

Model Inversion using Extended Gradual Intervals Arithmetic

Reda Boukezzoula¹ Laurent Foulloy¹ Sylvie Galichet¹

¹ LISTIC – University of Savoie – BP. 80439 – 74944 Annecy-le-vieux Cedex France

Abstract

Recently, gradual numbers have been introduced as a means of extending standard interval computation methods to fuzzy and gradual intervals. However, it is well known that the practical use of standard interval arithmetic operators, just as their fuzzy extension, gives results more imprecise than necessary. In this paper, we combine the concepts of gradual numbers and Kaucher arithmetic on extended intervals to define extended gradual interval arithmetic where subtraction and division operators are respectively the inverse operators of the addition and the multiplication. They are applied to a control problem based on the inversion of a linear model.

Keywords: Fuzzy and Gradual Interval Arithmetic, Extended Gradual Intervals, Exact Inverse operators, Mid-point-Radius Representation, Gradual Interval Model Inversion.

1. Introduction

Inverting systems is an important issue in engineering, especially in control and diagnostic problems [4], [5], [6]. However, in many applications the model of a system may contain parameters whose values are not precisely known. Fuzzy numbers and fuzzy intervals [2], [7], [11], [17] are frequently proposed to represent such parameters, especially in decision making, control theory or approximate reasoning problems.

Recently, Fortin *et al.* introduced the concept of gradual numbers which provides a new outlook on fuzzy intervals [13]. Indeed, a gradual number is defined by an assignment function able to represent the essence of graduality. A fuzzy interval can be represented as a pair of gradual numbers. These lower and upper bounds are called left and right profiles. This approach allows the extension of standard interval computation methods to fuzzy intervals.

Extending usual arithmetic operations on real numbers to intervals is not a new problem either for the crisp case [22], [23] or the fuzzy one [12], [21]. It is well-known that Zadeh's extension principle is computationally expensive due to the requirement of solving a nonlinear programming problem [8], [24]. Considering the fuzzy numbers or intervals as a collection of α -cuts is a conventional approach [1], [14]. It is also well-known that fuzzy arithmetic operations give results usually too large or imprecise (the so-called "dependence effect")

[3], [17], [18], [22], [23]. It follows that these arithmetic operations may produce some counterintuitive results where addition and subtraction (multiplication and division) are not inverse operations.

It is important to address this problem for applications where the inversion of a model is required. Additional constraints can be added on variables to obtain exact inverse operators as proposed in [17], [25]. Another approach was developed in the 70's by Orloff then Kaucher. They proposed a new type of interval arithmetic where the order condition between the lower bound and the upper bound of a conventional interval is no more required (the original work was published in German, see [16] for a paper in English). It is known as Kaucher arithmetic or Extended Interval Arithmetic. Since then, it has been investigated by several authors either theoretically or at the application level (see among others [9], [14], [19], [20]).

This paper merges the gradual interval approach and Kaucher arithmetic. It leads to Extended Gradual Intervals and the associated arithmetic operators which provide a means for computing inverse solution to linear equations. In section 2, basic notions about gradual intervals are introduced. Section 3 is devoted to arithmetic operations associated with gradual intervals and extended gradual intervals. Applications to right model inversion are provided in section 4. Concluding remarks are finally given in Section 5.

2. Proper and Improper Gradual Intervals

A gradual number is defined by an assignment function from $(0,1] \rightarrow \mathbb{R}$. In other words, a gradual number is simply a number which is parameterized by $\lambda \in (0,1]$. In this context, a real interval a becomes a gradual interval $a(\lambda)$ when its bounds are gradual numbers. It is then represented by the ordered pair of its two gradual bounds, denoted $[a^-(\lambda), a^+(\lambda)]$ with $a^-(\lambda) \leq a^+(\lambda)$.

In the same way as a conventional interval, the Mid-point, the Radius and the relative extent of a gradual interval $a(\lambda)$ are given by:

$$\begin{aligned} M_a(\lambda) &= (a^-(\lambda) + a^+(\lambda)) / 2 \\ R_a(\lambda) &= (a^+(\lambda) - a^-(\lambda)) / 2 \\ RX_a(\lambda) &= R_a(\lambda) / M_a(\lambda), \text{ for: } M_a(\lambda) \neq 0 \end{aligned} \quad (1)$$

The boundaries of a conventional interval a are ordered, i.e. $a^- \leq a^+$. In Kaucher arithmetic, a conventional interval is called a Proper Interval while an interval with a lower bound greater than the upper one is called an Im-

proper Interval. Let us denote PI the set of Proper Intervals (PI^* is the set of intervals which do not contain zero). In this context, it is well known that the subsystems $(PI, +)$ and (PI, \times) are semi-groups [19], [23]. It means that the reciprocal operations for $+$ and \times do not exist. In this framework, it is obvious that the solution of the interval equations $a + x = b$ and $a \times x = b$ cannot always be expressed in terms of proper intervals in PI .

As mentioned in [19], the equality $a + x = b$ has no solution in \mathfrak{R}^+ for $a > b$. In this case, the primary use of negative numbers is to make the equation $a + x = b$ always solvable, i.e. to switch from the additive semi-group $(\mathfrak{R}^+, +)$ to the group $(\mathfrak{R}, +)$. The set PI with the addition is a semi-group which can be extended to a group with new elements, called Improper Interval belonging to a set II . This set together with the set PI forms the set $EI = PI \cup II$ of Extended Intervals. In other words, like \mathfrak{R}^+ is completed by the set of negative numbers, PI can be completed by the set II of improper intervals having their radius negative.

This concept can be directly extended to gradual intervals. Indeed, a gradual interval can be interpreted as a conventional interval in a space of functions (i.e. the interval bounds are gradual number). Therefore, the set PGI of proper gradual intervals can be completed by the set IGI of improper gradual intervals, having their radius negative and $EGI = PGI \cup IGI$ is the set of Extended Gradual Intervals.

The set EGI is useful for solving inverse problems. However, the results can be not realistic for real applications, i.e. when improper intervals are obtained. An analogy can be made with complex numbers when they are used for computation but real numbers are required for the result. Therefore, an important address is to find conditions which guarantee that the result belongs to the set PGI .

3. Arithmetic Operators

3.1. PGI Arithmetic

The conventional interval arithmetic operations defined for intervals in the Midpoint-Radius (MR) space [20] can be directly extended to the gradual ones [10], [13].

3.1.1 Addition

$$\forall a(\lambda) \in PGI, \forall b(\lambda) \in PGI, \\ a(\lambda) + b(\lambda) = (M_a(\lambda) + M_b(\lambda), R_a(\lambda) + R_b(\lambda)) \quad (2)$$

3.1.2 Negation

$$\forall a(\lambda) \in PGI, \\ \text{neg}(a(\lambda)) = -a(\lambda) = (-M_a(\lambda), R_a(\lambda)) \quad (3)$$

3.1.3 Subtraction

It is defined as the addition of $a(\lambda)$ with the negation of $b(\lambda)$, i.e.:

$$\forall a(\lambda) \in PGI, \forall b(\lambda) \in PGI, \\ a(\lambda) - b(\lambda) = a(\lambda) + \text{neg}(b(\lambda)) \\ = (M_a(\lambda) - M_b(\lambda), R_a(\lambda) + R_b(\lambda)) \quad (4)$$

3.1.4 Multiplication by a Scalar

$$\forall a(\lambda) \in PGI, \forall \sigma \in R, \\ \sigma \times a(\lambda) = (\sigma \times M_a(\lambda), |\sigma| \times R_a(\lambda)) \quad (5)$$

3.1.5 Multiplication

$$\forall a(\lambda) \in PGI, \forall b(\lambda) \in PGI, \\ a(\lambda) \times b(\lambda) = c(\lambda) = (M_c(\lambda), R_c(\lambda)) \quad (6)$$

The expression of $c(\lambda)$ can be simplified according to the relative extent of the operands. In this case, three cases can be distinguished:

- If $|RX_a(\lambda)| \leq 1$ and $|RX_b(\lambda)| \leq 1$

$$M_c(\lambda) = M_a(\lambda) \times M_b(\lambda) + s(\lambda) \times R_a(\lambda) \times R_b(\lambda) \\ R_c(\lambda) = |M_a(\lambda)| \times R_b(\lambda) + |M_b(\lambda)| \times R_a(\lambda) \quad (7) \\ \text{with : } s(\lambda) = \text{sign}(M_a(\lambda)) \times \text{sign}(M_b(\lambda))$$

- If $|RX_a(\lambda)| > 1 \geq |RX_b(\lambda)|$ or $|RX_a(\lambda)| \geq |RX_b(\lambda)| > 1$

$$M_c(\lambda) = \theta_b(\lambda) \times M_a(\lambda), R_c(\lambda) = |\theta_b(\lambda)| \times R_a(\lambda) \quad (8) \\ \text{with : } \theta_b(\lambda) = \text{sign}(M_b(\lambda)) \times (|M_b(\lambda)| + R_b(\lambda))$$

- If $|RX_b(\lambda)| > 1 \geq |RX_a(\lambda)|$ or $|RX_b(\lambda)| \geq |RX_a(\lambda)| > 1$

$$M_c(\lambda) = \theta_a(\lambda) \times M_b(\lambda), R_c(\lambda) = |\theta_a(\lambda)| \times R_b(\lambda) \quad (9) \\ \text{with : } \theta_a(\lambda) = \text{sign}(M_a(\lambda)) \times (|M_a(\lambda)| + R_a(\lambda))$$

3.1.6 Reciprocal

The reciprocal of a gradual interval is reduced to the multiplication by a scalar.

$$\forall a(\lambda) \in PGI^*, \\ \text{rec}(a(\lambda)) = 1/a(\lambda) = \delta_a(\lambda) \times a(\lambda) \quad (10) \\ \text{with } \delta_a(\lambda) = (M_a(\lambda)^2 - R_a(\lambda)^2)^{-1}$$

3.1.7 Division

The division of $a(\lambda)$ by $b(\lambda)$ is defined as the multiplication of $a(\lambda)$ by the reciprocal of $b(\lambda)$, i.e.:

$$\forall a(\lambda) \in PGI, \forall b(\lambda) \in PGI^*, \\ a(\lambda) / b(\lambda) = a(\lambda) \times \text{rec}(b(\lambda)) = \delta_b(\lambda) \times a(\lambda) \times b(\lambda) \quad (11)$$

3.2. Remarks and Discussions

According to these operators, it can be easily shown that:

$$b(\lambda) + (a(\lambda) - b(\lambda)) \neq a(\lambda) \quad \text{and} \quad b(\lambda) \times (a(\lambda) / b(\lambda)) \neq a(\lambda).$$

Indeed, while being based on standard gradual interval arithmetic, these results are not controversial. Let $\mathbf{0}$ and $\mathbf{1}$ be thin gradual intervals whose representation is based on constant functions, that is:

$$\forall \lambda \in [0, 1], 0^-(\lambda) = 0^+(\lambda) = \mathbf{0} \\ \forall \lambda \in [0, 1], 1^-(\lambda) = 1^+(\lambda) = \mathbf{1} \quad (12)$$

It can also be shown that $a(\lambda) - a(\lambda) \neq \mathbf{0}$ and $a(\lambda) / a(\lambda) \neq \mathbf{1}$, which are counterintuitive results. Indeed, the usual division and multiplication operators give a result more imprecise than necessary. This prob-

lem is related to the lack of inverses in the calculus of fuzzy and gradual quantities. In this context, as the addition and subtraction (resp. multiplication and division) are not inverse operations, it is not possible to solve exactly inverse problems using these operators. Thus, a way around this problem must be searched for outside standard *PGI* arithmetic operations.

3.3. EGI Arithmetic

The extended interval arithmetic operations [16] can also be defined for gradual intervals. In the following, *EGI** denotes the set of extended gradual intervals which do not contain 0.

3.3.1 Dual

The dual operator is defined by symmetry with respect to the radius:

$$\forall a(\lambda) \in EGI, \text{dual}(a(\lambda)) = (M_a(\lambda), -R_a(\lambda)) \quad (13)$$

In other words, the dual operator makes a permutation between the interval bounds in the *EP* representation. It transforms a proper gradual interval into an improper one and reciprocally.

$$\text{dual}(a(\lambda)) = \text{dual}([a^-(\lambda), a^+(\lambda)]) = [a^+(\lambda), a^-(\lambda)] \quad (14)$$

3.3.2 Opposite

The opposite of $a(\lambda)$ is defined as the negation of the dual of $a(\lambda)$:

$$\forall a(\lambda) \in EGI, \text{opp}(a(\lambda)) = \text{neg}(\text{dual}(a(\lambda))) \\ = (-M_a(\lambda), -R_a(\lambda)) \quad (15)$$

3.3.3 Inverse

The inverse of $a(\lambda)$ is reduced to the multiplication of $\text{dual}(a(\lambda))$ by a scalar:

$$\forall a(\lambda) \in EGI, \\ \text{inv}(a(\lambda)) = 1 / \text{dual}(a(\lambda)) = \delta_a(\lambda) \times \text{dual}(a(\lambda)) \quad (16)$$

3.3.4 Subtraction

The subtraction operator is defined from the addition and the opposite:

$$\forall a(\lambda) \in EGI, \forall b(\lambda) \in EGI, \\ a(\lambda) - b(\lambda) = a(\lambda) + \text{opp}(b(\lambda)) \\ = (M_a(\lambda) - M_b(\lambda), R_a(\lambda) - R_b(\lambda)) \quad (17)$$

In this case, we have $a(\lambda) - a(\lambda) = \mathbf{0}$. As already mentioned, $c = a - b$ have no solution in \mathfrak{R}^+ for $a < b$. Similarly, the equation $c(\lambda) = a(\lambda) - b(\lambda)$ given by (17) has no solution in *PGI* if the radius of $c(\lambda)$ is not positive or null which leads to:

$$R_c(\lambda) \geq 0 \Rightarrow R_a(\lambda) \geq R_b(\lambda), \forall \lambda \in [0, 1] \quad (18)$$

The subtraction operator can be interpreted as an adapted version of the Hukuhara difference definition for intervals [26].

3.3.5 Division

The division of $a(\lambda)$ by $b(\lambda)$ is defined as the multiplication of $a(\lambda)$ by the inverse of $b(\lambda)$:

$$\forall a(\lambda) \in EGI, \forall b(\lambda) \in EGI^*, \\ a(\lambda) / b(\lambda) = a(\lambda) \times \text{inv}(b(\lambda)) \\ = \delta_b(\lambda) \times a(\lambda) \times \text{dual}(b(\lambda)) \quad (19)$$

In this case, it can be stated that $a(\lambda) / a(\lambda) = 1$. Furthermore, $c(\lambda) = a(\lambda) / b(\lambda) \in \text{PGI}$ if and only if the radius of $c(\lambda)$ is positive or null. As $b(\lambda) \in EGI^*$, i.e. $|RX_b(\lambda)| < 1$, it can be proved that $a(\lambda) / b(\lambda) \in \text{PGI}$ if and only if $|RX_a(\lambda)| \geq |RX_b(\lambda)|$, i.e. $a(\lambda)$ must be more extended (imprecise) than $b(\lambda)$.

4. Application to Model Inversion

The inverse of a system can be intuitively understood as being another system. When it is cascaded with the original one, it produces outputs which are the inputs of the original system. Two classes of inversion techniques are generally studied: left inverse and right inverse. For a given system S , the left inverse is defined as a system which computes some inputs of S from the knowledge of its outputs. This inverse is related to fault detection, diagnosis problems and reconstruction of inaccessible inputs. For example, when a mapping which describes the input-output relationship between the failure root causes and their symptoms is identified, then, if the symptoms values are known, the root causes that lead to these symptoms should be identified. This objective can be carried out by the left input-output mapping inversion. On the other hand, the right inverse of a system S is a second one which computes the required input in order to ensure that S have a certain desired output. This inversion technique is especially used in a control context and deeply related to regulation and tracking problems. Since the plant model represents links between the inputs, the current outputs and the future ones, one can use the inverse mapping as a design procedure for the control problem. That is the approach adopted in this paper where an inverse control strategy is proposed.

4.1. Linear Model-Based Inverse Control

Let us consider a SISO (Single Input-Single Output) linear plant whose representation is given by the following discrete-time dynamic equation:

$$y_{k+1}(\lambda) = \sum_{i=1}^n a_i(\lambda) \times y_{k-i+1}(\lambda) + \sum_{j=0}^{m-1} b_j(\lambda) \times u_{k-j}(\lambda) \quad (20)$$

where $y_k(\lambda)$ is the current output, $u_k(\lambda)$ is the current control input and $a_i(\lambda)$, $i = 1, \dots, n$, $b_j(\lambda)$, $j = 0, \dots, m-1$ denote the gradual interval model parameters. In the sequel, in order to simplify the notations, the expression (20) is rewritten as:

$$y_{k+1}(\lambda) = \Psi_k(\lambda) + b_0(\lambda) \times u_k(\lambda); \text{ where:}$$

$$\Psi_k(\lambda) = \sum_{i=1}^n a_i(\lambda) \times y_{k-i+1}(\lambda) + \sum_{j=1}^{m-1} b_j(\lambda) \times u_{k-j}(\lambda)$$

In this representation, for a given degree of preference λ , the nominal best value (the Midpoint) of any entity is considered together with an upper bound of its uncer-

tainty (the Radius). So, the gradual interval model can be understood as a family of interval systems. Usually, the control objective consists in forcing the plant output to follow a specified desired trajectory. In other words, one has to determine the control action which guarantees that the plant output converges to the desired trajectory. Due to the presence of uncertainties and the flexibility of the model, the control objective can be viewed as determining a gradual interval controller $u_k(\lambda)$ able to maintain the system output within an envelope of tolerance [4], [5] around the reference trajectory $r_k(\lambda) = (M_{r_k}(\lambda), R_{r_k}(\lambda))$. In this case, the stability is ensured in the way that the controller maintains the system output evolution within the specified desired tolerance envelope. From theoretical point of view, if the system (20) is invertible ($u_k(\lambda) \in PGI^*$) then its inverse can be implemented as an analytical inversion where the output is replaced by the desired trajectory:

$$u_k(\lambda) = (r_{k+1}(\lambda) - \Psi_k(\lambda)) / b_0(\lambda) \quad (21)$$

The basic scheme of the interval control structure is illustrated in Fig. 1 where the controller is directly obtained by model inversion.

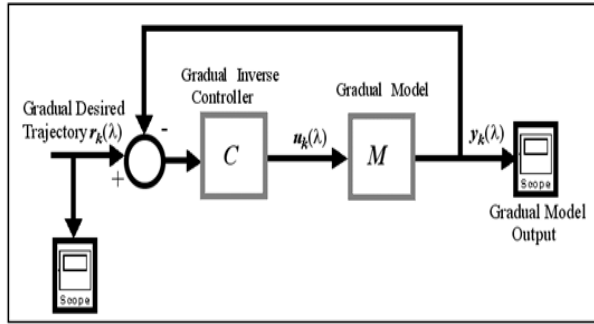


Fig. 1: Interval Inverse Control Structure

The inverse control action given by (21) is a proper gradual interval if the following condition is satisfied:

$$R_{r_{k+1}}(\lambda) \geq R_{\Psi_k}(\lambda) \text{ and } |RX_{r_{k+1}-\Psi_k}(\lambda)| \geq |RX_{b_0}(\lambda)|$$

leading to:

$$R_{r_{k+1}}(\lambda) \geq R_{\Psi_k}(\lambda) + |M_{r_{k+1}}(\lambda) - M_{\Psi_k}(\lambda)| \times |RX_{b_0}(\lambda)| \quad (22)$$

It is important to note here that the proposed control structure aims at determining, for any given uncertain desired trajectory (envelope of tolerance), the interval of possible values of the control input. So, primarily this control methodology is concerned with the system performance analysis in presence of interval uncertainties. Indeed, if a tolerance envelope is specified such that the condition (22) is satisfied, any crisp control input chosen in the interval control ensures that the system output is within the interval specified envelope. From practical implementation point of view, if we want to implement this control strategy to a real plant, a crisp control value must be chosen. In this case, for a crisp desired trajectory, the control value can be selected by a defuzzification method according to a desired control performance criterion.

4.2. Illustrative Example

Assume that the plant is a SISO linear system described by the following difference equation with varying parameters $a_0(\lambda), a_1(\lambda) \in PGI$ and $b_0(\lambda) \in PGI^*$:

$$y_{k+1}(\lambda) = a_0(\lambda) \times y_k(\lambda) + a_1(\lambda) \times y_{k-1}(\lambda) + b_0(\lambda) \times u_k(\lambda)$$

Let us also assume that system parameters are expressed by the triangular gradual intervals given by:

$$a_0(\lambda) = (0.1, 0.05 - 0.05\lambda), \quad a_1(\lambda) = (-0.5, 0.1 - 0.1\lambda) \text{ and } b_0(\lambda) = (1, 0.05 - 0.05\lambda).$$

Let us consider a triangular gradual reference trajectory $r_k(\lambda)$ given by the following MR representation :

$$r_k(\lambda) = (2, 1.5 - 1.5\lambda).$$

The initial conditions are taken equal to the initial desired trajectory. The behaviour of the system output and the control input (which is constant and not dependent on k), both obtained when applying the interval control method are illustrated in Fig. 2 and Fig. 3.

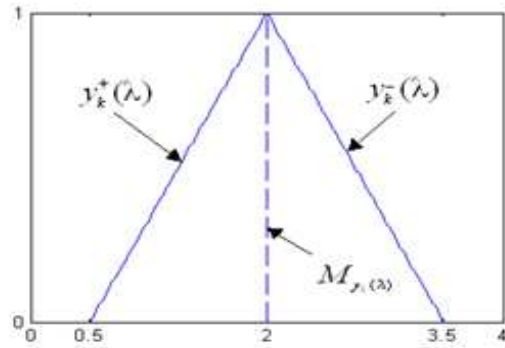


Fig. 2: Gradual System Output

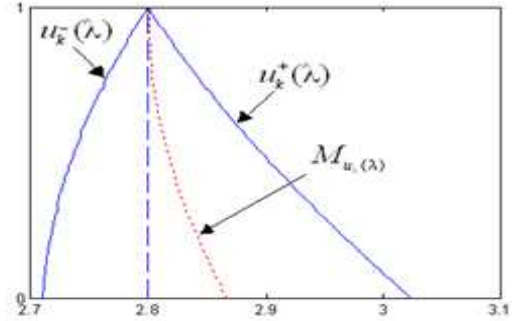


Fig. 3: Gradual Control Input

From these results, it can be stated that the system output tracks the desired trajectory perfectly. The controller maintains the system output within the tolerance envelope around the nominal desired trajectory.

When $\lambda = 1$ (precise input), the system tracks the nominal and exact desired trajectory. It is important to note here that the obtained control input can be always computed but sometimes not represented by a gradual interval. In this case, in order to ensure a PGI controller output, the condition (22) must be guaranteed. In this case, as the controller imposes that $y_k(\lambda) = r_k(\lambda)$, equation (22) is simplified to:

$$M_{r_k}(\lambda) \geq (1.24 - 1.24\lambda).$$

In this case all gradual trajectories leading to PGI or IGI controller outputs are illustrated in Fig. 4.

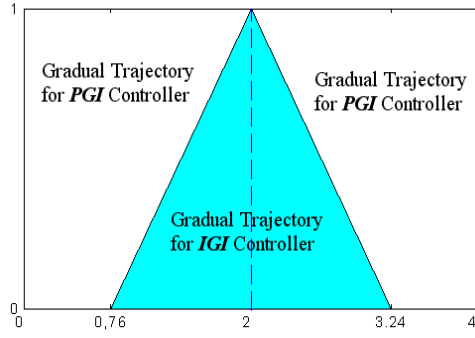


Fig. 4: PGI and IGI trajectories for the controller.

The developed control method is not limited to constant trajectories, but can be applied to time varying ones. Let us now consider that the constant nominal desired trajectory is replaced by the following one:

$$M_{r_k}(\lambda) = \sin(2k\pi/25) + \sin(2k\pi/50).$$

In this case, the gradual desired trajectory is rewritten as:

$$\begin{aligned} r_k(\lambda) &= (M_{r_k}(\lambda), R_{r_k}(\lambda)) \\ &= (\sin(2k\pi/25) + \sin(2k\pi/50), 1.5 - 1.5\lambda). \end{aligned}$$

The initial condition is taken equal to desired trajectory, parameters $a_0(\lambda)$ and $a_1(\lambda)$ being unchanged. The simulation results are given in Fig. 5 and Fig. 6.

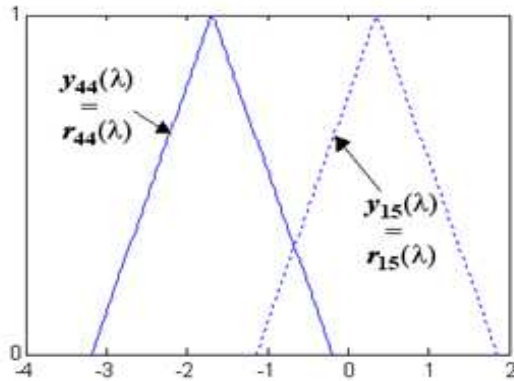


Fig. 5: Gradual Outputs and desired trajectories ($k=15$ and $k=44$)

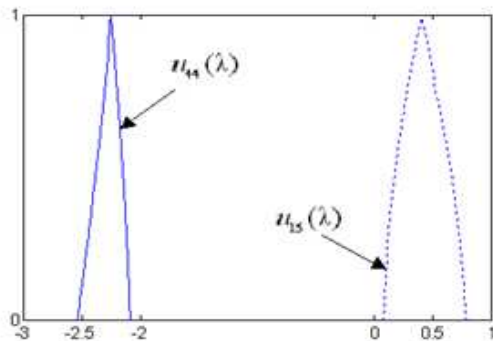


Fig. 6: Gradual Control inputs ($k=15$ and $k=44$).

According to these results, good control performances are then achieved, illustrating the efficiency of the proposed technique. Knowing that the midpoint of the trajectory is not constant, at each sampling time, the control input is a gradual interval as illustrated in Fig. 6. Moreover, the inverse controller output always belongs

to PGI if the desired trajectory uncertainty is chosen so that:

$$R_{r_{k+1}}(\lambda) \geq \max_k (R_{\psi_k}(\lambda) + |M_{r_{k+1}}(\lambda) - M_{\psi_k}(\lambda)| \times |RX_{b_0}(\lambda)|)$$

In this method, the controller is designed for all possible varying parameters values; the proposed technique presents a robustness property with regard to system parameter variations.

5. Conclusion

This paper has presented a methodology for the implementation of the subtraction and the division operators between gradual intervals. Based on these operators, an optimistic counterpart of the usual fuzzy arithmetic has been proposed. The proposed operators are exact inverses of the addition and multiplication operators and can solve the overestimation problem well-known in conventional fuzzy and gradual interval arithmetic. In this paper, academic illustrative examples have been used for illustrations. More complicated and realistic cases must be further tested. For example, the proposed operators may be used in the context of sensitivity analysis, linear and nonlinear equations solving, uncertain aggregation operators (Weighted Average, OWA,...), determining cluster centres for linguistic fuzzy C-means or in the aggregation of Sugeno-like rule consequents.

References

- [1] Bodjanova S., "Alpha-bounds of fuzzy numbers", Information Sciences, Vol. 152, pp. 237-266, 2003.
- [2] Boukezzoula R., Galichet S., Foulloy L., "MIN and MAX Operators for Fuzzy Intervals and their Potential Use in Aggregation Operators", IEEE Trans. on Fuzzy Systems Vol. 15, N° 6, pp. 1135-1144, 2007.
- [3] Boukezzoula R., Galichet S., Foulloy L., "Inverse arithmetic operators for fuzzy intervals", EUS-FLAT'07, Ostrava, 2007.
- [4] Boukezzoula R., Foulloy L., Galichet S., "Inverse Controller Design for Interval Fuzzy Systems", IEEE Transactions On Fuzzy Systems, Vol. 14, N° 1, pp. 111-124, 2006
- [5] Boukezzoula R., Galichet S., Foulloy L., "Nonlinear Internal Model Control: Application of Inverse Model Based Fuzzy Control", IEEE Transactions on Fuzzy Systems Vol. 11 N° 6, pp. 814-829, 2003.
- [6] Cabrera J.B.D. and Narendra K.S., "Issues in the Application of Neural Networks for Tracking Based on Inverse Control", IEEE Transactions on Automatic Control, Vol. 44, No. 11, November 1999.
- [7] Deschrijver G., "Generalized arithmetic operators and their relationship to t-norms in interval-valued fuzzy set theory", Fuzzy Sets and Systems, N° 160, pp. 3080-3102, 2009.
- [8] Dong W. M. and Wong F. S., "Fuzzy weighted averages and implementation of the extension principle", Fuzzy Sets and Systems, No 1, pp. 183-199, 1987.

- [9] Dimitrova N., Markov S., Popova E., "Extended interval arithmetics: new results and applications", in *Computer Arithmetic and Enclosure Methods* (Eds. Atanassova L., Herzberger J.), Elsevier Sci. Publishers, pp. 225-232, 1992.
- [10] Dubois D. and Prade H., "Gradual elements in a fuzzy set", *Soft Computing*, No12, pp. 165-175, 2008.
- [11] Dubois D., Kerre E., Mesiar R. and Prade H., "Fuzzy interval analysis", in: D. Dubois, H. Prade (Eds.), *Fundamentals of Fuzzy Sets, The Handbooks of Fuzzy Sets Series*, Kluwer, Boston, pp. 483-581, 2000.
- [12] Dubois D. and Prade H., "Operations on fuzzy numbers", *Journal of Systems Science*, No. 9, pp. 613-626, 1978.
- [13] Fortin J. Dubois D. and Fargier H., "Gradual numbers and their application to fuzzy interval analysis", *IEEE Trans. on Fuzzy Systems*, Vol. 16, No. 2, pp. 388-402, 2008.
- [14] Goldsztejn A., "A Right-Preconditioning Process for the Formal-Algebraic Approach to Inner and Outer Estimation of AE-Solution Sets", *Reliable Computing*, 11, pp. 443-478, 2005.
- [15] Guerra M.L. and Stefanini L., "Approximate fuzzy arithmetic operations using monotonic interpolations", *Fuzzy Sets and Systems*, Vol. 150, pp. 5-33, 2005.
- [16] Kaucher E., "Interval Analysis in Extended Interval Space \mathbb{IR} ", *Computing, Suppl. 2*, pp. 33-49, 1980.
- [17] Klir G.J., "Fuzzy arithmetic with requisite constraints", *Fuzzy Sets and Systems*, Vol. 91, No. 2, pp. 165-175, 1997.
- [18] Kulpa Z., "Diagrammatic representation for interval arithmetic", *Linear Algebra and its Applications*, Vol. 324, No. 1-3, pp. 55-80, 2001.
- [19] Markov, S., "On directed interval arithmetic and its applications", *Journal of Universal Computer Science*, Vol. 1, No. 7, pp. 514-526, 1995.
- [20] Markov, S., "Computation of Algebraic Solutions to Interval Systems via Systems of Coordinates", *Scientific Computing, Validated Numerics, Interval Methods* Eds. W. Kraemer, J. Wolff von Gudenberg), Kluwer, pp. 103-114, 2000.
- [21] Mizumoto M. and Tanaka K., "The four operations of arithmetic on fuzzy numbers", *Systems Comput. Controls*, Vol. 7 (5), pp. 73-81, 1976.
- [22] Moore R.E., "Methods and applications of interval analysis", SIAM, philadelphia, 1979.
- [23] Moore R.E., "Interval Analysis", Prentice-Hall, Englewood Cliffs, NJ. 1966.
- [24] Oussalah M. and De Schutter J., "Approximated fuzzy LR computation", *Information Sciences*, Volume 153, pp. 155-175, 2003.
- [25] Piegat A., "Cardinality Approach to Fuzzy Number Arithmetic", *IEEE Transactions on Fuzzy Systems*, Vol. 13, N°. 2, pp. 204-215, 2005.
- [26] Stefanini L. and Bede B., "Generalization of Hukuhara differentiability of interval-valued functions and interval differential equations", *Nonlinear Analysis*, Vol. 71, pp. 1311-1328, 2009.