Presented as late breaking results contribution
21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Hybrid MODE-SVR Algorithm for Nonparametric Dynamic System Identification of Autonomous Helicopter

Ismaila B. Tijani*, Rini Akmeliawati**

*Electrical Engineering, ADMC, Higher Colleges of Technology,
Abu Dhabi, UAE (e-mail: tismail@hct.ac.ae).
**School of Mechanical Engineering, University of Adelaide, Australia.
(e-mail: rini.akmeliawati@adelaide.edu.au)

Abstract: Practical application of SVR in nonparametric modeling requires not only achievement of acceptable model accuracy but also optimal reduction of the model complexity in terms of the associated support vectors. Attaining these performance metrics is not only challenging due to inherently conflicting nature of the duo performances, but also as a result of several structural parameters needed to be tuned in SVR deployment. In order to address this problem, a hybrid algorithm of SVR based on Multi Objective Differential Evolution (MODE-SVR) is proposed to search for the SVR structural parameters that provides Pareto-based optimal solutions for both model complexity and accuracy. The proposed algorithm is evaluated on nonparametric model of a UAV helicopter yaw dynamics. Performance analysis and comparative study with an existing method in MATLAB shows the effectiveness of the proposed hybrid algorithm. This is expected to simplify and enhance the practical application of SVR in machine learning applications.

Keywords: Hybrid system identification, Nonparametric methods, Support vector regression

1. INTRODUCTION

Support Vector Regression is known as an extension of Support Vector Machine (SVM) to regression problems with introduction of ε-insensitivity loss function by Vapnik (1995). It is an interesting tool that is often used for nonparametric modeling based on experimental or continuous data. The applications are ranging from runoff forecasting (Chu et.al, 2016), stock price prediction in stock market (Rustam, 2018) to friction modeling for a DC-motor driven rotary motion system (Tijani, 2012a).

In this paper, we propose a hybrid algorithm MODE-SVR that provides optimal search for the structural parameters of SVR for nonparametric modelling of the dynamics of a small-scale helicopter. Such optimal search will lead to Pareto-based optimal solutions to the model complexity and accuracy that are the typical conflicting performance specification in system modeling. The MODE-algorithm performs the optimal tuning of the SVR parameters.

The rest of this paper is organized as follows. Section 2 presents the proposed hybrid MODE-SVR algorithm and the illustration on an autonomous helicopter. Section 3 presents the results of the MODE-SVR nonparametric modeling.

1.1 Support Vector Regression

SVR is based on the principle of structure risk minimization, which minimizes an upper bound on the expected risk. This is different from traditional learning algorithms for function estimation, such as Neural Network that minimizes the error on the training data based on the principle of empirical risk minimization. Thus, SVR provides better ability to generalize, and at the same time less prone to the problems of overfitting and local minimal. The following describes SVR algorithm (Smola, 2001).

Given a set of N input/output data $\{x_i, y_i\}_{i=1}^{N}$ such that $x_i \in \Re^n$ and $y_i \in \Re$, the goal of learning theory is to find a function f which minimizes the expected risk (1):

$$R[f] = \int L(y, f(x)) dP(x, y) \quad (1)$$

where $L(y, f(x))$ is a loss is function, and $P(x, y)$ is unknown probability measure which is assumed to be responsible for the generation of the data.

Since function P is unknown, expression (1) cannot be directly computed, hence unlike traditional ERM principle that minimizes only the empirical risk (training error), statistical learning theory seeks to obtain a small risk in terms of both training error and model complexity by minimizing the regularized risk function (structural risk function) (2):

$$R_{reg}[f] := C.R_{emp}[f] + \frac{1}{2}\|w^2\| \quad (2)$$

where C is a constant determining the trade-off with the complexity penalizer, $\frac{1}{2}\|w^2\|$ is the regularization term (or complexity penalizer) used to find the flattest function with sufficient approximation qualities, $R_{emp}[f]$ is empiric risk defined as (3):

$$R_{emp}[f] := \frac{1}{N}\sum_{i=1}^{N} L(y_i, f(x_i)) \quad (3)$$

Employing ε-insensitivity SVR (ε-SVR), the loss function $L(y_i, f(x_i))$ is replaced by the Vapnik's ε-insensitivity loss function given as (Vapnik, 1995):

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

Therefore,

$$R^{\varepsilon}{}_{emp}[f] := \frac{1}{N}\sum_{i=1}^{N}|y_i - f(x_i)|_{\varepsilon} \tag{5}$$

and the goal of the function estimation is thus to minimizes (6.6):

$$R_{reg}[f] := C.\frac{1}{N}\sum_{i=1}^{N}|y_i - f(x_i)|_{\varepsilon} + \frac{1}{2}\|w^2\| \tag{6}$$

For non-linear regression in the primal weight space the model is of the form:

$$f(x) = \omega^T \varphi(x) + b \tag{7}$$

where for the given training set $\{x_i, y_i\}_{i=1}^{N}$, $\varphi(\cdot): \Re^n \to \Re^{n_h}$ is a mapping to a high dimensional feature space by the application of the kernel trick define as (8) (Sch¨olkopf, 2002):

$$K(x_i, x_j) = \varphi(x_i)^T \varphi(x_j) \tag{8}$$

The constrain optimization problem in the primal weight space is given by (9):

$$\min_{\omega,b,\xi,\xi^*} J_P(\omega,\xi,\xi^*) = \frac{1}{2}\omega^T\omega + C\sum_{i=1}^{N}(\xi_i + \xi_i^*) \tag{9}$$

Subject to: $y_i - \omega^T\varphi(x) - b \le \varepsilon + \xi_i \quad i = 1,2\cdots,N$

and $\omega^T\varphi(x) + b - y_i \le \varepsilon + \xi_i^* \quad i = 1,2,\cdots,N$

where ξ_i, ξ_i^* are the slack variables for soft margin.

By defining the Lagrangian and applying the conditions for optimality solution, one obtains the following dual optimization problem (10):

$$\max_{\alpha,\alpha^*} J_D(\alpha,\alpha^*) = -\frac{1}{2}\sum_{i,j=1}^{N}(\alpha_i + \alpha_i^*)(\alpha_j + \alpha_j^*)K(x_i,x_j) - \varepsilon\sum_{i=1}^{N}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{N}y_i(\alpha_i + \alpha_i^*) \tag{10}$$

Subject to: $\sum_{i=1}^{N}\left(\alpha_i + \alpha_i^*\right) = 0$,

and $0 \le \alpha_i, \alpha_i^* \le C$ for all $i = 1,2,\cdots,N$

The regression estimates is expressed as (11):

$$f(x) = \sum_{i=1}^{N}(\alpha_i - \alpha_i^*)K(x_i, x_j) + b \tag{11}$$

where α_i, α_i^* are the Lagrange multipliers which are the solution to the Quadratic optimization problem, and b follows from the complementary Karush-Kuhn-Tucker (KKT) conditions which state that at the point of the solution, the product between dual variables and constraints has to vanish as follows:

$$\alpha_i(\varepsilon + \xi_i - y_i + \langle\omega, \varphi(x_i)\rangle + b) = 0, \tag{12a}$$

$$\alpha_i^*(\varepsilon + \xi_i^* + y_i - \langle\omega, \varphi(x_i)\rangle - b) = 0, \tag{12b}$$

$$(\tfrac{C}{N} - \alpha_i^*)\xi_i^* = 0, \tag{12c}$$

$$b = y_i - \langle\omega, \varphi(x_i)\rangle - \varepsilon \text{ for } \alpha_i \in \left(0, C/_N\right), \tag{12d}$$

$$b = y_i - \langle\omega, \varphi(x_i)\rangle - \varepsilon \text{ for } \alpha_i^* \in \left(0, C/_N\right). \tag{12e}$$

From the foregoing review, apart from the choice of the Kernel function, selection and tuning of associated optimization parameters such as kernel parameter, ε-loss function, conditioning parameter for the quadratic programming, known as the regularization parameter, λ (Vapnik, 1995), and bound on the Lagrangian multipliers, *C*, play important roles in overall performance of the regression process.

2. PROPOSED HYBRID MODE-SVR

2.1 Proposed Algorithm

The proposed hybrid MODE-SVR is shown in Fig.1. The figure indicates that the MODE sub-algorithm is deployed to override the conventional manual tuning of the SVR structural parameters, i.e. C, ε, σ, and λ, where σ is the Gaussian kernel parameter selected in this study. Hence, the MODE sub-algorithm searches for the SVR structure parameters that minimizes both the complexity of the model in terms of number of support vector (nsv) and the prediction error between the actual and predicted output given as

$$V_N = \frac{1}{2\chi}\sum_{n=1}^{\chi}\varepsilon^T\varepsilon \tag{13}$$

where ε is prediction error expressed as $\varepsilon = y(k) - \hat{y}(k)$.

The hybrid algorithm was developed in MATLAB using the SVR sub-algorithm in (Canu et.al., 2012). Detailed description of the MODE algorithm is reported in (Tijani, 2012a)

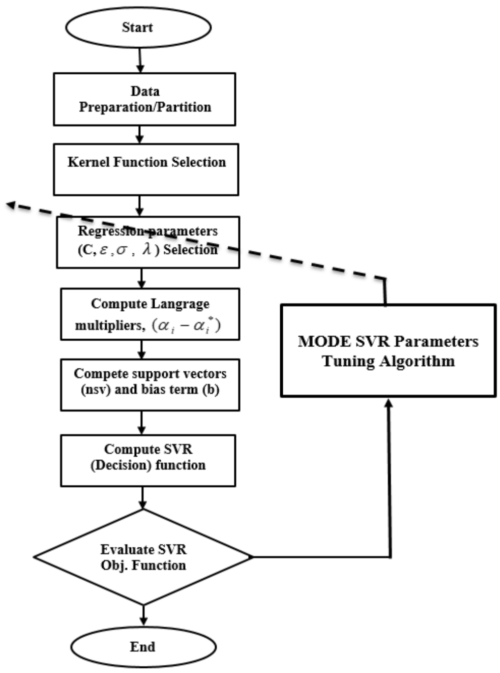

Fig. 1. Proposed hybrid MODE-SVR.

2.2 Helicopter Yaw Dynamics Flight Data

The proposed MODE-SVR technique is illustrated on yaw dynamics data of an autonomous small-scale helicopter, Hirobo SDX50 which is reported in (Tijani et al., 2012b).

Presented as late breaking results contribution
21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

The general physical specifications of the helicopter are given in Table 1.

Flight data was collected through the excitation of each of the system channels (roll, pitch, yaw and heave) with a sinusoidal input of varying frequency while keeping the system at desired hovering operating point as much as possible. The yaw dynamics data pair (pedal control input, *uped*, and yaw rate response, *r*) needed for this study.

Table 1. General specifications of Hirobo SDX50 Helicopter

Specifications		Specifications	
Fuselage length	1220 mm	Tail rotor diameter	258 mm
Fuselage width	186 mm	Gear ratio	8.70 : 4.71
Height	395 mm	Dry weight	3,400g
Main rotor diameter	1348 mm	Engine	50 class

The algorithm was implemented with the following pre-specified MODE parameters: number of decision variables, , D = 4; population size as multiple of variable dimension NP = 20*D; generation size, GEN = 40; crossover constant, CR = 0.5. The lower bound on the decision variables, L = 0.0000001, and the upper bound, U = 10. The process was repeated three times (RUNs) to evaluate consistence and convergence of algorithm. Fig. 2 shows the comparison of the resulting non-dominated Pareto solutions' front for the three RUNs. The detail solutions for the three RUNs comprises of objective functions/fitness, FitOB_NDS (nsv and mse), structure parameters, POP_NDS and constraint value, FitNC_NDS are presented in Table 2, 3 and 4.

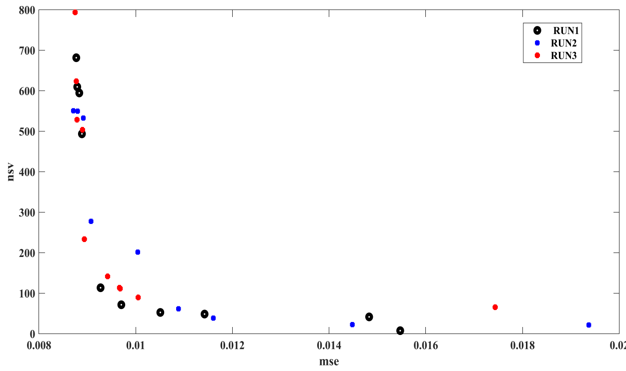

Fig. 2. Comparison of the Non-dominated Pareto solutions' front for the three RUNS

Table 2. First RUN

	FitOB_NDS		POP_NDS				FitNC_NDS
	mse	nsv	C	ϵ	σ	λ	mse
C1	0.00879	610	0.37933	0.04240	0.08630	0.57479	0.00879
C2	0.00889	494	0.93167	0.06216	0.01269	0.70845	0.00889
C3	0.00883	595	0.46957	0.04240	0.95287	0.04264	0.00883
C4	0.00877	682	0.16268	0.03351	0.44762	0.24202	0.00877
C5	0.01142	49	0.66856	0.22772	0.24282	0.27339	0.01142
C6	0.01483	42	0.95671	0.26337	0.86871	0.02433	0.01483
C7	0.01547	8	0.91621	0.32126	0.29328	0.01886	0.01547
C8	0.00927	114	0.84361	0.15205	0.05342	0.02792	0.00927
C9	0.00970	72	0.88726	0.17904	0.15362	0.03115	0.00970
C10	0.01051	53	0.09753	0.20197	0.19440	0.02330	0.01051

For further analysis, sample candidates were selected from the RUN1 (Table 2) to represent solution with highest complexity, RUN1-C4, solution with lowest complexity, RUN1-C7 and solution with average complexity, RUN1-C2. Fig. 3 and Fig. 4 show the comparative results of the three candidates with the experimental data.

Table 3. Second RUN

	FitOB_NDS		POP_NDS				FitNC_NDS
	mse	nsv	C	ϵ	σ	λ	mse
C1	0.00892	533	0.27324	0.05121	0.37306	0.60662	0.00892
C2	0.01937	22	0.53904	0.32047	0.28144	0.20487	0.01937
C3	0.01448	23	0.44797	0.30017	0.31580	0.05706	0.01448
C4	0.00908	278	0.29857	0.09032	0.16692	0.19459	0.00908
C5	0.00871	551	0.32733	0.04646	0.07737	0.05720	0.00871
C6	0.00880	550	0.27080	0.04834	0.36073	0.19459	0.00880
C7	0.01004	202	0.68524	0.11561	1.00768	0.08764	0.01004
C8	0.01161	39	0.33432	0.25375	0.21525	0.10198	0.01161
C9	0.01089	62	0.60723	0.19939	0.39517	0.23473	0.01089

Table 4. Third RUN

	FitOB_NDS		POP_NDS				FitNC_NDS
	mse	nsv	C	ϵ	σ	λ	mse
C1	0.01005	90	0.58102	0.15890	0.67499	0.07162	0.01005
C2	0.00875	794	0.56737	0.02042	0.14648	0.10860	0.00875
C3	0.00968	112	0.56951	0.15517	0.08426	0.33212	0.00968
C4	0.00890	504	0.57661	0.05684	0.06474	0.90165	0.00890
C5	0.00879	529	0.57377	0.04889	0.91778	0.01273	0.00879
C6	0.00894	234	0.56068	0.10170	0.07220	0.12888	0.00894
C7	0.00877	624	0.56784	0.04049	0.08646	0.51586	0.00877
C8	0.00942	142	0.57444	0.13680	0.19832	0.01290	0.00942
C9	0.00966	114	0.57321	0.15067	0.19832	0.25516	0.00966
C10	0.01743	66	0.71709	0.25005	0.27098	1.04060	0.01743

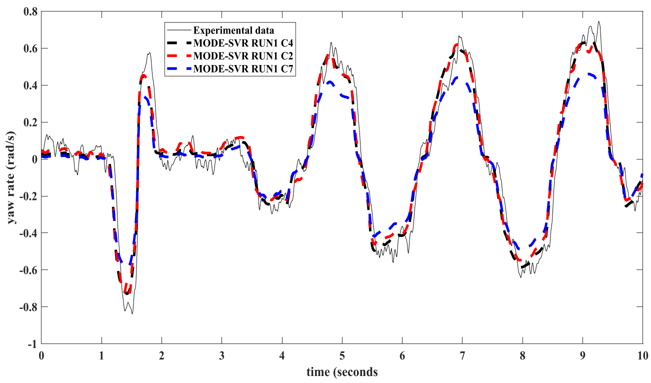

Fig. 3. Experimental data and MODE-SVR RUN1-C4, C2 and C7 predicted data with training dataset.

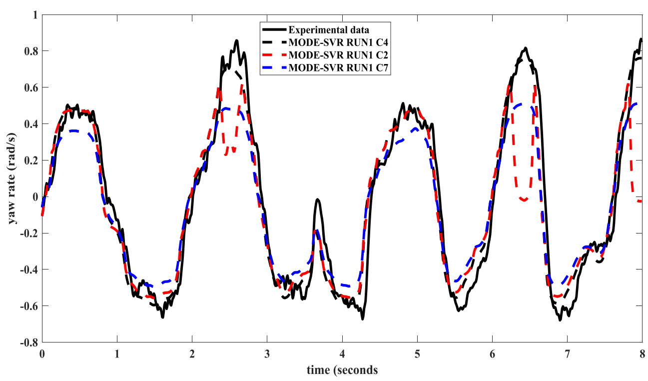

Fig. 4. Experimental data and the MODE-SVR RUN1-C4, C2 and C7 predicted data with validation data

Presented as late breaking results contribution
21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

3. RESULTS AND ANALYSIS

The performance of the proposed algorithm is benchmarked with MATLAB 2019b built-in SVR algorithm (Matlab, 2020). Candidate, C4 of RUN1 is selected for this benchmark purpose. Fig. 5 shows the MATLAB-SVR optimization process performance. The comparison of both MODE-SVR and MATLAB-SVR with experimental data is shown in Fig. 6 and Fig. 7 for training and validation dataset, respectively. Table 5 summarizes the performance parameters in terms of model complexity and prediction accuracy. First, in terms of the model complexity represented by the number of support vector (nsv), the proposed algorithm yielded lower complex model up to 30% reduction in model nsv. The performance is evaluated in terms of mean square error (mse). Though, both methods have similar prediction error on training dataset, the proposed hybrid algorithm outperformed the MATLAB-SVR on validation dataset which is an indication of better generalization. In general, without compromising the prediction accuracy, the hybrid MODE-SVR yielded simpler and more effective model for practical application

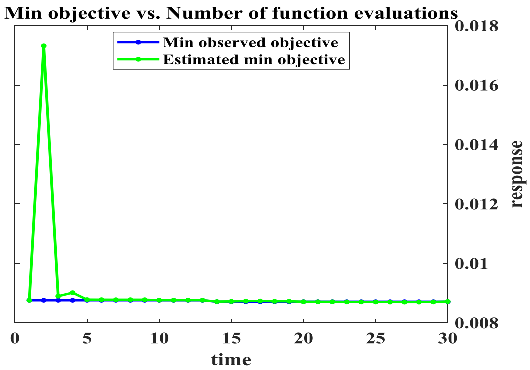

Fig. 5. MATLAB SVR Optimization process performance

Table 5. Performance Comparison of MODE-SVR and built-in MATLAB SVR

Technique	nsv	Prediction error (mse)	
		Training	Validation
MODE-SVR	682	0.0088	0.0121
Matlab-SVR	995	0.00880	0.136

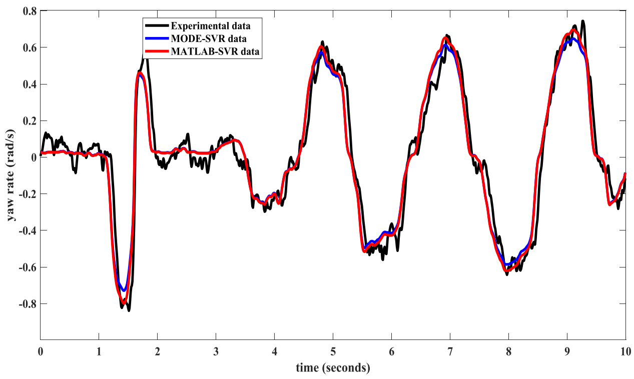

Fig. 6. Comparison of MODE-SVR and built-in MATLAB-SVR with training dataset

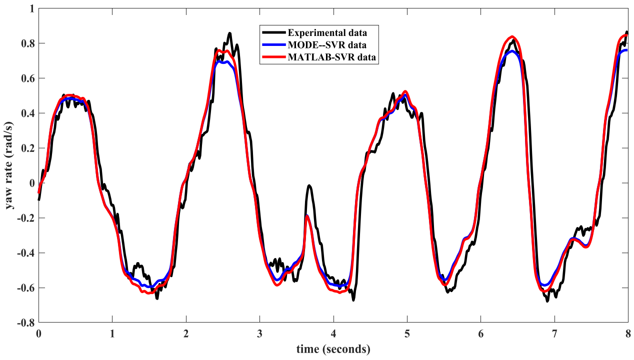

Fig. 7. Comparison of MODE-SVR and built-in MATLAB-SVR with validation dataset

4. CONCLUSIONS

The proposed hybrid MODE-SVR has demonstrated its effectiveness in tuning optimal SVR parameters in modeling of helicopter yaw dynamics using real-time flight data. The technique has been evaluated and compared with the typical SVR, MATLAB-SVR, and has demonstrated that it reduced the model complexity by 30% (represented by the number of support vector) and the increase the prediction accuracy by 10%.

REFERENCES

Canu S., Grangyalet, Y., Guigue, V., and A. Rakotomamonjy (2005) SVM and Kernel Methods Matlab Toolbox. *Perception Systemes et Information*, INSA de Rouen, Rouen, France.

Chu H., Wei J., Li T., and Jia K. (2016). Application of Support Vector Regression dor Mid- and Long-Term Runoff Forecasting In "Yellow River Headwater" Region. *Procedia Engineering* 154, pp. 1251 – 1257.

Rustam Z., Kintandani P. (2019) Application of Support Vector Regression in Indonesian Stock Price Prediction with Feature Selection Using Particle Swarm Optimisation. *Modelling and Simulation in Engineering*. Article ID 8962717.

Tijani I.B. and Akmeliawati, R. (2012a). Support vector regression based friction modeling and compensation in motion control system. *Eng. Appl. of Artif. Intell.*, vol. 25, issue 5, Elsevier.

Tijani I.B., Akmeliawati, R., Legowo, A., and Tun Muzamil L.A. (2012b). Matlab-based low-cost autopilot for UAV helicopter deployment. *Proc. 8th International Conference on Intelligent Unmanned Systems (ICIUS-2012),* SIM university, Singapore.

Tijani I.B. (2012c). *Flight control system with MODE based H-infinity for small scale autonomous helicopter*. PhD thesis, IIUM, Malaysia.

Smola A.J. and Scholkopf, B. (2001). *A tutorial on Support vector Regression*, NeuroCOLT Technical Report NC-TR-98-030.

Sch¨olkopf, B. and Smola A. (2002). *Learning with Kernels*, MIT Press, Cambridge MA.

Vapnik V. (1995). *The Nature of Statistical Learning Theory*, Springer, New York.