DINAME 2019 - Proceedings of the XV International Symposium on Dynamic Problems of Mechanics,
M.A. Savi, T.G. Ritto and W.M. Bessa (Editors), ABCM, Buzios, RJ, Brazil, March 10th to 15th, 2019

Robust Design Support using Fuzzy Simulation of Uncertain Dynamic System: A Self-Balancing Robot Case Study

Ugo Chouinard [1], Sofiane Achiche [1], Ilmar Santos[2], and Luc Baron [1]

[1] Polytechnique Montréal, 2900 blvd Édouard-Montpetit, Montréal, Qc, H3T 1J4, Canada

[2] Mechanical Department, Denmark Technical University, Nils Koppel Allé, 404, Kongens Lyngby, 2800 Denmark

Abstract: This paper proposes a fuzzy simulation method for dynamic systems having uncertain physical parameters. The paper first introduces a fuzzy arithmetic that can be used to have a non-diverging simulation, which would not be the case if the standard arithmetic is used. Then the fuzzy simulation is tested on a self-balancing robot system. The simulation method is shown to result in a good approximation of the dynamical response, but by being more computationally efficient than Monte-Carlo simulations. The paper advances potential uses of the fuzzy simulation such as being used in an optimization loop during the robust design of a mechatronic system.

Keywords: Fuzzy Simulation, Robotics, Robustness

INTRODUCTION

Uncertainty plays a large role in the modelling of physical systems. Indeed, it is usually impossible to obtain the exact value of parameters. Moreover, simplifications made to the modelling also accounts for the uncertainty in the response of the system. It is thus of utmost importance to consider uncertainties when designing dynamic systems as their effect could lead to decreased performance. Taking into consideration those uncertainties results in carrying out a robust design methodology. A design is said to be robust if it limits the variation in the response without necessarily removing the uncertainty in the design variables (Park et al., 2006). A robust design methodology can thus be seen as an optimization which tries to minimize the variance of the performance function. It is also possible to consider optimality during a robust methodology if both the mean and variance of the performance function are considered in a multi-objective optimization (Ghanmi et al., 2011).

One important aspect that is considered in this work is the use of simulations as a means of obtaining the performance function of the system instead of an analytical solution. Simulation is often more convenient in a highly complex system and allows to obtain information that is not necessarily available analytically, such as the total energy consumption of a system over a given period. In a previous work by the authors, it was suggested to use a double-loop Monte Carlo optimization (a Monte-Carlo simulation embedded in an optimization loop) for the robust design of a quadcopter drone (Coulombe et al., 2017). The Monte-Carlo simulation (MCS) was used on the dynamical simulation of the drone to obtain the response distribution with respect to uncertain physical parameters. It was shown that using a double loop Monte-Carlo optimization along the dynamical simulation could effectively be used to reduce the energy consumption and improve the robustness of the drone without removing the variance on the design variables. However, since the optimization used a MCS to generate the required data for statistical analysis, the process was computationally expensive even for a single-objective optimization composed of the mean and variance of the performance function.

To deal with the drawbacks of using MCS as a means of obtaining the uncertain performance of the system, especially while using simulations to compute the performance function, we suggest supporting the early design process using fuzzy numbers. The fuzzy numbers are used to represent the uncertainty within the system's design variables. It was shown that the use of fuzzy numbers to treat uncertainty in robust design methodology could be effective in the design of structures (Marano and Quaranta, 2008; Möller et al., 2000) or shock absorbers (Silva et al., 2016). However, (Möller et al., 2000; Silva et al., 2016) make use of alpha-level optimization, which although reported to be faster than when using Monte Carlo simulation, remains computationally heavy. Indeed, alpha-level optimization discretize fuzzy numbers on their membership function and requires calculating the extremum of the performance function based on the combination of the input variable's own extremum, at each discretization level. Moreover, alpha-level optimization, or the transformation method (Hanss, 2002), use the extension principle and thus usually evaluate one function with all the variables simultaneously. Using these methods in complex design problems would usually results in a large search space (Hanss, 2005).

Furthermore, traditional methods using fuzzy numbers will usually result in diverging simulation (i.e. time increasing uncertainty) due to the fuzzy numbers properties. Diverging fuzzy simulations, which bounds often quickly tends towards infinity, cannot be used in optimization as there would not be any possibility of calculating the variance of the performance function. Hence, in this work we explore the dynamical simulation aspect using a constrained fuzzy arithmetic, which should avoid obtaining diverging results. This work thus lay down the basis to carry out more time-efficient simulation of uncertain mechatronic systems using constrained fuzzy arithmetic instead of running multiple iterations through MCS or using alpha-level optimization, thus saving simulation time and allowing more time for the early design process.

At first, the mathematical foundations, used in this paper, for computing fuzzy number arithmetic are presented. Then, to showcase the efficiency of the method we use a self-balancing robot as a case study. The modeling and control of the robot is thus described, and then the results of the simulation of the uncertain model of both the fuzzy method and MCS are compared. Finally, this paper discusses on ways that the fuzzy simulation can be employed during the early design phases to robustly design mechatronic devices.

MATHEMATICAL PRELIMINARIES

Fuzzy Numbers

Fuzzy numbers are bounded fuzzy sets which also have the properties of being normal, convex, and upper semicontinous (Bede, 2013; Hanss, 2005). For instance, some of the widely used ones are the triangular and trapezoidal shaped fuzzy numbers. Fuzzy numbers can also have Gaussian membership that would allow to represent the uncertainty of a variable, which can be used in functions instead of a Monte-Carlo simulation (Hanss, 2005). Moreover, the use of the quasi-Gaussian membership function is more practical since the spread of the Gaussian fuzzy number would be theoretically infinite. Thus the membership function $\mu(x)$ of the quasi-Gaussian fuzzy number is given in Eq. (1), with c being a cut-off parameter which is suggested to be $c=3$ in (Hanss, 2005) because it would allow to capture 99% of the number membership.

$$\mu(x)=\begin{cases} 0 & \text{if} \quad x<\overline{x}-c\sigma_l \\ L\left(\exp\left(-\dfrac{(x-\overline{x})^2}{2\sigma_l^2}\right)\right) & \text{if} \quad \overline{x}-c\sigma_l \le x<\overline{x} \\ R\left(\exp\left(-\dfrac{(\overline{x}-x)^2}{2\sigma_r^2}\right)\right) & \text{if} \quad \overline{x} \le x<\overline{x}+c\sigma_r \\ 0 & \text{if} \quad \overline{x}+c\sigma_r \le x \end{cases} \tag{1}$$

Fuzzy Arithmetic

There are two approaches that are usually defined for the arithmetic operations on fuzzy numbers. The first one being the use of interval arithmetic on the $\alpha-cuts$ of the fuzzy numbers, and the second one being Zadeh's extension principle (Bector and Chandra, 2005). In this work we use the interval arithmetic approach, which is computationally efficient, but with using the principle of constrained interval instead of standard operations. The operations on fuzzy numbers as proposed by interval arithmetic on $\alpha-cuts$ have multiple drawbacks, especially if used in numerical simulation. Indeed, since simulation involves multiple operations, subsequent iterations of the simulation would necessarily increase the spread of the result to a point where it will diverge. Indeed, this would be related to the fact that for a give fuzzy number a , according to standard interval arithmetic principle $a-a\neq\{0\}$. Thus, instead of using the standard definition of fuzzy arithmetic, we will use the constrained arithmetic such as described in (Chalco-Cano et al., 2014) which then allows to have $a-a=\{0\}$, $a/a=\{1\}$. Therefore, for two given fuzzy numbers u,v having membership functions μ_u,μ_v and $\alpha-cuts$ $[u]_\alpha,[v]_\alpha,\alpha\in[0,1]$, the constrained arithmetical operations are given by Eq.(2)-(6) (Chalco-Cano et al., 2014).

Addition

$$[u]_\alpha \oplus [v]_\alpha = [u_\alpha^- + v_\alpha^-, u_\alpha^+ + v_\alpha^+] \tag{2}$$

Scalar multiplication

$$[ku]_\alpha = \begin{cases} [ku_\alpha^-, ku_\alpha^+] & \text{if} \quad k \ge 0 \\ [ku_\alpha^+, ku_\alpha^-] & \text{if} \quad k<0 \end{cases} \tag{3}$$

Multiplication

$$[u]_\alpha \otimes [v]_\alpha = \left[\min\{u_\alpha^- v_\alpha^-, u_\alpha^+ v_\alpha^+\}, \max\{u_\alpha^- v_\alpha^-, u_\alpha^+ v_\alpha^+\}\right] \tag{4}$$

Difference

$$[u]_\alpha \ominus [v]_\alpha = \left[\min\left\{u_\alpha^- - v_\alpha^-, u_\alpha^+ - v_\alpha^+\right\}, \max\left\{u_\alpha^- - v_\alpha^-, u_\alpha^+ - v_\alpha^+\right\}\right] \tag{5}$$

Division

$$[u]_\alpha \oslash [v]_\alpha = \left[\min\left\{u_\alpha^- / v_\alpha^-, u_\alpha^+ / v_\alpha^+\right\}, \max\left\{u_\alpha^- / v_\alpha^-, u_\alpha^+ / v_\alpha^+\right\}\right] \tag{6}$$

with u_α^-, u_α^+ being the lower and upper bounds of the fuzzy number's interval at a given $\alpha - cut$.

CASE STUDY: SELF-BALANCING ROBOT

A self-balancing robot is an underactuated mechatronic dynamical system which has found use in different application such as the well-known "Segway". It is an interesting case study since it is inherently unstable due to the system being an inverted pendulum. Moreover, the system makes for a good case study due to its relatively low complexity while still remaining a challenge control wise. The self-balancing robot has been widely studied in term of its dynamics and control (Chan et al., 2013; Dai et al., 2015; Ha and Yuta, 1996) and thus we shall only provide the results of the dynamic modelling and control for the paper to be self-explanatory.

System Model and Control

To demonstrate the fuzzy simulation method, we use a simplified model of the system for the simulation. Figure 1 shows the robot's physical model and parameters. In the model's simplifications, the motor's transfer function is not taken into consideration. The control is done directly on the torque provided by the motors, and not on the voltage supplied to the motors. A no-slip condition is also imposed on the wheels. Moreover, the system is linearized around the vertical equilibrium position $\theta = 0$. Therefore, the robot can be modelled by the state-space equation provided in Eq.(7)

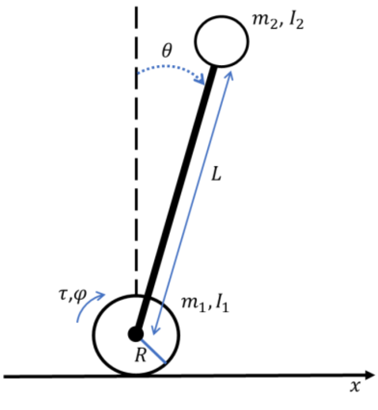
$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}u \\ \mathbf{y} &= \mathbf{C}\mathbf{x} + \mathbf{D}u \end{aligned} \tag{7}$$

where the state vector is $\mathbf{x} = \left[x, \dot{x}, \theta, \dot{\theta}\right]^T$ and the matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ are given by Eq. (8).

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \dfrac{-m_2 g P_2}{(P_2^2 - P_1 P_3)} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \dfrac{P_1 m_2 g}{(P_1 P_3 - P_2^2 R)} & 0 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ \dfrac{P_3}{(P_2^2 - P_1 P_3)} \\ 0 \\ \dfrac{P_2}{(P_1 P_3 - P_2^2 R)} \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad \mathbf{D} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{8}$$

with $P_1 = R(m_1 + m_2) + I_1 / R \quad P_2 = m_2 l \quad P_3 = I_2 + l m_2$

In this paper, the self-balancing robot is controlled using a linear quadratic regulator (LQR) controller. There is a wide body of literature concerning the LQR and the control of a self-balancing robot, or of the closely related inverted pendulum on a cart. For more information. the reader is referred for instance to (Fang, 2014).



m_1 : Wheel mass
I_1 : Wheel inertia
R : Wheel radius
ϕ : Wheel rotation
m_2 : Robot's mass
I_2 : Robot's inertia
L : Distance from wheel center to robot's center of gravity
θ : Angle of the robot from the vertical
τ : Torque provided by the motors on the wheel
x : Robot's linear displacement

Figure 1: Self-Balancing Robot Model

Fuzzy Simulation of the Robot

Simulation Set-up:

The first step in the simulation is to calculate the gains from the LQR controller using the nominal parameter values, which are provided in Tab. 1. The controller gains are found using the *LQR()* function in MATLAB. Once the gains are found, a variation of 10% is then set on the design variables of the self-balancing robot. For the fuzzy simulation, each of the design variables are instantiated as being a fuzzy number, using the uncertainty in the parameter value, and following the membership function of the gaussian fuzzy number provided in Eq. (1). As an example, we show the fuzzy controlled state matrix $\mathbf{A}-\mathbf{KB}$ in Fig. 2, where the shape of each uncertain entries can be observed. Therefore, each operation during the fuzzy simulation will be carried out using the fuzzy arithmetic presented in the previous section.

Table 1: Nominal Parameter Value of the Self-Balancing Robot

Parameter	m_1	m_2	L	R	I_1	I_2
Nominal Value	0.036 kg	0.9605 kg	0.15 m	0.0235 m	1.9881e^{-5} kg m^2	0.0013 kg m^2

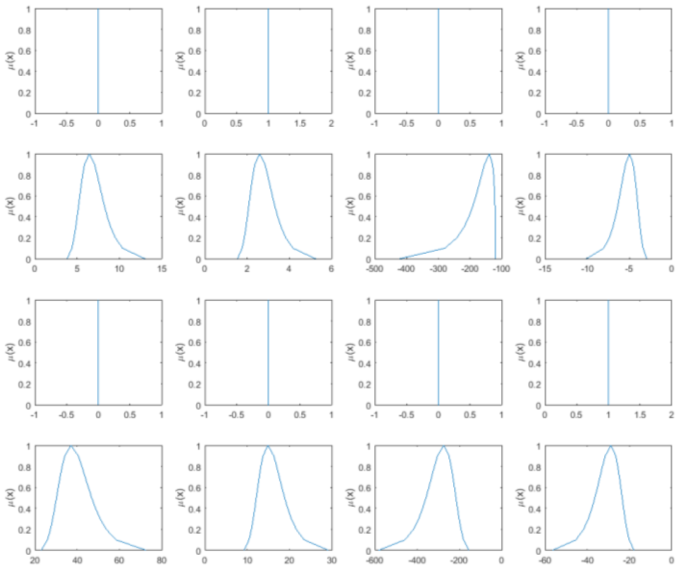


Figure 2: Fuzzy Controlled State Matrix ($\mathbf{A}-\mathbf{KB}$) Visual Representation with $\mu(x)$ being the membership of the fuzzy numbers

Simulation Results:

The result of the simulation is provided in Fig. 3 for both the Fuzzy method and MCS (with 5000 iterations). Furthermore, the simulation times are compared in Tab.2. From Fig. 3, it can be seen that the fuzzy simulation correctly calculates the nominal response of the system. The level curves around the nominal response represent the possibility of the robot having a certain position/orientation at a given time. Therefore, it should be seen as the possibility of the response being bounded at a given time, and not the actual response. From Fig. 3, it is clear that the fuzzy simulation does not provide the full bounds as it can be seen by the MCS. This is mainly a result of the constrained arithmetic operations that do not consider the full breadth of combinations. Although, the fuzzy simulation is far from being perfect, it still provides a good estimate of the potential behavior of the system in a much more computationally efficient way, such as shown in Tab.2.

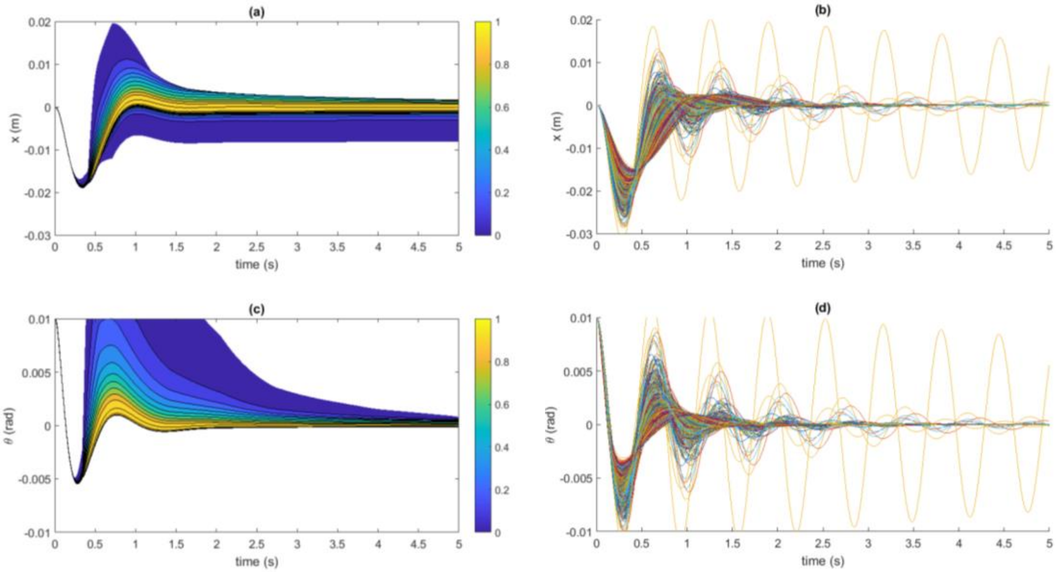


Figure 3: Simulation Result of Fuzzy Method vs MCS with the color bar ((a), (c)) representing the possibility of the system having a certain response bound

Table 2: Comparison of simulation time between Fuzzy method and Monte-Carlo

	Fuzzy Simulation	MCS (x5000)
Time (s)	0.45	3.85

POTENTIAL USE OF FUZZY SIMULATION IN DESIGN SUPPORT

As it was mentioned previously, the main goal of this paper is to lay down the basis for efficient optimization of uncertain dynamic systems during the early design stages. So far, we have introduced the required elements for the fuzzy simulation of a dynamic system, which was exemplified with the case study on the self-balancing robot. There are many ways that this type of simulation can be used in the optimization of dynamic system, especially of mechatronic devices. We present here two potential uses of the fuzzy simulation in design support: Carrying efficient robust design robust control and handling negative dependencies in mechatronic systems.

Achieving Robust Design Robust Control Efficiently

First, using fuzzy simulation, it should be possible to carry out efficient robust design robust control (RDRC) of mechatronic devices. For instance (Alyaqout et al., 2011) introduced the RDRC on a DC motor and mentioned that carrying out RDRC was computationally heavy even for a simple case such as the DC motor. For achieving RDRC, it would be required to optimize both the controller gains and system variables. Current method for RDRC are based on analytical solution (Alyaqout et al., 2011; Lu and Huang, 2013; Villarreal-Cervantes et al., 2013) which might not be achievable for highly complex systems, and thus simulation would be more suited to provide the required system performance information. However, including simulation in RDRC would usually require the use of a Monte-Carlo simulation to obtain the distribution of the performance functions, which would be computationally intensive.

An efficient RDRC could be achieved by using the fuzzy simulation in a multi-objective optimization loop, such as a genetic algorithm or particle swarm algorithm, and trying to find the set of pareto solutions that reduce uncertainty in the response. Obviously, the optimization would also require ensuring that the system is stable in the presence of uncertainties. This could be achieved by using the (fuzzy) extension of Kharitonov's theorem (Bhiwani and Patre, 2011) or by calculating the fuzzy eigenvalues (Buckley et al., 2002) of the state matrix. A potential RDRC optimization process is displayed in Fig. 4. However, no matter how the optimization is carried out, the difference in simulation time between the fuzzy and Monte-Carlo simulation should prove to be significant, thus allowing to search for a much larger design space.

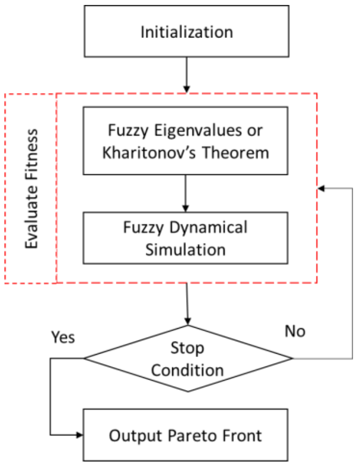


Figure 4: Robust Design Robust Control Optimization Using Fuzzy Simulation

Handling Negative Dependencies

The second way that fuzzy simulation could be used is when dealing with negative dependencies within the system. Negative dependencies are for instance the noise induced by functioning components to others, and may appear in the form of vibration, heat, or electro-magnetic field. Those negative dependencies can lead to decreased system performance. The work in (Chouinard et al., 2017) suggests a method to identify negative dependencies early during the design process by modelling the dependencies as fuzzy numbers. The information obtained by this method could be interpreted as a fuzzy noise that could be added to the simulation model. Doing so could allow to identify if the system would still be able to perform in the presence of those negative dependencies, or if changes to the physical design, control method, or filter design need to be done to make the system more robust. We show how this fuzzy noise could be introduced in the system model in Fig. 5. Once the result of dependency analysis is introduced in the model, it should be possible to carry out the simulation as described previously.

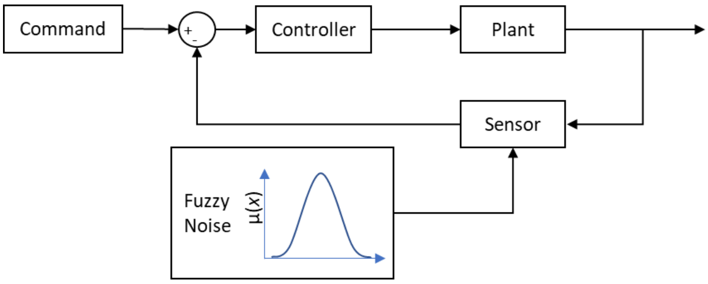


Figure 5: Introducing Fuzzy Noise in Simulation Model

CONCLUSION

This paper introduced a simulation method for uncertain systems using fuzzy numbers. The fuzzy simulation has been tested on a self-balancing robot and compared to the Monte-Carlo simulation. It is shown that the fuzzy simulation using constrained fuzzy arithmetic provides a good estimate of the behavior of the robot under uncertainty. The fuzzy simulation is also shown to be more computationally efficient than the Monte-Carlo simulation. The fuzzy method is then expected to be a good means of robustly optimising a dynamic system subject to uncertainties. Indeed, using the fuzzy simulation and trying to reduce the mean and variance of the response during an optimization process should lead to also reduce the mean and variance of the response from the real system. The paper then advances two potential use of the fuzzy simulation method to robustly optimize mechatronic systems: carrying out robust design robust control and dealing with negative dependencies in the system. The optimization of the self-balancing robot using the fuzzy method will be tested in future work.

ACKNOWLEDGMENTS

This research was funded by the Natural Science and Engineering Research Council of Canada

REFERENCES

Alyaqout, S.F., Papalambros, P.Y. and Ulsoy, A.G. (2011), "Combined Robust Design and Robust Control of an Electric DC Motor", *IEEE/ASME Transactions on Mechatronics*, Vol. 16 No. 3, pp. 574–582.

Bector, C.R. and Chandra, S. (2005), *Fuzzy Mathematical Programming and Fuzzy Matrix Games*, Springer-Verlag Berlin Heidelberg, Berlin, Heidelberg, available at: http://0-dx.doi.org.fama.us.es/10.1007/3-540-32371-6 (accessed 20 December 2017).

Bede, B. (2013), *Mathematics of Fuzzy Sets and Fuzzy Logic*, Springer, Heidelberg ; New York.

Bhiwani, R.J. and Patre, B.M. (2011), “Stability analysis of fuzzy parametric uncertain systems”, *ISA Transactions*, Vol. 50 No. 4, pp. 538–547.

Buckley, J.J., Eslami, E. and Feuring, T. (2002), “Fuzzy Eigenvalues”, *Fuzzy Mathematics in Economics and Engineering*, Vol. 91, Physica-Verlag HD, Heidelberg, pp. 199–228.

Chalco-Cano, Y., Lodwick, W.A. and Bede, B. (2014), “Single level constraint interval arithmetic”, *Fuzzy Sets and Systems*, Vol. 257, pp. 146–168.

Chan, R.P.M., Stol, K.A. and Halkyard, C.R. (2013), “Review of modelling and control of two-wheeled robots”, *Annual Reviews in Control*, Vol. 37 No. 1, pp. 89–103.

Chouinard, U., Achiche, S., Leblond-Ménard, C. and Baron, L. (2017), “Assessment of dependencies in mechatronics conceptual design of a quadcopter drone using linguistic fuzzy variables”, Vol. 4, presented at the 21st International Conference on Engineering Design, Vancouver, Canada, pp. 031–040.

Coulombe, C., Gamache, J.-F., Mohebbi, A., Chouinard, U. and Achiche, S. (2017), “Applying robust design methodology to a quadrotor drone”, Vol. 4, presented at the 21st International Conference on Engineering Design, Vancouver, Canada, pp. 395–404.

Dai, F., Gao, X., Jiang, S., Guo, W. and Liu, Y. (2015), “A two-wheeled inverted pendulum robot with friction compensation”, *Mechatronics*, Vol. 30, pp. 116–125.

Fang, J. (2014), “The LQR Controller Design of Two-Wheeled Self-Balancing Robot Based on the Particle Swarm Optimization Algorithm”, *Mathematical Problems in Engineering*, Vol. 2014, pp. 1–6.

Ghanmi, S., Guedri, M., Bouazizi, M.-L. and Bouhaddi, N. (2011), “Robust multi-objective and multi-level optimization of complex mechanical structures”, *Mechanical Systems and Signal Processing*, Vol. 25 No. 7, pp. 2444–2461.

Ha, Y.-S. and Yuta, S. (1996), “Trajectory tracking control for navigation of the inverse pendulum type self-contained mobile robot”, *Robotics and Autonomous Systems*, Vol. 17 No. 1–2, pp. 65–80.

Hanss, M. (2002), “The transformation method for the simulation and analysis of systems with uncertain parameters”, *Fuzzy Sets and Systems*, Vol. 130 No. 3, pp. 277–289.

Hanss, M. (2005), *Applied Fuzzy Arithmetic: An Introduction with Engineering Applications*, Springer-Verlag, Berlin ; New York.

Lu, X.J. and Huang, M.H. (2013), “Nonlinear-Measurement-Based Integrated Robust Design and Control for Manufacturing System”, *IEEE Transactions on Industrial Electronics*, Vol. 60 No. 7, pp. 2711–2720.

Marano, G.C. and Quaranta, G. (2008), “Fuzzy-based robust structural optimization”, *International Journal of Solids and Structures*, Vol. 45 No. 11–12, pp. 3544–3557.

Möller, B., Graf, W. and Beer, M. (2000), “Fuzzy structural analysis using α-level optimization”, *Computational Mechanics*, Vol. 26 No. 6, pp. 547–565.

Park, G.-J., Lee, T.-H., Lee, K.H. and Hwang, K.-H. (2006), “Robust Design: An Overview”, *AIAA Journal*, Vol. 44 No. 1, pp. 181–191.

Silva, A.D.G., Cavalini Jr, A.A. and Steffen Jr, V. (2016), “Fuzzy Robust Design of Dynamic Vibration Absorbers”, *Shock and Vibration*, Vol. 2016, pp. 1–10.

Villarreal-Cervantes, M.G., Cruz-Villar, C.A., Alvarez-Gallegos, J. and Portilla-Flores, E.A. (2013), “Robust Structure-Control Design Approach for Mechatronic Systems”, *IEEE/ASME Transactions on Mechatronics*, Vol. 18 No. 5, pp. 1592–1601.