Neural Net Model Predictive Controller for Adaptive Active Vibration Suppression of an Unknown System

Mateusz Heesch and Ziemowit Dworakowski(✉)

AGH University of Science and Technology, Kraków, Poland
zdw@agh.edu.pl.com

Abstract. Active vibration suppression is a well explored area when it comes to simple problems, however as the problem complexity grows to a time variant system, the amount of researched solutions drops by a large margin, which is further increased with the added requirement of very limited knowledge about the controlled system. These conditions make the problem significantly more complicated, often rendering classic approaches suboptimal or unusable, requiring a more intelligent approach - such as utilizing soft computing. This work proposes a Artificial Neural Network (ANN) Model Predictive Control (MPC) scheme, inspired by horizon techniques which are used for MPC. The proposed approach aims to solve the problem of active vibration control of an unknown and largely unobservable time variant system, while attempting to keep the controller fast by introducing several methods of reducing the amount of calculations inside the control loop - which with proper tuning have no negative impact on the controller's performance. The proposed approach outperforms the multi-input Proportional-Derivative (PD) controller preoptimized using a genetic algorithm.

Keywords: Neural network · Adaptive control · Unknown system · System identification · Vibration suppression

1 Introduction

1.1 Vibration Suppression

Vibrations have always been a problem for both structures and machines. They hamper the usability of the object by reducing user comfort, negatively impacting precision of machines, causing quicker wear or even putting people in direct danger. While taking them into account during design phase of the object is often enough, in some cases they need to be further reduced or eliminated altogether. In consequence, the necessity of reducing or eliminating vibrations [12] inevitably led to the development of vibration suppression technologies [9].

The work presented in this paper was supported by the National Science Centre in Poland under the research project no. 2016/21/D/ST8/01678.

L. Rutkowski et al. (Eds.): ICAISC 2019, LNAI 11508, pp. 102–112, 2019.
https://doi.org/10.1007/978-3-030-20912-4_10

There is a multitude of ways to address vibration suppression, the most common ones being stiffening, isolation and damping. In addition, some semi-active (or semi-passive) methods have also been developed - e.g. magnetorheological (MR) fluid damper, with adjustable viscosity of oil [14]. However, this type of a solution is still inherently passive, in opposition to active vibration cancelling, in which an actuator is effectively adding energy into the system to counteract the vibrations, due to which an inadequately controlled active suppression may destabilize the system - but a well controlled one offers higher performance than passive suppression [11].

It is hardly a new approach [4,7], but it is heavily limited in its applications due to several factors, amongst them unknown and changing system dynamics, which in many cases are also complex - e.g. a bridge with constantly shifting mass distribution and multiple excitation points as cars drive through it. Naturally, controlling a system with close to no knowledge of it is a non-trivial task.

1.2 Soft-Computing-Based Adaptive Control Systems

There are two major approaches to design of control algorithms for time-varying systems: robust approach in which the algorithm should work reasonably well under all circumstances possible and adaptive approach in which the control algorithm is able to adjust itself to the changing behavior of the object. In general a robust approach provides lower efficiency and is possible to use when changes in the system are relatively small [16]. For that reason, when either the system changes significantly or the precise control is required, adaptive algorithms are preferred. A schematic view of the adaptive control can be seen in Fig. 1: a regular control system with the addition of adaptation loop that gauges the system performance and adjusts it if necessary [5]. In cases when there is little to no knowledge regarding the system dynamics, a soft-computing approaches can be used to adapt to system changes on-the-fly. While there are numerous advances in utilizing soft computing in many engineering branches for control - including model identification [3] and vibration suppression [9] - the field is far from saturated as these solutions often concern a very limited area of applications, leaving many topics unexplored, or even untouched. Few examples of Artificial-neural-network-based applications of vibration reduction include manipulators [17], buildings [13], beams [15] or spacecrafts [6] but in most of these examples the system is either partially known, assumed to be stationary or is subjected to other artificial constraints that limit the possibility of applying particular control scheme to other cases. In particular, lack of classic solutions to adaptive vibration control did not trigger development of efficient soft-computing ones, even though these methods are naturally suited for such problems.

1.3 Contribution and Organization of the Article

The article introduces a novel adaptive control algorithm based on the Artificial Neural Network. The work uses an approach known as a Neural Net Model Predictive Control (MPC) scheme, inspired by horizon techniques which are used for MPC. The method is designed to work for systems with unknown dynamics with large variability of system parameters. Several ways of computational time reduction were proposed as well.

The article is organized as follows: Sect. 1 provides a brief introduction to the subject; Sect. 2 introduces the adaptive control algorithm, Sect. 3 describes the problem in which the algorithm is to be applied, Sect. 4 provides the results of the numerical experiment, finally, Sect. 5 summarizes and concludes the article.

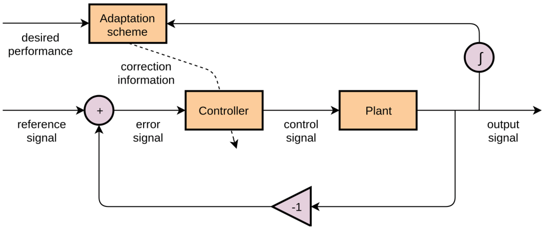

Fig. 1. Basic schematic of an adaptive controller consisting of distinct control and adaptation loops

2 The Method

2.1 Problem Assumptions

While the algorithm wasn't designed for one specific problem, some assumptions regarding the system and available knowledge about it had to be made - as these factors can severely limit viable approaches, and in consequence render some types of algorithm unfeasible. The assumptions for the controlled system are as follows:

- the system is a Multi-Degree-of-Freedom (MDOF) system with at least 3 degrees of freedom
- no knowledge of the system parameters or degrees of freedom is available
- only two points are observable - the object of suppression, and the point of application of the actuator
- these two points are separate, no information about their relation is available
- the system is subject to an unknown external excitation that causes vibrations, applied at a third point
- the system may change over time

2.2 Algorithm Design

The control problem at hand is a complex one, owing it's difficulty to three major factors:

1. The system is completely unknown
2. The system is changing over time
3. The actuator force is not applied directly on the suppression target

In consequence the algorithm design process was centered around addressing each of these issues.

The resulting control algorithm is a variation of neural net based model predictive control - a control approach that makes use of the model of the system to simulate the system response over a set time horizon, and then provide a control signal which best realizes the control goal over this horizon. In other words: the algorithm predicts damped system response for variety of possible inputs and then selects the input which minimizes the response over selected time period. In order to do that the algorithm needs to learn the system so the prediction of responses to particular inputs would be possible.

The usage of such a controller is normally split into 2 phases - model identification, where the neural net is trained using data from the system to reflect its behaviour, and then predictive control, where at each timestep the optimizer - neural net loop - simulates the possible future outcomes and returns optimal control signal.

This type of controller solves issues 1 and 3, as identifying the system via neural network requires no prior knowledge of the system, and giving the net proper information on outputs (such as state of both suppressed and actuated objects) should lead to accurate modelling of the transmission path between these 2 points. Issue no. 2 is then solved by the addition of adaptation loop within which the neural net is periodically retrained.

2.3 Algorithm Schematic

The proposed algorithm is composed of 2 major elements - controller and adaptation control - as seen in Fig. 2 - the former of which provides the input values for the controlled system, while the latter handles the controller's adaptation as well as online-learning capabilities. The schematic does not include reference signal since for vibration control it is simply a static 0 - however if this algorithm was to be applied to a problem where supplying reference signal is necessary, it would be passed into the selection algorithm block inside the controller.

2.4 Controller

The controller itself is built around a neural net, which aims to be a representation of the controlled system. The control algorithm is based on a streamlined predictive control, bearing similarities to fixed and receding horizon optimizations [1]. In both of these, several time steps for the systems output are estimated sequentially, and a control strategy is picked to best realize control goal over this horizon - the difference between these two is in applying this strategy. For fixed horizon, the strategy is realized for however many time-steps the horizon was set for, while in receding horizon only the first time step input is used. If the

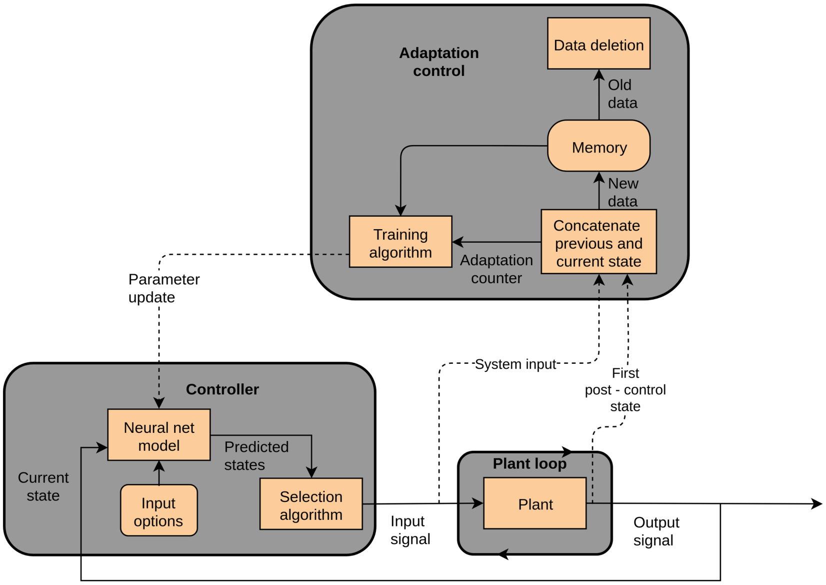

Fig. 2. Schematic of the proposed predictive control algorithm depicting the Controller, responsible for supplying Plant loop with proper control inputs, as well as Adaptation Control which harvests data from Plant loop in order to perform parameter updates on Controller's neural network core. For suppression task the reference signal is static 0

model used for predicting future states was a perfect reflection of the physical model, both of these methods would result in the same control signal, with receding horizon controller requiring more calculations (as they are necessary in each step). However since the identified models are generally not perfect, the extra calculations pay off in higher confidence of the predictions, minimizing the effect of error stacking in recurrent predictions.

The proposed algorithm goes in a different direction to avoid error stacking, as well as reduce the necessary calculations. Instead of performing series of recurrent predictions, the neural net predicts the outcomes for a single timestep, and the optimal input for this timestep is then used as control signal, as seen in Fig. 3.

To further improve performance, a naive approach of control signal persistence (CSP) is proposed - the control signal value which was deemed optimal for the single timestep persists for multiple ones, eliminating the need to recalculate it at every step. The rationale behind this approach is that for a sufficiently small persistence window the state of the controlled system does not undergo a significant change. Logically, if the system state does not change in a big way, neither does the optimal control signal. The calculation amount reduction is given by Eq. 1.

$$1 - \frac{1}{window\_size} \tag{1}$$

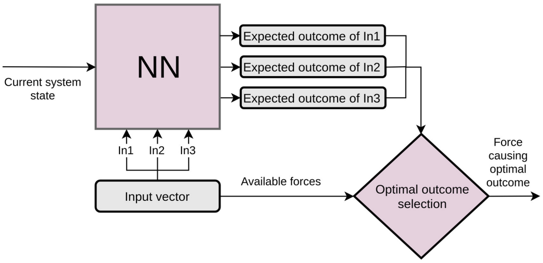

Fig. 3. Schematic of the inner workings of the neural Controller, which predicts outcomes of pre-defined available input forces, and proceeds to select the optimal outcome.

An alternative to CSP was also explored, where instead of predicting 1 step ahead, and setting the control signal for x steps, a single (non-iterative) prediction would be made for x steps ahead, to answer the question of finding the optimal static control signal for these x steps. However this approach was discarded since the controllers simply failed to reach satisfying level of suppression. The assumed reason behind this is that predicting x steps ahead is a considerably more difficult problem to model and - if possible at all - would require much larger neural nets, and in consequence much more training time and slower operation. This stands in opposition to the reason behind implementing such method in the first place - which is to speed the controller up.

In addition to that, the controller is also constrained in what control signals it can use - a quantized vector of possible inputs is specified to avoid free (and time consuming) iterative exploration of options.

A single cycle for the controller is as follows:

1. The controller receives current state of the system.
2. Current state + quantized input vector is fed into the neural net.
3. The net returns predicted system states for each of the proposed inputs, at the end of control time window.
4. The selection algorithm picks the control signal for the time window based on the estimated states.
5. The static (over the duration of the control time window) control signal is passed to the model.

While the computational weight of the control algorithm is lower than horizon-based model predictive control methods, it does require selection of CSP window length, as well as the quantized input vector. Selection of the latter is straightforward, as the actuator output range is generally limited, picking several values from this range is a simple precision vs speed tradeoff. While reducing the continuous range of inputs to a discrete list may seem like a bad idea at first glance, the importance of having a continuous range needs to be reconsidered - e.g. in many cases the difference between applying the force of 0.6N, and 0.603N is very likely to be negligible. Needless to say, if the controlled system is a highly sensitive, the input vector quantization should reflect that.

As for the CSP window length selection, it is dependent on both the controlled system as well as the actuator used, with window size increasing controller speed and theoretically decreasing its effectiveness due to its approximation implications:

1. the window needs to be small enough to still reflect relatively small time steps, not making large leaps and omitting crucial parts of the signal - e.g., if the oscillation frequency is 50 Hz, then picking a window of size 0.05 s will render the controller useless.
2. the window needs to be small enough to fit controller speed requirements.
3. lastly, the window needs to be long enough to allow for actuator reaction.

2.5 Adaptation Control

To deploy the controller, firstly the net is trained on some measured data, preferably one previously controlled with a different method, so that some effects of the input to the system are known. The controller is then operational - however depending on the quality of data provided for initial training, as well as changes that happen within the model - the control may very well be suboptimal, which is why another module is necessary. The adaptation control module of the proposed algorithm is not only devoted to the adaptation of the controller in case of change in the model, but is also responsible for its online learning capabilities.

After each control step it receives the data about it - what inputs were used, and what effects they had. These are then retained in the system's simple "memory" model with a set fixed size, where old entries are replaced with new ones. Periodically, the neural net will undergo a brief re-training process, fitting the neural net to the data stored in its memory, essentially "forgetting" how the model acted in past, and reinforcing knowledge on how it is behaving currently. This process leads to general self-improvement of the model during its operation, including adaptation.

2.6 PD Controller

PD-based controllers are often used in vibration suppression tasks [8,10]. Here, a simple multi-input PD control scheme was included [2] as a reference point for an ANN-based algorithm. The PD method operates based on a following equation:

$$F(t) = \sum P_1, i * y_i(t) + P_2 i * (y_i(t-1) - y_i(t)) \tag{2}$$

where P_i and $P_{2,i}$ refer to adjustable parameters, i refer to the observable mass under investigation, $y_i(t)$ is a displacement of i-th mass at time t while $y_i(t-1)$ is a displacement of i-th mass in time $t-1$. $P_{1,i}$ thus refer to adjustable proportional parameters of a controller while $P_{2,i}$ refer to adjustable derivative parameters of a controller. Depending on the parameters' values, the controller can either be treated as a robust or adaptive one. In this scenario it is pre-trained for a particular system state and then compared with adaptive ANN solution.

3 Problem Definition and Simulation Setup

While the system's parameter change over time is not of concern, the system on which the tests are to be carried out should comply with the other 2 main issues of the control problem that the controller was designed for - the knowledge of the system should be severely limited, and the system itself ought to be a complex MDOF one, with separate points for external disturbance, active component actuation and suppression target. The problem is an extension of the 3-degree-of-freedom unknown system for which the evolutionary-optimized PD controller was proposed before [2]. The resulting system can be seen in Fig. 4.

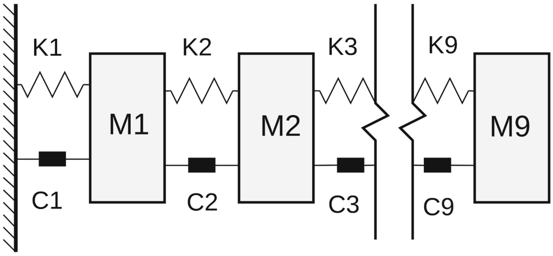

Fig. 4. Schematic of the 9-DOF mechanical system used for suppression simulations

The simulations themselves had following parameters:

- Total time - 15 s
- Simulation timestep - 0.001 s
- External disturbance of a 0.005 s long impulse starting at 0.1 s with the magnitude of 100N applied to m_1
- Suppression target set - m_3
- Actively actuated mass - m_5

The controller was tested in a volatile environment, with random changes to each parameter at every simulation cycle, leaving no time to properly adapt. The system's parameters were selected in following way:

- All masses were initialized at 5 kg, changed by a random number (uniform distribution) between −1 and 1 at every simulation cycle. Masses were kept within 0.1–10 boundaries).
- All damping ratios were initialized at 2.5, changed by a random number (uniform distribution) between −0.2 and 0.2 at every simulation cycle. Damping ratios were kept within 0.01–5 boundaries).
- Spring constants linearly decayed by 20% over the simulation length.

The NN predictive control was initialized with pre-trained net based on PD control of a time-invariant version of this system, reducing the magnitude of weight changes necessary properly fit the constantly changing system during the first few simulation cycles as the net starts at a point where it is familiar with the dynamics of the system in broad sense. The predictive control performance is compared to static multi-input PD control tuned with the use of a simple genetic algorithm: the system state at the beginning of simulation was copied for the genetic algorithm so the adjustable parameters of the PD algorithm were set to optimum for particular state of the system. Although this step would not be possible in practical scenario (usually there is no possibility to freeze the system so the PD controller would be able to learn it properly), this step was performed to provide a worst-case-scenario for the ANN-based algorithm to compete with.

4 Results

Since the changes in the system were quite significant, it comes as no surprise that the static multi-input PD controller destabilized the system on multiple occasions. Results of the simulation are plotted in Fig. 5. Logarithmic scale is used so the results of all the three algorithms could be compared in one plot. It is worth noting that simulation starts with a PD approach being comparable with the NN-based one. That is because of the pre-optimization performed before the experiment. As the experiment progressed, the PD control started to obtain significantly worse results - often destabilizing the system. The fact that all the three solutions seem to be correlated is due to the fact, that the parameters of the system contribute significantly to the vibration suppression. In some configurations (e.g. - higher damping ratios) the algorithms have "less work to do" - therefore they are compared against each other and against the uncontrolled system behavior instead of absolute RMS values.

Various metrics allowing for comparison of all the three solutions are given in Table 1. The superiority of the proposed approach over the pre-optimized multi-input PD one is well pronounced in all the metrics used.

Table 1. Results of the experiment averaged for the whole simulation.

Metric	No control	Multi-input PD	Predictive NN
Total RMS worse than uncontrolled [%]	n/a	36.25	0.65
RMS relative to the uncontrolled [Fraction]	n/a	5.58	0.81
Destabilized system [%]	0	31.90	0.05
Average RMS [m]	0.0286	0.2366	0.0227

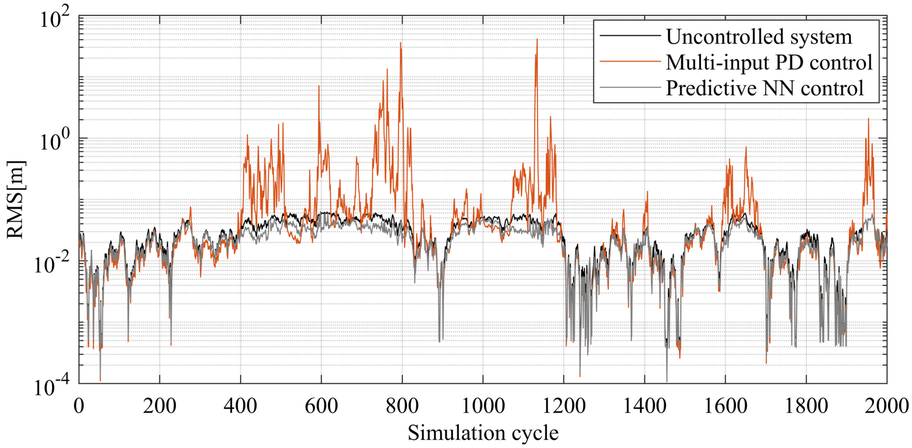

Fig. 5. Total RMS results of the adaptability test. Logarithmic scale was used for y axis for the purpose of better readability.

5 Conclusions

The results of the test bring several important takeaways:

- The proposed algorithm can very quickly fit to an unknown model which is constantly changing.
- The proposed algorithm struggles to achieve a very good performance - at multiple points its control results were worse than those of the multiple-input PD controller.
- While generally not providing optimal control per se, the algorithm was very capable at making adaptations that didn't result in the controller amplifying the systems vibrations, with only 0.65% of the cycles ending up in that state, as opposed to the fixed controller, for which this happened in 36.25% of the cases.
- Even though the stability analysis was not performed in this case, obtained results clearly show the proposed approach is generally stable, and safe enough for practical applications.

Acknowledgement. The work presented in this paper was supported by the National Science Centre in Poland under the research project no. 2016/21/D/ST8/01678.

References

1. Camacho, E., Bordons, C., Alba, C.: Model Predictive Control. Advanced Textbooks in Control and Signal Processing. Springer, London (2004). https://books.google.pl/books?id=Sc1H3f3E8CQC

2. Dworakowski, Z., Mendrok, K.: Indirect structural change detection based on control algorithm's performance. In: Proceedings of European Workshop on Structural Health Monitoring 2018, pp. 1–10 (2018). https://www.ndt.net/article/ewshm2018/papers/0024-Dworakowski.pdf
3. Hossain, M.S., et al.: Artificial neural networks for vibrationbased inverse parametric identifications: a review. Appl. Soft Comput. J. **52**, 203–219 (2016). https://doi.org/10.1016/j.asoc.2016.12.014. http://linkinghub.elsevier.com/retrieve/pii/S1568494616306329
4. Landau, I., Lozano, R., M'Saad, M., Karimi, A.: Adaptive Control. Springer, London (2011). https://doi.org/10.1007/978-0-85729-664-1
5. Landau, I.D., Lozano, R., M'Saad, M.: Adaptive Control. Springer, Heidelberg (1998). https://doi.org/10.1007/978-0-85729-343-5
6. Leeghim, H., Kim, D.: Adaptive neural control of spacecraft using controlmoment gyros. Adv. Space Res. **55**(5), 1382–1393 (2015). https://doi.org/10.1016/j.asr.2014.06.038
7. Mal'tsev, A.A., Maslennikov, R., Khoryaev, A., Cherepennikov, V.: Adaptive active noise and vibration control. Acoust. Phys. **51**(2), 195–208 (2005)
8. Milovanović, M.B., Antić, D.S., Milojković, M.T., Nikolić, S.S., Perić, S.L., Spasić, M.D.: Adaptive PID control based on orthogonal endocrine neural networks. Neural Netw. **84**, 80–90 (2016). https://doi.org/10.1016/j.neunet.2016.08.012
9. Muhammad, B.B., Wan, M., Feng, J., Zhang, W.H.: Dynamic damping of machiningvibration: a review. Int. J. Adv. ManufacturingTechnology (2016). https://doi.org/10.1007/s00170-016-9862-z
10. Pan, Y., Er, M.J., Sun, T., Xu, B., Yu, H.: Adaptive fuzzy PD control with stable H tracking guarantee. Neurocomputing (2016). https://doi.org/10.1016/j.neucom.2016.08.091
11. Preumont, A.: Vibration Control of Active Structures: An Introduction. Solid Mechanics and Its Applications. Springer, Netherlands (2011). https://doi.org/10.1007/978-94-007-2033-6. https://books.google.pl/books?id=MUQUQyB4bEUC
12. Rao, S.S., Fah, Y.F.: Mechanical Vibrations; 5th edn. in SI units. Prentice Hall, Singapore (2011). https://cds.cern.ch/record/1398617
13. Subasri, R., Suresh, S., Natarajan, A.M.: Discrete direct adaptive ELM controller for active vibration control of nonlinear base isolation buildings. Neurocomputing **129**, 246–256 (2014). https://doi.org/10.1016/j.neucom.2013.09.035
14. Terasawa, T., Sano, A.: Fully adaptive semi-active control of vibration isolation by Mr Damper. IFAC **38** (2002). https://doi.org/10.3182/20050703-6-CZ-1902.00254
15. Valoor, M., Chandrashekhara, K., Agarwal, S.: Self-adaptive vibration controlof smart composite beams using recurrent neural architecture. Int. J. Solids Struct. **38**(44–45), 7857–7874 (2001). https://doi.org/10.1016/S0020-7683(01)00125-1. http://linkinghub.elsevier.com/retrieve/pii/S0020768301001251
16. Zak, S.H.: Systems and Control. Oxford University Press, Oxford (2002)
17. Zhao, Z.l., Qiu, Z.c., Zhang, X.m., Han, J.d.: Vibration control of a pneumatic driven piezoelectric flexible manipulator using self-organizing map based multiple models. Mech. Syst. Signal Process. **70–71**, 345–372 (2016). https://doi.org/10.1016/j.ymssp.2015.09.041, http://linkinghub.elsevier.com/retrieve/pii/S0888327015004537, www.sciencedirect.com/science/article/pii/S0888327015004537