2017 10th International Symposium on Computational Intelligence and Design

Fast Modeling Methods for Complex System with Separable Features

Chen Chen[†‡], Changtong Luo[*†], Zonglin Jiang[†‡]
[†]*State Key Laboratory of High Temperature Gas Dynamics, Institute of Mechanics, Chinese Academy of Sciences, Beijing 100190, China*
[‡]*School of Engineering Sciences, University of Chinese Academy of Sciences, Beijing 100049, China*
[*]*Email: luo@imech.ac.cn*

***Abstract*—Data-driven modeling plays an increasingly important role in different areas of engineering. For most of existing methods, such as genetic programming (GP), the convergence speed might be too slow for large scale problems with a large number of variables. Fortunately, in many applications, the target models are separable in some sense. In this paper, we analyze different types of separability and establish a generalized separable model (GSM). In order to get the structure of the GSM, a multi-level block search method is proposed, in which the target model is decomposed into a number of blocks, further into minimal blocks and factors. Compare to the conventional GP, the new method can make large reductions to the search space. The minimal blocks and factors are optimized and assembled with a global optimization search engine, low dimensional simplex evolution (LDSE). An extensive study between the proposed method and a state-of-the-art data-driven fitting tool, Eureqa, has been presented with several man-made problems. Test results indicate that the proposed method is more effective and efficient under all the investigated cases.**

***Keywords*-data-driven modeling; genetic programming; generalized separable model; multi-level block search**

I. Introduction

Data-driven modeling has emerged as a powerful technique in different areas of engineering, such as industrial data analysis [9], circuits analysis and design [13], signal processing [14], system identification [5], etc. Data-driven modeling aims to find a function that best explains the relationship between independent variables and the objective value based on a given set of sample data. Among the existing methods, genetic programming (GP) [7] is a classical approach. GP can get an optimal solution provided that the computation time is long enough. However, the computational cost of GP for a large scale problem is still very expensive. Hence, how to use a appropriate method to solve such problem is considered as a kaleidoscope in this research field [1].

In many scientific or engineering problems, the target model are separable. Luo et al. [8] have presented a divide-and-conquer (D&C) method for GP. The authors indicated that the solving process could be accelerated by detecting the correlation between each variable and the target function [3]. In [8], a special method, bi-correlation test (BiCT), was proposed to divide a concerned target function into a number of sub-functions. Compared to conventional GP, D&C method could reduce the computational effort by orders of magnitude.

In this article, different types of separability are discussed, and a generalized separable model (GSM) is established. In order to get the structure of the GSM, a multi-level block search method is proposed, in which the target model is decomposed into a number of blocks, further into minimal blocks and factors. The new method is an improved version of [8] and [2]. The performance of the proposed method is compared with the results of Eureqa, which is a state-of-the-art data-driven fitting tool. Numerical results show that the proposed method is effective, and is able to recover all the investigated cases rapidly and reliably.

II. Types of separability

In this section, three examples of real-world problems are given as follows to illustrate several common types of separability in practical problems.

Example 1. *When developing a rocket engine, it is crucial to model the internal flow of a high-speed compressible gas through the nozzle. The closed-form expression for the mass flow through a choked nozzle is*

$$\dot{m} = \frac{p_0 A^*}{\sqrt{T_0}} \sqrt{\frac{\gamma}{R}\left(\frac{2}{\gamma+1}\right)^{(\gamma+1)/(\gamma-1)}}. \tag{1}$$

In Eq. (1), the five independent variables, p_0, T_0, A^*, R and γ are all separable. Eq. (1) can be called a multiplicatively separable model, which can be re-expressed as follows

$$\begin{aligned}\dot{m} &= f\left(p_0, A^*, T_0, R, \gamma\right)\\ &= \varphi_1\left(p_0\right) \times \varphi_2\left(A^*\right) \times \varphi_3\left(T_0\right) \times \varphi_4\left(R\right) \times \varphi_5\left(\gamma\right).\end{aligned} \tag{2}$$

Example 2. *In aircraft design, the lift coefficient of a whole aircraft can be expressed as*

$$C_L = C_{L\alpha}\left(\alpha - \alpha_0\right) + C_{L\delta_e}\delta_e \frac{S_{HT}}{S_{ref}}, \tag{3}$$

where the variable $C_{L\alpha}$, $C_{L\delta_e}$, δ_e, S_{HT} and S_{ref} are separable. The variable α and α_0 are not separable, but their combination (α, α_0) can be considered separable. Hence,

2473-3547/17 $31.00 © 2017 IEEE
DOI 10.1109/ISCID.2017.144

Eq. (3) can be re-expressed as

$$\begin{aligned} C_L &= f\left(C_{L\alpha}, \alpha, \alpha_0, C_{L\delta_e}, \delta_e, S_{HT}, S_{\mathit{ref}}\right) \\ &= \varphi_1\left(C_{L\alpha}\right) \times \varphi_2\left(\alpha, \alpha_0\right) \\ &+ \varphi_3\left(C_{L\delta_e}\right) \times \varphi_4\left(\delta_e\right) \times \varphi_5\left(S_{HT}\right) \times \varphi_6\left(S_{\mathit{ref}}\right). \end{aligned} \tag{4}$$

Example 3. *The flow past a circular cylinder is a classical problem in fluid dynamics. A valid stream function for the inviscid, incompressible flow over a circular cylinder of radius R is*

$$\psi = \left(V_\infty r \sin\theta\right)\left(1 - \frac{R^2}{r^2}\right) + \frac{\Gamma}{2\pi}\ln\frac{r}{R}, \tag{5}$$

which can be re-expressed as

$$\begin{aligned} \psi &= f\left(V_\infty, \sin\theta, R, r, \Gamma\right) \\ &= \varphi_1\left(V_\infty\right) \times \varphi_2\left(\sin\theta\right) \times \boxed{\varphi_3\left(r, R\right)} \\ &+ \varphi_4\left(\Gamma\right) \times \boxed{\varphi_5\left(r, R\right)}. \end{aligned} \tag{6}$$

Eq. (5) can be considered as a qusi-separable model. Note that the variable r and R appear twice. In other words, variable r and R have two sub-functions, namely $\varphi_3\left(r, R\right) = \left(1 - R^2/r^2\right) \cdot r$ and $\varphi_5\left(r, R\right) = \ln\left(r/R\right)$.

The models of Example 1 and 2 have been well studied in [8] and [2], respectively. The authors indicated that detecting the correlation between each variable and the target function could accelerate the solving process. This article aims to establish the mathematical model of a qusi-separable function, which is given in Example 3.

III. Generalized separable model

The definition of a generalized separable model is given as follows.

Definition 1. *The Generalized separable model $f\left(X\right)$ with n continuous variables $X = \{x_i : i = 1, 2, \cdots, n\}$, ($f : \mathbb{R}^n \mapsto \mathbb{R}$, $X \subset \Omega \in \mathbb{R}^n$, where Ω is a closed bounded convex set, such that $\Omega = [a_1, b_1] \times [a_2, b_2] \times \cdots \times [a_n, b_n]$) is defined as*

$$\begin{aligned} f\left(X\right) &= f\left(X^r, \bar{X}^r\right) = c_0 + \sum_{i=1}^{m} c_i \varphi_i\left(X_i^r, \bar{X}_i^r\right) \\ &= \sum_{i=1}^{m} c_i \tilde{\omega}_i\left(X_i^r\right) \tilde{\psi}_i\left(\bar{X}_i^r\right) \\ &= c_0 + \sum_{i=1}^{m} c_i \prod_{j=1}^{p_i} \omega_{i,j}\left(X_{i,j}^r\right) \prod_{k=1}^{q_i} \psi_{i,k}\left(\bar{X}_{i,k}^r\right), \end{aligned} \tag{7}$$

where the variable set $X^r = \{x_i : i = 1, 2, \cdots, l\}$ is a proper subset of X, such that $X^r \subset X$, and the cardinal number of X^r is card$\left(X^r\right) = l$. $\bar{X}^r$ is the complementary set of X^r in X, i.e. $\bar{X}^r = \complement_X X^r$, where card$\left(\bar{X}^r\right) = n - l$. X_i^r is the subset of X^r, such that $X_i^r \subseteq X^r$, where card$\left(X_i^r\right) = r_i$. $X_{i,j}^r \subseteq X_i^r$, such that $\bigcup_{j=1}^{p_i} X_{i,j}^r = X_i^r$, $\bigcap_{j=1}^{p_i} X_{i,j}^r = \emptyset$, where card$\left(X_{i,j}^r\right) = r_{i,j}$, for $i = 1, 2, \cdots, m$, $j = 1, 2, \cdots, p_i$ and $\sum_{j=1}^{p_i} r_{i,j} = r_i$. $\bar{X}_i^r \subset \bar{X}^r$ ($\bar{X}_i^r \neq \emptyset$), such that $\bigcup_{i=1}^{m} \bar{X}_i^r = \bar{X}^r$, $\bigcap_{i=1}^{m} \bar{X}_i^r = \emptyset$, where card$\left(\bar{X}_i^r\right) = s_i$, for $s_i \geqslant 1$, $\sum_{i=1}^{m} s_i = n - l$. $\bar{X}_{i,k}^r \subseteq \bar{X}_i^r$, such that $\bigcup_{k=1}^{q_i} \bar{X}_{i,k}^r = \bar{X}_i^r$, $\bigcap_{k=1}^{q_i} \bar{X}_i^r = \emptyset$, where card$\left(\bar{X}_{i,k}^r\right) = s_{i,k}$, for $k = 1, 2, \cdots, q_i$ and $\sum_{k=1}^{q_i} s_{i,k} = s_i$. Sub-functions φ_i, $\tilde{\omega}_i$, $\tilde{\psi}_i$, $\omega_{i,j}$ and $\psi_{i,k}$ are scalar functions, such that $\varphi_i : \mathbb{R}^{r_i + s_i} \mapsto \mathbb{R}$, $\tilde{\omega}_i : \mathbb{R}^{r_i} \mapsto \mathbb{R}$, $\tilde{\psi}_i : \mathbb{R}^{s_i} \mapsto \mathbb{R}$, $\omega_{i,j} : \mathbb{R}^{r_{i,j}} \mapsto \mathbb{R}$ and $\psi_{i,k} : \mathbb{R}^{s_{i,k}} \mapsto \mathbb{R}$, respectively. $c_0, c_1, \cdots, c_m$ are constant coefficients.

The function structure of GSM is defined as follows.

Definition 2. *In Eq. (7), the variables belong to X^r and $\bar{X}^r$ are called repeated variables and non-repeated variables, respectively. The sub-function $\varphi_i\left(\cdot\right)$ is called the i-th minimal block of $f\left(X\right)$, for $i = 1, 2, \cdots, m$. Any combination of the minimal blocks is called a block of $f\left(X\right)$. The sub-functions $\omega_{i,j}\left(\cdot\right)$ and $\psi_{i,k}\left(\cdot\right)$ are called the j-th and k-th factors of the repeated variables and non-repeated variables in i-th minimal block $\varphi_i\left(\cdot\right)$, respectively, for $j = 1, 2, \cdots, p_i$ and $k = 1, 2, \cdots, q_i$.*

IV. Separability detection and model determination

In order to detect the separability of the GSM, we aim to divide GSM into a suitable number of minimal blocks, and further into factors as the typical Example 3. This technique can be considered as an improved version of [8] and [2]. The modeling process of GSM mainly includes two parts, namely inner optimization and outer optimization. The inner optimization will be invoked to determine the function model and coefficients of the factors $\omega_{i,j}\left(X_{i,j}^r\right)$ and $\psi_{i,k}\left(\bar{X}_{i,k}^r\right)$. Fortunately, many state-of-the-art optimization techniques, e.g., parse-matrix evolution [11], low dimensional simplex evolution [10], artificial bee colony programming [6], etc. can all be easily used to optimize the factors. Then, the optimized factors of each minimal block are multiplied together to produce minimal blocks. The outer optimization aims at combining the minimal blocks together with the proper global parameters c_i. The whole process for modeling a GS system can be briefly described as follows:

1) (Minimal block detection) Partition a GS system into a number of minimal blocks with all the repeated variables fixed;
2) (Factor detection) Divide each minimal block into factors;
3) (Factor determination) Determine the factors by employing an optimization engine;
4) (Global assembling) Combine the optimized factors into minimal blocks multiplicatively, further into an optimization model linearly with proper global parameters.

The flowchart of the modeling process could be briefly illustrated in Fig. 1. The proposed technique is described with functions with explicit expressions. While in practical applications, no explicit expression is available. In fact, for data-driven modeling problems, a surrogate model [4] of black-box type could be established as the underlying target function in advance.

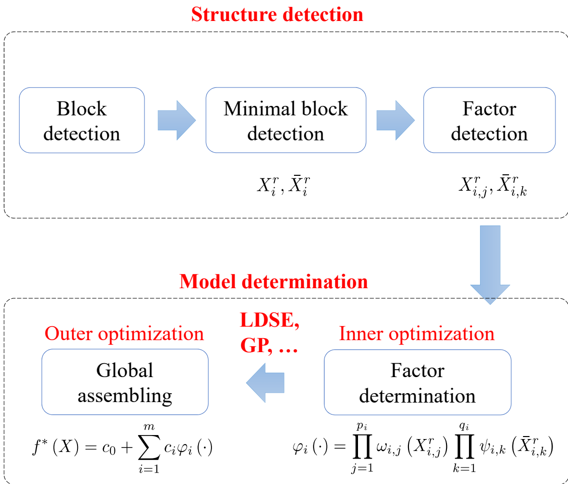

Figure 1. Flowchart of modeling process.

V. Numerical results and discussion

In our implementation, a kind of global optimization method, low dimensional simplex evolution (LDSE) [10], is chosen as the optimization engine. LDSE is a hybrid evolutionary algorithm for continuous global optimization. The performances including 'structure optimization' and 'coefficient optimization' capabilities of the proposed method are tested by comparing with a state-of-the-art software, Eureqa [12], which is a data-driven fitting tool based on genetic programming (GP). Eureqa was developed at the Computational Synthesis Lab at Cornell University by H. Lipson. 10 test cases are taken into account.

The calculation conditions are set as follows. The number of sampling points for each independent variable is 200. The regions for cases 1-5 and 7-10 are chosen as $[3, 3]$, while case 6 is $[1, 3]$. The control parameters in LDSE are set as follows. The upper and lower bounds of fitting parameters is set as 50 and 50. The population size N_p is set to $N_p = 10 + 10d$, where d is the dimension of the problem. Sequence search and optimization method is suitable for global optimization strategy. The search will exit immediately if the mean square error is small enough (MSE $\leqslant \varepsilon_{\text{target}}$), and the tolerance (fitting error) is $\varepsilon_{\text{target}} = 10^{-6}$. In order to reduce the effect of randomness, each test case is executed 20 times.

The computing time (CPU time) consists three parts, $t = t_1 + t_2 + t_3$, where t_1 is for the separability detection, t_2 for factors modeling, and t_3 for global assembling. In [8], authors have demonstrated that both the separability detection and function recover processes are double-precision operations and thus cost much less time than the factor determination process. That is, $t \approx t_2$. It is very easy to see that the computational efficiency of the proposed method is higher than Eureqa's. Note that our method is executed on a single processor, while Eureqa is executed in parallel on 8 processors.

VI. Conclusion

In this article, different types of separability are discussed, and a generalized separable model (GSM) is established. In order to get the structure of the GSM, a multi-level block search method is proposed, in which the target model is decomposed into a number of blocks, further into minimal blocks and factors. Compare to the conventional GP, the new method can make large reductions to the search space. The proposed method is an improved version of [8] and [2]. The minimal blocks and factors are optimized and assembled with a global optimization search engine, low dimensional simplex evolution (LDSE). The proposed method is tested on 10 man-made test cases. Remarkable performance is concluded after comparing with a state-of-the-art data-driven fitting tool, Eureqa. Numerical results show the algorithm is effective, and can get the target function more rapidly and reliably.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (Grant No. 11532014).

References

[1] Chen, C., Luo, C., & Jiang, Z. (2017). A multilevel block building algorithm for fast modeling generalized separable systems. arXiv: 1706.02281.

[2] Chen, C., Luo, C., & Jiang, Z. (2017). Block building programming for symbolic regression. arXiv: 1705.07877.

[3] Chen, C., Luo, C., & Jiang, Z. (2017). Elite Bases Regression: A Real-time Algorithm for Symbolic Regression. In: 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery. Guilin, China.

[4] Forrester, A., Sobester, A., & Keane, A. (2008). Engineering design via surrogate modelling: a practical guide. John Wiley & Sons.

[5] Guo, H., & Li, Z. (2012). Structural damage identification based on bayesian theory and improved immune genetic algorithm. Expert Systems with Applications, 39, 6426-6434.

[6] Karaboga, D., Ozturk, C., Karaboga, N., & Gorkemli, B. (2012). Artificial bee colony programming for symbolic regression. Information Sciences, 209, 1-15.

[7] Koza, J. R. (1992). Genetic programming: On the programming of computers by means of natural selection. (5th ed.). Cambridge, MA: MIT Press.

[8] Luo, C., Chen, C., & Jiang, Z. (2017). A divide and conquer method for symbolic regression. arXiv: 1705.08061.

Table I
10 TEST CASES.

No.	Target model
1	$f(\mathbf{x}) = 0.5 * e^{x_1} * \sin 2x_2$
2	$f(\mathbf{x}) = 2 * \cos x_1 + \sin(3x_2 - x_3)$
3	$f(\mathbf{x}) = 1.2 + 10 * \sin 2x_1 - 3 * x_2^2 * \cos x_3$
4	$f(\mathbf{x}) = x_3 * \sin x_1 - 2 * x_3 * \cos x_2$
5	$f(\mathbf{x}) = 2 * x_1 * \sin x_2 * \cos x_4 - 0.5 * x_4 * \cos x_3$
6	$f(\mathbf{x}) = 10 + 0.2 * x_1 - 0.2 * x_5^2 * \sin x_2 + \cos x_5 * \ln(3x_3 + 1.2) - 1.2 * e^{0.5x_4}$
7	$f(\mathbf{x}) = 2 * x_4 * x_5 * \sin x_1 - x_5 * x_2 + 0.5 * e^{x_3} * \cos x_4$
8	$f(\mathbf{x}) = 1.2 + 2 * x_4 * \cos x_2 + 0.5 * e^{1.2x_3} * \sin 3x_1 * \cos x_4 - 2 * \cos(1.5x_5 + 5)$
9	$f(\mathbf{x}) = 0.5 * \frac{\cos(x_3 x_4)}{e^{x_1} * x_2{}^2} * \sin(1.5x_5 - 2x_6)$
10	$f(\mathbf{x}) = 1.2 - 2 * \frac{x_1 + x_2}{x_3} * \cos x_7 + 0.5 * e^{x_7} * x_4 * \sin(x_5 x_6)$

Table II
COMPARATIVE RESULTS OF THE MEAN PERFORMANCES BETWEEN THE PROPOSED METHOD AND EUREQA FOR MODELING 10 TEST CASES.

Case No.	Dim	No. samples	Our method					Eureqa		
			Repeated variable	No. block	No. factor	CPU time	MSE	CPU time	MSE	Remarks
1	2	400	None	1	2	**7s**	$\leqslant \varepsilon_{\text{target}}$	7s	$\leqslant \varepsilon_{\text{target}}$	Solutions are all exact
2	3	600	None	2	2	**9s**	$\leqslant \varepsilon_{\text{target}}$	1m 9s	$[4.05, 7.68] \times 10^{-4}$	2 runs failed
3	3	600	None	2	3	**9s**	$\leqslant \varepsilon_{\text{target}}$	1m 9s	$\leqslant \varepsilon_{\text{target}}$	2 runs failed
4	3	600	x_3	2	4	**11s**	$\leqslant \varepsilon_{\text{target}}$	55s	$\leqslant \varepsilon_{\text{target}}$	Solutions are all exact
5	4	800	x_4	2	5	**14s**	$\leqslant \varepsilon_{\text{target}}$	2m 28s	$\leqslant \varepsilon_{\text{target}}$	3 runs failed
6	5	1000	x_5	4	6	**21s**	$\leqslant \varepsilon_{\text{target}}$	$\gg$ 6m 25s	$[4.79, 14.2] \times 10^{-6}$	All runs failed
7	5	1000	x_4, x_5	3	7	**16s**	$\leqslant \varepsilon_{\text{target}}$	$\gg$ 8m 38s	$[4.05, 7.68] \times 10^{-4}$	All runs failed
8	5	1000	x_4	3	6	**15s**	$\leqslant \varepsilon_{\text{target}}$	$\gg$ 8m 38s	$[4.05, 7.68] \times 10^{-4}$	All runs failed
9	6	1200	None	1	4	**9s**	$\leqslant \varepsilon_{\text{target}}$	$\gg$ 8m 38s	$[4.05, 7.68] \times 10^{-4}$	All runs failed
10	7	1400	x_7	2	6	**11s**	$\leqslant \varepsilon_{\text{target}}$	$\gg$ 8m 38s	$[4.05, 7.68] \times 10^{-4}$	All runs failed

[9] Luo, C., Hu, Z., Zhang, S.-L., & Jiang, Z. (2015). Adaptive space transformation: An invariant based method for predicting aerodynamic coefficients of hypersonic vehicles. Engineering Applications of Artificial Intelligence, 46, 93-103.

[10] Luo, C., & Yu, B. (2012). Low dimensional simplex evolution a new heuristic for global optimization. Journal of Global Optimization, 52, 45-55.

[11] Luo, C., & Zhang, S.-L. (2012). Parse-matrix evolution for symbolic regression. Engineering Applications of Artificial Intelligence, 25, 1182-1193.

[12] Schmidt, M., & Lipson, H. (2009). Eureqa (version 1.24.0) [software]. Available from www.nutonian.com; [accessed: 2017.05.01].

[13] Shokouhifar, M., & Jalali, A. (2015). An evolutionary-based methodology for symbolic simplification of analog circuits using genetic algorithm and simulated annealing. Expert Systems with Applications, 42, 1189-1201.

[14] Yang, Y. W., Wang, C., & Soh, C. K. (2005). Force identification of dynamic systems using genetic programming. International Journal for Numerical Methods in Engineering, 63, 1288-1312.