# 14th Edition of SWIM2023

## Summer Workshop on Interval Methods 2023

27-29 Jun 2023 Angers (France)

## *In memory of Nicolas Delanoue*

*« Un jour j'irai vivre en Théorie, car en Théorie tout se passe bien. »*

## Scope and Topics

SWIM aims at gathering researchers working on/with interval methods and their applications. The goal is to review the state-of-the-art in this field. Contributions can be for example in the domain of

- Verification and Validation
- Robust and Nonlinear Control Systems
- State Estimation
- Interval Observer Design
- Parameter Identification
- Fault Detection and Diagnosis, Fault Tolerant Systems
- Stability, Reachability, Observability
- Reliable Software Design
- Robotics
- Mathematics
- Verified Solution of Algebraic and Dynamic System Models
- Verified Numerics and Scientific Computing
- Linear Algebra
- and any other applications of interval methods, verified numerics, and other related set-membership techniques (*e.g:* affine arithmetics, polytopes, *etc*).

## Previous Editions of SWIM

The SWIM workshop series was initiated by the French MEA working group on Set Computation and Interval Techniques of the French research group on Automatic Control GDR MACS, where the MEA group especially aimed at promoting interval analysis techniques and applications to a broader community of researchers. Since 2008, SWIM has become an annual keystone event for researchers dealing with various aspects of interval and set-membership methods.

Previous editions of SWIM were held in:

- Hannover, Germany in 2022
- Paris, France in 2019
- Rostock, Germany in 2018
- Manchester, UK in 2017
- Lyon, France in 2016
- Prague, Czech Republic in 2015
- Uppsala, Sweden in 2014
- Brest, France in 2013
- Oldenburg, Germany in 2012
- Bourges, France in 2011
- Nantes, France in 2010
- Lausanne, France in 2009
- Montpellier, France in 2008

## Tuesday June 27

09:00 Welcome, inscriptions and coffee

09:50 Opening speech by Sébastien LAHAYE

10:00 **Plenary session by Andreas RAUH**

- **Offline and Online Use of Interval and Set-Based Approaches for Control and State Estimation: A Review of Methodological Approaches and Their Application**

11:00 **Parameter Identification | Verified Numerics and Scientific Computing**

- **Interval-based Exhaustive 2D Rigid Registration with Uniform Scaling**
  *Radwan Verlein, Rohou Simon, Trombettoni Gilles*
- **Interval Methods for the GPU**
  *Gillner Lorenz , Auer Ekaterina*

12:00 Lunch

14:00 **Robotics**

- **Acceleration of contractor algebra on RISCV in the context of mobile robotic**
  *Filiol Pierre, Jaulin Luc, Le Lann Jean-Christophe, Bollengier Theotime*
- **Hybrid Interval-Probabilistic Localization in Urban Environments**
  *Ehambram Aaronkumar, Jaulin Luc, Rohou Simon, Wagner Bernardo*
- **Union of adjacent contractors**
  *Brateau Quentin, Le Bars Fabrice, Jaulin Luc*

15:30 Coffee break

16:00 **Robotics**

- **Manifold decomposition for interval analysis**
  *Ignazi Arthur, Guyonneau Rémy, Lagrange Sébastien, Delanoue Nicolas*
- **Brunovsky decomposition for dynamic interval localization**
  *Rohou Simon, Jaulin Luc*

17:00 End of the day

## Wednesday June 28

09:00 Welcome and inscriptions

09:30 **Plenary session by Luc JAULIN**

- **A contractor which is minimal for narrow boxes**

10:30 Coffee break

11:00 **Reliable Software Design**

- **Possible preliminary test for the optimality condition in Interval Branch and Bound method**

  *Mihály Gencsi, Boglárka G.-Tóth*

- **Sparse tensors and subdivision methods for finding the zero set of polynomials**

  *Moroz Guillaume*

- **A new type of intervals for solving problems involving partially defined functions**

  *Revol Nathalie*

12:30 Lunch

14:00 **Mathematics**

- **Interval-Based Uncertainty Propagation for Multimodal Response Functions**

  *Sotiropoulos Dimitris, Adam Stavros*

- **Numerical Scheme of Generalised Moment Problem Based on Interval Analysis**

  *Algassimou Diallo*

- **Explaining neural classifiers decisions: A parameter estimation problem solved using intervals**

  *Adam Stavros, Likas aAristidis, Samartzis Ioannis*

15:30 Coffee break

16:00 **State Estimation**

- **State-Estimation of Uncertain Timed Event Graphs : an SMT approach**

  *Espindola Winck Guilherme, Hardouin Laurent, Lhommeau Mehdi*

- **Interval Observer Design for State Estimation of Lithium-Ion Batteries**

  *Lahme Marit, Rauh Andreas, Defresne Guillaume*

17:00 End of the day

19:00 Gala Dinner

## Thursday June 29

08:30 Welcome and inscriptions

09:00 **Verification and Validation | Verified Solution of Algebraic and Dynamic System Models**

- **Propagation of degenerate ellipsoids towards computer-assisted proofs**

  *Louédec Morgan, Christophe Viel, Jaulin Luc*

- **Set-Based Simulation Approaches for the Numerical Analysis of Dynamic Spiking Neuron Models**

  *Rauh Andreas, Kretzberg Jutta*

- **Computing invariants for discrete-time dynamical systems using polynomial zonotopes**

  *Chiche Nathan, Goubault Eric, Putot Sylvie*

10:30 Coffee break

11:00 **Robust and Nonlinear Control Systems**

- **Number of connected components using interval analisys**

  *Hugo Remin, Sébastien Lagrange*

- **Imprecise Bayesian Neural Networks**

  *Caprio Michele, Dutta Souradeep, Jang Kuk, Lin Vivian, Ivanov Radoslav, Sokolsky Oleg, Lee Insup*

12:00 Closing speech

# Tuesday June 27

# Offline and Online Use of Interval and Set-Based Approaches for Control and State Estimation: A Review of Methodological Approaches and Their Application

Andreas Rauh

Carl von Ossietzky Universität Oldenburg
Distributed Control in Interconnected Systems
26111 Oldenburg, Germany
andreas.rauh@uol.de

**Keywords:** Interval methods, Set-based state estimation, Robust control, Optimization under uncertainty, Neural networks, Fuel cells

## Introduction

Control and state estimation procedures need to be robust against imprecisely known parameters, uncertainty in initial conditions, and external disturbances. Interval methods and other set-based techniques form the basis for the implementation of powerful approaches that can be used to identify parameters of dynamic system models in the presence of the aforementioned uncertainties. Moreover, they are applicable to a verified feasibility and stability analysis of controllers and state estimators [1,3,7].

In addition to offline approaches for analysis, interval and set-based methods have also been developed in recent years which are allow to solve the associated design tasks and to implement reliable techniques that are applicable online. The latter approaches include online parameter adaptation techniques for nonlinear variable-structure controllers, interval observers, and fault diagnosis techniques [3,4,5,7]. In this talk, an overview of the methodological background will be presented, together with a review of practical applications for which interval and set-valued approaches have been employed successfully.

## Modeling, Parameter Identification, and Verified State Estimation

Although, for example, many dynamic system models in (control) engineering, especially in the frame of thermo-fluidic applications, are described after a first-principle modeling by state equations that have certain monotonicity properties, other applications in the domain of mechanics as well as for electro-chemical energy storage may require specific changes of coordinates to obtain these properties. In the domains of parameter identification as well as state and disturbance estimation, the most important monotonicity property that allows for a simplification of the aforementioned tasks is the cooperativity of the state equations. As far as the application domains mentioned above are concerned, these properties originate from the conservation of mass or energy [4].

In such cases, a decoupling of lower and upper bounding systems — that enclose all possible state trajectories — can be obtained. This property does not only allow for the simplification of the task of parameter identification but also for the implementation of real-time capable state estimation procedures.

For systems with periodically recurring trajectories (and also disturbance profiles), recent investigations have shown that the corresponding procedures can also be extended to a learning-type technique. This technique especially allows for enhancing the bounds of estimated state trajectories in each successive execution of the same task and exploits a formulation that uses the iteration counter as a second independent dimension in addition to time [2].

## Verified Control Implementation and Robust Model-Predictive Control

On the basis of the set-based state estimates described in the previous section, real-time capable robust control implementations can be derived that prevent the violation of state constraints with certainty.

Moreover, it is possible to implement robust predictive control laws in a similar manner. For the case of a nonlinear state feedback, interval extensions of sliding mode and backstepping control approaches have been published which allow for a guaranteed stabilization of the system dynamics and for a guaranteed prevention of overshooting certain thresholds for the state variables under constraints on the inputs and their respective variation rates. A practical application of this technique is the temperature control of a solid oxide fuel cell stack [3,7].

For the second class of controllers, a novel combination of set-based and neural network modeling was recently developed and integrated into a sensitivity-based predictive control scheme that maximizes the degree of fuel utilization of a fuel cell. The approach can be implemented for time-varying desired electric power profiles so that operating points stay within the region of Ohmic polarization, which is crucial for preventing accelerated aging of the fuel cell stack [5].

## Combination of Set-Based and Stochastic Uncertainty Representations

In the final part of this talk, a combination of stochastic and set-based (in this case, ellipsoidal) uncertainty representations will be considered. This approach allows, on the one hand, for a rigorous quantification of predefined confidence levels in stochastic state estimation procedures. On the other hand, it allows for handling nonlinearities in such a way that the previously mentioned tolerance bounds are definitely not determined in an overly optimistic manner [6].

## References


[1] E. Auer, L. Senkel, S. Kiel and A Rauh, Control-Oriented Models for SO Fuel Cells from the Angle of V&V: Analysis, Simplification Possibilities, Performance, *Algorithms*, 10, 140, 2017.

[2] T. Chevet, A. Rauh, T.N. Dinh, J. Marzat, T. Raïssi, Robust Interval Observer for Systems Described by the Fornasini-

Marchesini Second Model, *IEEE Control Systems Letters*, 6, 1940–1945, 2021.

[3] N. Cont, W. Frenkel, J. Kersten, A. Rauh and H. Aschemann, Interval-Based Modeling of High-Temperature Fuel Cells for a Real-Time Control Implementation Under State Constraints, *IFAC-PapersOnLine*, 53, 12542–12547, 2020.

[4] S. Ifqir, A. Rauh, J. Kersten, D. Ichalal, N. Ait-Oufroukh and S. Mammar. *Interval Observer-Based Controller Design for Systems with State Constraints: Application to Solid Oxide Fuel Cells Stacks*, Proceedings of the Intl. Conf. on Methods and Models in Automation and Robotics, MMAR 2019, Miedzyzdroje, Poland.

[5] A. Rauh and E. Auer, *Comparison of Stochastic and Interval-Based Modeling Approaches for the Online Optimization of the Fuel Efficiency of SOFC Systems*, Proceedings of the 9th Intl. Conf. on Systems and Control (ICSC), 2021, Caen, France.

[6] A. Rauh, T. Chevet, T.N. Dinh, J. Marzat, T. Raïssi, *Robust Iterative Learning Observers Based on a Combination of Stochastic Estimation Schemes and Ellipsoidal Calculus*, Proceedings of the 25th Intl. Conf. on Information Fusion (FUSION), 2022, Linköping, Sweden.

[7] A. Rauh, L. Senkel and H. Aschemann, Interval-Based Sliding Mode Control Design for Solid Oxide Fuel Cells With State and Actuator Constraints, *IEEE Transactions on Industrial Electronics*, 62(8), 5208–5217, 2015.

# Interval-based Exhaustive 2D Rigid Registration with Uniform Scaling

Verlein Radwan[1], Simon Rohou[2] and Gilles Trombettoni[1]

[1] LIRMM - Laboratoire d'informatique, de robotique et de microélectronique de Montpellier
161 Rue Ada, 34095 Montpellier, France
{vradwan,gilles.trombettoni}@lirmm.fr
[2] ENSTA Bretagne
2 Rue François Verny, 29200 Brest, France
simon.rohou@ensta-bretagne.fr

**Keywords:** Registration, Separators

## Motivation

The registration of two sets aims at recovering all the transformation parameters that map the two sets together. This work covers the case of bounded 2D sets (bi-partitions of $\mathbb{R}^2$) and transformations consisting of a composition of uniform scaling, rotation and translation.

As opposed to state-of-the-art local optimization methods for registration like ICP [**1**], we use a set-membership (interval) approach that can ensure completeness and compute multiple solutions, if any. Our method also handles uncertainties, whether they arise during computations or, for a future work, are already present in initial data.

Sets are manipulated using methods to describe them: separators.

## Set description

All the set entities, from the initial sets to the parameters of the transformation are managed by algorithmic operators called *separators*.

Using the CODAC library offering a catalogue of contractors, it is possible to define a new one as a sequence of separators handling constraints involving sets and set operations.

Used by a branch and separate algorithm, a separator is an algorithmic operator capable of constructing a set: from an initial domain, it can remove non-solution parts (contraction) as well as parts containing only solutions. Hence, a given separator defines a specific set and in the following, we mix up the notations for sets and separators.

## Set registration: overview

The algorithmic sequence used to perform the registration of two sets $\mathbb{A}$ and $\mathbb{B}$ starts by centering and normalization steps, followed by rotational mapping.

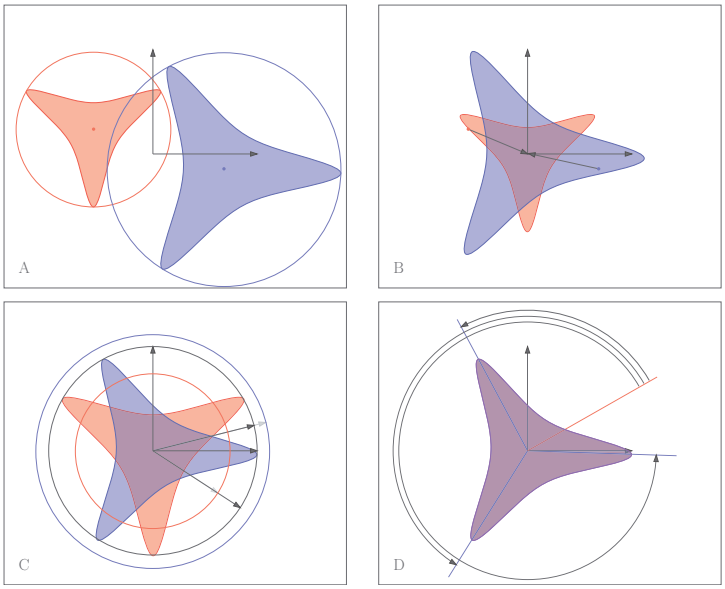



Figure 1: Algorithm in 4 steps: A: Identification of smallest circles (centroids and sizes). B: Centering of sets. C: Normalization of sets. D: Identification of rotation parameters.

The parameter for the centering is the centroid $\mathbf{c}^\star$ and the normalization factor is the size $r^\star$.

The whole algorithm consists of the following steps:

1. Find the centroids $\mathbf{c}^\star_{\mathbb{A}}, \mathbf{c}^\star_{\mathbb{B}}$ and sizes $r^\star_{\mathbb{A}}, r^\star_{\mathbb{B}}$ (Fig. 1.A),
2. Describe the centered sets $\mathbb{A}_c = \mathbb{A} - \mathbf{c}^\star_{\mathbb{A}},\ \mathbb{B}_c = \mathbb{B} - \mathbf{c}^\star_{\mathbb{B}}$ (Fig. 1.B),
3. Describe the normalized sets $\mathbb{A}_n = \mathbb{A}_c/r^\star_{\mathbb{A}},\ \mathbb{B}_n = \mathbb{B}_c/r^\star_{\mathbb{B}}$ (Fig. 1.C),
4. Describe the set of possible rotations $\Theta = \{\theta \mid \mathbb{B}_n = \mathrm{R}_\theta \mathbb{A}_n\}$ (Fig. 1.D).

For Step 1, both the centroid and size of a set will be determined using the parameters of the smallest enclosing circle.

**Proposition 1.** *The smallest (enclosing) circle of a set is unique, its center* $\mathbf{c}$ *and radius* $r$ *are respectively the centroid and size of the set.*

## Set registration: separators

The corresponding domains to previous variables fall in two categories. Vector variables have set domains while set variables have *thick set* [**2**] domains. A thick set is a interval of sets for the inclusion and are manipulated with two separators, one for each bound.

Step 1: the identification of the smallest circle of a set $\mathbb{X}$ is performed in two stages. First, the description of $\mathbb{C}$, the set of all enclosing circles of $\mathbb{X}$, then the identification of the smallest one.

$\mathbb{C}$ can be described using a distance equation, as follows:

$$\begin{aligned} g\colon \mathbb{R}^5 &\to \mathbb{R} \\ (\mathbf{c}, r, \mathbf{x}) &\mapsto (x_1 - c_1)^2 + (x_2 - c_2)^2 - r^2 \end{aligned}$$

$$\begin{aligned} \mathbb{C} &= \{(\mathbf{c}, r) \mid \forall \mathbf{x} \in \mathbb{X}, g(\mathbf{c}, r, \mathbf{x}) \leq 0\} \\ &= \neg\{(\mathbf{c}, r) \mid \exists \mathbf{x} \in \mathbb{X}, g(\mathbf{c}, r, \mathbf{x}) > 0\} \\ &= \neg\left\{(\mathbf{c}, r) \mid \exists \mathbf{x} \in \mathbb{R}^2, (\mathbf{c}, r, \mathbf{x})^\intercal \in \mathbf{g^{-1}}([0, +\infty]) \cap (\mathbb{R}^3 \times \mathbb{X})\right\} \end{aligned}$$

Thus, the separator $\mathcal{S}_{\mathbb{C}}$ can be obtained by the following operations:

$$\mathcal{S}_1 := \mathcal{S}_{\text{inv}}(g, [0, +\infty])); \ \mathcal{S}_2 := \mathcal{S}_1 \cap (\mathcal{S}_{\mathbb{R}^3} \times \mathcal{S}_{\mathbb{X}}); \ \mathcal{S}_3 := \mathcal{S}_{\text{proj}}(\mathcal{S}_2, (\mathbf{c}, r))$$
$$\mathcal{S}_{\mathbb{C}} := \neg\mathcal{S}_3$$

The second stage can also be achieved by separators, in a sequence detailed during the presentation. Let us call SmallestCircle the algorithm returning the desired separators.

Note that, from Step 2 onwards, we showcase only one way of the problem: inclusion of $\mathbb{A}$ in $\mathbb{B}$ by transformation.

1. $(\mathcal{S}_{\mathbf{c}^\star_{\mathbb{A}}}, \mathcal{S}_{r^\star_{\mathbb{A}}}) := \text{SmallestCircle}(\mathcal{S}_{\mathbb{A}})$;
   $(\mathcal{S}_{\mathbf{c}^\star_{\mathbb{B}}}, \mathcal{S}_{r^\star_{\mathbb{B}}}) := \text{SmallestCircle}(\mathcal{S}_{\mathbb{B}})$,
2. $\mathcal{S}^-_{\mathbb{A}_c} := \mathcal{S}_{\mathbb{A}} \ominus \mathcal{S}_{\mathbf{c}^\star_{\mathbb{A}}}$; $\mathcal{S}^+_{\mathbb{B}_c} := \mathcal{S}_{\mathbb{B}} \oplus \mathcal{S}_{-\mathbf{c}^\star_{\mathbb{B}}}$,
3. $\mathcal{S}^-_{\mathbb{A}_n} := \mathcal{S}^-_{\mathbb{A}_c} \oslash \mathcal{S}_{r^\star_{\mathbb{A}}}$; $\mathcal{S}^+_{\mathbb{B}_n} := \mathcal{S}^+_{\mathbb{B}_c} \otimes \mathcal{S}_{(r^{\star -1}_{\mathbb{B}})}$.

where the operations $\otimes$, $\oslash$ are extensions of Minkowski sum and difference operations for the multiplication. Separators for these operations can be found in [**3**].

As for Step 4, we use the property that the rotation is a sum in polar coordinates, so that we can use the Minkowski sum and difference operations. A specific separator $\mathcal{S}_{\text{polar}}$ for a Cartesian to polar change of coordinates is designed for the occasion.

$$\mathcal{S}^-_{\mathbb{A}^P} := \mathcal{S}_{\text{polar}}(\mathcal{S}^-_{\mathbb{A}_n}); \ \mathcal{S}^+_{\mathbb{B}^P} := \mathcal{S}_{\text{polar}}(\mathcal{S}^+_{\mathbb{B}_n})$$
$$\mathcal{S}^+_{\Theta} := \mathcal{S}^+_{\mathbb{B}^P} \ominus \mathcal{S}^-_{\mathbb{A}^P}$$

As rotation solutions forms a cyclic group, $\Theta$ is either discrete or the whole space, justifying the lack of $\mathcal{S}^-_{\Theta}$ and ending the approach.

## References

[**1**] P.J. Besl and N.D. McKay, Method for registration of 3-D shapes, *Sensor Fusion IV: Control Paradigms and Data Structures* vol 1611, 586–606, 1992.

[**2**] L. Jaulin and B. Desrochers, Thick Separators, *Decision Making under Constraints* vol 276, 125–131, 2020.

[**3**] B. Desrochers, *Simultaneous localization and mapping in unstructured environments : a set-membership approach*, 2018.

# Interval Methods for the GPU

Lorenz Gillner and Ekaterina Auer

Department of Electrical Engineering and Computer Science
University of Applied Sciences Wismar
Philipp-Müller-Straße 14, 23966 Wismar, Germany
lorenz.gillner@hs-wismar.de

**Keywords:** GPGPU, Programming, Interval Software, Optimization

## Interval libraries: beyond sequential workloads

Since many decades, interval methods have been used in the context of scientific computing for obtaining verified solutions to many problems featuring bounded uncertainty from such diverse areas as computer graphics or engineering. Over 20 libraries for interval arithmetic alone are available today in any of the major programming languages. With the advent of multi-core processors, the parallelization of interval methods using well established libraries such as C-XSC, Profil/Bias or Boost::Interval has been tested in multi-threaded and distributed systems [1,2,3]. However, while high-performance computing industry is moving towards developing specialized hardware to accelerate repetitive tasks, interval libraries have yet to properly adapt to specific co-processors, for example, graphics processing units (GPUs), data processing units (DPUs) or field-programmable gate arrays (FPGAs).

In recent years, general-purpose graphics processing units (GPGPUs) have inspired much interest among the scientific community, in part because of their low cost and high availability compared to conventional supercomputers or large-scale CPU clusters. The application of GPUs for computations with the focus on uncertainty has been investigated, for example, in [4,5]. Nonetheless, popular interval libraries cannot typically be used with the accelerator devices directly, reasons

for which are, on the one hand, the heterogeneity of application programming interfaces (APIs) across different hardware vendors and, on the other hand, the co-processors' architecture differing from that of conventional CPUs. Notable examples of custom interval libraries for the GPU are given in [6,7]. A very promising candidate to overcome the necessity to adapt existing libraries to hardware APIs is the Julia language [8]. In C++, the API adaptation problem is still not fully solved so that the corresponding software for interval arithmetic or algorithmic differentiation is quite rudimentary or not accessible. Here, to use the Compute Unified Device Architecture (CUDA) technology provided by NVIDIA is one possibility to port interval methods to NVIDIA-manufactured GPUs.

## Parallelization by example

In this talk, we study what is currently possible in the area of interval computations on the GPU using a parallel implementation of a simple yet effective parameter identification scenario. We consider an initial value problem for $m$ ordinary differential equations with $n$ unknown but bounded parameters $\mathbf{p} \in \mathbb{IR}^n$ having the closed-form exact solution $\hat{y}$ for which measurement data $y_{t_i}$ are available in time points $t_i \in [0, t]$. The parameters closest to the measured data can be identified by minimizing the least squares error

$$\Phi(\mathbf{p}_k) = \sum_{t_i=0}^{t} \sum_{j=1}^{m} (y_{t_i,j} - \hat{y}_j(t_i, \mathbf{p}_k))^2 \ ,$$

where $\mathbf{p}_k$ are sub-boxes obtained by bisecting $\mathbf{p}$ until a pre-defined maximum width $w$ so that $\forall \mathbf{p}_k \in \mathbf{p} : diam(\mathbf{p}_k) \leq w$ and $\mathbf{p} = \mathbf{p}_1 \cup \mathbf{p}_2 \cup \cdots \cup \mathbf{p}_N$. Out of these $N$ boxes, feasible ones can be computed, for example, by using the monotonicity test from [9]. The convex hull of the best boxes $\mathbf{p}^*$ can be then employed to find an enclosure of the solution $\hat{y}(t_i, \mathbf{p}^*)$. Since every box is tested individually, this scenario turns out to be "perfectly parallelizable" and is suitable for the execution on the GPU.

We compare two implementations of the parallel monotonicity test described above: a high-level version written in Julia using the `IntervalArithmetic.jl` and `CUDA.jl` packages and an implementation based on a CUDA-native interval library in C++ [6]. Results are evaluated in terms of computation speed, accuracy (i.e., enclosure widths) and ease of use for programmers. To conclude the talk, we take a look at the possible future of interval analysis on the GPU from the point of view of modern programming environments with emphasis on interactivity.

## References

[**1**] M. Zimmer, Using C-XSC in a Multi-Threaded Environment, *Universität Wuppertal Preprint*, 2011.

[**2**] M. Grimmer, An MPI Extension for the Use of C-XSC in Parallel Environments, *Universität Wuppertal Preprint*, 2011.

[**3**] M. Pilarek, R. Wyrzykowski, Solving Systems of Interval Linear Equations in Parallel Using Multithreaded Model and "Interval Extended Zero" Method, *PPAM 2011*, 206-214, 2012.

[**4**] E. Auer, A. Rauh, J. Kersten, Experiments-based parameter identification on the GPU for cooperative systems, *Journal of Computational and Applied Mathematics* 26, 532–540, 2020.

[**5**] G. Rebner, M. Beer, CUDA Accelerated Fault Tree Analysis with C-XSC, *SUM 2012*, 539-549, 2012.

[**6**] S. Collange, M. Daumas, D. Defour, Interval Arithmetic in CUDA, *GPU Computing Gems Jade Edition* (chap. 9), 99-107, 2012.

[**7**] G. Kozikowski, B. Kubica, Interval Arithmetic and Automatic Differentiation on GPU Using OpenCL, *Applied Parallel and Scientific Computing*, 489-503, 2013.

[8] D. Sanders, V. Churavy, Branch-and-bound interval methods and constraint propagation on the GPU using Julia, *SCAN 2020*, 66, 2021.

[9] W. Tucker, *Validated Numerics: A Short Introduction to Rigorous Computations*, Princeton University Press, United Kingdom, 2010.

# Acceleration of contractor algebra on RISCV in the context of mobile robotic

Filiol Pierre[1], Jaulin Luc [1], Le Lann Jean-Christophe [1], Bollengier Theotime [1]

[1] ENSTA Bretagne, Lab-STICC
Brest, France

**Keywords:** Interval arithmetic, Contractor algebra, RISCV, FPGA

## Introduction

Interval arithmetic and contractor algebra are widely used in robotic as they allow to tackle recurrent tasks like localisation and parameter estimation in an efficient way. The interval community can rely on some well known software libraries (ibex, gaol, mpfi ...) which perform accurate forward and backward contractions to solve most of the common problems. However, such libraries always revolve around the use of low-level maths and floating point unit (FPU) functions that are inherently dependent of the compilation platform and the execution environment. This becomes problematic in the area of mobile robotic where the code is often run on exotic embedded systems which may not support them at all or produce incorrect behaviours [1]. This area has also its own constraints when using interval arithmetic. The precision of the computation is often less of an issue compared to being fast, guaranteed (even when pessimistic) and energy-efficient. Most interval algorithms based on contractor algebra expose some data and instruction parallelism. An hardware implementation specifically tailored for the need of mobile robotic could be a solution to deal with the aforementioned limitations.

## Scope of the presentation

The RISCV architecture is an open ISA (instruction set architecture) that has gained a wide popularity over the recent years [2]. It offers multiple standard extensions from which a chip manufacturer can chose to build a ASIP (application specific instruction set processor).

The main focus of the presentation is to demonstrate the feasability of adding interval-specific instructions to the RISCV instruction set. The design proposed in this paper is based on standard RISCV extensions I,M,F,D to have access to native integer/float/double support. We also added a custom-made extension called *xinterval* which accelerates all the interval-related computations. The paper explains how a riscv emulator was implemented to run mobile robotic applications making use of *xinterval* instructions as well as providing benchmarking tools.

### Methodology

The following instructions have been chosen as a basis to create the *xinterval* extension :

- convert/move instructions between 32 bit words and intervals.
- arithmetic forward and backward contractors (add, sub, ...)
- math function forward and backward contractors (log, exp, cos, sin, sqrt, sqr ...)
- union/intersections

The RISCV toolchain has been modified for the aforementioned instructions to be recognized as valid assembly instructions.

```
// inline function to allow the use of xinterval instruction addfwctc
// This instruction add 2 intervals stored in double registers and stores the result in a double register
inline interval __attribute__((always_inline)) _addFwCtc(interval itv1, interval itv2) {
  interval result;
  asm("addfwctc %0,%1,%2" : "=f"(result) : "f"(itv1), "f"(itv2));
  return result;
}
```

Listing 1: "Using the forward addition contractor in C"

Lastly, a RISCV emulator implementing I,M,F,D standard extensions and *xinterval* custom extension has been designed in order to run guest C interval applications using our custom instructions.

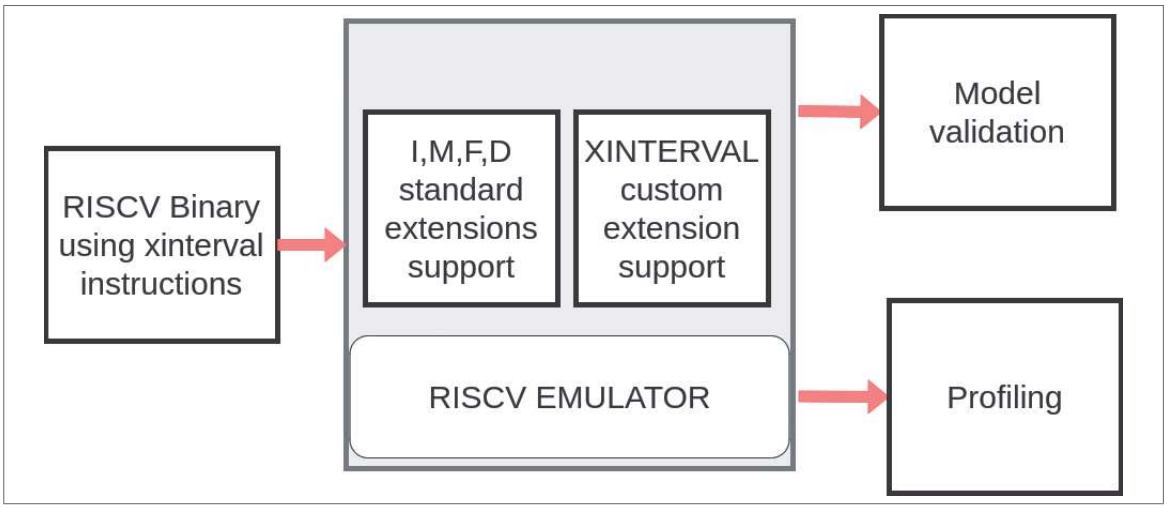



**Figure 1:** Design workflow

A test application linked to the area of mobile robotic has been written in order to evaluate our *xinterval* extension. The aim is to compute the set $\mathbb{S}$ of all the possible values for a robot position which is at a measured distance $r$ of a landmark with known location $(c_x, c_y)$.

$$\mathbb{S} = \{(x, y) \in \mathbb{R} \mid (x - c_x)^2 + (y - c_y)^2 \in [r]\} \quad (1)$$

Using a separator $S$ for $\mathbb{S}$ and the SIVIA (Set Inversion Via Interval Analysis) algorithm we are able to draw a paving for possible positions [3]. This produces the following figure when run from inside the emulator :

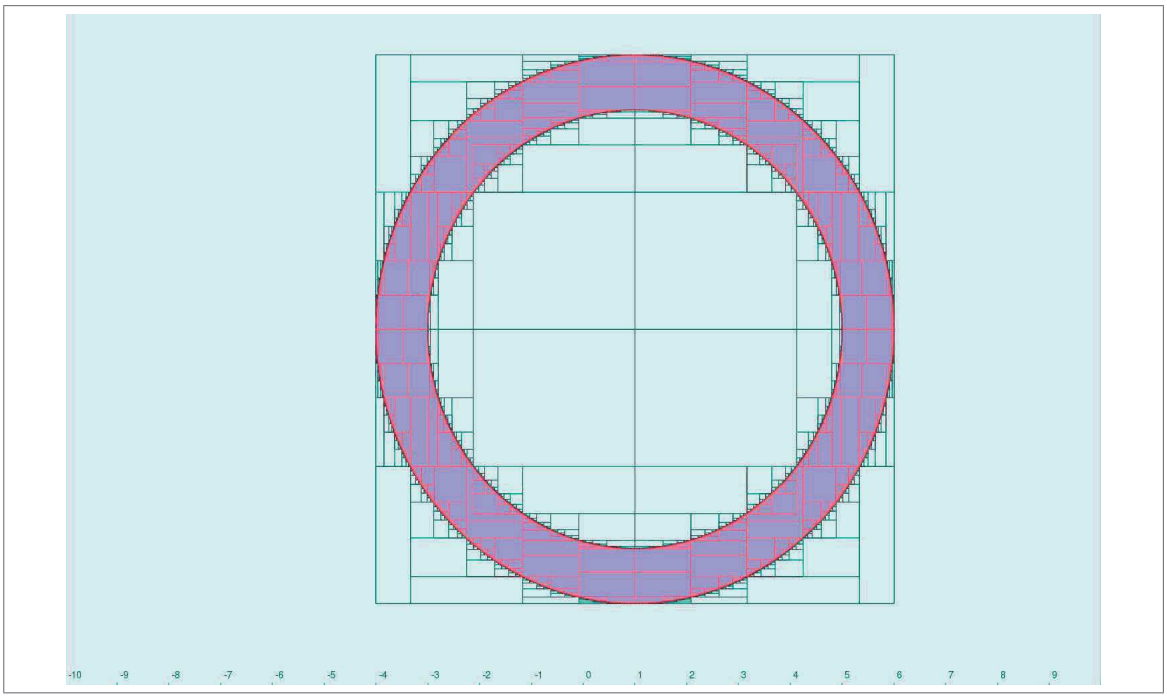



**Figure 2:** Paving for $\mathbb{S}$ obtained by using *xinterval* instructions

## Future works

The goal of our future work is to find the best compromises for an hardware implementation of the interval arithmetic as there are a lot of interesting trade-off to consider. We first plan to study the interval FPU model which can be largely optimised in the spectrum of mobile

robotic. For example, it is unnecessary to implement all the rounding modes that a traditional x86 FPU typically offers [4]. Using only RoundDown for lower bound and RoundUp for upper bound may be enough to remain guaranteed. Such simplification may save us a lot of clock cycles and hardware resources. The platform presented in this paper will allow us to monitor how each trade-off impact the overall execution speed and resource utilisation.

# References

[1] P. Filiol, T. Bollengier, L. Jaulin, and J.-C. Le Lann, "A new type of intervals for solving problems involving partially defined functions," in *SWIM'22, Hannover, Germany*, 2022.

[2] A. W. David Patterson, *The RISC-V Reader: An Open Architecture Atlas*. Strawberry Canyon, 2017. ISBN 978-0999249116.

[3] O. D. E. W. Luc Jaulin, Michel Kieffer, *Applied Interval Analysis*. Springer London Ltd, 2001. ISBN 9781447110675.

[4] J.-M. Muller, N. Brunie, F. de Dinechin, C.-P. Jeannerod, M. Joldes, V. Lefèvre, G. Melquiond, N. Revol, and S. Torres, *Handbook of Floating-Point Arithmetic, 2nd edition*. Birkhäuser Boston, 2018. ACM G.1.0; G.1.2; G.4; B.2.0; B.2.4; F.2.1., ISBN 978-3-319-76525-9.

# Hybrid Interval-Probabilistic Localization in Urban Environments

Aaronkumar Ehambram[1], Luc Jaulin[2], Simon Rohou[2] and Bernardo Wagner[1]

[1] Leibniz Universität Hannover, Real Time Systems Group (RTS)
Appelstraße 9A, 30167 Hannover, Germany
{ehambram,wagner}@rts.uni-hannover.de
[2]ENSTA Bretagne, Lab-STCC, Brest, France
lucjaulin@gmail.com, simon.rohou@ensta-betagne.fr

**Keywords:** Autonomous Driving, Localization in Building Maps, Hybrid Interval-Probabilistic Methods

## Introduction

With the rise of highly automated vehicles, safety needs to be assured. This requires the assessment of the localization uncertainty. Two predominant approaches in the literature, probability and set-membership theory, offer mathematical tools for the uncertainty assessment. Probabilistic approaches yield accurate point-valued results but may underestimate the uncertainty, while set-membership approaches reliably estimate the uncertainty but tend to be overly pessimistic without providing point-valued results. In the scope of this work, a novel **H**ybrid **P**robabilistic- **a**nd **S**et-Membership-based **Co**arse and **Re**fined (HyPaSCoRe) Localization method is introduced which enables safe operation while not being overly pessimistic regarding the uncertainty estimation by combining both approaches. This method localizes a robot in a building map in real-time and considers two types of hybridizations. First, set-membership approaches are used to robustify and control probabilistic approaches. Second, probabilistic approaches are used to reduce the pessimism of set-membership approaches by augmenting them with further probabilistic constraints.

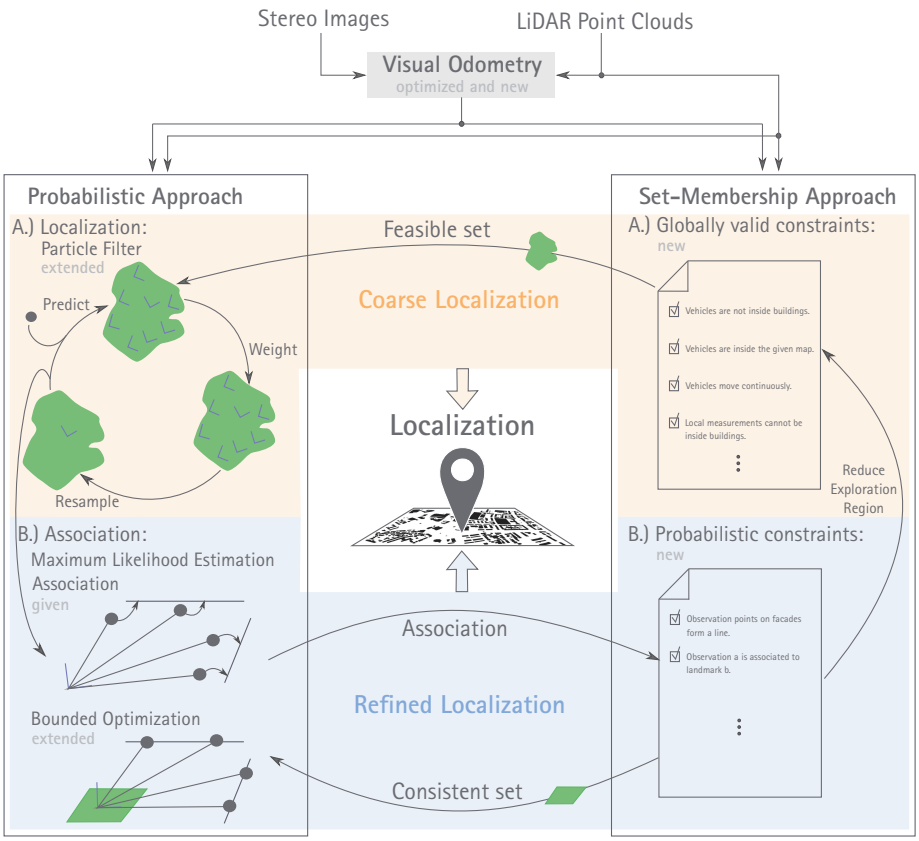



Figure 1: Method overview to the HyPaSCoRe Localization pipeline.

## Method

The HyPaSCoRe Localization consists of three modules – visual odometry, coarse localization, and refined localization (cf. Fig. 1). The method uses a stereo camera system, a LiDAR sensor, and GNSS data, focusing on localization in urban canyons where GNSS data can be inaccurate. The visual odometry module computes the relative motion of the vehicle. The coarse localization, visualized in orange in Fig. 1, first narrows down the feasible set of poses using globally valid but less restrictive constraints. Using a probabilistic particle filter approach, the module also provides the most likely poses inside the feasible set. The refined localization module, visualized in blue in Fig. 1, improves the uncertainty estimation by selecting the best-fitting particle from the coarse localization results. Employing the particle, the local Li-

DAR data is associated with building facades on the map. Those associations provide probabilistic constraints that determine a consistent set of poses, which provides bounds for a modified bounded optimization approach. This approach identifies the most likely pose within the consistent set, preventing significant divergence in the optimization by constraining the solution space. [1]
As indicated by the right arrow in Fig. 1, in the case of high reliability of the consistent set, we contract the feasible set to the consistent set to reduce the pessimism of the uncertainty estimate. From the HyPaSCoRe Localization, we obtain a feasible set as the uncertainty estimation and the most likely pose as the best point-valued localization result. The HyPaSCoRe Localization is real-time capable and is operable in different environments as long as enough buildings are visible.
The experimental evaluation based on author-collected and the KITTI datasets [2] shows that the HyPaSCoRe Localization maintains the integrity of the uncertainty estimation while providing accurate, most likely point-valued solutions in real-time. The implementation is publicly available on GitHub:
`https://github.com/AaronEhambram/hypascore_localization`

**Acknowledgement**

This work was supported by the German Research Foundation (DFG) as part of the Research Training Group i.c.sens [RTG 2159].

## References

[**1**] A. Ehambram, L. Jaulin and B. Wagner, Hybrid Interval-Probabilistic Localization in Building Maps, *IEEE Robotics and Automation Letters* vol. 7, no. 3, pp. 7059–7066, 2022.

[**2**] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, Vision meets robotics: The KITTI dataset, *The International Journal of Robotics Research*, vol. 10, no. 11, pp. 1231–1237, 2013.

# Union of adjacent contractors

Quentin Brateau[1], Fabrice Le Bars [1] and Luc Jaulin[1]

[1]ENSTA Bretagne, UMR 6285, Lab-STICC, IAO, ROBEX
2 rue François Verny, 29806 Brest CEDEX 09, France

quentin.brateau@ensta-bretagne.org
fabrice.le_bars@ensta-bretagne.org
lucjaulin@gmail.com

**Keywords:** Set union, Contractor programming, Geometric contractors, Localization

## Introduction

Set theory provides a fundamental structure for interval analysis, which must conform to its formalism [1,2]. Trivial operations are defined on sets, such as union, intersection, deprivation, cartesian product, and projection. They should be applicable to intervals and therefore to contractors.

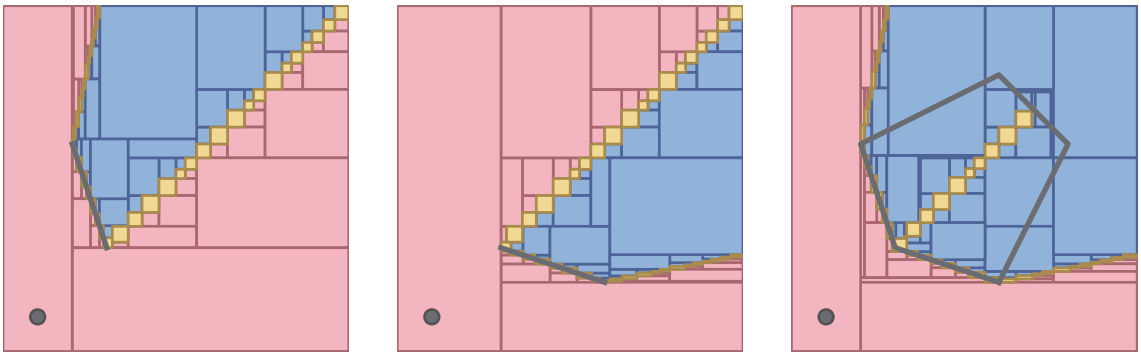
In the case of union of adjacent contractors, typical paving algorithm bisects unnecessarily the boxes and then reveals the common boundary between the two sets as shown in Figure 1. This behavior noticed on contractors union is not consistent with the set union as defined in set theory.

## Geometric contractors

Geometric contractors are a class of contractors based on geometric constraints, particularly used for localization in robotics [2]. They can be used to characterize all possible robot states based on measurements. Geometric contractors are often defined for segments, which

are one of the most simple geometric shapes. Then, by using set operators, more complex contractors can be built, such as contractors based on polygons [3].

By defining more complex contractors in this way, adjacent boundary-overlapping sets appear at each vertex. Figure 1 shows the paving of a visibility separator from a point, implemented by Rémy Guyonneau [3]. The visibility separator works well on two individual segments but fails to characterize inner subpaving when dealing with polygons.



(a) Visibility separator with a segment obstacle

(b) Visibility separator with a segment obstacle

(c) Visibility separator with a polygon obstacle

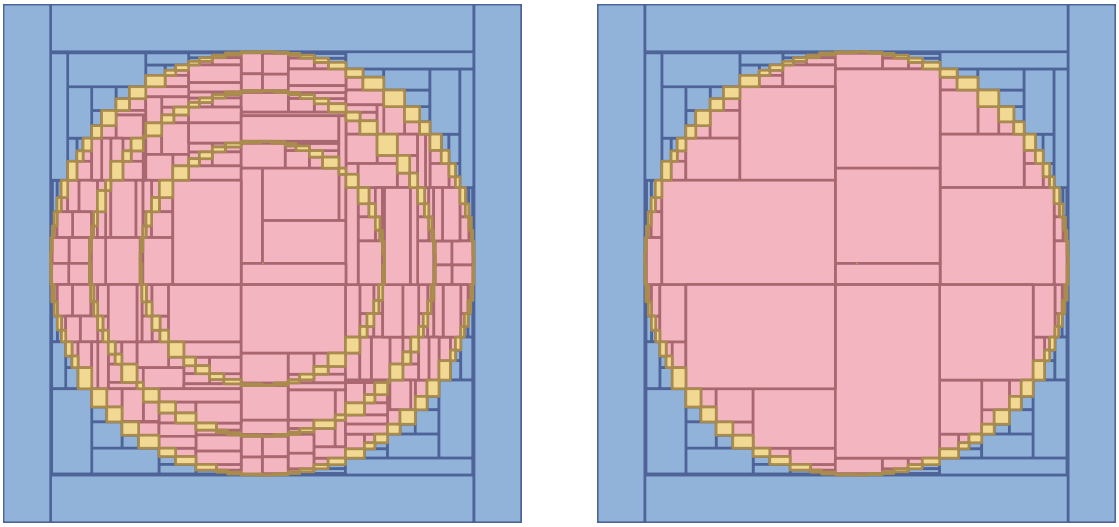
Figure 1: Visibility separator for a point relative to obstacles. In red the set visible from the point, in blue the invisible set, in yellow the uncertain set.

## Main results

Solutions can be found for geometric contractors to prevent the apparition of this common boundary. These solutions rely on contractor-specific solutions to avoid the use of contractors union when building more complex contractors, and some of these will be presented.

However, this problem lays the foundations for a larger issue of the characterization of the union of two adjacent contractors in the general case, for which a solution has not yet been found. Figure 2

shows a paving of the union of adjacent contractors on rings, on which the common boundary is appearing.



(a) Union of three adjacent ring separators

(b) Expected union of three adjacent ring separators

Figure 2: Union of adjacent boundary-overlapping separators. In red the inner set, in blue the outer set, in yellow the uncertain set.

## References

[**1**] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter, Applied Interval Analysis: With Examples in Parameter and State Estimation, Robust Control and Robotics. Springer Science & Business Media, 2012.

[**2**] L. Jaulin and B. Desrochers, "Introduction to the algebra of separators with application to path planning," Engineering Applications of Artificial Intelligence, vol. 33, pp. 141–147, Aug. 2014

[**3**] R. Guyonneau, "Méthodes ensemblistes pour la localisation en robotique mobile," These de doctorat, Angers, 2013.

# Manifold decomposition for interval analysis

Arthur Ignazi[1], Nicolas Delanoue[1], Remy Guyonneau[1] and Sébastien Lagrange [1]

[1] Polytech Angers, Université d'Angers
62 Avenue Notre-Dame du Lac, France
{arthur.ignazi, nicolas.delanoue, remy.guyonneau, sebastien.lagrange}@univ-angers.fr

**Keywords:** Interval analysis, Atlas, Manifold

## Introduction

The concept of a manifold is central to modern mathematical robotics [1].

The charts notion allows to generalize properties on Euclidean spaces to bstract manifolds: Smooth functions, vector fields, differential forms [2] ... Most interval analysis algorithms [3] [4] consider boxes which are Euclidean objects.

This paper presents a novelty approach to create charts that are compatible with box notion. According to this work, every interval based methods can be applied to the considered manifolds.

## Manifolds

**Definition 1** (Manifold)**.** *A manifold is a second countable Hausdorff space that is locally homeomorphic to Euclidean space.*

It exists three approaches to deal with manifolds: charts, submanifolds of $\mathbb{R}^n$ and quotient spaces [2].

We want to build an atlas of this manifold, so that we can solve problems on it using interval analysis, for example with set inversion problem (with SIVIA [4]), counting the solutions of an equation (with

newton interval algorithm), etc. Therefore, we propose to derivate a classical notion of atlas to box atlas.

**Definition 2** (Atlas). *An atlas of a manifold $M$ is an indexed family $\{(U_i, \varphi_i) : i \in I\}$ of charts on $M$ which covers $M$ (that is $\bigcup_{i\in I} U_i = M$). It is completed with transition maps $\tau_{ij}$ between two intersected charts $V_i$ and $V_j$, where $V_i = \varphi_i(U_i)$ and $V_j = \varphi_j(U_j)$. The transition map $\tau_{ij} : \varphi_i(U_i \cap U_j) \to \varphi_j(U_j \cap U_i)$ is the map defined by $\tau_{ij} = \varphi_j \circ \varphi_i^{-1}$.*

**Definition 3** (Box atlas). *A box atlas is an atlas $\{\varphi_i : U_i \to V_i \subset \mathbb{R}^n\}_{i\in I}$ of a manifold $M$, such that $\forall i \in I, V_i = \varphi_i(U_i)$ is a box.*

**Example 1** (Torus). *The torus is a representation of the cartesian product of two circles $T = S^1 \times S^1$. On figure 1 is the torus $T$ split on 4 maps. Each map is a box, where we can do interval analysis computations. As we have all functions $\varphi_i$, we can build a box atlas of the torus.*

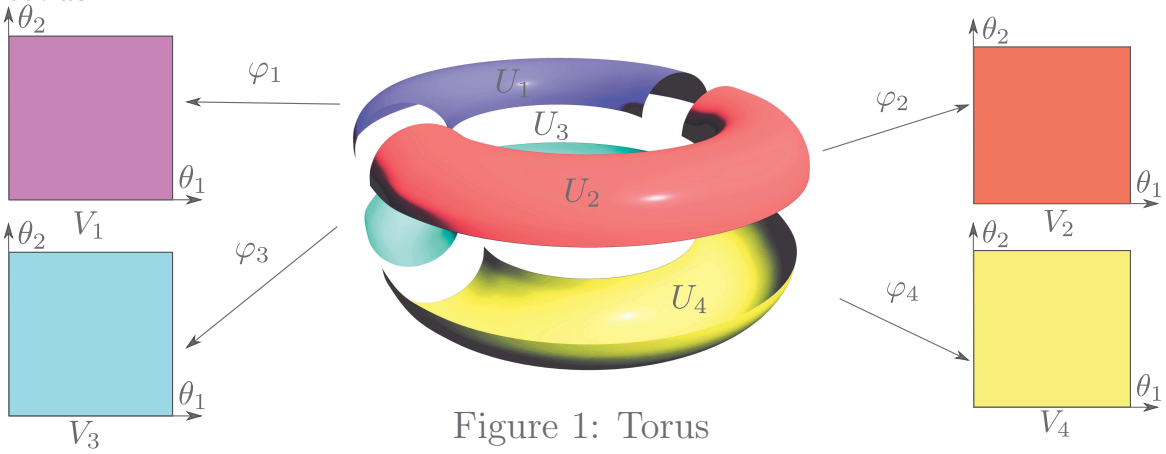



Figure 1: Torus

## Robotic application

In this example, the 2R robot is considered. This robot has two revolute joints. Each coordinate joint belongs to $[0, 2\pi]$ quotiented by the relation $0 = 2\pi$. Therefore, the configuration space of this robot is the Cartesian product of two circles, that is to say the torus $T$. Let us consider the environment composed with just one obstacle as depicted in figure 2.

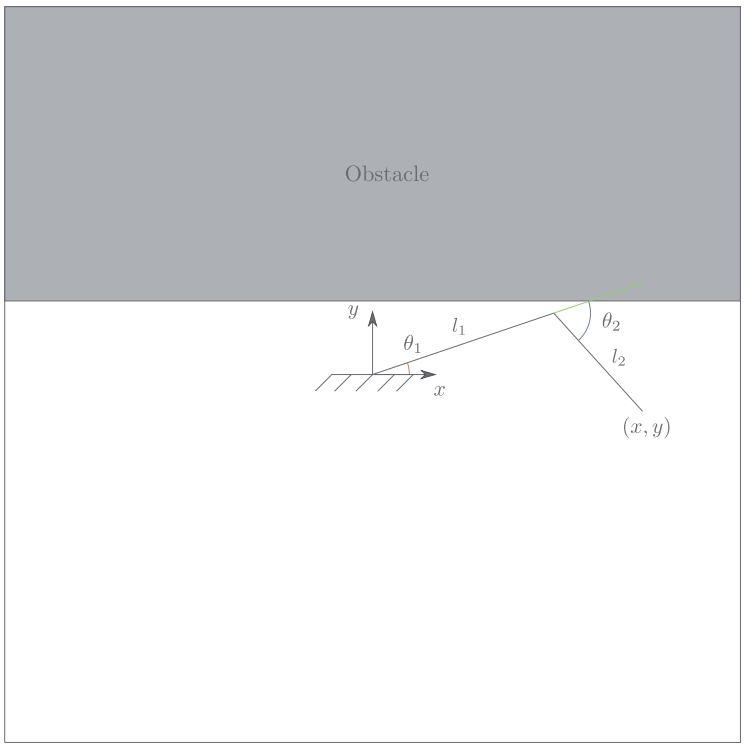



Figure 2: Robot with obstacle

The geometrical model of this robot is described by the following functions:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} l_1 cos(\theta_1) + l_2 cos(\theta_1 - \theta_2) \\ l_1 sin(\theta_1) + l_2 sin(\theta_1 - \theta_2) \end{pmatrix}. \tag{1}$$

The free configuration space is a subset of the torus. It represents all the angles for which the robot does not intersect the obstacle. This free configuration space can be characterized by applying the SIVIA algorithm on each chart of a torus box atlas. The constraints $\{g(\boldsymbol{\theta}) \leq 0, \boldsymbol{\theta} \in T\}$ related to the obstacle expressed on the torus are rewritten on each chart according to

$$\{g(\varphi_i^{-1}(v_i)) \leq 0, v_i \in V_i\}. \tag{2}$$

Figure 3 presents the result of the SIVIA algorithm applied to a torus box atlas composed of 4 charts.

As said on definition 3, a box atlas is a set of charts $V_i$ and functions $\varphi_i$, and a set of transition functions $\tau_{ij}$. The charts can be seen on figure 3. The functions are the following :

$$\begin{cases} \varphi_1 : & [0,\pi] \times [0,\pi] & \to [0,1] \times [0,1] \\ & \theta_1, \theta_2 & \mapsto \theta_1/\pi, \theta_2/\pi \\ \varphi_2 : & [\pi,2\pi] \times [0,\pi] & \to [0,1] \times [0,1] \\ & \theta_1, \theta_2 & \mapsto (\theta_1 - \pi)/\pi, \theta_2/\pi \\ \varphi_3 : & [0,\pi] \times [\pi,2\pi] & \to [0,1] \times [0,1] \\ & \theta_1, \theta_2 & \mapsto \theta_1/\pi, (\theta_2 - \pi)/\pi \\ \varphi_4 : & [\pi,2\pi] \times [\pi,2\pi] & \to [0,1] \times [0,1] \\ & \theta_1, \theta_2 & \mapsto (\theta_1 - \pi)/\pi, (\theta_2 - \pi)/\pi \end{cases} \tag{3}$$

$\tau$ generic : $\tau_{ij} : \varphi_i(U_i \cap U_j) \to \varphi_j(U_j \cap U_i)$ is the map defined by $\tau_{ij} = \varphi_j \circ \varphi_i^{-1}$

In this 4 charts example, let's take one example of transition map $\tau$ :

$$\begin{aligned} \tau_{12} : [0,1] \times \{0,1\} &\to [0,1] \times \{0,1\} \\ \theta_1, \theta_2 &\mapsto \theta_1, |\theta_2 - 1| \end{aligned}$$

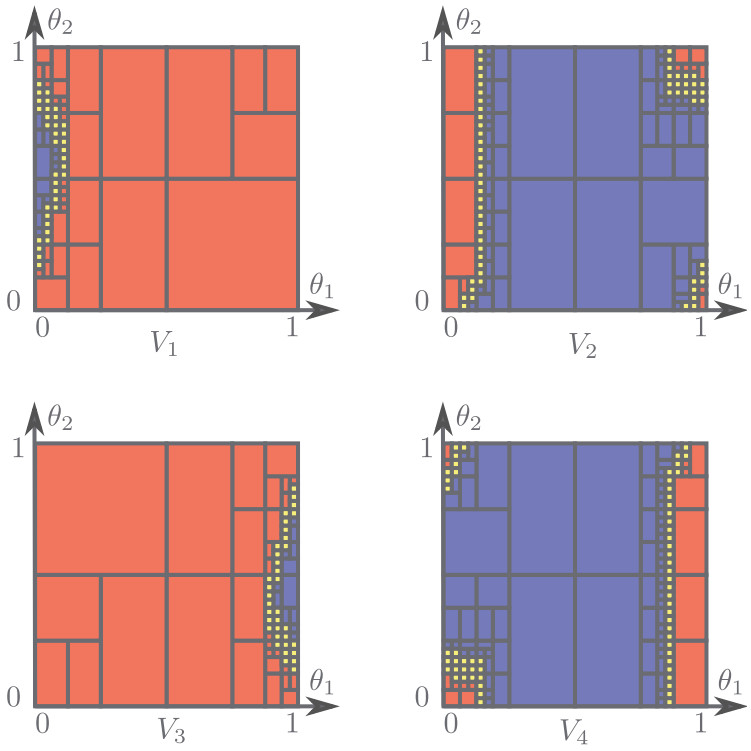



Figure 3: Sivia applied on a torus box atlas composed of 4 charts

Let us consider the graph whose nodes are the red boxes and edges exists between two nodes if the corresponding red boxes are adjacent. Two boxes $B_1 \subset V_1$ and $B_2 \subset V_2$ are adjacent if $\varphi_1^{-1}(B_1) \cap \varphi_2^{-1}(B_2) \neq \emptyset$.

With this representation, we can solve a path planning problem, running the Dijkstra algorithm on this graph [5], or we can compute the number of connected components. For this example, the graph only has one connected component. It can be easily seen on the torus representation figure 4 or the charts representation figure 3. On a classical representation, we could consider three components.

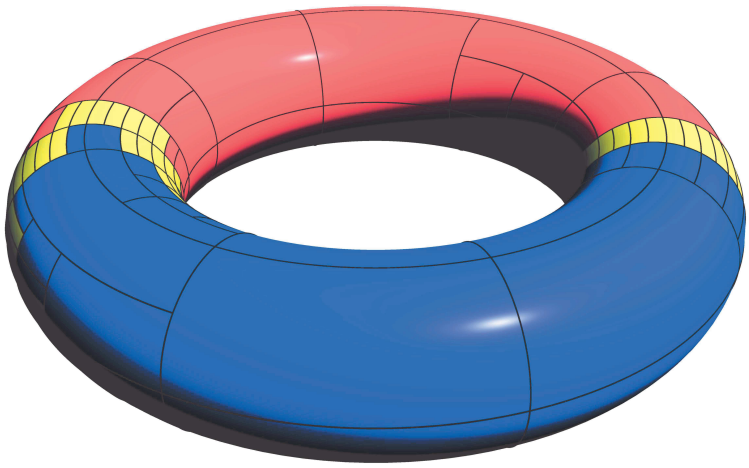


Figure 4: SIVIA on a torus

## References


[1] J. M. Selig, *Geometric fundamentals of robotics*, Springer,vol. 128, 2005.

[2] L. Tu, On the Fundamental Electromagnetic Equations for Moving Bodies, *Annalen der Physik* (ser. 4) 26, 532–540, 1908.

[3] L. Jaulin, M. Kieffer, O. Didrit, and É. Walter, *Interval Analysis*, Springer London, 2001, pp. 11-43.

[4] L. Jaulin and E. Walter, *Set inversion via interval analysis for nonlinear bounded-error estimation*, Automatica, vol 29, no. 4, pp. 1053-1064, 1993.

[5] L. Jaulin, *Path planning using intervals and graphs*, Reliable Computing, vol. 7, no. 1, pp. 1–15, 2001.

# Brunovsky decomposition for dynamic interval localization

Simon Rohou[1], Luc Jaulin[1]

[1] ENSTA Bretagne, Lab-STICC, UMR CNRS 6285, Brest, France
simon.rohou@ensta-bretagne.fr

**Keywords:** non-linear system, localization, differential equations

## Introduction

We present a new set-membership method [1] for estimating the trajectories of dynamical systems $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$, when the states are completely unknown and only observations $\mathbf{y}$ are available under the form of $\mathbf{y} = \mathbf{g}(\mathbf{x})$, with $\mathbf{f}$ and $\mathbf{g}$ non-linear functions.

When the states are completely unknown, conventional methods such as Kalman filters run into difficulties, as it is difficult to find a linearization point, or to perform prediction steps. Particle filters will employ algorithms with high complexity without ensuring a reliable convergence. In contrast, the use of set-membership approaches avoids the need for linearization and is more suited to large uncertainties by not removing consistent solutions. They will however badly behave in our context considering that the states are completely unknown. In order to overcome this problem, interval methods from state-of-the-art apply some branch-and-prune techniques such as shaving methods for reducing the state sets by performing bisections. The counterpart is obviously the increasing complexity of these algorithms. As a consequence, the current existing tools are not sufficient for addressing the considered problem both in a reliable and an efficient way.

Our contribution is to tackle this state estimation efficiently without performing bisections. This can be achieved by rewriting the system into a Brunovsky form.

## First part: symbolic Brunovsky decomposition

The first part of the proposed method is symbolic and follows the decomposition of Brunovsky [2], *i.e.*, it rewrites the set of differential equations into two blocks of constraints: one block gathers non-linear equations that do not involve differential operators, while the other block is composed of linear chains of integrators.

For instance, a differential flat system with flat outputs $z_1, \ldots, z_m$ and sensor outputs $\mathbf{y}$, admits the following Brunovsky decomposition:

$$\begin{cases} \mathbf{y} = \mathbf{g}(\mathbf{x}) \\ \begin{pmatrix} z_1 \\ \dot{z}_1 \\ \vdots \\ z_m^{(\kappa_m)} \end{pmatrix} = \lambda \begin{pmatrix} \mathbf{x} \\ \mathbf{u} \end{pmatrix} \\ z_1^{(\kappa_1)} \xrightarrow{\int} \cdots \xrightarrow{\int} \dot{z}_1 \xrightarrow{\int} z_1 \\ \vdots \\ z_m^{(\kappa_m)} \xrightarrow{\int} \cdots \xrightarrow{\int} \dot{z}_m \xrightarrow{\int} z_m \end{cases} \tag{1}$$

The first block with non-linear equations and no differential relations corresponds to the functions $\lambda$ and $\mathbf{g}$. The second block is only made of chains of integrators, for which an optimal operator will be at hand.

## Second part: contractor approach

The second part of the method is numerical and based on a contractor method. It relies on the previous decomposition and encloses the variables into boxes and tubes. Then, contractor operators are used for narrowing the sets of feasible solutions. In particular, a new contractor is provided for dealing with the chains of integrators, that gather all the differential aspects of the dynamical system.

## Application to robot localization

A robot measures some distances to known landmarks, in addition to known inputs $\mathbf{u}$ of the system, but without any prior knowledge about the states. This problem is known to be difficult to solve, and methods from state-of-the-art usually come into bisection procedures of the heading and position values [3], which implies a strong computation burden. We are able to provide a bounded estimate of the trajectory of the states, by using contractors and without performing bisections.

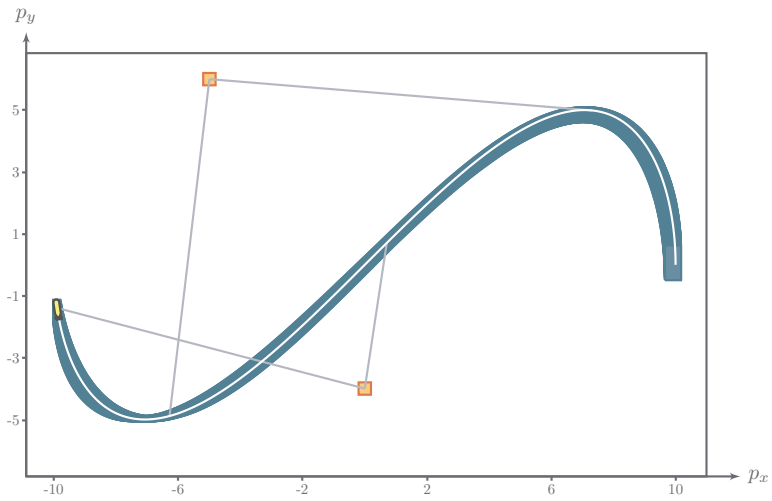



Figure 1: Guaranteed computation of a tube enclosing the feasible trajectories of a robot measuring bounded distances from two landmarks, without prior knowledge about its states (positions, velocity, heading).

## References

[1] S. Rohou and L. Jaulin, Brunovsky decomposition for dynamic interval localization, *IEEE Trans. on Automatic Control*, 2023.

[2] P. Brunovsky, A classification of linear controllable systems, *Kybernetika*, vol. 6, no. 3, pp. 173–188, 1970.

[3] Z. Wang, Guaranteed localization and mapping for autonomous vehicles, *Ph.D. dissertation*, 2018.

# Wednesday June 28

# A contractor is minimal for narrow boxes

Luc Jaulin

Robex, Lab-STICC, ENSTA-Bretagne, Brest, France lucjaulin@gmail.com

**Keywords:** Interval analysis, Contractors, Centered form

## Introduction

Interval analysis is an efficient tool used for solving rigorously complex nonlinear problems involving bounded uncertainties [1] [2] [3]. Many interval algorithms are based on the notion of *contractor* [5] which is an operator which shrinks an axis-aligned box $[\mathbf{x}]$ of $\mathbb{R}^n$ without removing any point of the solution set $\mathbb{X}$. Combined with a paver which bisects boxes, the contractor builds an outer approximation of the set $\mathbb{X}$. The resulting methodology can be applied in several domains of engineering such as localization [7], SLAM [8] [9], reachability [10], etc.

This paper proposes a new interval-based contractor for nonlinear equations which is minimal for narrow boxes. The method is based on the centered form combined with a Gauss Jordan band diagonalization preconditioning.

# 1 Illustration

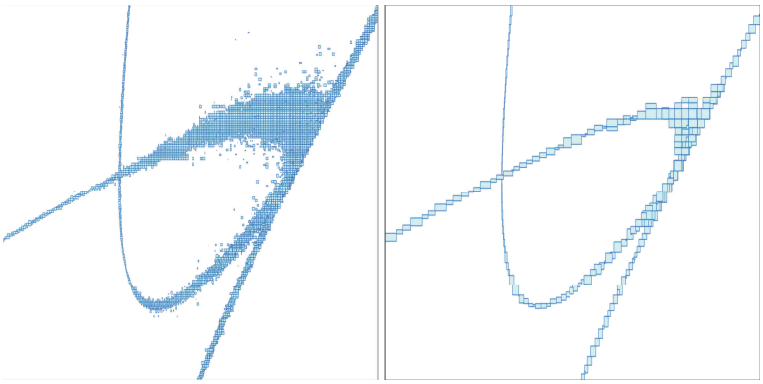
Consider the set

$$\mathcal{P} = \{\mathbf{p} \,|\, \exists \omega > 0,\, \mathbf{f}(\mathbf{p}, j\omega) = 0\}. \tag{1}$$

where

$$\mathbf{f}(p_1, p_2, j\omega) = \begin{pmatrix} -\omega^2 + 2\omega \sin(\omega p_1) + \cos(\omega p_2) \\ 2\omega \cos(\omega p_1) - \sin(\omega p_2) \end{pmatrix} \tag{2}$$

Take $[p_1] = [0, 2.5]$, $[p_2] = [1, 4]$,$[\omega] = [0, 10]$ and let us characterize the set $\mathcal{P}$. Using a branch and prune algorithm with a accuracy of $\varepsilon = 2^{-8}$ with an HC4 algorithm [1][12] (the state of the art), we get the paving of Figure 1, left. The number of boxes of the approximation is 43173. Similar results where obtained were obtained on the same example in [13]. With an accuracy of $\varepsilon = 2^{-4}$, our centered contractor yields Figure 1, right. The number of boxes of the approximation is 282 (instead of 43173), for a more accurate approximation.



A video illustrating the asymptotic minimality of the contractor is given at:

`https://youtu.be/nM9rR4jDj74`

# References

[1] M. Cébério and L. Granvilliers, "Solving nonlinear systems by constraint inversion and interval arithmetic," in *Artificial Intelligence and Symbolic Computation*, vol. 1930, LNCS 5202, 2001, pp. 127–141.

[2] V. Kreinovich, A. Lakeyev, J. Rohn, and P. Kahl, *Computational Complexity and Feasibility of Data Processing and Interval Computations.* Springer, 1997.

[3] A. Rauh and E. Auer, *Modeling, Design, and Simulation of Systems with Uncertainties.* Springer, 2011.

[4] G. Chabert and L. Jaulin, "Contractor Programming," *Artificial Intelligence*, vol. 173, pp. 1079–1100, 2009.

[5] R. Guyonneau, S. Lagrange, and L. Hardouin, "A visibility information for multi-robot localization," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013.

[6] M. Mustafa, A. Stancu, N. Delanoue, and E. Codres, "Guaranteed SLAM; An Interval Approach," *Robotics and Autonomous Systems*, vol. 100, pp. 160–170, 2018.

[7] S. Rohou, L. Jaulin, L. Mihaylova, F. L. Bars, and S. Veres, *Reliable Robot Localization.* Wiley, dec 2019. [Online]. Available: https://doi.org/10.1002%2F9781119680970

[8] E. Goubault and S. Putot, "Robust under-approximations and application to reachability of non-linear control systems with disturbances," *IEEE Control. Syst. Lett.*, vol. 4, no. 4, pp. 928–933, 2020. [Online]. Available: https://doi.org/10.1109/LCSYS.2020.2997261

[9] F. Benhamou, F. Goualard, L. Granvilliers, and J. F. Puget, "Revising hull and box consistency," in *Proceedings of the International Conference on Logic Programming*, Las Cruces, NM, 1999, pp. 230–244.

[10] R. Malti, M. Rapaić, and V. Turkulov, "A unified framework for robust stability analysis of linear irrational systems in the parametric space," *Automatica*, 2022, second version, under review (see also https://hal.archives-ouvertes.fr/hal-03646956). [Online]. Available: https://hal.archives-ouvertes.fr/hal-03646956/

# Possible preliminary test for the optimality condition in Interval Branch and Bound method

Mihály Gencsi, Boglárka G.-Tóth

Department of Computational Optimization
Faculty of Science and Informatics, University of Szeged
Szeged, Hungary
{gencsi,boglarka}@inf.u-szeged.hu

**Keywords:** Global Optimization, Interval Arithmetic, Branch and Bound method, Optimality condition, Fritz-John condition

## Introduction

In this study, we focus on solving the constrained nonlinear programming problems with inequality and general bound constraints. We consider the following $n$-dimensional nonlinear problem,

$$\begin{array}{ll} \underset{x \in \boldsymbol{y} \subseteq \mathbb{R}^n}{\text{minimize}} & f(x) \\ \text{subject to} & g_j(x) \leq 0, \quad j \in M_c, \end{array} \tag{1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $g_j : \mathbb{R}^n \to \mathbb{R}, j \in M_c$ are continuously differentiable nonlinear functions, and the interval box $\boldsymbol{y} = [\underline{y}, \overline{y}]$ denotes a general bound constraint , that can be formulated as $p_{i_u}(x) = x_i - \overline{y_i} \leq 0$ and $p_{i_l}(x) = \underline{y_i} - x_i \leq 0$. We search for the global optimum using a guaranteed method, the Interval Branch and Bound (IBB) method. In the IBB method [3, 5], we replace the problem with smaller subproblems. We try to discard a subproblem by computing upper and lower bounds and checking for feasibility or optimality using Interval Arithmetic. Nowadays, several implementations of the IBB can be found in the literature. However, many of them do not use the Fritz-John (FJ)

optimality conditions [4] to discard non-optimal subproblems. This implies solving an interval-valued system of equations. In this study, we investigate the interval FJ optimality conditions from a geometric point of view. More information on interval arithmetic can be found in [2, 1].

## The interval FJ Condition System

The FJ conditions are necessary conditions for a solution to be optimal in nonlinear programming. For problem (1), following [1], the straightforward extension of the FJ optimality conditions for a given box $\boldsymbol{x}$ are the interval-valued system of equations

$$\boldsymbol{\mu}_0 \nabla \mathbf{f}(\boldsymbol{x}) + \sum_{i \in M_b} \boldsymbol{\mu}_i \nabla \mathbf{p}_i(\boldsymbol{x}) + \sum_{j \in M_c} \boldsymbol{\mu}_j \nabla \mathbf{g}_j(\boldsymbol{x}) = 0 \tag{2}$$

$$\boldsymbol{\mu}_i \mathbf{p}_i(\boldsymbol{x}) = 0, \quad i \in M_b \tag{3}$$

$$\boldsymbol{\mu}_j \mathbf{g}_j(\boldsymbol{x}) = 0, \quad j \in M_c \tag{4}$$

$$\boldsymbol{\mu}_i \geq 0, \quad i \in M_b \cup M_c \cup \{0\}, \tag{5}$$

where $\mathbf{f}(\boldsymbol{x})$, $\mathbf{p}_i(\boldsymbol{x})$, $\mathbf{g}_j(\boldsymbol{x})$ are the inclusion functions, $\nabla \mathbf{f}(\boldsymbol{x})$, $\nabla \mathbf{p}_i(\boldsymbol{x})$, $\nabla \mathbf{g}_j(\boldsymbol{x})$ are the inclusions of the gradients of $f(x), p_i(x), g_j(x)$ and $M_b$, $M_c$ are the set of bound constraints and constraints, respectively. Note that we can reduce the number of equations in the system by considering only the active constraints. We consider a constraint to be active if the inclusion of the constraint, $\mathbf{p}_i(\boldsymbol{x})$, $\mathbf{g}_j(\boldsymbol{x})$, contains zero. Let $B$ and $C$ be the set of active bound constraints and active constraints, respectively. We can formalize the equations (2)-(5) for active constraints by replacing $M_b$ with $B$ and $M_c$ with $C$.

## The geometrical interpretation of the FJ conditions

Figure 1 shows the graphical meaning of the optimality condition. In this case there are two active constraints $g_1(x^*)$, $g_2(x^*)$. The feasible area is highlighted. We examine the optimality condition at the point

$x^*$. Graphically, the necessary condition is that $-\nabla f(x^*)$ has to be in the conic hull of the gradients of the active constraints $g_1(x^*)$, $g_2(x^*)$.

In the interval world, instead of the point $x^*$ there is the box $\boldsymbol{x}$, and instead of the gradients there are the enclosures of the gradients. The graphical interpretation of the interval optimality condition is shown in Figure 2. Note that if any enclosure of the gradient of the constraints contains zero inside, then the conic hull is full. So it contains all directions, also the enclosure $-\nabla\mathbf{f}(\boldsymbol{x})$. Moreover, if $0 \in int(\nabla\mathbf{f}(\boldsymbol{x}))$, all directions could be decreasing, so it always intersects with the conic hull. Otherwise, we can create the conic hull and reduce or discard the box $\boldsymbol{x}$ by intersecting the conic hull with the negative enclosure of the objective function.

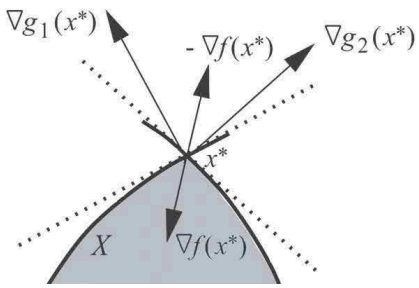



Figure 1: Graphical meaning of the optimality conditions.

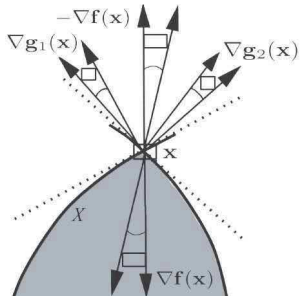



Figure 2: Graphical meaning of the interval optimality conditions.

Instead of directly applying the interval FJ Condition System (2)-(5), we want to build a method which returns as soon as we know that the test cannot succeed.

The main aim of the proposed method is based on the geometric point of view. By intersecting the obtained conic hull with the $-\nabla\mathbf{f}(\boldsymbol{x})$, we decide whether to apply the FJ optimality test or not. If the intersection is empty, no point satisfies the optimality condition, and the box can be rejected. Instead of computing the conic hull, we want to compute its section with a separating hyperplane $h$. If it exists, the intersection is a convex bounded polytope. Next, we have to check that the central projection of $-\nabla\mathbf{f}(\boldsymbol{x})$ onto $h$ is contained in it.

In more detail, we compute the normal vector $v$ of the hyperplane

$h$, which separates the gradients of the constraints from the origin. The existence of such a separating hyperplane is necessary to discard any part of the box $\boldsymbol{x}$ by the FJ optimality test. If the separating hyperplane does not exist, the conic hull is full. Therefore, every direction is included in it, so also every direction of $-\nabla \mathbf{f}(\boldsymbol{x})$ as well. If the method finds that the separating hyperplane does not exist, it returns immediately. Otherwise, it returns the normal vector $v$. The next step is to take the central projection of the constraints' gradients onto the hyperplane $h$. After that, we compute the convex hull $C$ of the projected boxes. If the central projection of the box $-\nabla \mathbf{f}(\boldsymbol{x})$ onto $h$ is contained in $C$, the optimality conditions are satisfied. Otherwise, the FJ optimality test may reduce or discard the box $\boldsymbol{x}$.

# References

[1] E. Hansen and G. W. Walster. *Global Optimization Using Interval Analysis: Revised And Expanded*. CRC Press, 2003.

[2] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter. *Applied Interval Analysis with Examples in Parameter and State Estimation, Robust Control and Robotics*. 08 2001.

[3] R. B. Kearfott. An interval branch and bound algorithm for bound constrained optimization problems. 2:259–280, 1992.

[4] O. Mangasarian and S. Fromovitz. The Fritz John necessary optimality conditions in the presence of equality and inequality constraints. *Journal of Mathematical Analysis and Applications*, 17(1):37–47, 1967.

[5] L. Pál and T. Csendes. Intlab implementation of an interval global optimization algorithm. *Optimization Methods and Software*, 24(4-5):749–759, 2009.

# Sparse tensors and subdivision methods for finding the zero set of polynomials

Guillaume Moroz

Université de Lorraine, CNRS, Inria, LORIA
F-54000 Nancy, France
guillaume.moroz@inria.fr

**Keywords:** Subdivision, sparse tensor, polynomials, root finding

## Introduction

Interval methods are well-suited to enclose the zero set of a function $f$. Subdivision algorithms in particular are widely used for such a task ([1, 2, 3, 4, 5] among others). They roughly consist in evaluating $f$ on boxes created along a subdivision tree. If the input function is a high degree polynomial, one of the bottleneck of those algorithms is the time required to evaluate $f$.

We propose a new approach that amortizes the evaluation cost over the boxes created in a subdivision algorithm. It combines on the one hand partial evaluations of the input polynomial and on the other hand sparse tensors [6, 7] to store the boxes created during the subdivision algorithm. This approach was implemented in the software `voxelize`[1]. As a result, this software can enclose the zero set of polynomial systems that were not reachable with state-of-the-art software.

## Amortized evaluation on a grid of boxes

The first idea to reduce the evaluation redundancies is to use partial evaluation. Assume that $f(x, y)$ is a bivariate polynomial of degree $d$.

[1] `https://gitlab.inria.fr/gmoro/voxelize`.

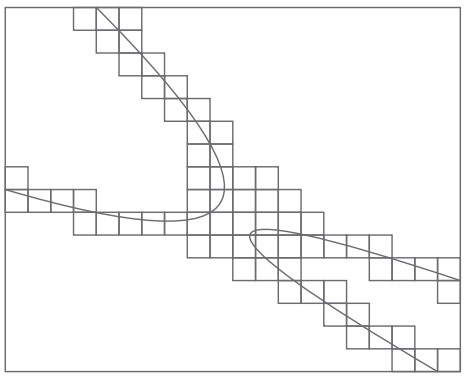


Figure 1: Boxes on the same level of the subdivision tree

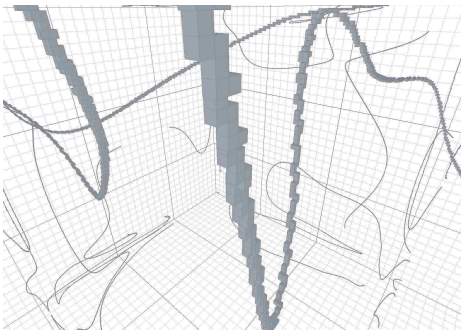


Figure 2: Enclosing of a curve defined by 2 trivariate polynomials of degree 100

Moreover, let $(x_i)_{0\leq i<n}$ and $(y_i)_{0\leq j<n}$ be two sequences of real intervals. Using the Hörner scheme, evaluating $f$ on a box requires $O(d^2)$ arithmetic operations, and evaluating $f$ on all the boxes $(x_i, y_j)$ for $0 \leq i, j < n$ requires $O(d^2n^2)$ arithmetic operations. By reorganizing the operations using partial evaluations, the number of arithmetic operations can be reduced to $O(dn(d+n))$.

More precisely the operations are reordered as follows. For a given $x_i$, the partial evaluation of $f$ in $x_i$ results in a univariate polynomial $f_i$ of degree $d$. This step requires $O(d^2)$ arithmetic operations. Then evaluating $f_i$ on $n$ intervals requires $O(dn)$ arithmetic operations. Finally, repeating these operations for all the $n$ intervals $x_i$, this allows us to evaluate $f$ on all the boxes of the grid with a total number of arithmetic operations in $O(dn(d+n))$. More generally, for higher dimensions, this leads to the following theorem.

**Theorem 1.** *Let $f$ be a polynomial in $k$ variables and of degree at most $d$ in each variable. Let $X_1, \ldots, X_k$ be $k$ sets of $n$ real intervals each. Then it is possible to evaluate $f$ on all the boxes of $X_1 \times \cdots X_k$ in $O(kd \max(nd^{k-1}, n^k))$ arithmetic operations.*

In the case where $n > d$, this approach results in a significant speedup since the amortized number of arithmetic operations to evaluate $f$ on each box of the grid is $O(kd)$ instead of $O(d^k)$.

## Amortized evaluation on a sparse subset of a grid

For the simple subdivision algorithm shown in introduction, if the boxes created are never discarded, then each level of the subdivision tree form a dense grid of boxes. In this case, the partial evaluation approach shown in the previous section can be applied directly to reduce the total number of arithmetic operations required to evaluate $f$ on each boxes with interval methods.

In the general case though, many boxes are discarded, and the boxes appearing in a given level of the subdivision tree form a subset of a grid, as shown in Figure 1. To handle the evaluation of $f$ on a subset of a grid, it is possible to design a variant of the partial evaluation presented in the previous section. The key idea is to encode this sparse subset of boxes with a sparse tensor in the Compressed Sparse Fiber format or CSF [6, 7]. The CSF is a generalization of the Compressed Row Format used to store the entries of a sparse matrix. Performing the partial evaluation approach on a set boxes in a CSF format leads to the following theorem.

**Theorem 2.** *Let $f$ be a polynomial in $k$ variables and of degree at most $d$ in each variable. Let $X_1, \ldots, X_k \subset \mathbb{R}$ be $k$ finite sets of intervals, and let $S$ be a subset of $X_1 \times \cdots \times X_k$. Finally, let $N_i$ be the number of boxes of the projection of $S$ on the first $i$ coordinates, for $i$ from 1 to $k$. Then it is possible to evaluate $f$ on all the points of $X_1 \times \cdots X_k$ in $O(kd \max_{1 \leq i \leq k}(d^{k-i} N_i))$ arithmetic operations.*

As a consequence, if each $X_i$ has more than $d$ intervals, and if the projection of $A$ on the $i$ first coordinates is $X_1 \times \cdots \times X_i$, then the amortized cost is in $O(kd^{k+1-i})$ instead of $O(d^k)$. As shown in the next section, this data structure is particularly well suited when combined with subdivision algorithms.

## Amortized subdivision algorithm

The boxes created in the subdivision algorithm can be handled in different orders. Using a breadth-first walk on the subdivision, the boxes

on the same level can be gathered and stored as a sparse tensor in the CSF format. Then $f$ can be evaluated on them using the amortized evaluation scheme sketched in the previous sections.

This approach was implemented in the library `voxelize`. Figure 2 shows the output boxes of the software `voxelize` enclosing an algebraic curve defined by two polynomial equations of degree 100, where the coefficients are randomly drawn from a normal law centered at zero.

## References

[**1**] J. M. Snyder, *Interval analysis for computer graphics*, in Proceedings of the 19th annual conference on Computer graphics and interactive techniques, SIGGRAPH '92, New York, NY, USA, 1992, ACM, pp. 121–130.

[**2**] S. Plantinga and G. Vegter, *Isotopic approximation of implicit curves and surfaces*, in SGP '04: Eurographics/ACM SIGGRAPH Symposium on Geometry Processing, 2004, pp. 245–254.

[**3**] R. B. Kearfott, *Rigorous global search: continuous problems*, Nonconvex optimization and its applications, Kluwer Academic Publishers, Dordrecht, Boston, 1996.

[**4**] A. Neumaier, *Interval methods for systems of equations*, Cambridge University Press, 1990.

[**5**] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter, *Applied Interval Analysis with Examples in Parameter and State Estimation, Robust Control and Robotics*, Springer London Ltd, Aug. 2001.

[**6**] S. Smith and G. Karypis, *Tensor-matrix products with a compressed sparse tensor*, in Proceedings of the 5th Workshop on Irregular Applications: Architectures and Algorithms, IA3 '15, ACM, 2015, pp. 5:1–5:7.

[**7**] S. Chou, F. Kjolstad, and S. Amarasinghe, *Format abstraction for sparse tensor algebra compilers*, Proc. ACM Program. Lang., 2 (2018), pp. 123:1–123:30.

# S01 E02: A new type of intervals for solving problems involving partially defined functions

Nathalie Revol

INRIA - LIP (UMR 5668 CNRS - ENS de Lyon - UCBL)
ENS de Lyon, 46 allée d'Italie, 69007 Lyon, France
Nathalie.Revol@inria.fr

**Keywords:** Interval Arithmetic, IEEE 1788-2015 Standard, Decoration, Reverse Operations

## Introduction

At SWIM 2022, Luc Jaulin and his co-authors presented a new interval arithmetic in [1], better suited than the usual interval arithmetic for approximating the inner enclosure of a set. It is assumed that this sought set $\mathcal{S}$ is defined by an equation $f(x) = y$ where $x$ belongs to some given interval $\mathbf{x}$, $y$ belongs to some given interval $\mathbf{y}$ and $f$ is defined by a mathematical expression involving the usual arithmetic operations $+$, $-$, $\times$, $/$ and mathematical functions such as exp, sin or $\sqrt{\ }$. The techniques presented in [2], based on the classical interval arithmetic introduced in [3,4], yield an outer enclosure of this set. To get an inner enclosure, that is, a set enclosed within the sought set $\mathcal{S}$, the approach proposed by Jaulin et al. consists in approximating an outer enclosure of the complementary of the set $\mathcal{S}$.

For this purpose, they define a so-called extended interval arithmetic: the extension consists in the addition of a new element in the set of numbers, that signals the fact that the function $f$ is evaluated outside of its domain. This is particularly useful for backward propagation: it enables the contracting algorithm to account for portions of

the set that lie outside of the domain and thus belong to the complementary of $\mathcal{S}$.

## Related work

The IEEE 1788-2015 standard for interval arithmetic [5] introduces the notion of decorations: a decoration is attached to an interval and sums up the history of the computations that led to this interval. In particular it signals the fact that an intermediate computation had an argument outside of its domain.

At first sight, it seems that this system of decorations could be used to solve the problem of determining a complementary set. However, even it is well defined for forward computations and thus for the forward step of Jaulin et al.'s algorithm, it lacks the required properties when it comes to reverse operations that are used during the backward step of this algorithm. Indeed, it propagates the "no information available" signal – called *trivial* decoration – during reverse computations.

The working group that developed the IEEE 1788-2015 was well aware that the decoration system defined by the standard was not satisfactory for these reverse operations, however it was the best it could come up with within the alloted time. Indeed, Section 11.7.1 of the standard [5] says: "No one way of decorating these operations gives useful information in all contexts. Therefore, a *trivial* decorated interval version is provided..."

## Tentative solution

Jaulin et al.'s paper shed light on this gap and offered a patch. Our proposal is an effort towards another patch that is more in line with the IEEE 1788-2015 standard and its system of decorations. It consists in the introduction of a new decoration, called *partial*. We do our best to define the rules that create and propagate it along the computations, in order to convey the desired infomation, which is "there were operations that were given arguments partially outside of their domain". Indeed there is a decoration that signals when the argument is completely out

of the domain and thus when the result is empty. The difficulty is thus to keep track of these "partially outside" arguments for both direct and reverse operations.

## Conclusion

As the IEEE 1788-2015 standard will undergo a revision before the end of 2015, such an insight into its limitations and its potential usages that were not yet considered is very timely and will be considered for the elaboration of the revised version of the standard.

## References

[1] P. Filiol, T. Bollengier, L. Jaulin and J.-C. Le Lann, A new interval arithmetic to generate the complementary of contractors, *SWIM 2022, Hanover, Germany*

[2] . L. Jaulin, M. Kieffer, O. Didrit and E. Walter, *Applied Interval Analysis, with Examples in Parameter and State Estimation, Robust Control and Robotics*, Springer-Verlag, 2001.

[3] . R.E. Moore, *Interval Analysis*, Prentice Hall, 1966.

[4] A. Neumaier, *Interval Methods for Systems of Equations*, Cambridge University Press, 1990.

[5] . IEEE Microprocessor Standards Committee, *IEEE 1788-2015 Standard for Interval Arithmetic.*

# Interval-Based Uncertainty Propagation for Multimodal Response Functions

Dimitris G. Sotiropoulos[1] and Stavros P. Adam[2]

[1] Department of Electrical and Computer Engineering
University of Peloponnese, GR26334 Patras, Greece
dg.sotiropoulos@uop.gr
[2] Department of Informatics and Telecommunications
University of Ioannina, GR47150 Arta, Greece
adamsp@uoi.gr

**Keywords:** Uncertainty Quantification, Interval Arithmetic,

## Introduction

In many engineering fields, a problem of fundamental importance in computing is finding sharp lower and upper bounds on the range of a function of several variables over an $n$-dimensional rectangle. Because the uncertainty problem is equivalent to bounding the range of a multivariate function over a box, interval mathematics is the most straightforward technique for modelling uncertainty and providing guaranteed enclosures. Unfortunately, this is not an easy problem to solve in general. Although there are numerous numerical methods for approximating the range of real functions, none guarantees the required accuracy of the result, and most of them (e.g., Monte Carlo) face computational time constraints.

In this work, an interval algorithm is proposed to address the fundamental problem of computing the range of values of a differentiable function over an $n$-dimensional rectangle to address the computation of uncertainty in the output. Our preliminary numerical results verify the applicability of the proposed method to large uncertainty levels (greater than 20%). Moreover, the method guarantees the required result accuracy as the user desires.

## Methodology description

Let us assume that $f : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable function defined over the box $X \subset \mathbb{R}^n$. In this scenario, the range of $f$, denoted as $F^* = [\min_{x \in X} f(x), \max_{x \in X} f(x)]$, can be accurately determined if verified enclosures for the global minimum are available. This is possible because the

maximization problem can be equivalently transformed into a minimization problem.

The proposed algorithm uses first-order information through an interval gradient evaluation. It takes advantage of the valuable properties of the optimal center of the mean value form (MVF) for both the bounding and the branching process [1] while employing two well-known accelerating devices: the cut-off test and the monotonicity test. The algorithm uses first-order information of the objective function utilizing an interval gradient evaluation to check monotonicity and then apply the optimal mean value form (MVF) for bounding the range of the function. For bounding the objective function, we take the intersection of the natural interval extension with optimal mean value form to obtain the best lower bound for the range of $f$. Moreover, the calculated derivative bounds are exploited to determine an optimal component (subdivision direction) to bisect the box during the branching phase of the algorithm.

In general, the search tree is incrementally expanded by iterating the following steps: (i) The initial search box $X^0$ is subdivided into smaller sub-boxes, (ii) the objective function and its derivatives are bounded over the sub-boxes, and (iii) sub-boxes that cannot contain a global minimizer with certainty are removed. We next summarize the basic interval techniques that accelerate the search process:

**Function range test:** A box $Y$ is discarded from further consideration when the lower bound $\inf F_Y$ is greater than the current upper bound $f_{\max}$. When the range test fails to remove it, it is stored in the working list $\mathcal{W}$ with candidate sub-boxes for further investigation.

**Cut-off test:** The function range test is applied for all candidate sub-boxes in the working list $\mathcal{W}$ when upper bound $f_{\max}$ is improved. Of course, the greater the improvement of $f_{\max}$ is, the more influential the cut-off test is.

**Monotonicity test:** Determines whether the objective function $f$ is strictly monotone in an entire sub-box $Y \subset X^0$ or at least one coordinate direction, in which case $Y$ cannot contain a global minimizer. Therefore, the whole sub-box is discarded, or its dimension is reduced as much as possible when $Y \subseteq X^0$.

Several examples have been investigated to test the algorithm's correctness and efficiency for large-range parameters. In [2], the algorithm has been applied to compute the mechanical properties of two composite unidirectional laminates.

## Numerical results

We present examples from the bibliography to demonstrate the practical behaviour of the proposed algorithm. The implementation was carried out in C++ using the C-XSC-2.5. The first example has only three uncertain parameters; the second is a highly nonlinear function with five interval parameters, and the third is a function with high-dimensional uncertain parameters and strong nonlinearity [4]. We have adopted the same interval parameters as in [4] for comparison reasons referred to as computational cost. At the same time, in our algorithm, we have set the prescribed accuracy $\varepsilon = 10^{-15}$ for the computation of the range bounds.

**Example 1.** $f_1(x) = e^{(0.2x_1x_3+1.5)} - \dfrac{1}{x_2^2}$, where the three interval parameters are produced under eight different levels of uncertainty, $\lambda = 10\%, \ldots, 80\%$ around the nominal value 2.5 and the results are reported in Table 1. In all

Table 1: Computed bounds and computational cost for various levels $\lambda$

| Input $\lambda$ | mid±rad | Output $\lambda$ | NFE | NGE | NB | LL |
|---|---|---|---|---|---|---|
| 10 | 16.172 ± 4.034 | 24.94 | 4 | 2 | 0 | 0 |
| 20 | 18.363 ± 8.639 | 47.04 | 4 | 2 | 0 | 0 |
| 30 | 22.453 ± 14.511 | 64.63 | 4 | 2 | 0 | 0 |
| 40 | 29.219 ± 22.635 | 77.47 | 4 | 2 | 0 | 0 |
| 50 | 40.021 ± 34.535 | 86.29 | 4 | 2 | 0 | 0 |
| 60 | 57.179 ± 52.705 | 92.18 | 4 | 2 | 0 | 0 |
| 70 | 84.634 ± 81.396 | 96.17 | 4 | 2 | 0 | 0 |
| 80 | 128.950 ± 128.238 | 99.45 | 4 | 2 | 0 | 0 |

runs, the algorithm required four (4) function calls, two (2) gradient calls, zero (0) bisections, and the necessary list length was also zero (0), while the CPU time was about 20 microseconds for each instance. In this example, the cost of our algorithm seems to be independent of the input uncertainty level of the parameters, where the uncertainty level $\lambda$ of an interval $X$ is computed as $\lambda = \dfrac{\mathrm{rad}(X)}{|\mathrm{mid}(X)|} \times 100\%$.

**Example 2.** $f_2(x) = \displaystyle\sum_{i=1}^{4} (-1)^i \left(x_1 x_i + 0.25\right)^i + \frac{1}{40 x_1^3 x_5} + 2$, where the five interval parameters are produced under eight different levels of uncertainty $\lambda$ as shown in Table 2, around the nominal value 0.5 for each variable.

**Example 3.** $f_3(x) = \displaystyle\sum_{1 \le i < j \le n} \frac{1}{100} \left(x_i^2 + x_i + 1\right) \left(x_j^2 + x_j + 1\right)$. The third example comprises $n = 100$ parameters, which can be extended to higher

Table 2: Computed bounds and computational cost for various levels $\lambda$

| Input $\lambda$ | mid±rad | Output $\lambda$ | NFE | NGE | NB | LL |
|---|---|---|---|---|---|---|
| 10 | 2.117 ± 0.256 | 12.10 | 4 | 2 | 0 | 0 |
| 20 | 2.225 ± 0.571 | 25.66 | 4 | 2 | 0 | 0 |
| 30 | 2.483 ± 1.039 | 41.86 | 10 | 6 | 2 | 1 |
| 40 | 3.089 ± 1.874 | 60.66 | 10 | 6 | 2 | 1 |
| 50 | 4.622 ± 3.663 | 79.27 | 10 | 6 | 2 | 1 |
| 60 | 9.085 ± 8.422 | 92.71 | 13 | 8 | 3 | 1 |
| 70 | 25.786 ± 25.467 | 98.76 | 13 | 8 | 3 | 1 |
| 80 | 125.884 ± 125.968 | 100.07 | 13 | 8 | 3 | 1 |

dimensions. We repeat the same experiment for this function by producing the interval parameters around the nominal value of 1.2 for each variable. The results are shown in Table 3.

Table 3: Computed bounds and computational cost for various levels $\lambda$

| Input $\lambda$ | mid±rad | Output $\lambda$ | NFE | NGE | NB | LL |
|---|---|---|---|---|---|---|
| 10 | 669.295 ± 147.609 | 22.05 | 4 | 2 | 0 | 0 |
| 20 | 709.736 ± 298.707 | 42.09 | 4 | 2 | 0 | 0 |
| 30 | 777.549 ± 456.785 | 58.75 | 4 | 2 | 0 | 0 |
| 40 | 873.349 ± 625.333 | 71.60 | 4 | 2 | 0 | 0 |
| 50 | 997.999 ± 807.840 | 80.95 | 4 | 2 | 0 | 0 |
| 60 | 1152.607 ± 1007.797 | 87.44 | 4 | 2 | 0 | 0 |
| 70 | 1338.528 ± 1228.692 | 91.79 | 4 | 2 | 0 | 0 |
| 80 | 1557.364 ± 1474.017 | 94.65 | 4 | 2 | 0 | 0 |

## References

[1] D.G. Sotiropoulos and T.N. Grapsa, Optimal Centers in Branch-and-Prune Algorithms for Univariate Global Optimization, *Applied Mathematics and Computation* 169(1), 247-277, 2005. [DOI]

[2] D.G. Sotiropoulos and K.I. Tserpes, Interval-Based Computation of the Uncertainty in the Mechanical Properties and the Failure Analysis of Unidirectional Composite Materials, *Mathematical and Computational Applications* 27(3):38, 2022. [DOI]

[3] T. Wei, F. Li, G. Meng, et al., Bounds for uncertain structural problems with large-range interval parameters, *Archive of Applied Mechanics*, 91, 1157–1177, 2021. [DOI]

# Numerical Scheme of Generalised Moment Problem Based on Interval Analysis

Algassimou Diallo[1], feu Nicolas Delanoue [1] and Daouda Niang Diatta[2]

[1] Universté d'Angers, LARIS, 49100 Angers, France
algassimou.diallo@etud.univ-angers.fr
[2] Université Assane Seck de Ziguinchor, LMA, 27000 Ziguinchor, Sénégal
dndiatta@univ-zig.sn

## 1 Introduction

Consider the following list of problems:

- Finding the global minimum of a function on a subset of $\mathbb{R}^n$,
- Solving equations,
- Computing the Lebesgue volume of a subset $S \subset \mathbb{R}^n$, Computing an upper bound on $\mu(S)$ over all measures $\mu$ satisfying some moment conditions,
- Pricing exotic options in Mathematical Finance,
- Computing the optimal value of an optimal control problem,
- Evaluating an ergodic criterion associated with a Markov chain,
- Evaluating a class of multivariate integrals,
- Computing Nash equilibria,
- With $\widehat{f}$ the convex envelope of a function $f$ , evaluate $\widehat{f}(\mathrm{x})$ at some given point $x$.

The above seemingly different and unrelated problems share actually a very important property: these problems can be seen as particular instances of a linear infinite dimensional optimization problem, called the Generalized

Moment Problem (GMP).

Let $K$ be a Borel subset of $\mathbb{R}^n$ and $\mathscr{M}(K)$ be the space of finite signed Borel measures on $K$, whose positive cone $\mathscr{M}(K)_+$ is the space of finite Borel measures $\mu$ on $K$. Given a set of indices $\Gamma$, a set of reals $\gamma_j \ : \ j \in \Gamma$, and functions $f, h_j : K \to \mathbb{R}, \ j \in \Gamma$ that are integrable with respect to every measure $\mu \in \mathscr{M}(K)_+$, the GMP is defined as follows:

$$\begin{aligned} \rho^*_{mom} = \inf_{\mu \in \mathscr{M}(K)_+} & \int_K f d\mu \\ s.t. & \int_K h_j d\mu \leqq \gamma_j, \quad j \in \Gamma. \end{aligned} \tag{1}$$

and it's dual is defined as ;

$$\begin{aligned} \rho^*_{pop} = \sup_{\lambda_j \in \mathbb{R}_+} & \sum_{j \in \Gamma_+} \lambda_j \gamma_j \\ s.t. \quad & H(\lambda, x) = f(x) - \sum_{j \in \Gamma_+} \lambda_j h_j(x) \geq 0, \quad \forall x \in K \end{aligned} \tag{2}$$

where $\gamma_+ \subset \Gamma$ stands for the set of indice $j$ for which the generalized moment constraint is the inequality $\int h_j d\mu \leq \gamma_j$.

It is known that the GMP has great modeling power with impact in several branches of Mathematics and also with important applications in various fields as illustrated by the above list. However, in its full generality the GMP cannot be solved numerically. According to Diaconis (1987)[1], "the theory [of moment problems] is not up to the demands of applications". One invoked reason is the high complexity of the problem: "numerical determination is feasible for a small number of moments, but appears to be quite difficult in general cases", whereas Kemperman (1987)[1] points out the lack of a general algorithmic approach. Indeed quoting Kemperman: "...a deep study of algorithms has been rare so far in the theory of moments, except for certain very specific practical applications, for instance, to crystallography, chemistry and tomography. No doubt, there is a considerable need for developing reasonably good numerical procedures for handling the great variety of moment problems which do arise in pure and applied mathematics and in the sciences in general...".

This contribution will propose an approach based on interval arithmetic to approximate the GMP with continuous function data and its dual.

We use the guaranteed approximation of the integration of a function with respect to a measure which says that if $\mu$ be a non negative measure on a compact set $K$, $f$ a continuous real-valued function defined on $K$ and $\mathcal{X}$ be a finite partition of $K$ and $[f]$ a convergent inclusion function of $f$, then

$$\sum_{X\in\mathcal{X}} \underline{f}(X)\mu(X) \leq \int_K f d\mu \leq \sum_{X\in\mathcal{X}} \overline{f}(X)\mu(X),$$

to relax the primal problem into a finite dimensional linear problem.
The relaxation of the dual problem is based on the observation that if a continuous function $f$ is strictly positive then, for a given convergent inclusion function of $f$, there exists a partition of $K$ for which the inclusion function is positive on all the elements of this partition.

## References

[**1**] Jean Bernard Lasserre, *Moments, positive polynomials and their applications*, volume 1. World Scientific, 2009.

[**2**] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter, *Applied Interval Analysis: With Examples in Parameter and State Estimation, Robust Control and Robotics*, Springer London, 2001.

[**3**] Nicolas Delanoue, Mehdi Lhommeau and Sébastien Lagrange, *Nonlinear optimal control: a numerical scheme based on occupation measures and interval analysis*, Computational Optimization and Applications, 77(3):307–334, 2020.

# Explaining neural classifiers decisions: A parameter estimation problem solved using intervals

Stavros P. Adam [1], Aristidis C. Likas [2], Ioannis T. Samartzis [1]

[1] Department of Informatics and Telecommunications, University of Ioannina
GR47150 Arta, Greece
adamsp@uoi.gr, johnsam@gmail.com
[2] Department of Computer Science and Engineering, University of Ioannina
GR45110 Ioannina, Greece
arly@cs.uoi.gr

**Keywords:** Set membership estimation, Interval analysis, Explainable machine learning, Neural classifiers

## Introduction

Machine learning (ML) systems have demonstrated their ability to successfully solve real world problems by means of constructing nonlinear models of the processes governing these problems. These models result from data analyzed by complex nonlinear ML approaches such as neural networks, kernel-based techniques, boosting, Gaussian processes and many more, while the most recent paradigm concerns Deep Learning (DL), a class of stacked neural architectures consisting of several interconnected layers with numerous neurons.

However, it is common knowledge that nonlinear classifiers and in particular neural classifiers act as black boxes and they provide no means of direct human understanding of their decision making. In consequence humans using these systems cannot rely upon them without understanding the learned elements of their reasoning in order to interpret and verify decisions. This proves to be a serious problem whenever safety requirements apply, especially in the case of critical systems operation. In addition, the General Data Protection Regulation (GDPR) of the European Union imposes explicit legal requirements on the use of Artificial Intelligence (AI) systems either for profiling or for automated decision-making purposes. Among these requirements *Transparency towards individuals* stands for providing meaningful information explaining the results of an AI system processing personal data [1].

Though interpretability of learned systems proved to be useful in several scientific and technological areas, it was the advent of DL architectures that, actually, promoted the topic known as eXplainable Artificial Intelligence (XAI). Moreover, recent research on XAI has received increased attention due to initiatives such as the DARPA XAI project and the Explainable Artificial Intelligence Workshops held during the IJCAI-2017 and IJCAI-2019. For a detailed survey of the methods and techniques proposed towards Explainable AI and Machine Learning systems the interested reader may refer to [2].

## Methodology description

The idea behind the proposed approach is not new. Actually, Adam et al. [3] first introduced it in order to derive the domain of validity of a trained Multi-layer Perceptron (MLP). This idea is adapted, here, to deal with Explainability matters of neural classifiers, as described hereafter.

After training a neural classifier implements a model of the underlying classification function which is learned given a number of training patterns. These patterns are supposed to consistently represent the input space as their sampling distribution is considered to match the distribution of the input data. When an unknown $n$-dimensional input pattern, $\mathbf{x} = (x_1, x_2, \ldots, x_N)^T$ is presented to the trained network a decision is taken by the network concerning the classification of the pattern. The network output, denoted here, $\mathbf{y} = \mathcal{N}(\mathbf{W}, \mathbf{X})$ depends on the values of the features $x_i$, $i = 1, \ldots, N$ and in consequence each one of the features can be seen as a parameter of the classification model. Note, that $\mathbf{y} = (y_1, y_2, \ldots, y_M)^T$ represents the vector of $M$ network outputs, one output per class, while $\mathbf{W}$ represents the matrix of the network synaptic weights computed during training.

Suppose that for some given pattern the $j$-th network output is triggered, which results in having $y_j$ in some interval $[y_j] = [1 - \beta_j, 1]$ where $0 \leqslant \beta_j$ is pattern dependent. In order to explain the decision of the network one should be able to analyze how each feature contributes to the decision $y_j$. This means that if $[x_k]$ denotes the interval within which range the values of the $k$-th feature then, with respect to the above hypotheses, the problem of deriving the contribution of that feature to the specific network output can be cast in terms of the following parameter estimation problem,

$$\mathcal{N}^{-1}(y_j(\mathbf{x})) \subseteq ([x_1], [x_2], \ldots, [x_k], \ldots, [x_N]). \tag{1}$$

In this formulation all intervals $[x_i]$, except for $i = k$, are degenerate, i.e. point intervals, corresponding to the specific values of the features for pattern $\mathbf{x}$.

It is obvious that SIVIA [4] is the appropriate method for solving this set inversion problem which results in defining the set of values of the $k$-th feature contributing to the specific prediction, denoted by

$$\hat{\mathrm{x}}_k = \{\hat{x} \in [x_k] : \mathcal{N}(x_1, x_2, \ldots, \hat{x}, \ldots, x_n) \in [y_j]\}. \tag{2}$$

It is this set $\hat{\mathrm{x}}_k$ and its relation with $[x_k]$ that hold all the information explaining how the specific prediction $\mathcal{N}(\mathbf{x}) = \mathbf{y}$ provided by the classifier is affected by the $k$-th feature. Moreover, $\hat{\mathrm{x}}_k$ is computed as a union of subsets of $[x_k]$ and so it can be connected or not. Figure 1, hereafter, displays examples of various possible configurations of $\hat{\mathrm{x}}_k$. These examples concern contribution of the four features of the Fisher Iris dataset to the prediction provided by a trained MLP with 2 hidden units for the 17-th pattern of the class *Iris-versicolor*.

In order to formulate a measure suitable for the feature relevance, let us first state that the degree of relevance of some feature is inversely proportional to the size of the set $\hat{\mathrm{x}}_k$. Hence, the narrower the union of intervals forming $\hat{\mathrm{x}}_k$, the higher the relevance of the $k$-th feature while a larger $\hat{\mathrm{x}}_k$ corresponds to a less relevant feature. Formula (3) defines a rough measure of the relevance, where $\mu(\cdot)$ is used to denote the length of an interval.

$$\mathcal{R}_{\mathcal{N},\mathbf{x},k} = 1 - \frac{\mu(\hat{\mathrm{x}}_k)}{\mu([x_k])}. \tag{3}$$

A number of important remarks have permitted to derive the more detailed formula (4) which is actually used in all our experiments.

$$\mathcal{R}_{\mathcal{N},\mathbf{x},k} = \left\{ \begin{array}{ll} 1 - \dfrac{\sum_{l=1}^{L} \mu(x^l) \mathbb{1}_{\mathcal{A}}(x^l)}{\mu_k} & \text{if} \quad \sum_{l=1}^{L} \mu(x^l) \mathbb{1}_{\mathcal{A}}(x^l) > 0 \\ \mathrm{H}\left(\sum_{l=1}^{L} \mu(x^l) \mathbb{1}_{\mathcal{U}}(x^l)\right) & \text{if} \quad \sum_{l=1}^{L} \mu(x^l) \mathbb{1}_{\mathcal{A}}(x^l) = 0 \end{array} \right\} \tag{4}$$

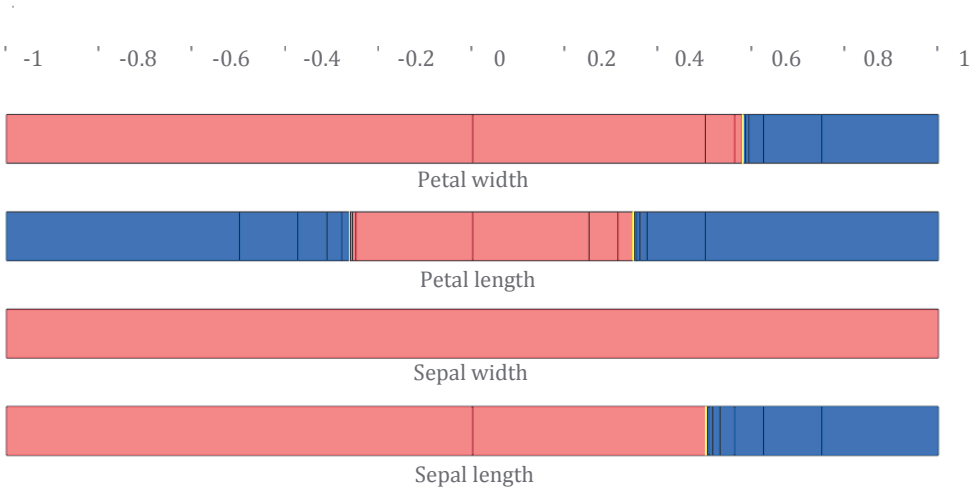



Figure 1: Contributing feature values form the red colored areas.

In Formula (4), $x^1, x^2, \ldots, x^L$ denote the intervals forming the partition of $[x_k]$ derived from SIVIA. The intervals of interest are those belonging either to $\mathcal{A}$ the class of intervals in the interior of $\hat{\mathrm{x}}_k$ or to $\mathcal{U}$ the class of intervals forming the border of $\hat{\mathrm{x}}_k$. Note that, $\mathbb{1}(\cdot)$ and $\mathrm{H}(\cdot)$ denote, respectively, the indicator and the Heaviside functions.

## Main results

The approach proposed is a deterministic method which succeeds to quantify uncertainty pertaining the output decision of some specific neural classifier (after training) in terms of the relevance of its input features. The method can be used to examine the feature relevance either in cases of correct or incorrect classification. Based on the above formula the relevance computed for each feature of the specific pattern $\mathbf{x}$ in the example of Figure 1 is:

| Sepal length | Sepal width | Petal length | Petal width |
|---|---|---|---|
| $\mathcal{R}_{\mathcal{N},\mathbf{x},1} = 0.249451$ | $\mathcal{R}_{\mathcal{N},\mathbf{x},2} = 0.0$ | $\mathcal{R}_{\mathcal{N},\mathbf{x},3} = 0.695282$ | $\mathcal{R}_{\mathcal{F},\mathbf{x},4} = 0.209869$ |

One is able to construct heat maps providing an outlook of the feature relevance for any class. An example of such heat maps is portrayed in the following Figure 2 where it is possible to examine whether there exist features which do not contribute to or they are irrelevant for some classification problem and thus they can be pruned.

Another experiment was carried out in order to test the performance of the approach in the case of image classification problems. This experiment used a shallow network with 784 inputs, 100 hidden nodes and 10 outputs which was trained with the MNIST benchmark. Each image tested is considered to be a pattern composed of 784 features, one per pixel of the $28 \times 28$ pixels of the image. Some indicative results are given in Figure 3 below. Note that the colored image, next to each original gray-scale image, displays for each pixel its relevance to the network output. According to the color map used, the pixels with higher relevance are brighter while the pixels that are irrelevant are colored black.

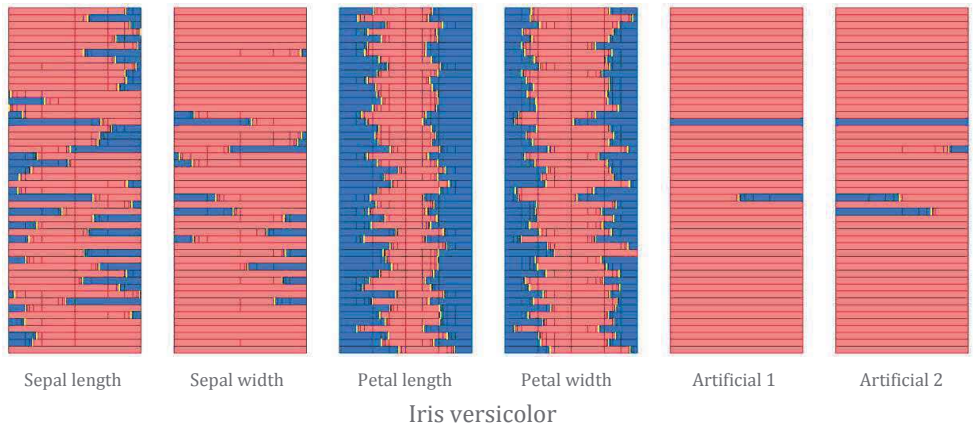



Figure 2: Heat map of feature relevance for Iris versicolor class with two artificial features.

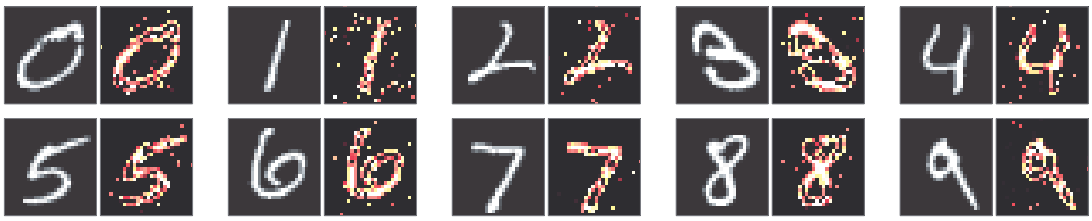


Figure 3: Performance of the proposed approach for a number of images from the MNIST dataset.

## References

[1] Europa.eu, General Data Protection Regulation (GDPR), *Official journal of the european union: Regulations*, URL https://gdpr-info.eu/, 2017.

[2] N. Burkart and M. F. Huber, A Survey on the Explainability of Supervised Machine Learning, *Journal of Artificial Intelligence Research* 70, 245–317, 2021.

[3] S. P. Adam, D. A. Karras, G. D. Magoulas, and M. N. Vrahatis, Reliable estimation of a neural network's domain of validity through interval analysis based inversion, *Proceedings of the International Joint Conference on Neural Networks, (IJCNN-2015)*, IEEE, 2015.

[4] L. Jaulin and E. Walter, Set Inversion Via Interval Analysis for nonlinear bounded-error estimation, *Automatica*, 729(4), 1053–1064, 1993.

# State-Estimation of Uncertain Timed Event Graphs : an SMT approach

Guilherme Espindola-Winck, Laurent Hardouin, Mehdi Lhommeau

Laboratoire Angevin de Recherche en Ingénierie des Systèmes, Université d'Angers, LARIS, Polytech Angers
49000, Angers, France

{guilherme.espindolawinck;laurent.hardouin;mehdi.lhommeau}@univ-angers.fr

**Keywords:** Timed Event Graphs, (max,+)-linear systems, State-Estimation, Satisfiability modulo theory

## Introduction

This work presents a new approach to use Satisfiability Modulo Theory (SMT) approach to handle state-estimation of Timed Event Graphs (TEGs) with uncertain place holding times, which are modeled by linear state-space equations in the (max,+) algebra [1]. The existing approaches in the literature are mentioned below.

**State observer :** A Luenberger-inspired observer introduced in [8] computes, using residuation theory [1], an estimation as close as possible, from below, to the true state. It is exclusively based on structural assumptions of the perturbations, i.e. without any probabilistic guarantee.

**Stochastic filtering :** Stochastic filtering of such systems has been studied in [9] for bounded random place holding times and extended to implicit forms in [7], exploiting live property of TEGs. The stochastic filtering algorithm is based on a two-fold calculation, as it is the case for the classical Bayesian filtering algorithm and uses the

mathematical conditional expectation and interval analysis tools of max-affine functions.

**Set-membership filtering :** Set-membership filtering, or set-estimation, for short, has been studied in [2, 5, 4]. Briefly, this kind of estimation approach computes, as a set, the exact (or over-approximation) support of the posterior probability density function (PDF) in such a way that a point in this set can be considered as an estimation with a probabilistic guarantee but no stochastic optimality, as in Kalman-like filtering [6].

Set-estimation for implicit forms is difficult to handle with existing approaches, and in this work we present an approach based on recent works of [10] to compute an estimation within the support of the posterior PDF.

# (max,+)-linear systems

The (max,+) algebra ($\mathbb{R}_{\max} = \mathbb{R} \cup \{-\infty\}, \oplus := \max, \otimes := +$) is defined in [1]. Implicit equations as $x = Ax \oplus b$ admit $x = A^\star b$ as smallest solution where

$$A^\star = \bigoplus_{k \in \mathbb{N}} A^{\otimes k}$$

is the Kleene star of $A$. In practice, $A$ is strictly lower triangular, hence $A^\star$ is well-defined.

It is well known that Timed Event Graphs, working under *earliest firing rule*, are modeled by (see [1, Sec. 2.5]) the following kind of equations:

$$\begin{aligned} x(k) &= B(k)x(k) \oplus A(k)x(k-1), \\ z(k) &= C(k)x(k), \\ & B(k) = B, A(k) = A, C(k) = C,\ k = 1, 2, \ldots. \end{aligned}$$

where $B, A \in \mathbb{R}_{\max}^{n \times n}$, $C \in \mathbb{R}_{\max}^{q \times n}$, $x(k) \in \mathbb{R}^n$ (continuous state-space) and $z(k) \in \mathbb{R}^q$ (continuous measurement-space). As it can be noted, this modeling considers implicit forms, i.e. $x(k)$ appears in both sides

of the dynamic evolution equation. Furthermore, this equation can be rewritten in explicit form thanks to the Kleene star of $B$, i.e. as $x(k) = B^{\star}Ax(k-1)$.

In practice, in order to consider uncertainties, the matrix entries have to be considered as random values, distributed according to PDFs. More specifically, each entry is assumed to take an arbitrary value within the support of the corresponding PDF $f(\cdot)$, and in this paper such supports have bounded domain. Mathematically, $B(k) \in [\underline{B}, \overline{B}]$, $A(k) \in [\underline{A}, \overline{A}]$ and $C(k) \in [\underline{C}, \overline{C}]$ and for instance, $a_{ij}(k) \sim f(a_{ij})$ is analogous to $a_{ij}(k) \in [\underline{a}_{ij}, \overline{a}_{ij}]$. Moreover, in the uncertain case, it is not possible to apply Kleene star operation if we are not interested in conservative results.

# Set-estimation

Let us define the prediction set based on the previous knowledge of the state $\hat{x}(k-1|k-1)$ for systems without implicit form, i.e. $x(k) = A(k)x(k-1),\ A(k) \in [A] = [\underline{A}, \overline{A}]$, as

$$X_{k|k-1} = \{Ax \in \mathbb{R}^n \mid A \in [A]\}$$

with $x = \hat{x}(k-1|k-1)$, which is equal to $[\underline{A}x, \overline{A}x]$. For practical implicit forms,

$$x_i(k) = \left( \bigoplus_{l=1}^{i-1} b_{il}(k) x_l(k) \right) \oplus \left( A(k)x(k-1) \right)_i$$

for $i = 1, 2, \ldots$. Hence, computing $X_{k|k-1}$ analytically turns out to be difficult to be handled with existing approaches.

The measurement likelihood set based on the measurement $z(k)$ is computed as

$$\begin{aligned} \tilde{X}_{k|k} &= \{x \in \mathbb{R}^n \mid \exists C \in [\underline{C}, \overline{C}] \ s.t.\ Cx = z(k)\} \\ &= \{x \in \mathbb{R}^n \mid \underline{C}x \leq z(k) \leq \overline{C}x\} \end{aligned}$$

being computed using [2]. Thus, $X_{k|k} = X_{k|k-1} \cap \tilde{X}_{k|k}$ is the support of the posterior PDF such that it is easy to define an estimation $\hat{x}(k|k) \in X_{k|k}$.

In this work, we consider the *symbolic representation* of (max,+)-linear systems using SMT techniques [10]. Briefly, each row of $B(k)x(k) \oplus A(k)x(k-1)$ is represented by

$$
\begin{aligned}
&Row^i_{Bx(k)} \times Row^i_{Ax(k-1)} \ : \\
&\left[\left(\bigwedge_{l\in\mathcal{G}_i} \mathtt{x}_i^{(k)} - \mathtt{x}_l^{(k)} \geq \mathtt{b}_{il}^{(k)}\right) \wedge \left(\bigwedge_{j\in\mathcal{F}_i} \mathtt{x}_i^{(k)} - \mathtt{x}_j^{(k-1)} \geq \mathtt{a}_{ij}^{(k)}\right)\right] \\
&\wedge \left[\left(\bigvee_{l\in\mathcal{G}_i} \mathtt{x}_i^{(k)} - \mathtt{x}_j^{(k)} = \mathtt{b}_{il}^{(k)}\right) \vee \left(\bigvee_{j\in\mathcal{F}_i} \mathtt{x}_i^{(k)} - \mathtt{x}_j^{(k-1)} = \mathtt{a}_{ij}^{(k)}\right)\right] \\
&\wedge \left[\left(\bigwedge_{l\in\mathcal{G}_i} (\mathtt{b}_{il}^{(k)} \geq \underline{b}_{il}) \wedge (\mathtt{b}_{il}^{(k)} \leq \overline{b}_{il})\right) \wedge \left(\bigwedge_{j\in\mathcal{F}_i} (\mathtt{a}_{ij}^{(k)} \geq \underline{a}_{ij}) \wedge (\mathtt{a}_{ij}^{(k)} \leq \overline{a}_{ij})\right)\right]
\end{aligned}
$$

with $\wedge$ (resp. $\vee$) the conjunction (resp. disjunction) operator. $\mathtt{x}_1^{(k)}, \ldots, \mathtt{x}_n^{(k)}$ and $\mathtt{a}_{ij}^{(k)}, \mathtt{b}_{il}^{(k)}$ are symbolic variables for each $k$ ; $\mathcal{F}_i \subseteq \{1, \ldots, n\}, \mathcal{G}_i \subseteq \{1, \ldots, i-1\}$ are sets of indices of the $i$-th rows of $A(k), B(k)$ that are different $-\infty$ (i.e. are finite). Thus $\mathbb{S}^{k,k-1} \ : \ \bigwedge_{i=1}^{n} \left(Row^i_{Bx(k)} \times Row^i_{Ax(k-1)}\right)$ represents symbolically the dynamic evolution equation. The same is applied to the measurement likelihood, representing the observation equation

$$
\begin{aligned}
\tilde{\mathtt{X}}^{k|k} \ : \ \bigwedge_{i=1}^{q} &\left[\left(\bigwedge_{j\in\mathcal{H}_i} \mathtt{c}_{ij}^{(k)} + \mathtt{x}_j^{(k)} \leq z_i(k)\right) \wedge \left(\bigvee_{j\in\mathcal{H}_i} \mathtt{c}_{ij}^{(k)} + \mathtt{x}_j^{(k)} = z_i(k)\right)\right. \\
&\left.\wedge \left(\bigwedge_{j\in\mathcal{H}_i} (\mathtt{c}_{ij}^{(k)} \geq \underline{c}_{ij}) \wedge (\mathtt{c}_{ij}^{(k)} \leq \overline{c}_{ij})\right)\right]
\end{aligned}
$$

where $\mathtt{c}_{ij}^{(k)}$ are symbolic variables and $\mathcal{H}_i$ with the same meaning as

for $\mathcal{F}_i$ but for $C(k)$. Finally, $\mathtt{X}^{k|k-1} : \hat{\mathtt{x}}^{k-1|k-1} \wedge \mathtt{S}^{k,k-1}$ represents $X_{k|k-1}$ with $\hat{\mathtt{x}}^{k-1|k-1} : \bigwedge_{i=1}^{n} \left(\mathtt{x}_i^{(k-1)} = \hat{x}_i(k-1|k-1)\right)$. Then $\mathtt{X}^{k|k} : \mathtt{X}_{k|k-1} \wedge \tilde{\mathtt{X}}^{k|k}$ is expected to be `SAT`, representing $X_{k|k}$. Using SMT solvers, for instance [3], we are able to verify the previous expression and return a *solution* that makes each asserted constraint true, defining then a value for $\hat{x}(k|k)$. As part of a filtering algorithm, a recursion is defined, i.e. $\hat{x}(k-1|k-1) \leftarrow \hat{x}(k|k)$ and the solver is called once again.

**Example 1.** *Let* $B(k) \in \begin{pmatrix} -\infty & -\infty \\ [1,2] & -\infty \end{pmatrix}$, $A(k) \in \begin{pmatrix} [4,6] & [3,5] \\ [3,7] & [4,5] \end{pmatrix}$ *and* $C(k) \in \begin{pmatrix} [0,1] & -\infty \end{pmatrix}$ *with* $\hat{x}(0|0) = (1,0)^T$, $x(1) = (6,7)^T$, $C(1) = \begin{pmatrix} 0.5 & -\infty \end{pmatrix}$ *implying* $z(1) = 6.5$. *Then* $X_{1|1}$ *is depicted in Figure 1, with* $X_{1|0}$ *computed using existing approaches.*

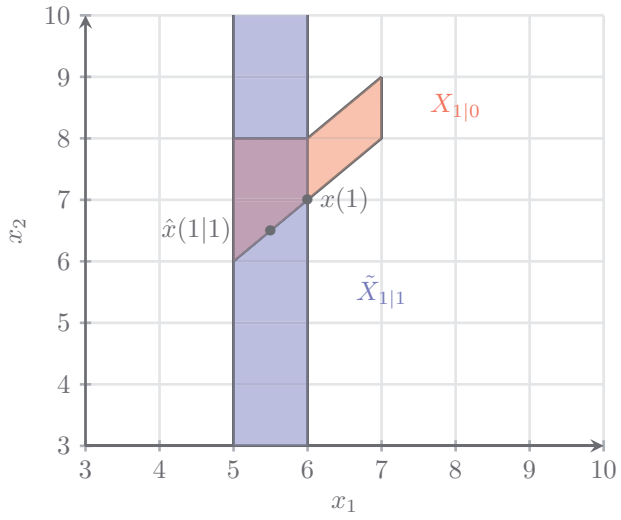



Figure 1: Computing an estimation

# Conclusions and perspectives

As part of future works, we aim at including stochastic constraints as symbolic variables and thus respect stochastic criteria.

# References

[1] F. Baccelli, G. Cohen, G. Olsder, and J. Quadrat. *Synchronization and Linearity : An Algebra for Discrete Event Systems*. Wiley and Sons, 1992.

[2] R. M. F. Candido, L. Hardouin, M. Lhommeau, and R. Santos-Mendes. An algorithm to compute the inverse image of a point with respect to a nondeterministic max plus linear system. *IEEE Transactions on Automatic Control*, pages 1–1, 2020.

[3] L. de Moura and N. Bjørner. Z3: an efficient smt solver. volume 4963, pages 337–340, 04 2008.

[4] G. Espindola-Winck, R. M. F. Cândido, L. Hardouin, and M. Lhommeau. Efficient state-estimation of uncertain max-plus linear systems with high observation noise. *IFAC-PapersOnLine*, 55(28):228–235, 2022. 16th IFAC Workshop on Discrete Event Systems WODES 2022.

[5] G. Espindola-Winck, L. Hardouin, and M. Lhommeau. Max-plus polyhedra-based state characterization for umpl systems. In *2022 European Control Conference (ECC)*, pages 1037–1042, 2022.

[6] G. Espindola-Winck, L. Hardouin, M. Lhommeau, and R. Santos-Mendes. Criteria stochastic filtering of max-plus discrete event systems with bounded random variables. *IFAC-PapersOnLine*, 55(40):13–18, 2022. 1st IFAC Workshop on Control of Complex Systems COSY 2022.

[7] G. Espindola-Winck, L. Hardouin, M. Lhommeau, and R. Santos-Mendes. Stochastic filtering scheme of implicit forms of uncertain max-plus linear systems. *IEEE Transactions on Automatic Control*, 67(8):4370–4376, 2022.

[8] L. Hardouin, C. A. Maia, B. Cottenceau, and M. Lhommeau. Observer design for (max,+) linear systems. *IEEE Trans. on Automatic Control*, 55 - 2:538 – 543, 2010.

[9] R. S. Mendes, L. Hardouin, and M. Lhommeau. Stochastic filtering of max-plus linear systems with bounded disturbances. *IEEE Transactions on Automatic Control*, 64(9):3706–3715, Sep. 2019.

[10] M. S. Mufid, D. Adzkiya, and A. Abate. Smt-based reachability analysis of high dimensional interval max-plus linear systems. *IEEE Transactions on Automatic Control*, 67(6):2700–2714, 2022.

# Interval Observer Design for State Estimation of Lithium-Ion Batteries

Marit Lahme[1], Andreas Rauh [1] and Guillaume Defresne[1,2]

[1] Carl von Ossietzky Universität Oldenburg,
Distributed Control in Interconnected Systems
D-26111 Oldenburg, Germany
{marit.lahme, andreas.rauh}@uol.de
[2] Institut Supérieur de l'Aéronautique et de l'Espace (ISAE-SUPAERO)
F-31055 Toulouse, France
guillaume.defresne@student.isae-supaero.fr

**Keywords:** interval observer design, LMI, lithium-ion battery, state and disturbance estimation

## Introduction

Estimating the dynamic behavior of lithium-ion batteries is crucial for a lot of applications, for example in battery management systems. The state variables can either not be measured directly or the measurement effort or costs would be too high. To accurately estimate the state variables, it is necessary to design observers with stable error dynamics and high estimation accuracy. Furthermore, the dynamic behavior is subject to process and measurement noise. The noise can be modeled using stochastic or interval techniques. In the latter case, the noise is considered to be unknown, but bounded. In this case, interval observer approaches are widely used to estimate state variables of uncertain dynamic systems [1]. Our goal is to design an interval observer for estimating the state variables of a lithium-ion battery that can subsequently be used for identification or control purposes.

The equivalent circuit model shown in Fig. 1 is employed to model the dynamic behavior of lithium-ion batteries. Here, $\sigma(t)$ is the state of charge, $v_{\mathrm{TS}}(t)$ and $v_{\mathrm{TL}}(t)$ are the voltages across the RC sub-networks

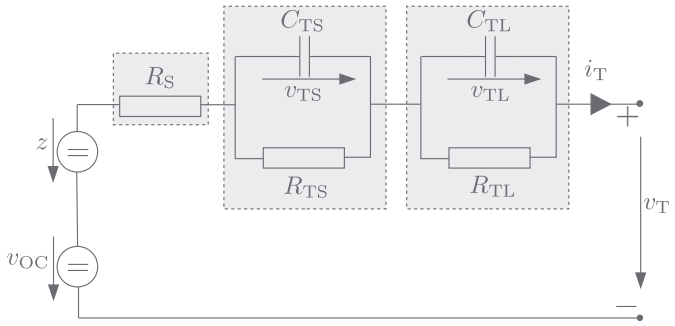



Figure 1: Equivalent circuit model of a lithium-ion battery, including a disturbance voltage source $z$.

and $z(t)$ represents a disturbance voltage source. This voltage source is included in the state vector to quantify model mismatches based on a scalar-valued integrator disturbance model. The lithium-ion battery can be modeled with the following state space system, as shown in previous work [2]

$$
\begin{aligned}
&\mathbf{x}(t) = \begin{bmatrix}\sigma(t) & v_{\mathrm{TS}}(t) & v_{\mathrm{TL}}(t) & z(t)\end{bmatrix}^T \in \mathbb{R}^n \ , \ n = 4 \ , \qquad (1)\\
&\dot{\mathbf{x}}(t) = \mathbf{A}(\sigma(t)) \cdot \mathbf{x}(t) + \mathbf{b}(\sigma(t)) \cdot u(t) \ ,\\
&\mathbf{y}^*(t) = \mathbf{y}(t) - \mathbf{D}(\sigma(t)) \cdot i_{\mathrm{T}}(t) = \mathbf{C}(\sigma(t)) \cdot \mathbf{x}(t) \in \left[\underline{\mathbf{y}}_{\mathrm{m}} \ ; \ \overline{\mathbf{y}}_{\mathrm{m}}\right] \ ,\\
&\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \frac{-1}{C_{\mathrm{TS}}(\sigma(t)) \cdot R_{\mathrm{TS}}(\sigma(t))} & 0 & 0 \\ 0 & 0 & \frac{-1}{C_{\mathrm{TL}}(\sigma(t)) \cdot R_{\mathrm{TL}}(\sigma(t))} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \ , \ \mathbf{b} = \begin{bmatrix} \frac{-1}{C_{\mathrm{Bat}}} \\ \frac{1}{C_{\mathrm{TS}}(\sigma(t))} \\ \frac{1}{C_{\mathrm{TL}}(\sigma(t))} \\ 0 \end{bmatrix} \ ,\\
&\mathbf{C} = \begin{bmatrix}\eta_{\mathrm{OC}}(\sigma(t)) & -1 & -1 & 1\end{bmatrix} \ , \ \mathbf{D} = -R_{\mathrm{S}}(\sigma(t)) \ ,
\end{aligned}
$$

where $\mathbf{y}^*(t)$ is a quasi-linear representation of the output equation. The state of charge dependent parameters are represented by nonlinear expressions [2,3]. In this model, the terminal current $i_T$ is used as the system input to charge or discharge the battery and the terminal voltage $v_T$ is measured. The measurement $\mathbf{y}_{\mathrm{m}}$ is subject to noise which is assumed to be bounded.

## Interval Observer Design

In previous work, an interval observer based on a Luenberger observer structure was used to estimate the lower and upper bounds of the state variables. The observer gain was designed in such a way, that stability was guaranteed for the observed system, but the observer gain matrix was not specifically optimized. The drawback of this approach is, that the estimation uncertainty may increase over time. However, as shown in [3], the estimation uncertainty can be reduced by using input current profiles that include phases of zero current. In contrast to [3], where $z(t) = 0$, the estimation of the disturbance voltage $z(t)$ is included in this work. The objective of this work is to estimate the state variables during system operation without using specified current profiles. Therefore, it is necessary to investigate other design approaches. A promising approach is a TNL observer design as derived in [4]. This approach includes two additional design parameters $\mathbf{T}$ and $\mathbf{N}$ besides the observer gain $\mathbf{L}$. Therefore, it provides more design degrees of freedom. Additionally, it includes an $H_\infty$ technique to reduce the influence of the bounded measurement noise on the estimated (state) variables. The approach is based on LMIs, to be evaluated offline which are designed so that stability is guaranteed for the observed systems. Additionally, a Metzler [5] structure is enforced in the system matrix and the observer gains are optimized.

Before implementing the observer, the system (1) has to be investigated, because it is not fully observable in this form, in contrast to the simplified version in [3] if a linear polytopic uncertainty model is employed to embed the nonlinearities. As mentioned in [2], the system is observable if it is considered as nonlinear, even if $z(t)$ is included in the state vector. Observability can be ensured for system (1) for example by considering a second output equation, which leads to an augmented output matrix. Aside from that, the output matrix is separated into a constant and a time-varying part, where the latter one is treated as a measurement uncertainty. This simplifies solving the LMIs.

Directly implementing the TNL approach for the system (1) leads to an infeasible problem. To solve this, the TNL observer as described

in [4] can either be combined with an iterative approach for solving the LMIs, similar to [6] and [7], or implemented as a cascaded interval observer.

## References

[**1**] T. Raïssi and D. Efimov, Some Recent Results on the Design and Implementation of Interval Observers for Uncertain Systems, *at - Automatisierungstechnik* (vol. 66), no. 3, 213–224, 2018.

[**2**] M. Lahme and A. Rauh, Combination of Stochastic State Estimation with Online Identification of the Open-Circuit Voltage of Lithium-Ion Batteries, In *Proc. of the 1st IFAC Workshop on Control of Complex Systems (COSY)*, 97–102, Bologna, Italy, 2022.

[**3**] M. Lahme and A. Rauh, Set-Valued Approach for the Online Identification of the Open-Circuit Voltage of Lithium-Ion Batteries, In *Proc. of the 13th Summer Workshop on Interval Methods (SWIM)*. Hanover, Germany, 2022, in press.

[**4**] Z. Wang, C.-C. Lim and Y. Shen, Interval Observer Design for Uncertain Discrete-Time Linear Systems, *Systems & Control Letters* (vol. 116), 41–46, 2018.

[**5**] T. Kaczorek, Positive 1D and 2D Systems. *Communications and Control Engineering*. Springer, London, 2002.

[**6**] R. Dehnert, M. Damaszek, S. Lerch, A. Rauh and B. Tibken, Robust Feedback Control for Discrete-Time Systems Based on Iterative LMIs with Polytopic Uncertainty Representations Subject to Stochastic Noise. *Frontiers in Control Engineering* (vol. 2), 2022.

[**7**] A. Rauh and S. Romig, Linear Matrix Inequalities for an Iterative Solution of Robust Output Feedback Control of Systems with Bounded and Stochastic Uncertainty, *Sensors* (vol. 21), no. 9, 2021. MDPI.

# Thursday June 29

# Propagation of degenerate ellipsoids towards computer-assisted proofs

Morgan Louédec[1], Luc Jaulin[1] and Christophe Viel[2]

[1] Lab-STICC, ENSTA-Bretagne,
2 rue François Verny 29200 Brest, France
morgan.louedec@centraliens-nantes.org
lucjaulin@gmail.com
[2] CNRS, Lab-STICC, F-29806, Brest, France
christophe.viel@ensta-bretagne.fr

**Keywords:** Ellipsoid, Non-linear, Stability, Degenerate

## Introduction

Over the last two decades, the research on controlling groups of mobile robots has been very active. It has led to high-dimensional nonlinear models which are complex to study. The increase in complexity is now making conventional nonlinear analysis methods challenging to use, such as the Lyapunov theory in [6, 5, 7]. Interval analysis may have the potential to propose general analysis methods to ease the analysis of complex systems, see for example [3, 2].

Computer-assisted proofs with interval analysis algorithms often consider specific shapes of sets to find a compromise between the computational complexity and the pessimism of the algorithm [1]. Ellipsoids are often described as a good compromise as they lead to polynomial complexity. Moreover, they are related to quadratic Lyapunov functions which helps their use for stability proofs. There exists some method to propagate ellipsoids over the evolution of a system, such as [4]. However, some problems can lead to systems with flatten ellipsoids, resulting in degenerate ellipsoids where the classic tools cannot find a solution.

This presentation proposes to illustrate the way degenerate ellipsoids propagate.

## Basic properties

Every evolution of a state vector $\boldsymbol{x} \in \mathbb{R}^n$ can be described by a nonlinear mapping

$$\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x}), \boldsymbol{f} : \mathbb{R}^n \mapsto \mathbb{R}^n, \tag{1}$$

where $\boldsymbol{y} \in \mathbb{R}^n$ is a future state of the system. This mapping can be a **discrete mapping**, where the analytical form of $\boldsymbol{f}$ is given. It can be associated with a **dynamical system**, where $\boldsymbol{f}$ is the flow function. It can be a **Poincare mapping**, where $\boldsymbol{f}$ is the projection of the state on a Poincare surface. Or it can be a mix of the three previous mappings

In the state space, a nondegenerate ellipsoid of the form $\mathcal{E}(\boldsymbol{\mu}, \boldsymbol{\Gamma})$ can be expressed as

$$\mathcal{E}(\boldsymbol{\mu}, \boldsymbol{\Gamma}) = \left\{ \boldsymbol{x} \in \mathbb{R}^n | (\boldsymbol{x} - \boldsymbol{\mu})^T \boldsymbol{\Gamma}^{-T} \boldsymbol{\Gamma}^{-1} (\boldsymbol{x} - \boldsymbol{\mu}) \leq 1 \right\} \tag{2}$$

with the midpoint $\boldsymbol{\mu} \in \mathbb{R}^n$ and the positive definite shape matrix $\boldsymbol{\Gamma}\boldsymbol{\Gamma}^T \succ 0$. While the ellipsoids are often described by the quadratic form (2), they can also be presented as an affine transformation of the unit sphere

$$\begin{aligned} \mathcal{E}(\boldsymbol{\mu}, \boldsymbol{\Gamma}) &= \{ \boldsymbol{x} \in \mathbb{R}^n | \exists \widetilde{\boldsymbol{x}} \in \mathbb{R}^n, \boldsymbol{x} = \boldsymbol{\mu} + \boldsymbol{\Gamma} \cdot \widetilde{\boldsymbol{x}}, \|\widetilde{\boldsymbol{x}}\| \leq 1 \} \\ &= \boldsymbol{\mu} + \boldsymbol{\Gamma} \cdot \mathcal{E}(\boldsymbol{0}, \boldsymbol{I}_n) \end{aligned} \tag{3}$$

A degenerate ellipsoid is an ellipsoid whose shape matrix $\boldsymbol{\Gamma}$ is not invertible. While degenerate ellipsoids can't be described by the quadratic form (2), they can be defined as a singular affine transformation with (3).

Given the initial domain $\boldsymbol{x} \in \mathcal{E}(\boldsymbol{\mu}_x, \boldsymbol{\Gamma}_x)$, the propagation of the ellipsoids consists in finding an enclosing ellipsoid $\mathcal{E}(\boldsymbol{\mu}_y, \boldsymbol{\Gamma}_y)$ such that $\boldsymbol{y} \in \mathcal{E}(\boldsymbol{\mu}_y, \boldsymbol{\Gamma}_y)$

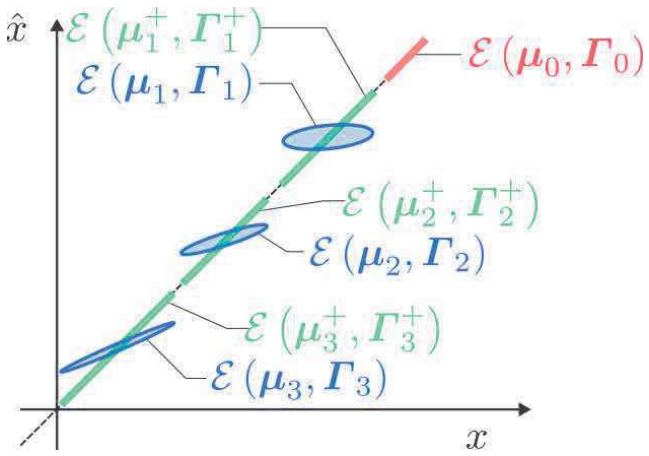



Figure 1: Ellipsoidal propagation for the system (4) with $\boldsymbol{x}\left(t_k\right) \in \mathcal{E}\left(\boldsymbol{\mu}_k, \boldsymbol{\Gamma}_k\right)$ and $\boldsymbol{x}\left(t_k^+\right) \in \mathcal{E}\left(\boldsymbol{\mu}_k^+, \boldsymbol{\Gamma}_k^+\right)$ for $k \in \{0, 1, 2, 3\}$

## Main results

While the propagation of ellipsoids with linear mapping is straightforward, some non-linear propagation methods as been proposed in the literature, such as [4]. This work proposes modifications of the method [4] to solve hybrid non-linear systems in all cases.

Here is a common example of a system with degenerate ellipsoids

$$\begin{aligned} \dot{x} &= f\left(x, \hat{x}, \alpha\right) \\ \dot{\hat{x}} &= f\left(\hat{x}, \hat{x}, 0\right) \\ \hat{x}\left(t_k^+\right) &= x\left(t_k\right) \\ t_k &= k \cdot T \end{aligned} \tag{4}$$

with the period $T > 0$ and where $\boldsymbol{x} \in \mathbb{R}^n$ is the state vector, $\hat{\boldsymbol{x}} \in \mathbb{R}^n$ is the state estimation and $\boldsymbol{\alpha} \in \mathbb{R}^n$ is a constant disturbance. The measurements at the times $t_k^+$ make the ellipsoids degenerate, as illustrated by Figure 1.

### Acknowledgement

This work has been supported by the Brittany region and the French Defense Innovation Agency (AID)

# References

[1] Matthias Althoff and Jagat Jyoti Rath. Comparison of guaranteed state estimators for linear time-invariant systems. *Automatica*, 130:109662, August 2021.

[2] Auguste Bourgois, Amine Chaabouni, Andreas Rauh, and Luc Jaulin. Proving the stability of the rolling navigation.

[3] Nacim Ramdani and Nedialko S. Nedialkov. Computing reachable sets for uncertain nonlinear hybrid systems using interval constraint propagation techniques. *IFAC Proceedings Volumes*, 42(17):156–161, January 2009.

[4] Andreas Rauh and Luc Jaulin. A computationally inexpensive algorithm for determining outer and inner enclosures of nonlinear mappings of ellipsoidal domains. *International Journal of Applied Mathematics and Computer Science*, 31(3):399–415, 2021.

[5] M. Trinh, Shiyu Zhao, Zhiyong Sun, D. Zelazo, B. Anderson, and H. Ahn. Bearing-Based Formation Control of a Group of Agents With Leader-First Follower Structure. *IEEE Transactions on Automatic Control*, 2019.

[6] Guoqing Xia, Yu Zhang, and Ying Yang. Control Method of Multi-AUV Circular Formation Combining Consensus Theory and Artificial Potential Field Method. In *2020 Chinese Control And Decision Conference (CCDC)*, pages 3055–3061, August 2020. ISSN: 1948-9447.

[7] Shiyu Zhao and D. Zelazo. Bearing Rigidity and Almost Global Bearing-Only Formation Stabilization. *IEEE Transactions on Automatic Control*, 2016.

# Set-Based Simulation Approaches for the Numerical Analysis of Dynamic Spiking Neuron Models

Andreas Rauh[1] and Jutta Kretzberg[2]

[1] Carl von Ossietzky Universität Oldenburg
Distributed Control in Interconnected Systems
26111 Oldenburg, Germany
andreas.rauh@uol.de
[2] Carl von Ossietzky Universität Oldenburg
Computational Neuroscience
26111 Oldenburg, Germany
jutta.kretzberg@uol.de

**Keywords:** Reachability analysis, Set-based integration, Neuron models, Remainder-form decomposition

## Introduction

The Hodgkin-Huxley model [1,2] is one of the most well-known models for the description of the current flow through ion-selective channels in neural membranes. Although this model is capable of representing a large number of dynamic effects such as spiking phenomena, the accurate identification of the associated parameters is inherently complex due to the fact that the required invasive measurements inevitably disturb the dynamic effects to be identified. Therefore, this contribution investigates different state-of-the-art approaches for a verified simulation of the Hodgkin-Huxley model, analyzes thoroughly the reason why they fail to determine enclosures for the temporal evolution of the state trajectories, and finally proposes novel enclosure techniques that remove the aforementioned shortcomings.

## Modeling of Intracellular Membrane Potentials Using the Hodgkin-Huxley Model

In their original work, Hodgkin and Huxley succeeded in describing the temporal behavior of intracellular membrane potentials by a set of coupled nonlinear ordinary differential equations for which they received the Nobel Prize in Physiology or Medicine in 1963. The differential equations of their model describe the variation of the electric voltage $v$ at the lipid bilayer represented by an electric capacitance as well as by accounting for voltage-gated ion channels [2].

For the latter, the additional state variables $n$, $m$, and $h$ are included in the model. These variables describe normalized probabilities associated with potassium channel subunit activation, sodium channel subunit activation, and sodium channel subunit inactivation, respectively. From a system theoretic point of view, the electric current $i$ is included in the model as a control input. Throughout this paper, the parameterization of the Hodgkin-Huxley model according to [1] is considered.

According to the discussion above, the state equations are given by

$$\begin{aligned} \dot{v} &= g_{\mathrm{Na}} m^3 h \cdot (e_{\mathrm{Na}} - (v+65)) + g_{\mathrm{K}} n^4 \cdot (e_{\mathrm{K}} - (v+65)) \\ &\quad + g_{\mathrm{L}} \cdot (e_{\mathrm{L}} - (v+65)) + i \\ \dot{m} &= \alpha_{\mathrm{M}}(v) \cdot (1-m) - \beta_{\mathrm{M}}(v) \cdot m \\ \dot{h} &= \alpha_{\mathrm{H}}(v) \cdot (1-h) - \beta_{\mathrm{H}}(v) \cdot h \\ \dot{n} &= \alpha_{\mathrm{N}}(v) \cdot (1-n) - \beta_{\mathrm{N}}(v) \cdot n \end{aligned} \tag{1}$$

with the voltage-dependent nonlinearities

$$\begin{aligned} \alpha_{\mathrm{M}}(v) &= \frac{2.5 - 0.1 \cdot (v+65)}{e^{2.5-0.1\cdot(v+65)} - 1}\ , \quad \alpha_{\mathrm{H}}(v) = 0.07 \cdot e^{-\frac{v+65}{20}}\ , \\ \alpha_{\mathrm{N}}(v) &= \frac{0.1 - 0.01 \cdot (v+65)}{e^{1-0.1\cdot(v+65)} - 1}\ , \quad \beta_{\mathrm{M}}(v) = 4 \cdot e^{-\frac{v+65}{18}}\ , \\ \beta_{\mathrm{H}}(v) &= \frac{1}{e^{3.0-0.1\cdot(v+65)} + 1}\ , \quad \beta_{\mathrm{N}}(v) = 0.125 \cdot e^{-\frac{v+65}{80}}\ . \end{aligned} \tag{2}$$

# Set-Based Simulation Approaches

Widely used set-based simulation approaches for dynamic systems represented in the form of ordinary differential equations typically rely on temporal Taylor series expansions of the solution with interval-valued remainders that are determined in such a way that the resulting discretization and method errors are included in a guaranteed manner [4]. Extensions of these approaches make use of so-called Taylor models in which the solutions are embedded into high-order polynomial enclosures. However, the application of both techniques as implemented in the MATLAB toolbox INTLAB 12 fails to compute guaranteed enclosures. This is either due to the failure in determining suitable integration step sizes (for low-order series expansions) or due to overestimation in the computed bounds for the Taylor coefficients (for higher order expansions) leading to divisions by zero.

Unfortunately, the Hodgkin-Huxley model violates also the sufficient criterion for cooperativity [5] which would allow for evaluating lower and upper bounding trajectories for the domains of reachable states in a decoupled manner. Moreover, also the exponential enclosure technique [6], as proposed by the first author of this contribution, is not efficiently applicable due to the fact that the linearized dynamics of this model are characterized by eigenvalues with positive real parts during the transient regime.

Therefore, this contribution proposes the following novel alternatives that allow to resolve the problems discussed above:

- Subdivision of the nonlinear model into a cooperative part and a set-based remainder, where the remainder is determined by exploiting a-priori known bounds for the range of the voltage-dependent functions (2). This approach is closely related to the remainder-form decomposition published in [3];

- Refinement of the bounds determined on the basis of the remainder-form decomposition above by implementing an iterative tightening scheme for the remainder bounds.

With the help of the verified enclosures of the state trajectories according to the proposed procedure, this contribution provides the basis for novel procedures for uncertainty quantification of dynamic models in computational neuroscience, where the point-valued parameters in (1)–(2) will be replaced by interval variables in future work.

## References


[1] R. Chandra, `hhrun`: Hodgkin Huxley Model Simulation for User Defined Input Current, `www.mathworks.com/matlabcentral/fileexchange/46740-hhrun-hodgkin-huxley-model-simulation-for-user-defined-input-current`, accessed: June 5, 2023.

[2] A.L. Hodgkin and A.F. Huxley, A Quantitative Description of Membrane Current and its Application to Conduction and Excitation in Nerve, *The Journal of Physiology* 117(4), 500–44, 1952.

[3] M. Khajenejad and S.Z. Yong, Tight Remainder-Form Decomposition Functions With Applications to Constrained Reachability and Guaranteed State Estimation, *IEEE Transactions on Automatic Control*, 2023. `doi.org/10.1109/TAC.2023.3250515`

[4] N.S. Nedialkov. *Interval Tools for ODEs and DAEs.* In Proceedings of the 12th GAMM-IMACS Intl. Symposium on Scientific Computing, Computer Arithmetic, and Validated Numerics SCAN 2006, Duisburg, Germany, IEEE Computer Society, 2007.

[5] T. Raïssi and D. Efimov. Some Recent Results on the Design and Implementation of Interval Observers for Uncertain Systems, *at-Automatisierungstechnik*, 66, 213–224, 2018.

[6] A. Rauh, R. Westphal, H. Aschemann and E. Auer. *Exponential Enclosure Techniques for Initial Value Problems with Multiple Conjugate Complex Eigenvalues.* In International Symposium on Scientific Computing, Computer Arithmetic, and Validated Numerics, 247–256, Springer, Cham, 2015.

# Computing invariants for discrete-time dynamical systems using polynomial zonotopes

Nathan Chiche, Eric Goubault and Sylvie Putot

Laboratoire d'informatique de l'Ecole Polytechnique
Palaiseau, France
{putot,goubault,chiche}@lix.polytechnique.fr

**Keywords:** Set-based methods, polynomial zonotopes, invariants

## Introduction

Invariants are a key notion in a lot of scientific fields such as mathematics, physics, biology and computer science. Our current work consists in the generation of numerical invariants for discrete dynamical systems, using a Kleene-like iteration over the domain of polynomial zonotopes [2], extending similar work on zonotopes, the geometric concretization of affine forms [3].

## Background

Polynomial zonotopes are a natural extension of zonotopes introduced in [1] in the context of reachability analysis of hybrid systems with nonlinear continuous dynamics.

**Definition 1** (Polynomial zonotope)**.** *A polynomial zonotope is the image of an n-dimensional hypercube by a vector valued polynomial function. In other words, $PZ \subset \mathbb{R}^p$ is a polynomial zonotope means that there exists a function $f : \mathbb{R}^n \to \mathbb{R}^p$ where every component of $f$ is a multivariate polynomial such that $PZ = f([-1; 1]^n)$*

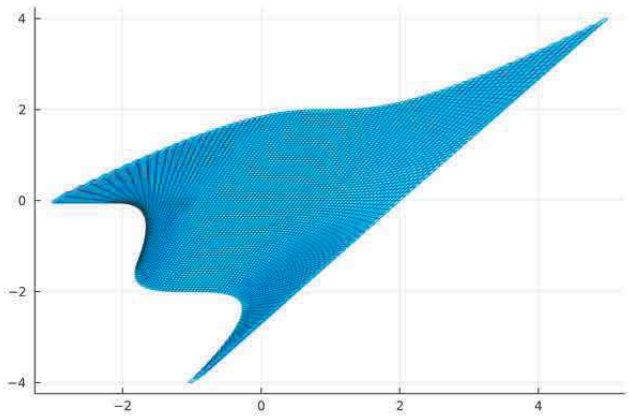



Figure 1: An example of polynomial zonotope

Figure 1 shows a polynomial zonotope that is defined by:

$$PZ = \left\{ \begin{pmatrix} 2\varepsilon_1 + \varepsilon_2 + 2\varepsilon_1^3\varepsilon_2 \\ 2\varepsilon_2 + 2\varepsilon_1^3\varepsilon_2 \end{pmatrix} \middle| (\varepsilon_1, \varepsilon_2) \in [-1; 1]^2 \right\}$$

The following Kleene's theorem explains how to compute a least-fixed point in a complete partially ordered set. It will motivate us to do value iteration using domains of specific sets.

**Theorem 1** (Kleene)**.** *If $f \in X \to X$ is a continuous operator in a complete partially ordered set $(X, \sqsubseteq, \sqcup, \bot)$, then $lfp(f)$ exists. Moreover, $lfp(f) = \sqcup\{f^i(\bot) | i \in \mathbb{N}\}$*

In general, we will rather compute a post fixed-point than a least fixed-point since we cannot iterate a infinite number of times.

## First Results

In the same way as [3], we want to use such a value iteration to generate invariants. However, similarly to the case of zonotopes, the domain of polynomial zonotopes is not a complete partially ordered set (least upper bound does not exist). We should at least provide a binary join

operator and a containment check for polynomial zonotopes. The join operator between two polynomial zonotopes should result in another polynomial zonotope which contains the other two (because giving the union cannot be expressed in a polynomial zonotope representation).

Another important part is to prove that our join has good properties regarding the convergence of value iteration (we did not investigate this part yet).

**Definition 2** (join operator)**.** *Let $A$ and $B$ denote two polynomial zonotopes*

$$A = \left\{ \begin{pmatrix} P_1(\varepsilon) \\ \vdots \\ P_p(\varepsilon) \end{pmatrix} \middle| \varepsilon \in [-1;1]^n \right\} \text{ and } B = \left\{ \begin{pmatrix} Q_1(\varepsilon) \\ \vdots \\ Q_p(\varepsilon) \end{pmatrix} \middle| \varepsilon \in [-1;1]^n \right\}$$

*We denote the join of $A$ and $B$ by:*

$$A \sqcup B = \left\{ \begin{pmatrix} (\frac{1}{2}\varepsilon_{n+1} + \frac{1}{2})Q_1(\varepsilon) + (\frac{1}{2} - \frac{1}{2}\varepsilon_{n+1})P_1(\varepsilon) \\ \vdots \\ (\frac{1}{2}\varepsilon_{n+1} + \frac{1}{2})Q_p(\varepsilon) + (\frac{1}{2} - \frac{1}{2}\varepsilon_{n+1})P_p(\varepsilon) \end{pmatrix} \middle| \varepsilon \in [-1;1]^n, \varepsilon_{n+1} \in [-1;1] \right\}$$

It is indeed a join since evaluating $\varepsilon_{n+1}$ in $-1$ gives every point in A and doing so in 1 gives every point in B, hence $A \cup B \subseteq A \sqcup B$. It only introduces one new variable but makes the total degree of each polynomial grow by one.

We need a stopping criterion in order to stop our value-iteration in the case a fixpoint is reached, which implies deciding whether a polynomial zonotope is included in another one.
Kochdumper introduced in his thesis a test for containment check. It consists in splitting a polynomial zonotope in several smaller ones, enclose these ones in boxes and finally check whether these boxes are included in another polynomial zonotope. However, this method does not provide a complete decision procedure which makes it unsuitable for our use.

We want a method to check if $f(A \sqcup B) \subseteq A \sqcup B$. Let $Q_1, \dots Q_p$ denote the polynomials involved in the left-hand side and $P_1, \dots P_p$

the ones for the other one ($A \sqcup B = \{(P_1(\varepsilon, \varepsilon_{n+1}), \dots P_p(\varepsilon, \varepsilon_{n+1}))^T | \varepsilon \in [-1; 1]^n, \varepsilon_{n+1} \in [-1; 1]\}$. Being interested in input/output relations, we need to check whether the following system has solutions in $\varepsilon'_{n+1}$ belonging to $[-1; 1]$

$$\begin{cases} Q_1(\varepsilon, \varepsilon_{n+1}) - P_1(\varepsilon, \varepsilon'_{n+1}) = 0 \\ \vdots \\ Q_p(\varepsilon, \varepsilon_{n+1}) - P_p(\varepsilon, \varepsilon'_{n+1}) = 0 \end{cases} . \tag{1}$$

which is equivalent to $\sum_{i=1}^{p} (Q_i \varepsilon, \eta') - P_i(\varepsilon, \eta))^2 = 0$

For this purpose, we want to make use of Sturm sequences which give a criteria to find real roots of a given polynomial within a certain interval. The Sturm theorem states that for $P = ax^2 + bx + c$ square-free, if the number of sign changes in $(sg(a - b + c), sg(-2a + b), sg(b^2/4a - c))$ is bigger than the number of sign changes in $(sg(a + b + c), sg(2a + b), sg(b^2/4a - c))$. In our case, $a, b, c$ are multivariate polynomials. We can determine their sign by using Bernstein expansion if the over-approximation does not contain 0. When this does not work, we can use SOS programming to solve the problem.

## References

[1] M. Althoff, *Reachability Analysis of Nonlinear Systems using Conservative Polynomialization and Non-Convex Sets*, Hybrid Systems: Computation and Control, 2013, pp. 173–182.

[2] N. Kochdumper and M. Althoff, *Sparse Polynomial Zonotopes: A Novel Set Representation for Reachability Analysis*, Hybrid Systems: Computation and Control, pp. 173–182, 2013.

[3] E. Goubault and S. Putot, *A zonotopic framework for functional abstractions*, Formal Methods in System Design volume 47, pages 302–360, 2015.

# Number of connected components using interval analisys

Hugo Rémin, Sébastien Lagrange

LARIS, France

**Keywords:** Topology, Connected Components, Contractibility

## Introduction

In topology, one of the simplest invariant is the number of components connected by arc [1]. This problem is relevant in the field of robotics for path planning. In this presentation we introduce a novel method counting the number of connected components of a set defined by inequalities. The main idea is to decompose the set in contractible subsets using interval analisys and then to reconnect the subsets using a graph. This method proposes an amelioration of the algorithm described in [2]. Next is an example to illustrate the method.

## Illustrative example

Given a set $S$ defined by : $\begin{cases} x^2 + y^2 - 6 \leq 0 \\ 0.2\cos(x - y) - \sin(yx) - 0.6 \leq 0 \end{cases}$

Using SIVIA we can visualize $S$ and see that it has three connected components, one big in the center and two small on the upper left and bottom right respectively.

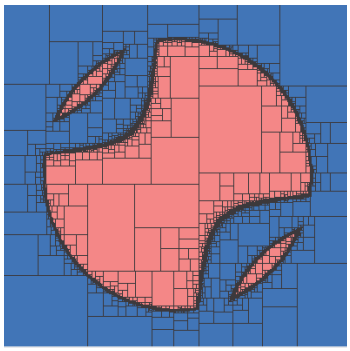


Figure 1: Set $S$ in red

The first step of the proposed method uses a splitting algorithm to decompose the space in three domains (blue/green/red). The ouptut of the first step is given in Figure 2. In blue we have the boxes that does not contain any points of $S$, in red boxes that are included in $S$ and in green, the boxes that are locally contractible to a point.

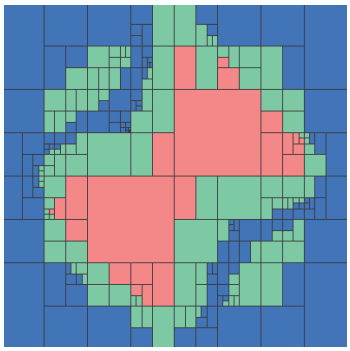


Figure 2: Result of the splitting algorithm

The second step consist in creating a graph from the previous decomposition where each neighboring green boxes are connected if they are contractible at a same point (see Figure 3) . Finally, the problem of counting the number of connected components of the set $S$ now becomes the problem of counting the number of connected component in a graph which is solvable in polynomial time [3].

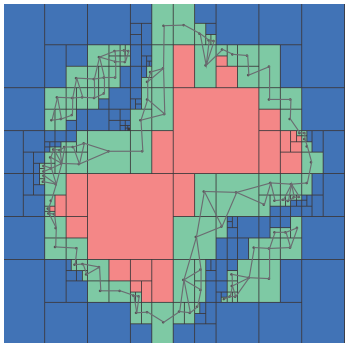


Figure 3: Resulting graph from the decomposition

## Robotic application

In robotics a concept commonly used is the configuration space [4] in which each point is associated with a position for the robot. Knowing the topology of this space is useful for path planning, with our method we can decide wether a path between two point is feasible (i.e. in the same connected component) and give us a path when one is feasible. The graph obtained from our algorithm is sufficiently rich to be used as a roadmap and using a shortest path algorithm like Dijkstra's [5] we can obtain a path between two points in the same connected component.

## References

**[1]** Janich, K., *Topology (Undergraduate Texts in Mathematics)*, Springer Verlag, 1980.

**[2]** Delanoue, N. and Jaulin, L. and Cottenceau, B., Counting the Number of Connected Components of a Set and Its Application to Robotics, *Applied Parallel Computing* vol. 3732, 93-101, 2006.

**[3]** Tarjan, E., Depth-first search and linear graph algorithms, *SIAM Journal on Computing* vol. 1, 146-160, 1972.

[**4**] LaValle, S., *Planning Algorithms*, Cambridge University Press, 2006.

[**5**] Dijkstra, E. W., A note on two problems in connexion with graphs, *Numerische Mathematik* vol. 1, 269–271, 1959.

# Imprecise Bayesian Neural Networks

Michele Caprio[1], Souradeep Dutta[1], Kuk Jang[1], Vivian Lin[1], Radoslav Ivanov[2], Oleg Sokolsky[1], Insup Lee[1]

[1] University of Pennsylvania, PRECISE Center, Department of Computer and Information Science
3330 Walnut Street, 19104 Philadelphia, Pennsylvania, USA
{caprio,duttaso,jangkj,vilin,sokolsky,lee}@seas.upenn.edu
[2] Rensselaer Polytechnic Institute, Department of Computer Science
110 Eighth Street, 12180 Troy, New York, USA
ivanor@rpi.edu

**Keywords:** Bayesian deep learning; imprecise probabilities; credal sets; epistemic and aleatory uncertainties; uncertainty quantification; machine learning robustness.

## Abstract

Uncertainty quantification and robustness to distribution shifts are important goals in machine learning (ML) and artificial intelligence (AI). Although Bayesian neural networks (BNNs [7]) allow for uncertainty in the predictions to be assessed, different sources of uncertainty are indistinguishable. We introduce imprecise Bayesian neural networks (IBNNs), a new deep learning technique grounded in imprecise probability (IP) theory [13]. Unlike other techniques in ML and AI involving IPs [2, 10] – that typically only focus on classification problems – IBNNs can be used for classification, prediction, and regression. They capture the ambiguity [4, 5] the designer faces when selecting which prior to choose for the parameters of a Bayesian neural network and which likelihood distribution to choose for the training data at hand. An IBNN can be defined as a stochastic NN trained using finitely generated credal sets (FGCSs) of priors and likelihoods. FGCSs are convex sets of probability measures having finitely many extreme points; we

use them to train IBNNs in order to overcome some of the drawbacks of BNNs. In particular, FGCSs allow to counter the criticism to the practice in (standard) Bayesian statistics of (i) using a single, arbitrary prior to represent the initial state of ignorance of the agent, (ii) using non-informative priors to model ignorance, and (iii) using a single, arbitrary likelihood to represent the agent's knowledge about the sampling model. IBNNs allow to achieve robustness in the sense of Bayesian sensitivity analysis [1], and to quantify and distinguish between epistemic and aleatoric uncertainties [6]. This is desirable in light of several areas of recent ML research, such as Bayesian deep learning [3, 9], adversarial example detection [12], and data augmentation in Bayesian classification [8]. IBNNs can also be used to compute sets of outcomes that enjoy PAC-like properties [11]. We apply them to two case studies. One, to model blood glucose and insulin dynamics for artificial pancreas control, and two, for motion prediction in autonomous driving scenarios. We show that IBNNs performs better when compared to an ensemble of BNNs benchmark.

**Acknowledgement**

The authors would like to acknowledge partial funding by the Army Research Office (ARO MURI W911NF2010080).

# References

[1] James O. Berger. The robust Bayesian viewpoint. In Joseph B. Kadane, editor, *Robustness of Bayesian Analyses*. Amsterdam : North-Holland, 1984.

[2] Giorgio Corani, Alessandro Antonucci, and Marco Zaffalon. *Bayesian Networks with Imprecise Probabilities: Theory and Application to Classification*, chapter 4 of Data Mining: Foundations and Intelligent Paradigms: Volume 1: Clustering, Association and Classification, pages 49–93. Berlin, Germany : Springer, 2012.

[3] Stefan Depeweg, Jose-Miguel Hernandez-Lobato, Finale Doshi-Velez, and Steffen Udluft. Decomposition of uncertainty in Bayesian deep learning for efficient and risk-sensitive learning. In *International Conference on Machine Learning*, pages 1184–1193. PMLR, 2018.

[4] Daniel Ellsberg. Risk, ambiguity, and the Savage axioms. *The Quarterly Journal of Economics*, 75(4):643–669, 1961.

[5] Itzhak Gilboa and Massimo Marinacci. Ambiguity and the Bayesian paradigm. In Daron Acemoglu, Manuel Arellano, and Eddie Dekel, editors, *Advances in Economics and Econometrics, Tenth World Congress*, volume 1. Cambridge : Cambridge University Press, 2013.

[6] Eyke Hüllermeier and Willem Waegeman. Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. *Machine Learning*, 3(110):457–506, 2021.

[7] Laurent Valentin Jospin, Hamid Laga, Farid Boussaid, Wray Buntine, and Mohammed Bennamoun. Hands-on Bayesian neural networks – A tutorial for deep learning users. *IEEE Computational Intelligence Magazine*, 17(2):29–48, 2022.

[8] Sanyam Kapoor, Wesley J Maddox, Pavel Izmailov, and Andrew Gordon Wilson. On uncertainty, tempering, and data augmentation in bayesian classification. *arXiv preprint arXiv:2203.16481*, 2022.

[9] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in neural information processing systems*, 30, 2017.

[10] Shireen Kudukkil Manchingal and Fabio Cuzzolin. Epistemic deep learning. *Available at arxiv:2206.07609*, 2022.

[11] Sangdon Park, Osbert Bastani, Nikolai Matni, and Insup Lee. PAC confidence sets for deep neural networks via calibrated prediction. In *8th International Conference on Learning Representations*, 2020.

[12] Lewis Smith and Yarin Gal. Understanding measures of uncertainty for adversarial example detection. *arXiv preprint arXiv:1803.08533*, 2018.

[13] Peter Walley. *Statistical Reasoning with Imprecise Probabilities*, volume 42 of *Monographs on Statistics and Applied Probability*. London : Chapman and Hall, 1991.

# TABLE DES AUTEURS

## *K*



## *L*



## *M*



## *P*



## *R*



## *S*



## *T*



## *V*



## *W*