Unconstrained Optimization), Trans. of JSME, Vol67,No.665C, 2001, pp.797-802 (in Japanese).

[4]Arakawa,M., Nakayama, H., Ishikawa,H.,Approximate Optimization Using RBF Network and Genetic Range Genetic Algorithms (Proposal of base function and basic consideration), Trans. of JSME, vol.70, No697C, 2004. (in Japanese)

[5]Arakawa, M.,Nakayama, H.,Ishikawa, H.,Approximate Optimization Using Radial Basis Function Networks and Genetic Range Genetic Algorithms,10th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, AIAA, Albany,(2004 - 9)

[6]Shirai, T.,Arakawa, M.,Nakayama,H,Ishikawa, H.,Approximate Optimization Using Combination of Regression and RBF network,The 3th China-Japan-Korea Joint Symposium on Optimization of Structural and Mechanical System, 2004, pp.107-112. , CJK-OSM, Kanazawa,(2004 - 11)

[7]Tokura,S, Arakawa,M. et.al, 2003,"Uniform Magnetic Field in Bending Magnet for the Compact Proton Synchrotron", IHI technical Report, 43-1,2-6.

[8]Shirai,T. Arakawa,M., Nakayama,H.,2006, Approximate Multi-Objective Optimization Using Satisficing Method and Radial Basis Function Networks, Proc. CJK-OSM4, 497-502

[9]Shirakawa,M., Kawai, K., Arakawa,M., Nakayama, H., 2006,"Intelligent Start-up Schedule Optimization System For A Thermal Power Plant", Prof. IFAC

[10]Deb, 2001, Multi-Objective Optimization Using Evolutionary Algorithms

[11]Nakayama, Tanino, 1994, Theory and Application of Multi-Objective Programming, (in Japanese).

[12]Arakawa, Miyashita, Ishikawa, Development of Genetic Range Genetic Algorithms (3rd Report, Proposal of Fitness Function to Search Multiple Local Optima), Trans. of JSME, 70-695C,(2062-2069), 2004. (in Japanese).