SNE SIMULATION NOTES EUROPE

2nd SNE Special Issue
EUROSIM Congress 2019

Journal on Developments and Trends in Modelling and Simulation

EUROSIM Scientific Membership Journal

Vol. 32 No.1, March 2022
ISSN Online 2306-0271
ISSN Print 2305-9974
DOI 10.11128/sne.32.1.1059
ISBN Print 978-3-903311-30-5

TU Verlag an der Technischen Universität Wien

ARGESIM

SNE - Aims and Scope

Simulation Notes Europe (SNE) provides an international, high-quality forum for presentation of new ideas and approaches in simulation - from modelling to experiment analysis, from implementation to verification, from validation to identification, from numerics to visualisation - in context of the simulation process.

SNE seeks to serve scientists, researchers, developers and users of the simulation process across a variety of theoretical and applied fields in pursuit of novel ideas in simulation and to enable the exchange of experience and knowledge through descriptions of specific applications. **SNE** follows the recent developments and trends of modelling and simulation in new and/or joining application areas, as complex systems and big data. **SNE** puts special emphasis on the overall view in simulation, and on comparative investigations, as benchmarks and comparisons in methodology and application. For this purpose, **SNE** documents the **ARGESIM Benchmarks** on *Modelling Approaches and Simulation Implementations* with publication of definitions, solutions and discussions. SNE welcomes also contributions in education in/for/with simulation.

A *News Section* in **SNE** provides information for **EUROSIM** Simulation Societies and Simulation Groups.

SNE, primarily an electronic journal, follows an open access strategy, with free download in basic layout. **SNE** is the official membership journal of **EUROSIM**, the *Federation of European Simulation Societies and Simulation Groups* – www.eurosim.info. Members of **EUROSIM** societies are entitled to download **SNE** in an elaborate and extended layout, and to access additional sources of benchmark publications, model sources, etc. **Print SNE** is available for specific groups of **EUROSIM** societies, and starting with Volume 27 (2017) as print-on-demand from TU Verlag, TU Wien. **SNE** is DOI indexed by CrossRef, identified by DOI prefix 10.11128, assigned to the **SNE** publisher **ARGESIM** (*www.argesim.org*).

Author's Info. Individual submissions of scientific papers are welcome, as well as post-conference publications of contributions from conferences of **EUROSIM** societies. **SNE** welcomes special issues, either dedicated to special areas and/or new developments, or on occasion of events as conferences and workshops with special emphasis.

Authors are invited to submit contributions which have not been published and have not being considered for publication elsewhere to the **SNE** Editorial Office.

SNE distinguishes different types of contributions (*Notes*), i.e.

- **TN** Technical Note, 6–10 p.
- **SN** Short Note, max. 5 p.
- **SW** Software Note , 4–6 p.
- **BN** Benchmark Note, 2–10 p.
- **ON** Overview Note – only upon invitation, up to 14 p.
- **EN** Education Note, 6–8 p.
- **PN** Project Note 6–8 p.
- **STN** Student Note, 4-6 p., on supervisor's recommendation
- **EBN** Educational Benchmark Note, 4–10 p.

Further info and templates (doc, tex) at **SNE**'s website.

www.sne-journal.org

SNE Editorial Board

SNE - Simulation Notes Europe is advised and supervised by an international scientific editorial board. This (increasing) board is taking care on peer reviewing of submission to SNE:

Editorial

***Dear Readers**, This first issue of SNE Vol. 32, 2022, the 2nd SNE Special Issue 'EUROSIM Congress 2019', continues publishing selected contributions presented at the EUROSIM Congress, July 2019, Logrono, Spain. The EUROSIM Congress is the tri-annual family meeting of the European simulation societies. The Special Issue Editorial Board suggested contributions from EUROSIM 2019 with mainly simulation-oriented topics, and with a variety in theory and application. We were surprised by the very positive reactions of many authors, who have sent their revised contribution for publication in SNE, seven published in this issue, and in sum nineteen outstanding contributions from the EUROSIM 2019 (details see Special Issue Editorial), an excellent promotion for the next EUROSIM Congress 2023, June 2023, Amsterdam – info at www.eurosim2023.eu*

This issue also continues a SNE tradition: as with SNE Vol. 30, 28, 25, 23, and Vol. 21, Vlatco Čerić, Past President of the Croatian Simulation Society, provides his artwork as cover pictures for SNE Vol. 32, 2022. 'Algorithms, mathematics and art are interrelated in an art form called algorithmic art. Algorithmic art is visual art generated by algorithms that completely describe creation of images. This kind of art is strongly related with contemporary computer technology, and especially computer programming, as well as with mathematics used in algorithms for image generation' – as Čerić defines (vceric.net). For this volume, the artist and simulationist Vlatko Čerić has chosen four algorithmic art pictures from the series AMULETS. Furthermore, Vlatco Čerić's algorithmic art pictures are a central impact for the posters and publication covers of two main simulation events in 2022: MATHMOD 2022 and ASIM 2022, both in July 2022 at TU Vienna – www.mathmod.at, www.asim-gi.org/asim2022

We thank the special issue editors for their excellent editorial work (for details, see the special issue editorial). I would like to thank all authors for their contributions to SNE 32(1) showing the broad variety of simulation – and the success of the EUROSIM Congresses. And last but not least thanks to the SNE Editorial Office for layout, typesetting, preparations for printing, electronic publishing, and much more.

Felix Breitenecker, SNE Editor-in-Chief, eic@sne-journal.org; felix.breitenecker@tuwien.ac.at

Contents SNE 32(1)
2nd Special Issue *EUROSIM Congress 2019*

Online SNE 32(1), DOI 10.11128/sne.32.1.1059
ARGESIM Publisher, Vienna, www.argesim.org
Print SNE 32(1) 978-3-903311-30-5
TU Verlag Vienna, Print-on-Demand, www.tuverlag.at

SNE Contact & Info

SNE Online ISSN 2306-0271, SNE Print ISSN 2305-9974

→ ***www.sne-journal.org***

office@sne-journal.org, eic@sne-journal.org

SNE Editorial Office
Johannes Tanzler (Layout, Organisation),
Irmgard Husinsky (Web, Electronic Publishing),
Felix Breitenecker (Organisation, Author Mentoring)
ARGESIM / Math. Modelling & Simulation Group,
Inst. of Analysis and Scientific Computing, TU Wien
Wiedner Hauptstrasse 8-10, 1040 Vienna, Austria

SNE SIMULATION NOTES EUROPE

WEB: → *www.sne-journal.org*, DOI prefix 10.11128/sne

Scope: Developments and trends in modelling and simulation in various areas and in application and theory; comparative studies and benchmarks (documentation of ARGESIM Benchmarks on modelling approaches and simulation implementations); modelling and simulation in and for education, simulation-based e-learning; society information and membership information for EUROSIM members (Federation of European Simulation Societies and Groups).

Editor-in-Chief: Felix Breitenecker, TU Wien, Math. Modelling Group
Felix.Breitenecker@tuwien.ac.at, *eic@sne-journal.org*

Print SNE and Print-on-Demand: TU-Verlag, Wiedner Hauptstrasse 8-10, 1040, Vienna, Austria – *www.tuverlag.at*

Publisher: ARGESIM ARBEITSGEMEINSCHAFT SIMULATION NEWS
c/o Math. Modelling and Simulation Group, TU Wien / 101, Wiedner Hauptstrasse 8-10, 1040 Vienna, Austria;
www.argesim.org, *info@argesim.org*
on behalf of ASIM *www.asim-gi.org* and EUROSIM → *www.eurosim.info*

Editorial SNE 32(1) – 2nd Special Issue *EUROSIM Congress 2019*

SNE 32(1), the **2nd SNE Special Issue** '*EUROSIM Congress 2019'*, comprises a selection of outstanding contributions of the **EUROSIM Congress 2019** which took place in July 2019 University of Logrono, La Rioja, Spain. The **EUROSIM Congress** is the tri-annual family meeting of the European simulation societies, among them **CEA-SMSG**, the Spanish, **ASIM** – the German, **DBSS** – The Dutch-Benelux, **SLOSIM** – the Slovenian, and **SIMS** – the Scandinavian simulation society.

For us, the members of the Special Issue Editorial Board, it was not an easy task to select contributions and suggest for post-conference publication in **SNE**: the area of modelling and simulation has become very broad, and so we decided for outstanding publications, which represent very different areas of modelling and simulation. After invitation for postconference publication in **SNE**, many authors have expressed their interest – and we were surprised by the very positive reactions of many authors, who have sent their revised contribution for publication in SNE: SNE 31(3), the (1st) SNE Special Issue 'EUROSIM Congress 2019 published eight congress contributions, this 2nd SNE Special Issue 'EUROSIM Congress 2019' publishes seven congress contributions, two congress contributions have already been published in SNE 29(4) and in SNE 31(4), and two further contributions are scheduled for SNE 32(2) – in sum nineteen outstanding contributions from the EUROSIM 2019, an excellent promotion for the EUROSIM Congress 2023, June 28-30, 2023, Amsterdam – www.eurosim2023.eu

For this **2nd SNE Special Issue** '*EUROSIM Congress 2019'* we had to decide for seven contributions, one *Overview Note* and six *Technical Notes*.

This issue stars with the Overview Note (an Invites Lecture to EUROSIM 2019 Congress) *Review on Monte Carlo Simulation Stopping Rules: How Many Samples Are Really Enough?* by M. Bicher, M. Wastian, D. Brunmeir, and N. Popper. The authors present four methods that allow calculating an optimal replication number for Monte Carlo simulation and getting an image about the error between the estimated and the real mean value. The methods are furthermore evaluated on a simple case study, a stochastic cellular automaton model for simulation of an infectious disease.

The series of *Technical Notes* starts with the contribution *Simulation Processes for Onboard State Estimation in a Small UAV Environment* by A.M. Omeri et al. It describes the processes for estimating and analysing the position states onboard small unmanned aerial vehicles in the low-altitude simulation environment. Those processes are explained using two simulation platforms.

The next two contributions deal with air traffic management. T. Koça and M.A. Piera report on *Guarantying Consistency of Spatio-temporal Regions that Solve Air Traffic Conflicts*. Among the various approaches towards this problem, the authors believe the ones that try to solve conflicts using spatio-temporal regions are the most adequate base for such systems, because of their unique ability to consider post-decitional uncertainties. And M. A. Piera et al. present *A Socio-technical Holistic ABM Simulation Framework to Assess Pilots Performance Variability* – a socio-technical approach to understand the causes of a degraded mode pilot performance while providing a simulation framework to predict the time windows at which supporting tools could be fired to lessen the pilot workload.

Next, J.L. Pitarch and C. de Prada Machine discuss in their contribution Machine *Learning and the Digital Era from a Process Systems Engineering Perspective* the particularities of large processing plant systems, that can make machine learning for general purpose fail in extracting the right information from data, leading thus to unreliable process models.

The following contribution *Extendable Hybrid Approach to Detect Conscious States in a CLIS Patient using Machine Learning* by Shang-Ju Wu et al. deals again with machine learning, but with a very different application. The authors propose a method for uncovering consciousness in complete locked-in syndrome (CLIS) patients – using a hybrid system based on the combination of complex coherence, sample entropy and Granger causality to uncover the underlying state of consciousness in a CLIS patient from electrocorticography signals.

The last contribution *Genetic Algorithms in the Domain of Personalized Nutrition* by P. Heinonen and E. K. Juuso makes use of AI techniques for analysing nutrition flow for personalised nutrition.

The editors express their gratitude to all authors for their great effort and cooperation. Last but not least the editors thank the SNE Editorial Office for the support in compiling this special issue. The editors hope that you will enjoy the **SNE Special Issue**, and that it will encourage you to participate actively in the next congress, which will take place in Amsterdam in summer 2023 – info see www.eurosim2022.eu/

Sincerely, the **SNE 32(1)** Special Issue Editors

Emilio Jiménez, Juan-Ignacio Latorre Biel (**CEA-SMSG**
Miguel Mujica-Mota (**DBSS**), *Borut Zupancic* (**SLOSIM**)
Esko Juuso (**SIMS**)
N. Popper, A. Körner, F. Breitenecker (**ASIM**)

Review on Monte Carlo Simulation Stopping Rules: How Many Samples Are Really Enough?

Martin Bicher[1*], Matthias Wastian[2], Dominik Brunmeir[2], Niki Popper[1,2,3]

[1] Institute for Information Software Engineering, TU Wien, Favoritenstraße 9–11, 1040 Vienna, Austria; *martin.bicher@tuwien.ac.at*

[2]dwh GmbH, Neustiftgasse 57–59,1070 Vienna, Austria;

[3]DEXHELPP, Association for Decision Support for Health Policy and Planning, Neustiftgasse 57–59, 1070 Vienna, Austria;

SNE 32(1), 2022, 1-8, DOI: 10.11128/sne.32.on.10591
Received: 2020-11-10 (selected EUROSIM 2019 Postconference Publication); Revised: 2021-05-10; Accepted: 2021-06-15
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna, ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. Due to extensive usage of stochastic simulation models correct execution of Monte Carlo simulation has become more and more important. Hereby the unknown real mean of the simulation result is estimated by the sample mean of a large number of simulation evaluations. Unfortunately, this procedure is often done carelessly. Modellers commonly use replication counts without scientific justification and sometimes underestimate the consequences of a bad or even wrong choice: if it is chosen too small, the sample mean is not a representative approximation for the regarded mean, and not only the simulation output, but also any kind of simulation analysis will not be representative at all. If the number is chosen too high, the Monte Carlo experiment will consume unnecessary computation time, which could, exemplarily, be invested into deeper model analysis instead. In this work, we present four methods that allow calculating an optimal replication number for Monte Carlo simulation and getting an image about the error between the estimated and the real mean value. The methods are furthermore evaluated on a simple case study, a stochastic cellular automaton model for simulation of an infectious disease.

Introduction

During the last decades stochastic microscopic simulation methods like agent-based modelling or discrete event simulation established as standard tools for decision support. Even though the actual distribution of the simulation results of such models carries extremely valuable information, usually the mean value of the simulation's state variable is regarded as *the* simulation output and approximated using the sample mean of Monte Carlo (MC) experiments with the simulation.

Unfortunately, this extremely important procedure is often regarded as trivial and therefore sidelined. Modellers often feel so annoyed by waiting for dozens of simulation runs to be finished and calculating the sample mean that they do not want to spend any effort on carefully choosing the only parameter of this procedure, namely the number of repetitions. Sentences like "*We repeated the simulation M times to flatten stochastic effects*" are often found in literature without any scientific justification, why M should be a meaningful number of repetitions. Some examples:

- [1] *Ten replications ... were performed ... Increasing the number of replications shows no significant effect on this error.*
- [2] *The averaging ... was carried out on* 500 *MC runs.*
- [3] *To reflect randomness 10 replicate runs were undertaken for each scenario, where the ... Random Seed parameter is changed in each replicate.*

It seems as if modellers are not really conscious about the consequences of a bad or even wrong choice: if M is chosen too small, the sample mean is not a representative approximation for the regarded mean, and not only the simulation output, but also any kind of simulation analysis (parameter variations, scenario tests, etc.) is not representative at all.

If the number is chosen too high, the Monte Carlo experiment will consume unnecessary computation time, which could, exemplary, be invested into deeper model analysis instead.

Summarising, modellers use highly sophisticated methods for identification, optimisation or calibration of model parameters, but still use a more or less arbitrary repetition count, probably some suitable power of ten, to perform the Monte Carlo experiment with the simulation. To overcome this imbalance, this work centers around the question: *How many times do I really need to repeat the simulation to make sure, the mean-value estimator lies within a certain tolerance from the real mean?*

Interpreting the single simulation-run results in a Monte Carlo simulation as a sequence of i.i.d. random numbers, the problem turns into a solely theoretic one. Hereby, the research question can be viewed completely detached from the process of Monte Carlo simulation and, on the first glance, appears like a problem that mathematicians should have solved long time ago: How well does the sample mean approximate the real one? Indeed a lot of researchers have successfully worked on it and found a lot of different formulas that describe the difference between those two quantities (see [4, 5, 6, 7, 8, 9]). What seems positive at first, contains one large disadvantage: *there is no unique answer to this problem*!

Consequently, modellers who start taking care about the correct number of replications for their simulation, not only have to search for a possibly very complicated formula, they also have to evaluate which of the derived concepts is the "best" or most "correct". In the course of this work, we take a closer look at the two most fundamental concepts to answer the research question and evaluated their value when applied to a Monte Carlo simulation case study.

1 Methods

Before going into details of the two mentioned methods, we clarify some notation which we will use henceforth. First of all, we will consider the output of the simulation as a random variable $X \in \mathbb{R}$ and define $X_i, i \in \mathbb{N}$ as i.i.d. copies of it. Clearly, this restricts the simulation output to be a scalar number, but most of the ideas can be extended to time-series or multidimensional output as well. As X is defined as the output of a simulation it is fair to assume that $\mu := \mathbb{E}(X)$ and $\sigma^2 := \mathbb{V}(X)$ are finite real numbers.

All methods in this work consider the empiric mean

$$\overline{X}_M := \frac{1}{M}\sum_{i=1}^{M}. \tag{1}$$

and the corresponding error $|\overline{X}_M - \mu|$, which itself is a random variable. Therefore, we cannot ask for a useful upper and lower bound for it, but seek for a confidence interval with predefined width δ_{abs} and confidence level p so that

$$P\left(|\overline{X}_M - \mu| \leq \delta_{\text{abs}}\right) \geq p. \tag{2}$$

Suppose, we are able to find a formula that describes the relation between sample size M, probability p and allowed absolute error δ_{abs}, we are able stop the Monte Carlo simulation whenever the allowed error δ_{abs} and the failure probability $(1-p)$ are sufficiently small. We combined these thoughts in a so called stopping rule:

Definition 1 (Stopping Rule)**.** *A real-valued function*

$$f : \mathbb{R}^+ \times (0,1) \times \mathbb{N} \to \mathbb{R} : (\delta_{abs}, p, M) \mapsto f(\delta_{abs}, p, M, \cdot)$$

is called stopping rule for the Monte Carlo simulation if

$$f(\delta_{abs}, p, M, \cdot) \geq 0 \Rightarrow P(\mu \in [\overline{X}_M - \delta_{abs}, \overline{X}_M + \delta_{abs}]) \geq p. \tag{3}$$

We will call the smallest positive integer M for which $f(\delta_{abs}, p, M, \cdot) \geq 0$ the stopping index of the stopping rule and use the label

$$M_{stop} = M_{stop}(\cdot). \tag{4}$$

The $\cdot$ notation indicates, that it is possible, that stopping index and stopping rule may depend on additional parameters, like (sample-) moments of the distribution (see later).

As mentioned, there are several approaches how such a stopping rule can be derived. We will take a look at the two fundamental ones.

1.1 Chebyshev Inequality Stopping Rule

The Chebyshev inequality or Bienaymé–Chebyshev inequality [10] was applied for determination of replication numbers in, exemplary, [11]. This inequality has become a standard tool in stochastics and gives a connection between a random number's variance and its expected value:

$$P(|X - \mathbb{E}(X)| \geq k) \leq \frac{\mathbb{V}(X)}{k^2}. \tag{5}$$

This inequality is valid for any random number X with finite expected value and variance and is independent of the distribution. Therefore it can be applied for the sample mean of independent simulation experiments. As $\mathbb{V}(\overline{X}) = \frac{\mathbb{V}(X)}{M} = \frac{\sigma^2}{M}$ and $\mathbb{E}(\overline{X}) = \mathbb{E}(X) = \mu$, we get

$$P(|\overline{X}_M - \mu| \geq k) \leq \frac{\sigma^2}{Mk^2} \Rightarrow P(|\overline{X}_M - \mu| < k) \geq 1 - \frac{\sigma^2}{Mk^2}. \tag{6}$$

With $k = \delta_{\text{abs}}$ and $1 - \frac{\sigma^2}{Mk^2} = p$ we get, that

Corollary 1 (Chebyshev Stopping Rule)**.**

$$f(\delta_{abs}, p, M, \sigma^2) = (1-p) - \frac{\sigma^2}{M\delta_{abs}^2}, \tag{7}$$

is a stopping rule for the Monte Carlo simulation, if X has a bounded first and second moment.

Unfortunately, the Chebyshev inequality is only sharp in very rare cases. Hence, we expect that the rule usually overestimates the iteration count.

1.2 Gauss-Distribution Stopping Rule

The second and even more frequently used stopping rule (e.g. [12] p. 119) is based on the Central Limit Theorem (CLT)

$$\frac{\overline{X}_M - \mu}{\sqrt{M}\sigma} \xrightarrow[M\to\infty]{P} Y \sim \mathcal{N}(1,0). \tag{8}$$

Consequently, in case M is large enough, $\overline{X}_M$ can be imagined as normally- (Gaussian-) distributed with parameters μ and $\sigma/\sqrt{M}$. Hence, we may use the percentiles of the standard normal distribution to estimate the probability. Let

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-s^2/2} ds \tag{9}$$

stand for the probability function of the normal distribution, then

$$P\left(\frac{\overline{X}_M - \mathbb{E}(\overline{X}_M)}{\sqrt{\mathbb{V}(\overline{X}_M)}} \leq k\right) = P\left(\frac{\sqrt{M}(\overline{X}_M - \mu)}{\sigma} \leq k\right) \approx \Phi(k). \tag{10}$$

and thus

$$P\left(|\overline{X}_M - \mu| \leq \frac{k\sigma}{\sqrt{M}}\right) \approx \Phi(k) - \Phi(-k) = 1 - 2\Phi(-k). \tag{11}$$

We conclude that

Corollary 2 (Gaussian Stopping Rule)**.**

$$f(\delta_{abs}, p, M, \sigma^2) = (1-p) - 2\Phi\left(-\frac{\sqrt{M}\delta_{abs}}{\sqrt{\sigma^2}}\right) \tag{12}$$

is an asymptotic stopping rule for the Monte Carlo simulation, in case X has a bounded first and second moment and M is sufficiently large.

Interestingly, the speed of convergence of the sample mean quantiles towards the quantiles of the Normal distribution depends on the skewness of the distribution of X (see Edgeworth Extension, [13] p. 538) and specifically if the distribution is symmetric or not. Consequently, the Gaussian estimator needs to be used with care if

- the number of samples M is small (e.g. if the allowed error is comparably large), and
- the distribution of X is skewed.

In these cases the Gaussian stopping rule might eventually underestimate the iteration count.

1.3 Estimation of the Variance

At this stage, neither of the two stopping rules can be applied directly as both depend on the unknown variance σ^2 of the random number, i.e. the fluctuations of the simulation output. Essential to overcome this problem is, that we do not necessarily need to know these quantities precisely, but need a feasible approximation for it. We need to make sure that $\forall p, \delta_{\text{abs}}, M : (\delta_{\text{abs}}, p, M, \sigma^2) \leq 0 \Rightarrow f(\delta_{\text{abs}}, p, M, s^2) \leq 0$ for a variance estimator s^2. In this case, the function remains a stopping rule and $M_{\text{stop}}(\sigma^2) \leq M_{\text{stop}}(s^2)$. This is achieved if either

(a) $s^2 \geq \sigma^2$ as the variance occurs in the denominator of both stopping rules, or

(b) $|s^2 - \sigma^2|$ is small enough to make sure that $M_{\text{stop}}(s_2) = M_{\text{stop}}(\sigma^2)$.

Ad (a) In case $X \in [a,b]$ for some known a,b, it is possible to state a crude upper bound for the variance. Knowing that the expected value μ minimises the mean-quadratic error $f(t) = \mathbb{E}(X-t)^2$, we get

$$\mathbb{V}(X) = \mathbb{E}(X-\mu)^2 \leq \mathbb{E}\left(X - \frac{b+a}{2}\right)^2 \leq \mathbb{E}\left(b - \frac{b+a}{2}\right)^2 = \frac{(b-a)^2}{4} =: s_c^2. \quad (13)$$

Unfortunately s_c^2 might overestimate the real variance by a large margin. Therefore the iteration count might also be overestimated. Moreover, if we cannot find an upper or lower bound for the simulation result, we cannot state a suitable upper bound for the variance as well which only leaves variant (b).

Ad (b) The most reliable estimator for the real variance is, of course, the sample variance

$$s_N^2 := \frac{1}{N-1}\sum_{i=1}^{N}\left(X_i - \overline{X}_N\right)^2, \quad (14)$$

which can be determined by N individual simulation replications. It makes sense including the evaluation of the sample variance into the Monte Carlo simulation itself, i.e. using $N = M$. Unfortunately, the sample variance gives no guarantee that the real variance is not underestimated, even if the iteration count is very large, which essentially makes every stopping rule an asymptotic one.

1.4 Application of Stopping Rules

Using crude variance bound s_c^2 mentioned above makes it possible to apply any of the two defined stopping rules (7) and (12) in advance. It is presented as pseudo-code in Algorithm 1.

Algorithm 1 Monte Carlo Simulation with A-Priori Variance Estimate

Require: $p \in [0,1], \delta_{\text{abs}} > 0, X \in [a,b]$
  $s_c^2 \leftarrow \frac{(b-a)^2}{4}$
  $M_{\text{stop}} \leftarrow \mathbf{argmin}_{M\in\mathbb{N}}\{f(\delta_{\text{abs}}, p, M, s_c^2) \geq 0\}$
  $X' \leftarrow 0$
  for $i \in [1, \ldots, M_{\text{stop}}]$ **do**
    $X_i \leftarrow$ Simulation()
    $X' \leftarrow X' + X_i$
  end for
  return X'/M_{stop}

Before stating the corresponding algorithm for applying stopping rules with sample moments, we need to mention, that both, sample mean and sample variance can be update dynamically. Clearly,

$$\overline{X}_{M+1} = \frac{M}{M+1}\overline{X}_M + \frac{1}{M+1}X_{i+1}. \quad (15)$$

Hence, $\overline{X}_{M+1}$ can be calculated from $\overline{X}_M$ with reasonable effort. Unfortunately this strategy can only be applied indirectly to the estimator of the sample variance. With $\overline{X^2}_M = \frac{1}{M}\sum_{i=1}^{M} X_i^2$,

$$\overline{s^2}_M = \frac{1}{M-1}\sum_{i=1}^{M}(X_i - \overline{X}_M)^2 = \frac{M}{M-1}\left(\overline{X^2}_M - \overline{X}_M^2\right). \quad (16)$$

As both, $\overline{X}$ and $\overline{X^2}$ can be updated on-the-fly using (15), also s^2 can be updated dynamically.

We conclude Algorithm 2 which is also proposed exemplarily in [12].

Algorithm 2 Monte Carlo Simulation with Dynamic Variance Estimate

Require: $p \in [0,1], \delta_{\text{abs}} > 0, M_0 \in \mathbb{N}$
  $\overline{X}_0 \leftarrow 0$
  $\overline{X^2}_0 \leftarrow 0$
  for $i \in \mathbb{N}$ **do**
    $X_i \leftarrow$ Simulation()
    $\overline{X}_i \leftarrow \frac{i-1}{i}\overline{X}_{i-1} + \frac{1}{i}X_i$
    $\overline{X^2}_i \leftarrow \frac{i-1}{i}\overline{X^2}_{i-1} + \frac{1}{i}X_i^2$
    if $i > M_0$ **then**
      $s_i^2 \leftarrow \frac{i}{i-1}\left(\overline{X^2}_i - \overline{X}_i^2\right)$
      if $f(\delta_{\text{abs}}, p, M, s_M^2) \geq 0$ **then**
        return $\overline{X}_i$
      end if
    end if
  end for

The initial guard $M_0 \geq 1$ prevents that the algorithm terminates prematurely. Quick analysis makes clear that the loop will run through three phases.

1. For $i < M_0$, the loop runs through a warm-up phase and is not allowed to stop. It prevents premature termination of the algorithm by a gross underestimation of the sample variance, caused by a few simulation results that lie very close together.
2. For $M_0 \leq i \ll M_{\text{stop}}(\sigma^2)$ the sample variance s_i has stopped fluctuating and represents a feasible approximation for σ^2. Therefore, $i \ll M_{\text{stop}}(s_i^2)$ holds as well.
3. For i close to $M_{\text{stop}}(\sigma^2)$ the sample variance approximates the real variance really well. Therefore, $M_{\text{stop}}(\sigma^2) \neq M_{\text{stop}}(s_i^2)$ only in very rare cases.

Consequently, the choice of M_0 is not too critical and its main purpose is preventing "accidents" like

$$X_1 \approx X_2 \Rightarrow \sigma^2 \gg s_2^2 \approx 0 \Rightarrow f(p, \delta_{\text{abs}}, 2, s_2^2) \geq 0 \Rightarrow M_{\text{stop}}(s_2) = 2. \quad (17)$$

Ross [12] proposes $M_0 \geq 100$.

Algorithm 2 is sometimes found in a slightly different version wherein the sample moments are not updated continuously, but in batches of predefined size (see [14]). What seems like an insignificant mathematical detail at first, is relevant for parallel computing.

2 Case Study

In order to test both algorithms and both stopping rules, we decided to use a simple, academic, yet for this purpose quite representative cellular automata (CA) model for simulation of a susceptible-infectious-recovered (SIR) epidemic. It brings the classic SIR differential equation model by Kermack and McKendrick [15] into a spatial context and can be thought of depicting a bacterial infection that spreads on a rectangular Petri-dish. It unites ideas of typical agent-based SIR models like [16] and models of bacteria growth on a homogeneous surface like [17]

- The cell space of the CA is chosen as a rectangular grid with $m \times n$ cells.
- The state mapping of the CA maps each cell onto any of three states: $0 \cong$ susceptible, $1 \cong$ infectious, $2 \cong$ recovered. For the sake of readability we say, a cell is in e.g. the infectious-state, if the state mapping currently maps it onto 1.
- The simulation starts with one randomly chosen cell in the infectious state, while all other $m \cdot n - 1$ cells are susceptible.
- The CA uses the Von Neumann neighbourhood, i.e. a cell's neighbourhood consists of the four cells directly above, below, left and right of it. The neighbourhood is restricted at the borders.
- The CA is updated synchronously using the following two stochastic rules:
  - A cell in the susceptible state becomes infected with probability α, if it has at least one infectious neighbour.
  - A cell in the infectious state recovers with probability β.
- The time-update terminates as soon as the number of infectious cells in the CA reaches zero. We regard the *maximum number of infectious cells* observed during the iteration as the outcome of the model.

Figure 1 shows snapshots of the CA at three different times during the execution. The infection spreads radially from the point of initial infection and leaves interesting patterns. Figure 2 shows the number of infected cells as a function of time for twenty different simulation runs. Easily seen, the outcome X of the simulation run is highly irregular and has an unusual distribution. In particular, about every twelfth simulation run the disease did not break out at all, leading to $X = 1$.

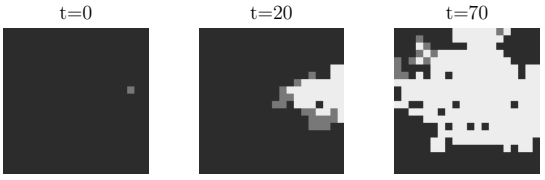


Figure 1: Three snapshots of the CA for different times during execution with $\alpha = \beta = 0.2$ and 20×20 cells. Infectious cells are marked red, susceptible blue and recovered ones yellow.

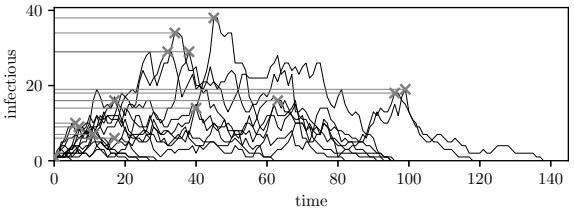


Figure 2: Results of twenty simulation runs with $\alpha = \beta = 0.2$ and 20×20 cells. The outcome value of every run, i.e. point with the highest value of infectious cells, is marked.

We chose this model for our test scenario as it is both simple and to some extent realistic. Hereby we do not mean, that the stated model is a realistic model for disease spread, but that the distribution of the model's outcome could be something we would also receive from a fully validated model. This feature distinguishes this study from tests found in standard literature about stopping of Monte Carlo experiments.

They use either classic, theoretical distributions (e.g. Exponential-, Pareto-, Uniform, Normal-inv. Gaussian-distribution are compared in [6]) or test the methods in the context of Monte Carlos integration (e.g. [8]).

3 Testing of Algorithms and Stopping Rules

3.1 Test Definition

As test scenario, $\alpha = \beta = 0.2$ as well as $N = M = 20$ were fixed. Both Monte Carlo algorithms and both stopping rules were applied in a Monte Carlos setting themselves with 10000 replications each.

In order to evaluate the failures of the algorithms, knowledge about the real mean value would be necessary. As the distribution of the simulation result cannot be determined analytically, we executed the simulation one million times and used the results of this procedure for bootstrapping. As this ridiculously large number is definitely large enough to approximate the distribution of X accurately, we replace the original simulation by drawing from a sampled list. This procedure has two key advantages:

- In contrast to the actual simulation, we know the moments of the distribution. They are precisely the empiric mean and variance of the huge data sample.
- Randomly drawing from the list executes much faster than evaluating the simulation. As we need to run each Monte Carlo test 10000 times, this is an essential feature.

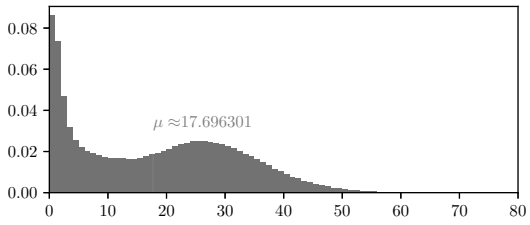


Figure 3: Histogram of the maximum number of infectious cells of one million simulation runs. The mean value is marked red.

The resulting distribution is seen in Figure 3. It has

$$\mu = \overline{X}_{10^6} = 17.696301,$$
$$\sigma^2 = s^2_{10^6} = 186.49197791739908. \quad (18)$$

To use Algorithm 1 and evaluate the stopping index M_{stop} in advance it is necessary to calculate an upper bound of the variance. As described this is only possible, if lower and upper bounds for the random number are available. Clearly, 1 poses for a lower bound of X as the maximum number of infectious cells cannot be smaller than its initial value. The total number of cells $m \times n = 400$ is a very pessimistic upper bound for X. Yet, in advance, it might be the only one available. We get

$$s_c^2 = \frac{(400-1)^2}{4} = 39800.25. \quad (19)$$

3.2 Test Results

Table 1 shows the results of tests with different values for p and δ_{abs}. The third column shows the sample mean of the stopping index $\overline{M_{\text{stop}}}$ and its sample standard deviation gained from 10000 reruns of the Monte Carlo algorithm with different random number settings. Note, that M_{stop} is deterministic in Algorithm 1, while it varies in Algorithm 2 depending on the value of the sample variance. The fourth column shows, how many of the 10000 algorithm executions failed to satisfy $\left|\overline{X}_{M_{\text{stop}}} - \mu\right| \leq \delta_{\text{abs}}$. As the confidence interval is supposed to make sure that the probability for such a failure is smaller than $1-p$, we define that the algorithm/stopping-rule failed as a whole, if the number of failures exceeded $(1-p)\cot 10000$.

The results of the case study are fascinating and allow highly interesting conclusions. First of all, the huge difference between the forecasted values for M_{stop} is remarkable. We find that the value for M_{stop} predicted by Algorithm 1 with the Chebyshev stopping rule is more than 1000 times higher than the average value gained from Algorithm 2 with the Gauss rule. In general, stopping indices predicted by Algorithm 1 are more than 200 times larger than those of Algorithm 2 which is, of course, a consequence of the crude variance bound being more than 200 times larger than the real variance.

In terms of comparing the two stopping rules we find that the stopping indices of the Chebyshev stopping rule are 5 to 8 times higher than the ones predicted by the Gauss stopping rule. This results from the general weakness of the Chebyshev inequality.

testcase	algorithm	stopping rule	$\overline{M_{\text{stop}}}$ ($\pm$ std)	failures (allowed)	failed?
	Algorithm 1	Chebyshev	199002	0 (500)	No
$p = 0.95$	Algorithm 1	Gauss	38223	0 (500)	No
$\delta_{\text{abs}} = 2$	Algorithm 2	Chebyshev	931.67($\pm$31.56)	1 (500)	No
	Algorithm 2	Gauss	178.49($\pm$14.11)	548 (500)	Yes
	Algorithm 1	Chebyshev	398003	0(250)	No
$p = 0.975$	Algorithm 1	Gauss	49988	0(250)	No
$\delta_{\text{abs}} = 2$	Algorithm 2	Chebyshev	1864.37($\pm$45.57)	0(250)	No
	Algorithm 2	Gauss	233.72($\pm$16.18)	273(250)	Yes
	Algorithm 1	Chebyshev	796005	0 (500)	No
$p = 0.95$	Algorithm 1	Gauss	152892	0 (500)	No
$\delta_{\text{abs}} = 1$	Algorithm 2	Chebyshev	3729.72($\pm$64.13)	0 (500)	No
	Algorithm 2	Gauss	715.76($\pm$28.12)	510 (500)	Yes

Table 1: Test results of 10000 Monte Carlo simulations stopped with the specified algorithm and stopping rule.

Finally, Algorithm 2 applied with the Gauss stopping rule failed in all three cases. Although they almost matched with the predicted ones, the resulting failure counts were slightly too high in all three test cases indicating that $P(|\overline{X}_{M_{\text{stop}}} - \mu| \leq \delta_{\text{abs}})$ is actually smaller than p. Therefore, the confidence level of the interval is not as high as required.

In summary, having a really non-asymptotic stopping algorithm for a Monte Carlo simulation comes at a price which is by far too high. On the one hand, performing hundreds of thousands of simulation runs is simply not necessary and too costly. Hence, the proposed dynamic algorithm with the sample variance is more practicable. On the other hand, asymptotic stopping rules may lead to an unreliable confidence interval.

4 Conclusion

In this work we presented and performed basic tests for the two most popular stopping rules for Monte Carlo simulation and investigated them in the context of two stopping algorithms. Considering the results presented in the last section, all four investigated methods essentially failed either by requiring too many or too little simulation runs. While the Gaussian approach needs to be corrected to become non-asymptotic, the Chebyshev concept needs to be made sharper. We found several attempts for both ideas in literature [9, 4, 6], yet the evaluation of the corresponding stopping strategies is work-in-progress.

We demonstrated that the seemingly simple problem of stopping Monte Carlo simulation at the right time is still not solved satisfactorily. Although the methods presented in this work are surely not optimal, we would yet still recommend using one of them in favour of just "guessing" a feasible iteration count.

References

[1] Duguay C, Chetouane F. Modeling and improving emergency department systems using discrete event simulation. *Simulation*. 2007;83(4):311–320.

[2] Purtov AN, Mamonova MV, Prudnikov VV, Prudnikov PV. Monte Carlo simulation of aging phenomena in multilayer magnetic structures. In: *EPJ Web of Conferences*, vol. 185. EDP Sciences. 2018; p. 3008.

[3] Saroj S, Rajput SJ. Composite smart mesoporous silica nanoparticles as promising therapeutic and diagnostic candidates: recent trends and applications. *Journal of Drug Delivery Science and Technology*. 2018; 44:349–365.

[4] Audibert JY, Munos R, Szepesvári C. Tuning bandit algorithms in stochastic environments. In: *International conference on algorithmic learning theory*. Springer. 2007; pp. 150–165.

[5] Maurer A, Pontil M. Empirical Bernstein bounds and sample variance penalization. *arXiv preprint arXiv:09073740*. 2009;.

[6] Bayer C, Hoel H, Von Schwerin E, Tempone R. On Nonasymptotic Optimal Stopping Criteria In Monte Carlo Simulations. *Siam Journal On Scientific Computing*. 2014;36(2):17. A869–A885.

[7] Gilman MJ. A Brief Survey of Stopping Rules in Monte Carlo Simulations. In: *Proceedings of the Second Conference on Applications of Simulations*. 1968; pp. 16–20.

[8] Hickernell F, Choi SC, Jiang L, Jiménez Rugama L. Monte Carlo Simulation, Automatic Stopping Criteria For. In: *Wiley StatsRef: Statistics Reference Online*. John Wiley & Sons. 2018;.

[9] Hickernell FJ, Jiang L, Liu Y, Owen AB. Guaranteed Conservative Fixed Width Confidence Intervals via Monte Carlo Sampling. In: *Monte Carlo and Quasi-Monte Carlo Methods 2012*, edited by Dick J, Kuo FY, Peters GW, Sloan IH, pp. 105–128. Berlin, Heidelberg: Springer Berlin Heidelberg. 2013;.

[10] Bienaymé IJ. *Considérations à l'appui de la découverte de Laplace sur la loi de probabilité dans la méthode des moindres carrés*. Imprimerie de Mallet-Bachelier. 1853.

[11] Ballio F, Guadagnini A. Convergence assessment of numerical Monte Carlo simulations in groundwater hydrology. *Water resources research*. 2004;40(4).

[12] Ross S. *Simulation*. Knovel Library. Elsevier Science. 2012.

[13] Feller W. *An introduction to probability theory and its applications*, vol. 2. John Wiley & Sons. 1957.

[14] Xie W, Nelson BL, Barton RR. Statistical uncertainty analysis for stochastic simulation. In: *Departmant of Industrial Engineering and Management Sciences, Northwestern University Working paper*. IEEE. 2014;.

[15] Kermack WO, McKendrick AG. A Contribution to the Mathematical Theory of Epidemics. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*. 1927;115(772):700–721.

[16] Miksch F, Haim C, Schneckenreither G, Breitenecker F. Comparison of Differential Equations and Cellular Automata for Epidemic Simulation. In: *ERK 2013 Proceedings.*, 22. Portoroz, Slovenia. 2013; pp. 137–140.

[17] Krawczyk K, Dzwinel W, Yuen DA. Nonlinear development of bacterial colony modeled with cellular automata and agent objects. *International Journal of Modern Physics C*. 2003;14(10):1385–1404.

Simulation Processes for Onboard State Estimation in a Small UAV Environment

Marsel Omeri*, Marko Radanovic, Ernesto Emmanuel Santana Cruz, Romualdo Moreno Ortiz

Department of Telecommunications and Systems Engineering, Autonomous University of Barcelona
Carrer dels Emprius, 2, 08202 Sabadell, Spain; *marsel.omeri@uab.cat

SNE 32(1), 2022, 9-14, DOI: 10.11128/sne.32.tn.10592
Received: 2020-11-10 (Selected EUROSIM 2019 Postconference Publication); Revised: 2021-10-18; Accepted: 2021-12-15
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. This paper describes the processes for estimating and analyzing the position states onboard small unmanned aerial vehicles in the low-altitude simulation environment. Those processes are explained using two simulation platforms, Robot Operating System and Gazebo. They comprise different system functionalities from trajectory generation, linear-kinematic trajectory conversion, path controllers to the modeling and configuration of various aerial vehicles, onboard positioning and navigation sensors, and creation and visualization of the simulation environment. We model and simulate the vehicle trajectories to determine 3-D positions from the GPS sensor, along with estimated positions from the fusion of GPS/IMU and Altimeter. The simulated results have provided a dataset on lateral and vertical trajectory profile guidance and prediction in the low altitude airspaces for follow-up research on the common reference altitude determination, as well as the definition of the well clear and collision states for the detect and avoid functions of the small unmanned aerial vehicles.

Introduction

To accommodate the future demand for low-altitude small unmanned aerial vehicles (sUAVs) [1], the previous air traffic management experiences indicate that this demand must be organized to balance traffic efficiency and safety [2,3]. Additionally, sUAV operators demand their missions to operate beyond visual line of sight (BVLOS), requiring autonomous capabilities [4]. This paper presents a test study on real-time simulation processes for estimating and analyzing future sUAV position states.

This estimation is based on the vehicle's trajectory inputs from the modeled sensor systems [7]. We use two simulation frameworks: Robot Operating System (ROS) [8, 9] and Gazebo [10, 11], to generate trajectories and simple pairwise traffic scenarios, together with the implementation of navigation, guidance, and control modules. In particular, the study has applied the GPS/ IMU sensor system with configured tracking rates [12], compliant with the sUAV performance model. In this paper, we use different system functionalities in both frameworks, from the trajectory generation, controllers to the modeling and configuration of different sUAVs, onboard positioning and navigation, and visualization of the simulation environment. The generated sUAV trajectories are further processed for extraction and data analysis to determine the 3-D positions in discrete moments for the kinematic trajectory profile. The simulated results have provided a dataset on the trajectory guidance and prediction in the low altitude airspaces, valid for follow-up research on the common reference altitude determination, definition of the well clear and collision states for the detect and avoid sUAV functionalities. Figure 1 illustrates a process flow of the simulated sUAV trajectory data exchange.

The rest of the paper is organized as follows. Section 1 explains in detail the ROS-Gazebo simulation framework with the main functionalities and the integrated sUAV performance and onboard sensor models, while Section 2 elaborates the data extraction and analysis method developed using the MATLAB Robotics System Toolbox, inclusive of multi-sensor fusion for position estimates. Section 3 describes the simulation results obtained for a given scenario with two sUAV trajectories and the output data analysis as a potential for follow-up research. Finally, concluding remarks and future directions are provided in Section 4.

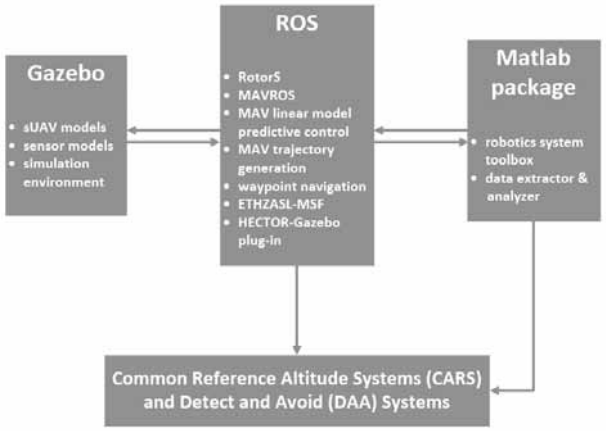


Figure 1: Process flow of the simulated sUAV trajectory data exchange.

1 Simulation Overview

Overall simulation is based on RotorS [14], a framework that enables simulation and tests different sUAVs, sensors, controllers, and state estimation algorithms. Also, it may be extended further to implement high-level operations like collision detection, avoidance, and so on. RotorS is developed based on ROS [15], a rapidly evolving middle-ware in robotics and automation, and Gazebo.

We extend RotorS with some other packages as mentioned above to design an encounter trajectory scenario. Each sUAV feeds odometry data from onboard sensors to the linear MPC and attitude PID controller. To generate and navigate the trajectory, we use the waypoint_navigation package [16], which also depends on the trajectory_generation_ros package. The former allows the user to input 3D ENU coordinates as trajectory input or GPS waypoints.

1.1 sUAV models

The Gazebo framework contains an extensive base of the sUAV performance types, from different MAV types, low-altitude-short-endurance (LASE), low-altitude-long-endurance (LALE), medium-altitude-long-endurance (MALE), to high-altitude-long-endurance (HALE) types. In this study, the simulation and testing are done on the LALE sUAV types. That sUAV is commonly referred to as a drone and is characterized by the local mission functions covering the flight ranges up to 10 km. The following three sUAV models are selected for the simulated trajectory guidance and testing scenarios: Firefly, Pelican, and Hummingbird (Figure 2). Table 1 lists their performance characteristics (AGL – Above Ground Level).

sUAV models from *the AscTec family*	Fire-fly	Peli-can	Hum-mingbird
maximum payload weight [kg]	0.6	0.65	0.2
maximum take-off weight [kg]	1.6	1.65	0.71
maximum flight time/endurance [min]	14	16	20
maximum airspeed [m/s]	15	16	15
maximum climb rate [m/s]	8	8	5
maximum range [m]	4500	4500	4500
maximum altitude AGL [m]	1000	1000	1000

Table 1: Performance characteristics of the selected sUAV models [17].

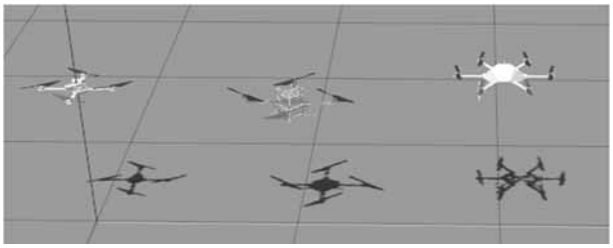


Figure 2: Three selected AscTec-family sUAV models: Firefly (on the left), Pelican (in the middle), and Hummingbird (on the right).

1.2 Sensor models

In this paper, the sUAV is equipped with the following onboard sensors:

- IMU-based on hector gazebo plugins
- GPS – based on hector gazebo plugins
- Altimeter- based on hector gazebo plugins

The simulation environment is designed as such; the GPS reference is the same as the launching point of sUAV. We implement a sensor fusion module for position estimation based on a ROS multi-sensor fusion (MSF) package [18]. This package uses EKF, where IMU is feeding data for the Prediction State, and GPS/Altimeter are the Update State sensors. Typically, GPS/IMU produces good results for outdoor environments, and it is the most common onboard sensors combination when it comes to commercial sUAVs. In addition, we include a Barometer to see how it affects altitude estimation. Table 2 illustrates the parameters for each sensor.

Sensor	Update Rate [Hz]	Simulated Noise (Gaussian) [m]
GPS	10	0.01, 0.01, 0.01
IMU	100	0.35, 0.35, 0.30
Altimeter	20	0.1

Table 2: Sensor parameters.

1.3 Simulation Environment

Gazebo allows a visual, 3-D simulation of a scenario consisting of cyber-physical systems, e.g., ground-rovers, UAVs, and other objects like simple obstacles, or surrounding environmental elements, together composing what it is called a Gazebo world. Thus, in Gazebo, realistic scenarios for cyber-physical systems, including the surrounding environment, can be created.

A sUAV model with its properties presents the main object in Gazebo. From the physical point of view, a single sUAV as point mass is linked to a reference coordinate system that can be set in the GPS frame or as the positive Euclidean space (i.e., ENU – East, Nord – Up). Figure 3 illustrates the launching phase of a single sUAV in the ENU coordinate system. The blue line presents the Up direction, i.e., the positive z-axis, the green one the Nord direction (i.e., the positive y-axis). In contrast, the red one denotes the East direction (i.e., the positive x-axis). Equivalently, in the GPS frame, the East and Nord directions are replaced by the geographic coordinates: longitude and latitude.

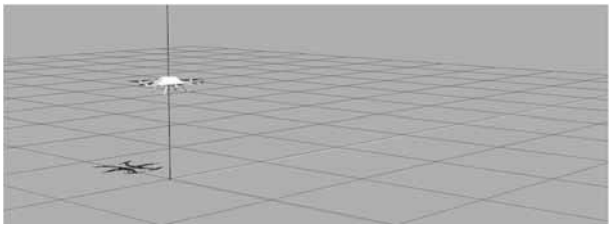


Figure 3: Single sUAV operate in ENU coordinate system.

By default, Gazebo allows a visual 3-D simulation of a single UAV trajectory only, focusing on the guidance and control aspects of trajectory profile configured by the inputs previously explained. Since the Gazebo framework is developed mostly for the sUAV types that operate over the local, short-range mission profiles (Figure 4), the environment is customized as per this type of trajectory. Instantaneous and significant heading changes characterize the local, short-range profiles.

The most crucial aspect of visualization is altitude control. Such controller onboard sUAV has one of the managing functions that replace the control of a remote human pilot on the ground. The trajectory profile, either created as the 4-D structure or 3-D structure with assigned airspeed, must be complied with the principle of creating:

1. starting/launching waypoint and following waypoint,
2. second last and last waypoint.

In both cases, the waypoints' geographic location (latitude and longitude) must be maintained constant while allowing an UAV to reach the required altitude AGL or descent to the ground, i.e., changing only the value of z-coordinate. When the altitude controller achieves the needed elevation, it activates to maintain the constant vertical profile or change it if required by the planned trajectory (Figure 4).

Gazebo framework allows the simulation of a single sUAV by default. However, it is possible to extend the simulation to two or more sUAV trajectories and create a potential traffic scenario. Regardless of the number of trajectories added in simulation (2, 3, etc.), each added sUAV is controlled and managed with respect to the reference sUAV, i.e., the one linked to the reference coordinate system.

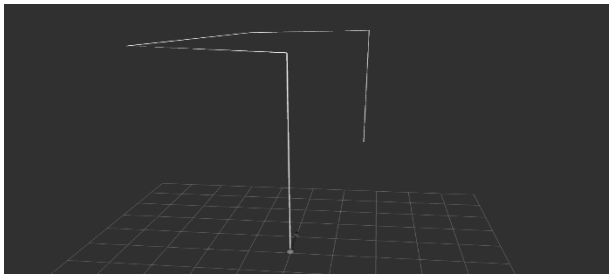


Figure 4: Local, short-range mission profile supported by the altitude controller onboard sUAV.

Nevertheless, the functional requirements of simulated sUAVs and the cooperative or non-cooperative task nature are out of this paper's scope. To conclude, in this study, only two sUAVs with their 3-D trajectory profiles have been further analyzed, creating a pairwise traffic scenario (Figure 5).

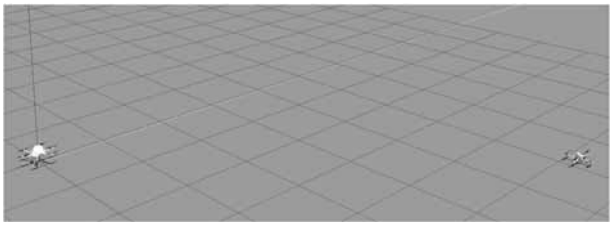


Figure 5: Generation of another sUAV in Gazebo with respect to the reference sUAV

2 Data Extraction and Analysis

This section explains in detail the data extraction procedure and analysis based on the GPS-IMU-Altimeter data capturing. It further elaborates on the multi-sensor fusion (MSF) 3D-position estimation method.

2.1 Data extraction and analysis

In this simulation study, we have recorded the published data from the sensors like GPS, IMU, Altimeter, and the data from the MSF method. These output data are saved as *rosbag* data files. For demonstration purposes, we saved the data from UAV1 and UAV2 in a file named *firefly_hummingbird_scenario1.bag.* To obtain the former, we have created a *ROS launch_file* containing the topics which we are interested in analyzing:

- firefly/fix and firefly/hummingbird – containing GPS position values in WGS84 coordinate frame,
- firefly/gps/points and hummingbird/gps/points – containing GPS position values in ENU coordinate frame,
- firefly/MSF/UpdatePose and hummingbird/MSF/UpdatePose - containing position values in ENU coordinate frame.

There have been used two different methods to extract the information from *firefly_scenario1.bag*:

- writing a *ros_python* node which converts bag file to a CSV file,
- using Robotics System Toolbox in Matlab.

By this procedure, we want to test the interoperability of ROS with other simulation frameworks, which we tend to use in our future research. Figure 6 illustrates the data extraction procedure. The functions have been generated based on the Robotics Toolbox to read messages from different ROS/Gazebo sensors. These messages are converted into Matlab files (.mat) for later processing. Also, we include a ROS *node* written in Python, which allows the output data to be saved in CSV format

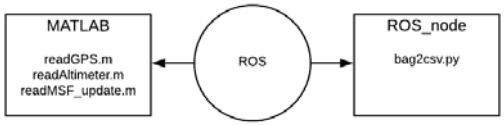


Figure 6: Data extraction procedure from ROS/Gazebo sensor.

2.2 Sensor fusion for 3-D position estimation

The main objective of this simulation work was to see how the altitude controller would preserve the altitude of the given trajectories. Therefore, we have also included onboard the sUAV an altimeter sensor to increase the accuracy of estimation. These data are quite sensitive for different applications related to inspection, coverage area, defining standard reference altitude for sUAVs, etc.

The estimation process has been fused a GPS/IMU sensors with an Altimeter using the MSF ROS package, illustrated in Figure 7. The MSF working procedure is explained in chapter 2.2, and, in this case, it gives 3D positions for both sUAVs.

Figure 7: Sensor fusion process.

3 Simulation Results

This section presents the data output as results obtained for the simulated trajectories of a pair of sUAVs. It analyses the sigma values, i.e., the lateral and vertical trajectory profile errors as differences between the planned and estimated 3-D positions, and discusses the future their potential use in the implementation in the operational domain. All simulations and data processing were run on a PC with Linux Ubuntu 18.04.2 (64bit) processor Intel(R) Core (TM) i7-8700 CUP @ 3.2GHz x 12, 16 GB RAM, and Nvidia Graphic Card Quadro P1000. The codes used for ROS/Gazebo were written in C++, Python, and XML, whereas MATLAB and Python functions were developed for data extraction.

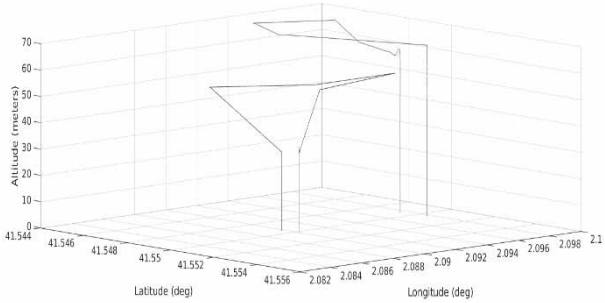


Figure 8: GPS 3-D tracking profile.

3.1 Simulated data output and estimation

For this simulation study, two LALE sUAV trajectories have been created in a 3-D configuration, governed by the initial airspeed value of 15 m/s and the initial acceleration of 3 m/s^2 (both for sUAV1 and sUAV2). The trajectory waypoints have been planned as per Table 3.

The simulation scenario output data are illustrated in Figure 8 and Figure 9. Input data files for two sUAVs are denoted as *drone1_scenario1.yaml* and *drone2_scenario1.yaml*, respectively.

sUAV_ID	Latitude [°]	Longitude [°]	Altitude AGL [m]
	41.549582	2.089708	0.0
	41.549582	2.089708	30.0
	41.553057	2.089743	60.0
sUAV1	41.553895	2.097228	70.0
	41.550498	2.097869	70.0
	41.550917	2.093766	60.0
	41.549370	2.090490	30.0
	41.549370	2.090490	0.0
	41.550493	2.097766	0.0
	41.550493	2.097766	65.0
	41.553465	2.092379	68.0
sUAV2	41.552516	2.089675	68.0
	41.550276	2.091821	68.0
	41.551607	2.094963	65.0
	41.550748	2.096422	60.0
	41.550748	2.096422	0.0

Table 3: Trajectory waypoints for sUAV1 and sUAV2.

After the simulation, the data were extracted from GPS/IMU and then plotted their trajectory profiles. It is shown that the scenario was designed to have changes in both the altitudes and heading directions in the case of both sUAVs. A subplot of the data extracted from the altimeter has been included to emphasize the fluctuations in altitude. Figure 10 reports the estimated 3-D positions of trajectories based on GPS/IMU and altimeter sensor fusion. These results are further analyzed to identify the deviations on the trajectory profiles.

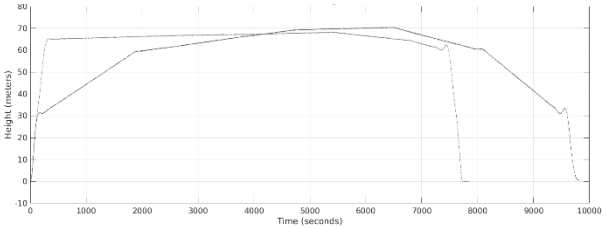


Figure 9: Altimeter tracking profile.

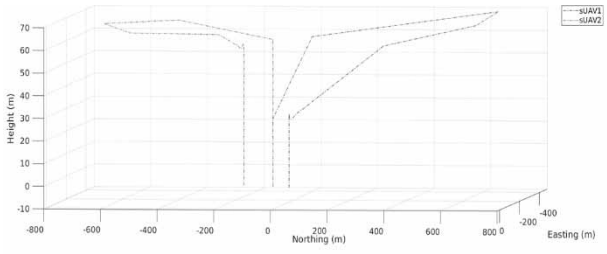


Figure 10: Estimated 3-D positions tracking profile.

3.2 Estimation analysis

This section describes and demonstrates the position estimation errors for vertical and horizontal profiles. The errors are defined as the difference between GPS_output and MSF_output data, where MSF is the module fusing GPS/IMU and Altimeter sensors.

We calculate and plot the lateral and 3-D errors by simply calculating deviations on trajectories. Moreover, a plot of error in altitude (*z*-axis) is provided to give a complete picture of the estimation (Figure 11, Figure 12, and Figure 13). Finally, the calculation is given as follows:

$$\sigma_{xy} = \sqrt{\sigma_x^2 + \sigma_y^2} \tag{1}$$

$$\sigma = \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2} \tag{2}$$

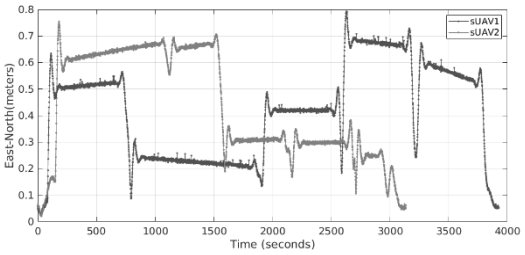


Figure 11: Horizontal profile errors over time for sUAV1 (blue) and sUAV2 (red).

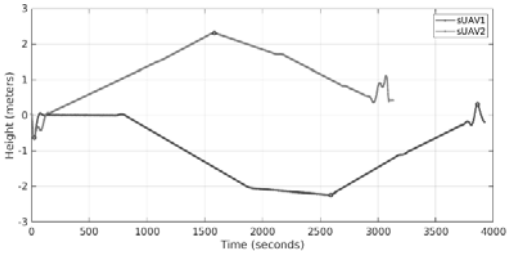


Figure 12: Vertical profile errors over time for sUAV1 (blue) and sUAV2 (red).

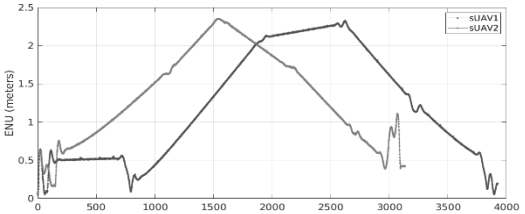


Figure 13: 3-D profile errors over time for sUAV1 (blue) and sUAV2 (red).

4 Conclusion and Future Work

This paper focuses on the test study on real-time estimation of the relative, pairwise sUAV 3-D trajectory positions in the extended ROS/Gazebo simulation framework.

The assessment is performed based on the planned trajectory inputs, using the modelled GPS-IMU-Altimeter sensor fusion to identify the lateral and vertical profile errors and resulting 3-D profile errors. Results indicate that the combined sensor fusion for the onboard tracking and guidance functions provides meaningful insight on the future investigation of the standard altitude reference for a multi-sUAV urban environment, along with a possible testing standard for the definition of the Well Clear and collision detection and avoidance thresholds in different scenario types, such as encounter, intersection or overtaking.

Acknowledgments

This work was supported by the Ministry of Economy, Industry, and Competitivity through the project "GESTION AUTOMATIZADA DE TRAFICO AEREO PARA RPAS" (TRA2017-88724-R).

References

[1] Ren L, Castillo-Effen M, Yu H, Johnson E, Yoon Y, Takuma N, Ippolito CA. Small unmanned aircraft system (sUAS) categorization framework for low altitude traffic services. In: Proceedngs of the 36th IEEE/AIAA Digital Avionics Systems Conference (DASC) 2017, pp. 1-10. St. Petersburg, USA, September 2017. doi: 10.1109/DASC.2017.8101996.

[2] Prevot T, Rios J, Kopardekar P, Robinson JE, Johnson M, Jung J. UAS Traffic Management (UTM) Concept of Operations to Safely Enable Low Altitude Flight Operations. In: 16th AIAA Aviation Technology Integration and Operation Conference, AIAA Aviation 2016. Washington, USA, June 2016. doi: 10.2514/6.2016-3292.

[3] Jiang T, Geller J, Ni D, Collura J. Unmanned Aircraft System traffic management: Concept of operation and system architecture. International Journal of Transportation Science and Technology 5(3), 123–135 (2016).

[4] Brooker P. Introducing Unmanned Aircraft Systems into a High Reliability ATC System. Journal of Navigation 66(5), 719-735 (2013).

[5] Zhan W, Wang W, Chen N, Wang C. Efficient UAV Path Planning with Multiconstraints in a 3D Large Battlefield Environment. Mathematical Problems in Engineering 2014, 1-12 (2014). doi: 10.1155/2014/597092.

[6] Weiss S, Achtelik MA, Lynen S, Achtelik MC, Kneip L, Chli M, Siegwart R. Monocular vision for long-term micro aerial vehicle state estimation: A compendium. Journal Field Roboics 30(5), 803-831 (2013). doi: 10.1002/rob.21466.

[7] Giovanneschi F, et al. An adaptive sensing approach for the detection of small UAV: first investigation of static sensor network and moving sensor platform. In: Proceedings of the Signal Processing, Sensor/Information Fusion and Target Recognition XXVII, 106460S. Orlando, USA, April 2018. doi: 10.1117/12.2304758.

[8] Badger J, Gooding D, Ensley K, Hambuchen K, Thackston A. ROS in space: A case study on robonaut 2. Robot Operating System (ROS), 343-373 (2016). Springer, Cham.

[9] Chen H, Kakiuchi Y, Saito M, Okada K, Inaba M. View-based multi-touch gesture interface for furniture manipulation robots. Advanced Robotics and its Social Impacts, IEEE, 39-42 (2011).

[10] Koenig N, Howard A. Design and use paradigms for Gazebo, an open-source multi-robot simulator. In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 2004, pp. 2149-2154. Sendai, Japan, September 2004. doi: 10.1109/IROS.2004.1389727

[11] Nagaty A, Saeedi S, Thibault C, Seto M, Li H. Control and navigation framework for quadrotor helicopters. Journal of intelligent and robotic systems 70(1-4), 1-12 (2013).

[12] Desa Hazry, Mohd Sofian, A. Zul Azfar. Study of Inertial Measurement Unit Sensor. In Proceedings of the International Conference on Man-Machine Systems (ICoMMS), Batu Ferringhi, Penang, Malaysia, October 2009.

[13] Caron F, Duflos E, Pomorski D, Vanheeghe P. GPS/IMU data fusion using multi-sensor Kalman filtering: introduction of contextual aspects. Information Fusion 7 (2), 221-230 (2006).

[14] Furrer F, Burri M, Achtelik M, Siegwart R. RotorS - A modular gazebo MAV simulator framework. Studies in Computational Intelligence 2016, 595-625 (2016).

[15] Quigley M, Conley K, Gerkey B, Faust J, Foote T, Leibs J, Wheeler R, Ng AY. ROS: an open-source Robot Operating System. In ICRA workshop on open source software, 3(3.2), p. 5 (2009).

[16] GitHub Website, 3. https://github.com/ehz-asl/waypoint_navigator.

[17] ASCENDING TECHNOLOGIES Website, http://www.asctec.de.

[18] http://wiki.ros.org/ethzasl_sensor_fusion/Tutorials/Introductory%20Tutorial%20for%20Multi-Sensor%20Fusion%20Framework

Guarantying Consistency of Spatio-temporal Regions that Solve Air Traffic Conflicts

Thimjo Koça*, Miquel Angel Piera

Logistic and Aeronautics Group, Department of Telecommunications and System Engineering, Autonomous University of Barcelona, Sabadell, Spain; *thimjo.koca@uab.cat

SNE 32(1), 2022, 15-21, DOI: 10.11128/sne.32.tn.10593
Received: 2020-11-10 (Selected EUROSIM 2019 Postconference Publication); Revised: 2021-09-21; Accepted: 2021-11-01
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna, ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. Separation management together with more efficient conflict detection & resolution are two of the main challenges that Air Traffic Management faces in its quest to modernize itself. This quest for modernization comes as a result of the necessity to adapt to the increment in demand and complexity of the projected future air traffic. Several approaches are proposed to the problem and several sets of properties that they should satisfy. We identify among them robustness, the ability to provide realistic solutions, and consideration of uncertainties the most critical ones. These properties should of course come at a reasonable computational cost. Among the various approaches towards the problem, we believe the ones that try to solve conflicts using spatio-temporal regions are the most adequate base for such systems, because of their unique ability to consider post-decitional uncertainties. In one of the two such methodologies, the construction of such regions, can produce several inconsistencies. We present in this work a methodology by which such inconsistencies can be taken care of.

Introduction

Air traffic management's (ATM) mission is to make air transportation possible. This is attained by the means of efficient, environmentally friendly and socially valuable systems, which have safety as their principal goal [1, 2]. On en-route traffic, safety is quantified through a minimum horizontal separation distance and a minimum vertical separation distance, that need to be maintained between aircraft. Current ATM provides minimum pairwise separation through a system with human air traffic controllers (ATC) at the core of its decision making.

In the quest to modernization of the airspace system to reduce congestion and delays and handle denser traffic flows, it is essential to develop, deploy, and maintain new decision support systems (DST) automation [3]. The DST fundamental function is a conflict resolution which is to provide aid in the process of resolving intruder's intent.

Such a DST, often called Conflict Detector & Resolver (CD&R), should demonstrate some properties that relate to the ATM's goals. Several properties have been proposed in literature [4]. Among them, the most basic ones are being robust (i.e. being able to always provide solutions), providing realistic solutions, and doing so in a computationally tractable, and resilient manner. Evidently solutions should be realistic, otherwise it will not be possible to fly them. Being computed in tractable manner is essential given the time criticality of the system. Being resilient "forces" the CD&R to consider uncertainties. Such uncertainties can be present before the time that a solution maneuver should be executed, or after that. Uncertainties present before the execution of a solution maneuver are mainly due to measurements errors, or wind uncertainties [5]. Some works count these kind of uncertainties [6, 7, 5]. Some others don't [8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20] with the explicitly, or implicitly expressed claim that such uncertainties in a short tactical time horizon, under the 4D trajectory concept [21], are insignificant, if not completely absent.

Uncertainties present after the time of the execution of a solution maneuver, which can also be human-caused (e.g. a pilot not executing the maneuver at the given fixed time, or with not the exact turning rate, etc.) however are of a higher criticality. The main

reason for this is the shorter given time to deal with them. Two works provide solution that consider post-decisional uncertainities, specifically [22, 23].

In [23], the use of spatio-temporal regions throughout the execution of a flight to count for uncertainties is proposed. In cases where the assigned regions could not provide solutions to predicted conflicts, one, or more trajectories were deviated and new regions were built around them.

Alternatively, in AGENT project [22], a methodology is proposed such that spatio-temporal regions are constructed only in situations where a loss of separation is predicted and a conflict resolution process is initialized. In this work we present a methodology to make sure that the regions proposed by the resolution process proposed in AGENT are consistent and safe trajectories can actually be constructed within them.

The rest of this work is organized as follows. In section I the key idea of spatio-temporal regions and the how the consistency problem arises are explained. Section II contains the proposed algorithms. A concrete study case is provided in section III and the concluded remarks are given in section IV.

1 Assigning Continuous Space-Time Regions

1.1 Trajectory dynamics model

We employ a widely used manner to model the aircraft dynamics [10, 24, 22, 25, 15, 26]. The trajectory of the flight is modeled as a series of 4D (space-time) waypoints. The aircraft is treated as a point mass in a 3D Euclidean space, evolving over time. We obtain its x and y coordinates by applying the stereographic projection [27] on the its latitude and longitude. The z coordinate represents the aircraft's altitude. During the flight, the involved aircraft are assumed to have piece-wise constant velocity between two consecutive waypoints. Moreover, planar maneuverability constraints are modeled by the impose of a maximum angle by which an aircraft can deviate.

Given the above, the flight state variables of the aircraft is specified as (x, y, z, v_x, v_y, v_z), where (x, y, z) are its coordinates and (v_x, v_y, v_z) its velocity components.

1.2 Continuous space-time regions

The core idea of continuous space-time regions lies in the observation that instead of trying to assign a single trajectory to each aircraft that must maneuver to solve a detected conflict, a space-time region can be given to each one of them.

Mathematically, classical approaches assign to each aircraft a function describing their motion:

$$\begin{cases} x = x(t) \\ y = y(t) \\ z = z(t) \end{cases} \tag{1}$$

Assigning a region instead, as suggested in [23, 22] could be expressed as:

$$[x(t), y(t), z(t)] \in V(t) \tag{2}$$

where $V(t)$ is a dynamic volume, evolving over time.

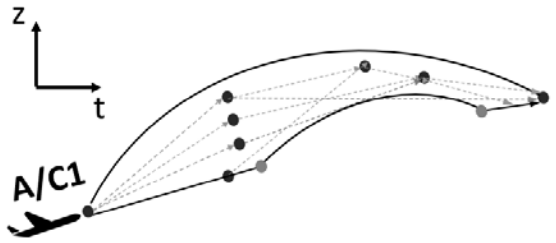


Figure 1: Assigned safe region for AC_1 and examples of various legs it can construct (green segments), or not (red segments).

Figure 1 illustrates a safe space-time region assigned to an aircraft in a world with a single spatial dimension (z coordinate) and time. The black continuous curves represent the border of AC_1 safe region (i.e. a guaranteed conflict-free area), the green dashed lines represent feasible legs that AC_1 can fly, the red dashed lines represent legs which might cause a loss of separation, i.e. a conflict and the black dots are feasible, conflict-free waypoints for AC_1.

1.3 Representation through moving polygons

Getting in to more details, we assume that we are in a well-structured traffic [21], and not under free-flight conditions [28], and also that the original trajectory is the optimal one, actual regions are constructed around the original trajectories of the aircraft.

The possible maneuvers that can be issued to resolve a conflict can be classified in two big categories, simple maneuvers and compound maneuvers. Simple maneuvers come in three flavors, alteration of the horizontal velocity component without a change of its module, alteration of the flight level, or alteration of the module of the horizontal component of the velocity. Compound maneuvers are maneuvers that are made of several simple maneuvers. In our implementation, each region is constructed based on simple maneuvers only, i.e. if we implement a region based on alteration of horizontal velocity direction, within it we can construct only trajectories that are based on horizontal deviations from the original trajectory.

Since we operate in a well-structured traffic, our goal is that after an aircraft deviates to avoid a loss of separation, it should return to its original trajectory. Concentrating in the case of the horizontal deviation this means that we will have at least two changes in the velocity direction, one to go out of the original trajectory and another to go back to it, as illustrated in Figure 2. Because of this last fact, it is convenient to represent a region as a series of moving polygons.

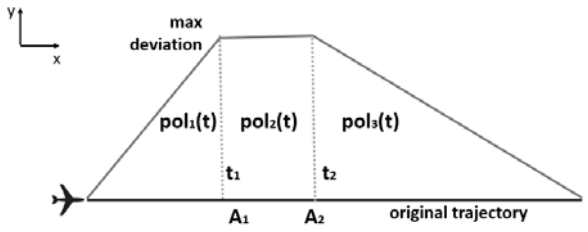


Figure 2: A spatio-temporal region for AC_1 is made up of at least three sub-regions, represented by moving polygons.

By moving polygon here we mean a convex polygon, made up of several spatial points, each one of them traveling in time through a constant (in direction and module) velocity. Note that each point can travel by a different velocity, as long as the convexity of the polygon is maintained throughout its movement.

Further on, since such regions need to be free of conflicts, some cuts might be performed on them. Because different parts of the region, i.e. different polygons have, in general, different conflicts, different cuts will be performed in them, as illustrated in Figure 3. Black lines represent the trajectories of each aircraft. In this scenario, AC_1 is asked to construct its spatio-temporal region, border by its original trajectory, the black segment and the other region limits, represented by the blue lines, to seek for a safe solution to the problematic situation. AC_1's region is made up of three moving polygons, pol_1, pol_2, and pol_3, separated by each other by the green dotted segments. As illustrated by the red dashed segments in the figure, pol_2 is in conflict with AC_2 and pol_3 is in conflict with AC_3. To make therefore this region safe some cuts will be performed resulting in the situation illustrated by Figure 4.

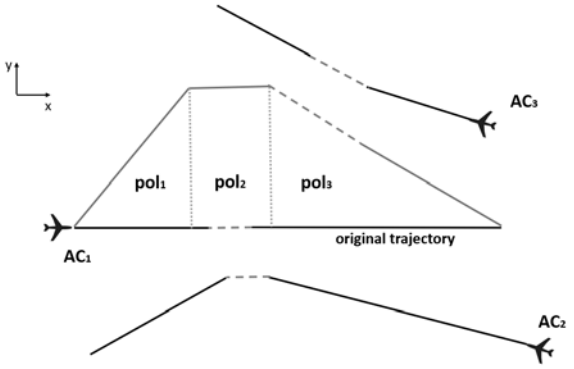


Figure 3: Situation illustrating case when different polygons have different conflicts.

In this figure we can see that AC_1 can travel within pol_2 and reach point A which is considered safe. As soon as it leaves this point however, it is outside the safe region and with no guarantee to be free of conflicts.

The algorithms presented in the following section of this work take these safe, but maybe inconsistent regions and transform them into safe and consistent ones.

2 Region Consistency Algorithm

Algorithm 1 is the main algorithm of this work and illustrated the general procedure that is being followed. Essentially what it expressed is that the intersection of all moving polygons should be calculated and used further to construct the modified, safe, consistent region.

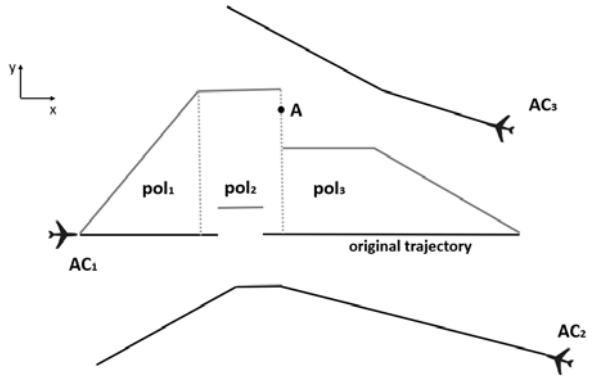


Figure 4: Situation illustrating case when different polygons get cut differently and are therefore inconsistent.

Algorithm 1 Region Consistency Algorithm

let pol_1 be the latest modified moving polygon
for all let pol_2 be a polygon in the range between pol_1 ancestor and the very first polygon of the region **do**
  pol = translate(pol_1, pol_2, backwards)
  $pol_1 := pol$
end for
initialize the array of moving polygons, *arrayPol*
Add pol_1 in *arrayPol*
for all pol_2 between pol_1 successor and the very last polygon of the region **do**
  let t_2 be the time interval during which pol_2 exists
  pol_1 = translate(pol_1, pol_2, forward)
  add pol_1 in *arrayPol*
end for
initialize new region using *arrayPol*

The main step in Algorithm 1 is the translate step, based on Algorithm 2. If the polygons would have been static, standard computational geometry clipping algorithms [29, 30, 31] could have been used to calculate their intersection. Instead the translation, for each pair of consecutive moving polygons, their state at the common time instance is calculated and then the intersection between these static polygons is performed using [29] and used further. More specifically, let pol_n be the moving polygon we are considering and t_n the time interval during which it exists. The static polygon s_n is calulcuated as the state of pol_n ant the beginning of t_n. At the same time, the static polygon s_{n-1} is calculated as the state of pol_{n-1}, i.e. ancestor of pol_n at the end of its time interval t_{n-1}. Note that that the end of t_{n-1} is equal to the beginning of t_n since the two polygons are consecutive. The intersection state s_i between s_{n-1} and s_n is calculated. As a next step the velocities, v_i, corresponding to each of the vertices of s_i are calculated and the new moving polygon pol_i is formed using s_i as its end state, v_i as its set of its velocities and t_{n-1} as its moving time interval.

Algorithm 2 Polygon Translation Algorithm

INPUT: pol_1 to be translated, pol_2 to constrain the translation
if forward translation **then**
  calculate s_1, the end state of pol_1
  calculate s_2, the starting state of pol_2
else
  calculate s_1, the starting state of pol_1
  calculate s_2, the end state of pol_2
end if
find their intersection, i_s
let t_2 be the time interval during which pol_2 exists
v = velocities(i_s, pol_2)
return polygon pol using i_s, v, and t_2

The last algorithm, Algorithm 3 shows how the velocities, v_i corresponding to the verties of s_i are calculated. In it, s_{n-1} is divided into triangles. For each vertex then of s_n the triangle within which it lies is identified. If we denote the vertex under consideration by $\vec{p} = (x, y)$ and the vertices of the triangle within which it lies by $\vec{p}_i = (x_i, y_i)$ for $i \in \{1, 2, 3\}$ then we have to solve the linear system:

$$\begin{cases} \sum_{i=1}^{3} \alpha_i \vec{p}_i = \vec{p} \\ \sum_{i=1}^{3} \alpha_i = 1 \end{cases}$$

Which is a linear system of three equations with three unknowns and a unique solution[1]. Then using the solution of this system, we can calculate the desired velocity $\vec{v}$ as follows:

$$\vec{v} = \sum_{i=1}^{3} \alpha_i \vec{v}_i$$

where $\vec{v}_i$ are the corresponding velocities for the triangles vertices $\vec{p}_i$.

[1] The guarantees for the uniqueness come from the fact that we are trying to express a point within a triangle as a convex combination of the triangle's vertices.

Algorithm 3 Velocity Initialization Algorithm

INPUT: $state_1$, pol_2
if forward translation **then**
  $state_2$ = state of pol_2 at its starting time
else
  $state_2$ = state of pol_2 at its end time
end if
divide $state_2$ into triangles
initialize *arrayV*
for all p, vertex of $state_1$ **do**
  let tr be the triangle of $state_2$ that contains p and p_i its vertices, $i \in 1,2,3$
  find the coefficient $\alpha_1, \alpha_2, \alpha_3$, s.t. $p = \sum_{i=1}^{3} \alpha_i p_i$
  $v := \sum_{i=1}^{3} \alpha_i v_i$, where v_i are the velocities corresponding to p_i
  add v to *arrayV*
end for
return *arrayV*

3 Study Case

In this section a real case is given. The studied region is a product of solving a conflict found in a traffic simulation based on historical flight data over Europe, taken from DDRII [2]. The predicted flying geometry before the conflict resolution is given in Figure 5. AC_1, with the black trajectory, will loose separation with AC_2, with the blue trajectory. The red segments denote the parts of the trajectories that will be in conflict. AC_3, with the green trajectory, is not in conflict with any other aircraft. However some possible solutions to the original conflict between AC_1 and AC_2 might cause a new conflict with AC_3.

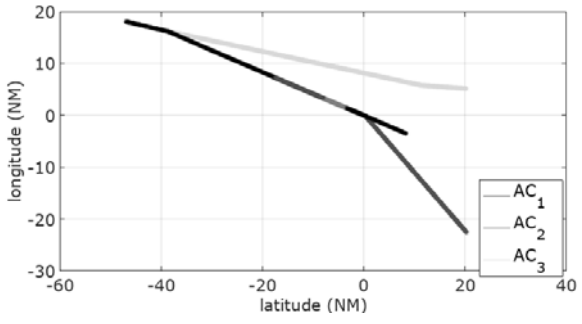


Figure 5: The predicted traffic geometry before the conflict resolution.

[2]DDRII is a data depository, provided by EUROCONTROL with extensive data regarding flights that pass over European sky

The resolution algorithm has chosen AC_2 as the aircraft that will need to maneuver to solve the conflict. Figure 6 contains the initial spatio-temporal region that AC_1 builds to seek a solution for the conflict. This region is not safe, having loss of separation with AC_2, as it contains the original trajectory of AC_1 as one of its borders, and also contains an induced conflict with AC_3. To avoid these conflicts the region needs to be cut, resulting in the shape depicted in Figure 7. There we can see that part of the expanding region leads to points that are not contained in the parallel region and therefore are not guaranteed to be safe. This is taken care by the methodology introduced in Section II and its results can be seen in Figure 8.

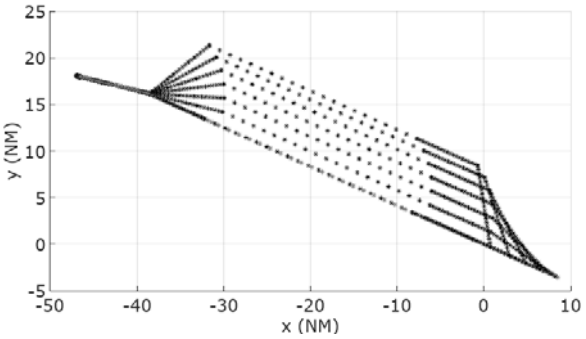


Figure 6: The region that AC_1 initializes to solve the conflict with AC_2.

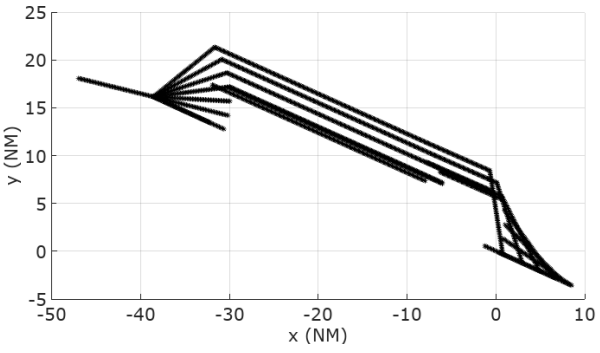


Figure 7: AC_1's cot region containing inconsistencies.

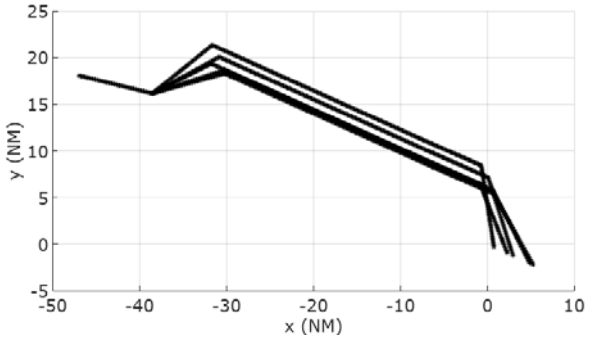


Figure 8: AC_1's consistent region.

4 Concluding Remarks

Separation management together with more efficient conflict detection & resolution are two challenges of importance that modern ATM faces in its attempt to adapt to the increment in demand and complexity. Among the various approaches towards the problem, we believe the ones that try to solve conflicts using spatio-temporal regions are an adequate base for such systems, because of their ability to consider post-decitional uncertainties. While producing such regions, according to the methodology proposed in [22], several inconsistencies can arise. We presented in this work a set of algorithms by which such inconsistencies can be eliminated.

Acknowledgement

This research is supported by the national Spanish project: "EU-TM" (ref. TRA2017-88724-R). Opinions expressed in this article reflect the authors' views only.

References

[1] Di Gravio G, Mancini M, Patriarca R, Costantino F. Overall safety performance of the air traffic management system: Indicators and analysis. *Journal of Air Transport Management*. 2015;.

[2] Cook A, Belkoura S, Zanin M. ATM performance measurement in Europe, the US and China. *Chinese Journal of Aeronautics*. 2017;.

[3] Tang J, Piera MA, Guasch T. Coloured Petri net-based traffic collision avoidance system encounter model for the analysis of potential induced collisions. *Transportation Research Part C: Emerging Technologies*. 2016;.

[4] Pritchett AR, Genton A. Negotiated Decentralized Aircraft Conflict Resolution. *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS*;1.

[5] Yang Y, Zhang J, Cai KQ, Prandini M. Multi-aircraft Conflict Detection and Resolution Based on Probabilistic Reach Sets. *IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY*. 2017;25(1).

[6] Visintini AL, Glover W, Lygeros J, Maciejowski J, Glover W, Maciejowski J. Monte Carlo Optimization for Conflict Resolution in Air Traffic Control. *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS*. 2006;7(4).

[7] Yang Y, Zhang J, Cai KQ, Prandini M. A stochastic reachability analysis approach to aircraft conflict detection and resolution. 2014;.

[8] Radanovic M, Eroles MAP. Spatially-Temporal Interdependencies for the Aerial Ecosystem Identification. In: *Procedia Computer Science*. 2016; .

[9] Piera M, Radanovic M, Leal X. Multi-Agent systems for air traffic conflicts resolution by using a causal analysis of spatio-Temporal interdependencies. In: *Simulation Series*. 2016; .

[10] Pallottino L, Feron EM, Bicchi A. Conflict Resolution Problems for Air Traffic Management Systems Solved With Mixed Integer Programming. *IEEE Transactions on Intelligent Transportation Systems*. 2002;.

[11] Christodoulou M, Costoulakis C. Nonlinear Mixed Integer Programming for Aircraft Collision Avoidance in Free Flight. *Tech. rep*. 2004.

[12] Vela AE, Solak S, Clarke JPB, Singhose WE, Barnes ER, Johnson EL. Near real-time fuel-optimal en route conflict resolution. *IEEE Transactions on Intelligent Transportation Systems*. 2010;.

[13] Cafieri S, Durand N. Aircraft deconfliction with speed regulation: New models from mixed-integer optimization. *Journal of Global Optimization*. 2014;.

[14] Alonso-Ayuso A, Escudero LF, Martín-Campo FJ, Mladenović N. A VNS metaheuristic for solving the aircraft conflict detection and resolution problem by performing turn changes. *Journal of Global Optimization*. 2015;.

[15] Alonso-Ayuso A, Escudero L, Martín-Campo F. Exact and approximate solving of the aircraft collision resolution problem via turn changes. *Transportation Science*. 2016;.

[16] Alonso-Ayuso A, Escudero LF, Martín-Campo FJ. An exact multi-objective mixed integer nonlinear optimization approach for aircraft conflict resolution. *TOP*. 2016;.

[17] Cafieri S, Omheni R. Mixed-integer nonlinear programming for aircraft conflict avoidance by sequentially applying velocity and heading angle changes. *European Journal of Operational Research*. 2017;.

[18] Maas J, Sunil E, Ellerbroek J, Hoekstra J. The Effect of Swarming on a Voltage Potential-Based Conflict Resolution Algorithm. 2016;.

[19] Dougui N, Delahaye D, Puechmorel S, Mongeau M. A light-propagation model for aircraft trajectory planning. In: *Journal of Global Optimization*. 2013; .

[20] Mitchell IM, Bayen AM, Tomlin CJ. A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games. *IEEE Transactions on Automatic Control*. 2005;.

[21] Wichman K, Lindberg L, Kilchert L, Bleeker O. Four-Dimensional Trajectory Based Air Traffic Management. 2012; .

[22] team A. Report on AGENT functional and non-functional requirements. *Tech. rep.* 2016.

[23] ONERA, Alenia Aeronautica, CIRA, DLR, ENAC, Erdyn Consultans, Israel Aerospace Industries, Monitor Soft, NLR, Technion Israel Institute of Technology, Thales Communications France, TsAGI, University of Patras. 4DCo-GC.
URL `http://www.4dcogc-project.org/`

[24] Hu J, Prandini M, Sastry S. Optimal Coordinated Maneuvers for Three-Dimensional Aircraft Conflict Resolution. *Journal of Guidance, Control, and Dynamics*. 2002;.

[25] Munoz C, Narkawicz A, Chamberlain J. A TCAS-II Resolution Advisory Detection Algorithm. *AIAA Guidance, Navigation, and Control (GNC) Conference*. 2013;.

[26] Koyuncu E, Uzun M, Inalhan G. Cross-entropy-based cost-efficient 4D trajectory generation for airborne conflict resolution. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*. 2016;.

[27] Kosel TH. Computational techniques for stereographic projection. *Journal of Materials Science*. 1984;.

[28] Free Flight in a Crowded Airspace? In: *Air Transportation Systems Engineering*. 2012;.

[29] Sutherland IE, Hodgman GW. Reentrant polygon clipping. *Communications of the ACM*. 1974;.

[30] Vatti BR. A generic solution to polygon clipping. *Communications of the ACM*. 1992;.

[31] Weiler K, Atherton P. Hidden surface removal using polygon area sorting. *ACM SIGGRAPH Computer Graphics*. 2005;.

A Socio-technical Holistic ABM Simulation Framework to Assess Pilots Performance Variability

Miquel Angel Piera[1*], Juan José Ramos [1], Gonzalo Martin[1], Jose Luis Muñoz[2], Jordi Manzano[3]

[1]Dep. of Telecommunications and Systems Engineering. Universidad Autonoma de Barcelona, Barcelona, Spain; *miquelangel.piera@uab.cat*

[2] ASLOGIC, Av Electricidad, 1-21, Planta 2/1, 08191Rubí, Spain; [3]Air Europa Líneas Aéreas, S.A.U, Madrid, Spain

SNE 32(1), 2022, 23-28, DOI: 10.11128/sne.32.tn.10594
Received: 2020-11-10 (Selected EUROSIM 2019 Postconference Publication); Revised: 2021-09-14; Accepted: 2021-10-05
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. One major limitation of developing cognitive computing cockpit supporting tools to maintain the pilots' workload between acceptable lower and upper thresholds is the time variability of humans to perform a well-trained action in a dynamic environment. There are several human factors related issues such as fatigue, stress and workload among others that are reported as the major contributor to human performance variability. Without a deep understanding of the mechanisms that affect pilot performance during the different phases of flight, any support such as a recommended action to improve aircraft stability can affect as an interruption to current cockpit task that increments the workload, forcing pilots to comprehend the consequences of the proposed actions and take a decision about accepting or rejecting the recommendation. This paper presents a socio-technical approach to understand the causes of a degraded mode pilot performance while providing a simulation framework to predict the time windows at which supporting tools could be fired to lessen the pilot workload.

Introduction

The aviation industry has experienced a huge technological evolution from early Clipper Model 314 with five crew positions (navigator, radio operators, flight engineer and two pilots), with each position having specific operating responsibilities (aviate, navigate, communicate and manage system) to today's fly-by-wire and computer systems with two flight crew members in the flight deck.

The introduction of new technologies in the flight deck has allowed important advances in flight control, communication, navigation, and engine management technologies.

This has resulted in a simplified and consolidated control mechanism that reduces flight crew workload for a variety of skill-based and rule-based tasks lessening the amount of manual or repetitive tasks that flight crew have to carry out.

Successful multicrew cockpit achievements through flight deck automatisms have fostered the need for further automatisms to move towards the Single Pilot Operations (SPO) challenge which can provide important cost savings while improving safety [1]. A review of the different conceptual approaches to only one pilot in the flight deck could be found in [6], where the authors recommended an architecture which combines human and automation agents both in air and ground.

However, despite the shift of skill-based and rule-based human operator tasks towards more knowledge-based tasks has been successful in different application fields such as Industry 4.0 [3], the increased system complexity that comes with the new technologies creates novel issues that could increase flight crew workload above their capabilities.

Some issues arise in the cockpit, because the new supporting tools do not simply replace the human in performing a cognitive task but also transforms the actions and introduces new tasks which cannot be predicted at design stages. Thus, for example, shortages of an adequate feedback and support arise at latest stages during validation experiments. Literature [5] describes several automation-related problems and surprises that usually are detected when the human operator is in-the-loop uncovering feedback problems and cognitive task load increments.

Figure 1 represents the task load in a multi-crew cockpit at different phases of a flight [8]. As it can be observed, the peak task load at landing phase could overload a pilot in SPO if an extra task appears, such as an interrupting event (i.e. Air Traffic Controller instruction or Electronic Centralized Aircraft Monitor signal), that requires the attention of the Pilot.

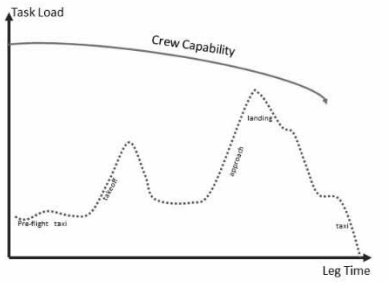


Figure 1: Crew Task load in nominal flying scenarios.

Considering the Wickens' multiple resource theory processing channels [9], and the effects of simultaneous demand on a single channel [2], it can be easily noted that an auditive or visual support to pilot during the peak task load could be counterproductive since task demand could exceed pilot cognitive capacity.

This article describes a socio-technical simulation model to better understand particular requirements for a supporting tool to lessen the cognitive task of a pilot. The remaining of this technical note is organized as follows: Section 1 introduces the socio-technical challenges to tackle the role of the pilot. Section 2 discusses the FRAM modeling formalism while Section 3 presents a FRAM model describing a crew task at approach phase. Section 4 illustrates some results validated during a simulation trial.

1 Socio-technical Modeling Challenges

Airline pilots are trained with formal written procedures acquiring the skills for setting switches, buttons or introducing data in flight systems at the different phases of flight. **Figure 2** represents the pilot situational awareness cognitive processes attending aviate tasks assuming the human in the loop behaviour.

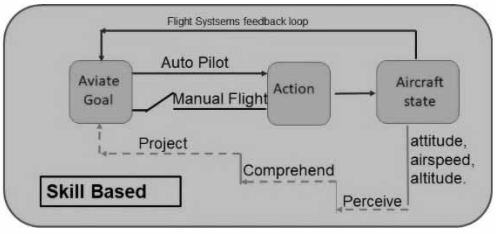


Figure 2: Aviate Pilot in the loop.

Assuming a human-in-the-loop pilot behaviour in which the pilot performs a sequence of actions according to linear procedures, a discrete event simulation model could be built in which an ordered sequence of events, each one described by a deterministic or stochastic time, could replicate the pilot task load and generate similar results to monitored times in training exercises.

Unfortunately, aviate tasks also co-exist with communicate and navigate tasks each one with different priorities that can interrupt the current task, forcing the pilot to attend the interruption, determine the priority with respect to the current task and decide which tasks should be postponed. Thus, pilot behaviour in the flight deck cannot be assumed as a human-in-the-loop attending a sequence of well-trained tasks, rather pilot must constantly attend different sources of aviate, communicate and navigate interruptions such as:

- Air Traffic Controllers (ATCo) can issue an auditive (radio communication) or a visual (data link) instruction at any time, which can cause an interruption to current aviate task.
- Aircraft: Flight deck aircraft are equipped with a warning system to inform crew about abnormal aircraft problems. Thus Boeing implements the Engine Indication and Crew Alerting System (EICAS) while Airbus implements the Electronic Centralized Aircraft Monitor (ECAM) to inform pilot about an aircraft component failure.
- Crew: Pilots can be interrupted at any time by crew through the Service Interphone chime. When it sounds, pilot must react and listen to the Flight Attendant.

In a realistic scenario pilots are frequently interrupted while performing a procedure. Regardless of the particularities of the interruption, pilots must carefully screen the information attached to the interruption and fire a cognitive task that consists of a set of mental actions to predict the future aircraft state if the ongoing task is prioritized to a convenient stopping point before responding to the interruption, or the state that would be reached if pilot attends the interruption and returns to the interrupted task later. Regardless of the pilot choice, there is an increment of pilot mental workload since he must constantly remember to return to the deferred task later. Furthermore, a pilot can perform maximal two concurrent actions, if they do not require the same cognitive channel (i.e. he can monitor a display at the same time he is performing a psychomotor action on a flap), but a third concurrent action usually forces to postpone the lowest priority action and generates a pending memory item that affects the performance, forcing the pilot a "remember to remember" action.

The design of cognitive computing flight deck supporting tools to assist pilots preventing peak workload requires a socio-technical model description of pilot behaviour to understand how and when the assistance should be provided to improve pilot performance [7] and avoiding a degraded mode due to pending memory items.

2 Functional Resonance Analysis Method (FRAM) Formalism

Lack of a modelling guideline to formalize the interaction between cockpit supporting tools and pilot behaviour is an important source of model maintenance problems when new changes must be introduced in the model to predict the impact on new cockpit functionalities.

Furthermore, a scarce understanding about the hidden dynamics on how the context affects interdependencies between human operator and automatisms affects not only the maintenance of the model but also the acceptability and transparency of the results. To overcome present modelling shortages, the FRAM approach (Functional Resonance Analysis Method; [4]) has been used, which provides an excellent functional structure to represent socio-technical systems supporting different abstraction levels in each FRAM component when implemented in an Agent Based Modelling framework. FRAM formalism enhance modellers with a socio-technical approach to formalize Procedures, Actor behaviour, Component Behaviour and the Interdependencies.

In the present implementation, the human cognitive tasks have been described by means of non-linear relationships which consider static attributes and operational context to change the dynamic attributes. Behavioural rules are used also to describe the decision making process considering the dynamic attributes which guides a trade-off between performance and workload. A functional entity is described in FRAM by the six following relations and represented graphically in **Figure 3**.

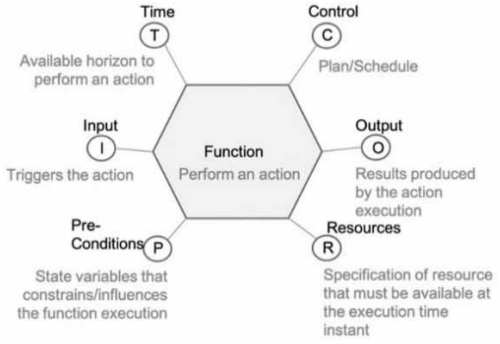


Figure 3: FRAM component.

The interface of FRAM components consists of:

- Input: Triggers an action to be implemented by a computer service, a machine or by a human.
- Time: Available time horizon to perform an action. It can be immediate or with a latency in the case of computer service, or can be a stochastic time parametrized by values of influence variables in case of a human action.
- Resources: Provides an estimation of resource availability at a particular time instant, required to perform the action.
- Control: An action usually requires the adjustment of a function that can be a plan, a procedure or a human task.
- Output: The results produced by an action.
- Preconditions: State variables that must be fulfilled to proceed with the action.

Note, that the FRAM approach is mainly oriented to resilience engineering, trying to determine how variability may interact within a system in a manner that leads to adverse performance outcomes.

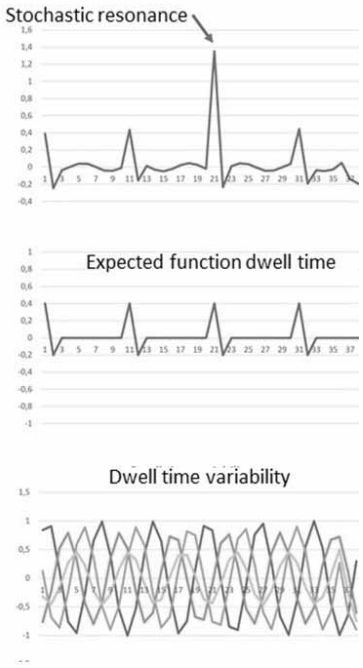


Figure 4: Dwell time variability.

Figure **4** represents graphically the effects of performance variability on the execution of tasks. As it can be observed, the coexistence of time variability when performing different concurrent tasks (lower part of the figure) can cause a peak resonance on the overall behaviour (upper part of the figure) which sometimes can be observed as a time-out (i.e. a task not finalized before a deadline).

This approach has been very useful to support a deep understanding of the overall behaviour of an aircraft pilot in the flight deck. Thus, it is possible to investigate the flight deck functional architecture and to provide an answer to relevant questions such as:

- Why can the combination of safety procedures be unsafe? The identification of the contextual conditions that can impact negatively on safety is an excellent information to guide the changes in the pilot-cockpit procedures to guarantee a resilient flow of tasks.
- Why can the combination of well performed tasks lessen the performance of the overall aircraft system? Note that small delays when performing critical-safe tasks in a fast changing environment can block the finalization of a procedure.

3 Flight Level Authorization FRAM Model

To illustrate the FRAM formalism, this section describes one of the procedures a flight crew should perform during the approach phase.

Flight level authorization procedures describes the main flow of actions and its alternatives the Pilot Flying (PF) performs when an aircraft is located above 8000 ft. and by 40 Nm to the airport, and is initiated by the ATCo which issues a clearance instruction.

The ATC issues the message through the radio to instruct a new FL (flight level), which should be listened by both pilots. Then Pilot Monitoring (PM) repeats the FL instruction to ATC for acknowledgement which is listened also by the PF (action 2-5.1). At that time PF sets FL in the FCU (action 1-5.1). Finally, when PF sets the new FL, then PM crosschecks in the FCU, that the FL is the same that ATC said and, after that, PM should check in the PFD (Primary Flight Display) that FL is blue – with subsequently, PM call-out to PF "nnn FL blue" (nnn is the FL cleared). Moreover, PF had to check that FL was correct in the FCU and has checked "nnn FL blue" in the PFD (action 1-5.3).

Action Code	Action Meaning	Time_ out	Time
1-5.1 PF	Interpret ATC message	15	4
1-5.2 PF	Select the FL	15	11
1-5.3 PF	Check FL blue	15	4
2-5.1 PM	Interpret ATC message	15	4
2-5.2 PM	Acknowledge	15	4
2-5.3 PM	Check FL in FCU	15	6

Table 1: Pilot Flying (PF) and PM (Pilot Monitoring) actions.

Figure 5 and **Figure 6** illustrate the main cockpit instruments to perform the FL Authorization task are the Flight Control Unit (FCU) and the PFD (Primary Flight Display) which are required resources to perform actions 1-5.2 and 1-5.3 respectively. In addition, in action 1-5.2 it is also formalized the resource HM that means a psychomotor action (i.e. Handmade), this mental resource is required because PF set the FL in FCU.

Furthermore, there are some actions such as 1-5.3 and 2-5.3 that are the result of an external process, such as the communication among the ATC and the PF or the communication between the PF and PM. Such external processes can be simple actions without any pre-condition, neither control nor required mental resource, firing the output action as consequence of receiving an input.

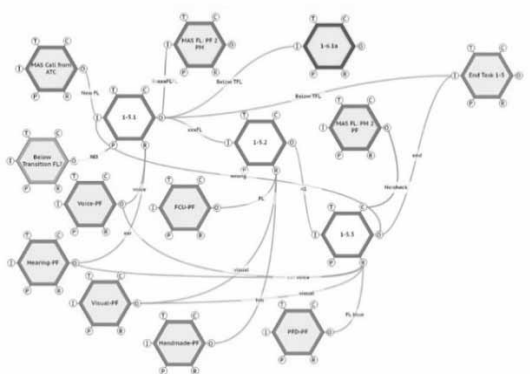


Figure 5: FRAM PF FL Authorization Model.

Main functionality of MAS actions is to introduce a delay that could be caused by the communication channel, or a human reaction time. Thus, action "MAS FL: PM 2 PF" is used to describe the PF reaction time to a communication from PM, while action "MAS FL: PF 2 PM" describes the PM reaction time to a communication from PF.

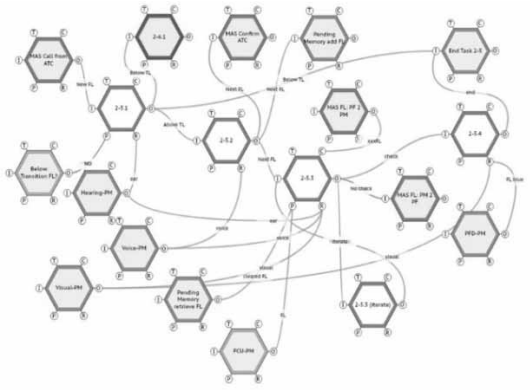


Figure 6: FRAM PM FL Authorization Model.

The triggering of the procedures is an external event driven by the ATC represented in purple, while the end of the task is represented in green and the triggering of a new task is represented in red. Grey colour is used to describe the inherent actions of the task already introduced in Table 1. Important to note, that despite the ATC call and the available resources (both cognitive and cockpit instruments), action 1-5.2 will not be fired if the aircraft is below transition FL (P connection to action 1-5.1).

4 Simulation Results

To illustrate the benefits of the socio-technical modelling approach, it has been validated three different scenarios.

4.1 Nominal FL Authorization Scenario above Transition FL

Figure 7 represents the different actions performed by the Pilot Flying (PF; top Gantt chart) and the Pilot Monitoring (PM; bottom Gantt chart).

Concurrent tasks are represented as a box in the first two rows, while a postponed task is represented by a black line at the third row, and a Pending Memory Item is represented by a yellow line at the 4th row. In this scenario, the aircraft is above the Transition Flight Level (i.e. from flight level to altitude) when the ATC issues the FL authorization at time 25 s.

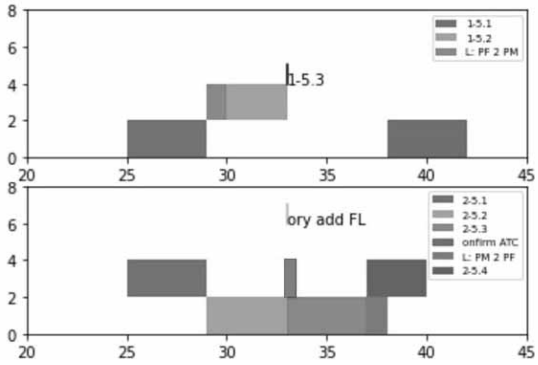


Figure 7: Flight Level authorization nominal scenario.

As it can be observed, PF and PM receive the instruction and perform action 1.5.1 and action 2.5.1 resp. A PF callout to PM checks the FL in the FCU, while at the same time PF is selecting the FL in FCU. As a result, PF must wait that PM confirms that the FL in the FCU is the same FL instructed by the ATC which occurs at time 38 s.

Both Gantt charts postpone active actions until all pre-conditions and mental resources are available. Thus PF action 1-5.3 is postponed until PM confirmation, which occurs in parallel to action 2.5.4 (Check FL blue). Worthwhile to note that action "Pending Memory add FL" is a memory action in which PM must remember the FL issued by the ATC and will be retrieved later to validate the FL selected by the PF.

4.2 Nominal FL Authorization Scenario below Transition FL

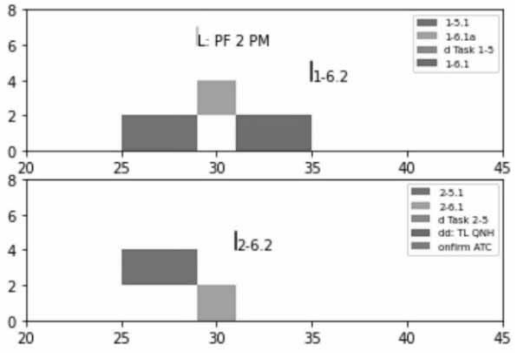


Figure 8: Flight Level authorization nominal scenario below transition level.

The FL authorization procedure considers two different aircraft states. The scenario described in section 4.1 represented the aircraft above the transition level, while in this scenario, the ATC issues the same authorization, but the aircraft is below the transition level.

Figure 8 represents the different actions performed by the Pilot Flying (PF; top Gantt chart) and the Pilot Monitoring (PM; bottom Gantt chart): the PF does not perform the sequential tasks "Select the FL" (1-5.2) and "Check FL blue" (1-5.3), instead he performs action 1-6.1a, which fires the "Altitude Authorization" procedure.

4.3 Interrupted FL Authorization Scenario above Transition FL

There are different sources of interruptions, affecting the actions that pilots are performing, such as an instruction from ATC, an ECAM Warning or a Crew Cabin Call.

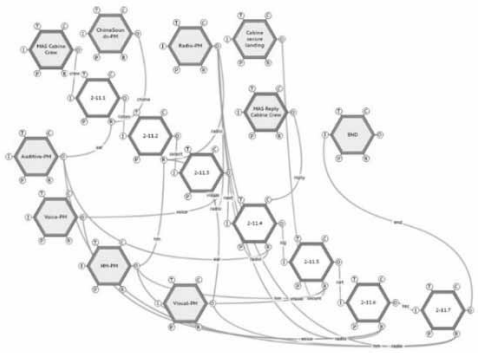


Figure 9: Cabin Crew communication.

Figure 9 represents the sequence of actions Cabin Crew to communicate with PM, while **Table 2** describes the meaning of each action.

Action Code	Action Meaning	Timeout	Time
2-11.1	Listen Call TCP (chime sounds)	60	7
2-11.2	Selector changed to CABIN	15	2
2-11.3	Reply Cabin Crew	60	2
2-11.4	Listen Cabin Secure	15	3
2-11.5	Cabin Secure	15	4
2-11.6	Reply Cabin Secure	15	2
2-11.7	Selector changed to VHF1	15	1

Table 2: PM actions attending Cabin Crew.

Figure 10 represents the different actions performed by the Pilot Flying (PF; top Gantt chart) and the Pilot Monitoring (PM; bottom Gantt chart) when a cabin crew interruption arises five seconds after receiving the ATC authorization. As it can be observed, the amount of PM concurrent actions is increased, and that impacts the performance, since workload is boosted with three pending memory actions that PM should remember to perform once the mental resources are available. The three actions not performed on time are:

- PM Confirms to ATC
- PM Pending Memory add FL
- PM Check FL blue

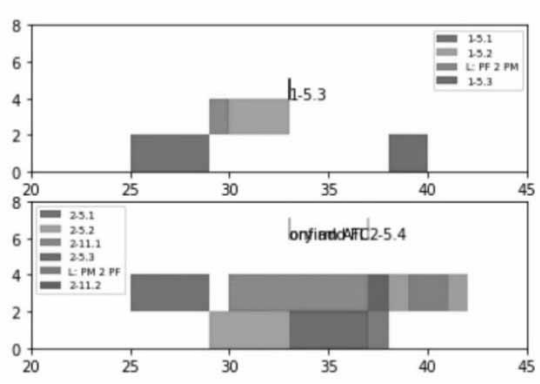


Figure 10: FL authorization with Cabin Crew interruptions.

5 Conclusions

This paper highlights the main modelling requirements of a socio-technical model to properly represent the human-machine behaviour when interacting in a dynamic context.

Functional Resonance Analysis Method (FRAM) has been used as a modelling formalism since its basic component allows the description of mental resources a human actor requires to implement an action, while at the same time it allows also the specification of technical requirements of the supporting tools to enhance human operator to perform the action. The paper illustrates a particular flight deck procedure, pilots should perform considering different aircraft status and potential interrupting events.

As a result, it has been described by means of a Gantt chart the different actions are executed considering the availability of mental resources and supporting tools, postponing some actions until all requirements are satisfied.

The model described has been developed and validated in the European project E-PILOTS (https://e-pilots.eu/), and provides the baseline to analyze how and when elaborated information should be provided (i.e. visual/auditive) avoiding the postponement of an action because the cognitive channel is busy or the human operator is attending 2 concurrent actions.

Acknowledgements

This research has received funding from the Clean Sky 2 Joint Undertaking (JU) under grant agreement Nº 831993 project "E-PILOTS: Evolution of cockPIt operations Levering on cOgnitive compuTing Services" and the national Spanish project: "EU-TM" (ref. TRA2017-88724-R). Opinions expressed in this article reflect the authors' views only.

References

[1] Comerford D, Brandt SL, Mogford P. NASA / CP — 2013 – 216513 NASA ' s Single -Pilot Operations, Technical Interchange Meeting : Proceedings and Findings. April, p. 300, 2013.

[2] Davies AK, Tomoszek A, Hicks MR, White J. AWAS (Aircrew Workload Assessment System): Issues of theory, implementation, and validation. In R. Fuller, N. Johnston, and N. McDonald (Eds.) Human Factors in Aviation Operations. Proceedings of the 21st Conference of the European Association for Aviation Psychology (EAAP), vol. 3, Chapter 48, 1995.

[3] Fantini P, Pinzone M, Taisch M. Placing the operator at the centre of Industry 4.0 design: Modelling and assessing human activities within cyber-physical systems. Computers & Industrial Engineering. V. 139, 2020.

[4] Hollnagel E. The ETTO Principle: Efficiency-Thoroughness Trade-off: Why Things that go Right Sometimes go Wrong. 2009, Ashgate Publishing Ltd.

Lee JD, Seppelt BD. Human Factors of Automation Design. In Handbook of Automation Design, S. Nof, Ed., Springer, New York, pp. 417–436.

[5] Neis SM, Klingauf U, Schiefele J. Classification and review of conceptual frameworks for commercial single pilot operations. AIAA/IEEE Digit. Avion. Syst. Conf. - Proc., vol. 2018-Septe, pp. 1–8, 2018.

[6] Tang J, Piera M, Baruwa O. Discrete-event modeling approach for the analysis of TCAS-induced collisions with different pilot response times. Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering 229(13). 2015.

[7] Silvagni SL, Napoletano I. Graziani, Le Blaye P, Rognin L. Concept for Human Performance Envelope. Futur. Sky Saf., 2015.

[8] Wickens, Situation awareness and workload in aviation. Current directions in psychological science, vol. 11, no. 4, 2002.

Machine Learning and the Digital Era from a Process Systems Engineering Perspective

José L. Pitarch[1*], César de Prada[2]

[1]Inst. de Automática e Informática Industrial (ai2), Universitat Politècnica de Valencia, Camino de Vera s/n, 46022 Valencia, Spain; **jlpitarch@isa.upv.es*

[2]Institute of Sustainable Processes (IPS), Universidad de Valladolid, C/Real de Burgos s/n, 47011 Valladolid, Spain

SNE 32(1), 2022, 29-36, DOI: 10.11128/sne.32.tn.10595
Received: 2020-11-10 (Selected EUROSIM 2019 Postconference Publication); Revised: 2021-09-14; Accepted: 2021-10-05
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. Modern sensorization, communication and computational technologies provide collecting and storing huge amounts of raw data from large cyber-physical systems. These data should serve as the basis to take better decisions at all levels (from the design to operation and management). Nevertheless, raw data need to be transformed in useful information, usually in the form of prediction models. Machine learning plays a key role in this task. Process industry is not alien to this digital transformation, although large processing plants present particularities that differentiate them from other systems. These differences, if neglected, can make machine learning for general purpose fail in extracting the right information from data, leading thus to unreliable process models. As such models are the basis on which the ideas towards the cognitive plant rely, this issue is of major importance for a successful full digitalization of the process industry. In this paper the authors discuss these aspects, as well as some suitable machine-learning approaches, through their experience gained from applying advanced engineering in an industrial case study.

Introduction

In the digital era, the impressing amount of data that can be stored, as well as the speed at which they can be stored, are expected to significantly impact the decision-making procedures at all levels of a factory: from the process design, through the operation and maintenance, to production scheduling and supply chain. Coordinating actions at all levels is the work towards reaching the full digitalized, cognitive and, ultimately, autonomous plant.

However, in the process industries (those that process bulk materials or resources to transform them into products), these expected advances will not come alone by just collecting huge amounts of data and presenting them in a nice view: data treatment and analytics is necessary to ensure the data quality. Moreover, models for reliable predictions need to be built upon such data, in order to be later used in advanced control, optimization and planning routines [1].

Once data quality is ensured, models are to be build, and the current trends from the big-data revolution seem to impose the wide set of machine-learning (ML) techniques in all sectors. However, as the authors will illustrate in this paper, the direct application of an ML approach to a modelling problem in the process industry needs to be evaluated carefully.

In this particular sector, production takes place in a set of complex (and expensive) process units, linking flows of materials and energy at large scale. Nonetheless, the process industry is not characterized by a scarce knowledge on the involved processes: researchers on Process Systems Engineering (PSE) [2] have been developing physical models (e.g., distillation columns) for design, simulation and decision-support solutions during several years. Although these models have limitations for use in real-time applications (computational complexity and/or fitness to actual plants), it is not sensible to throw out all this deep knowledge and replace it by deep learning machines [3]. Thus, one of the key challenges of ML to successfully penetrate in the process industry is developing methods and tools that are able to naturally embed the existing physical knowledge on the underlying processes.

Researchers in PSE have already taken some steps forward in this path:

a) developing hybrid or grey-box models (combination of first-principles laws and regression equations) which get a high matching level with the actual plant [4];
b) proposing methodologies for robust data analysis/ reconciliation [5]; and
c) presenting approaches/tools for data-driven modeling that are tailored to the features of the process industry [6], [7].

In the following sections, the authors discuss the above-mentioned issues with ML through an industrial case study that consists of building a prediction model for the fouling accumulation in the heat exchangers of a multiple effect evaporation plant. Some of the recently proposed methodologies and software are tested on this case study, trying to give the reader a clearer vision on the potential advantages as well as the existing limitations.

1 Description of the Case Study and Motivation

The case study is an evaporation plant in a cellulose fiber production factory, whose objective is to continuously remove certain amount of water from an acid liquid inlet (called *spinbath* hereinafter) that comes from spinning machines, the place where cellulose pulp is recovered into fibers of desired properties.

The plant layout, simplified in Figure 1, makes use of several heat exchangers in serial connection to heat the spinbath up to a temperature suitable to start the evaporation by pressure drop. This pressure drop is first created in the evaporation chambers by induced vacuum, and later by further condensation in an attached surface condenser, creating thus a multiple-effect evaporation. The efficiency of these type of plants (live-steam consumption per amount of evaporated water) is mainly determined by: 1) the performance of the cooling system and 2) the fouling state in the heat exchangers (due to deposition of organic residues present in the spinbath) [8]. Therefore, representative, but of limited complexity, models of these systems are needed to predict online the impact that the operation will have on the plant performance over time.

For such a task, a set of experiments were performed running the plant and the cooling system in different operating conditions (setting different values for the main control variables: spinbath flow, temperature set point, cooling water flow). Moreover, in order to get information on the fouling degradation in the exchangers over time, an extensive dataset corresponding to several months of operation (including stops for cleaning) has been also recovered from the collected plant historian.

In this way, the modeler may be tempted to directly try to find black-box models which relate the live-steam consumption with the input variables through raw measurements. This involves some risks and limitations, as we will see later on.

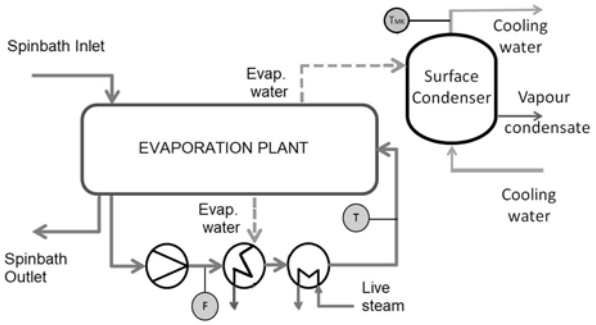


Figure 1: Schema of the evaporation plant with attached surface condenser as cooling system.

2 Data Conditioning and Variable Estimation

Everybody in the machine learning and data-analytics research community claims that ensuring the quality of data is essential to extract sensible information: process measurements need to be coherent and reliable. In industrial practice, however, it is not common to go beyond the standard filters to exclude faulty instrumentation (out of range sensors, communication loss, etc.) and to average data with the aim of mitigating the effect of noise to account for steady state in large-scale systems.

A systematic method to detect and assign the quality of process data can be proposed from the Spanish AENOR-UNE norm 500540 [9], used to analyze data in meteorological stations. This method is based on several progressive levels of tests where each datum is associated to the highest quality level being passed, see Figure 2. Note that the more restrictive tests (thus, the ones ensuring higher confidence data) are model based.

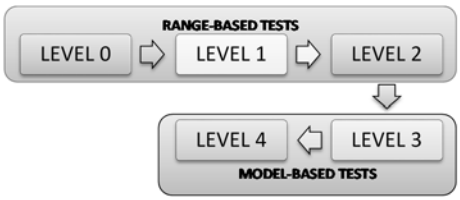


Figure 2: Data quality and validation levels.

Each level depicted in Figure 2 corresponds to the following quality tests:

- **Level 0: Communications.** Check whether the data are recorded or not at the expected sampling time (problems in the sensor or in communications).
- **Level 1: Limits.** Check that the datum is within instrument span and/or physical range. E.g., the maxi-

mum values expected of the flowmeters will be determined by a simple analysis of the flow capacity limit of the pipes.

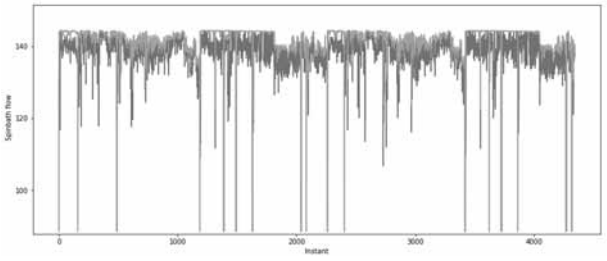
- **Level 2: Trends.** Consider the time changes of the data in consecutive sampling times. E.g., the level in a big tank cannot change faster than several centimetres by minute.
- **Level 3: Data reconciliation.** With a basic first-principle model of the plant, apply methods of (dynamic) data reconciliation (DR) and gross-error detection [5]. This provides a reliable set of measurements as well as estimations of unmeasured variables and parameters that are coherent with the process physics. E.g., mass balances need to be fulfilled in each time instant.
- **Level 4: Time series & correlations.** Consider the time series of the collected values for each variable [10]. E.g., a time-series model can be derived by analyzing the historical data of the flows in a pipe, relating them with valves, and the model output is later used to compare and validate newly recorded data.

Fuzzy logic and set theory can be used to develop filters for the three first levels, based on comparison rules which are able to remove inconsistent data [11]. Different strategies and rules can be used, such as range and speed of change of the measurements, etc. Nevertheless, what really makes the difference in the authors' opinion are the model-based tests, because they include process knowledge in the data processing. Of course, these involve higher engineering effort for implementation, as relatively complex models of the plant/process (either first principles or time series) need to be previously build.

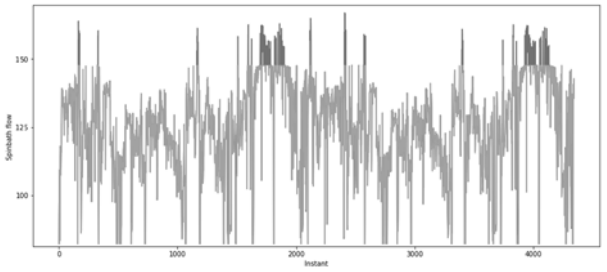
After these quality tests, resource and key efficiency indicators can be defined upon reliable sets of measurements to monitor the plant efficiency in real time [12].

2.1 Instrumentation Issues and DR in the Evaporation Plant

When retrieving sensor data from the historian, the first issues usually arise: many of the collected flow measurements were either "upper bounded" by the instrument range (span-related issue) or they were showing values higher than the actual flow, see Figure 3. In particular, this last problem was not caused by a biased instrument, but because of the improper location of the instrument itself: there was a bypass valve in the pipe after the flowmeter, so a (non-constant) undetermined part of the spinbath was sent to another equipment. Hence, the actual flow was usually lower.



a) Wrong measurement due to wrong sensor placement.



b) Actual values exceeding the instrument range.

Figure 3: Flow-measurement issues. Orange line: sensor values. Blue line: actual values.

Realizing of such wrong values and the explanation took the authors a significant amount of time and several failed modeling attempts. However, most of these data passed the tests of range-based filters. Here we highlight the importance of the model-based tests, because suitable DR of these wrong measurements with mass-balance equations plus the rest of plant measurements provided the corrected values depicted in blue in Figure 3.

Moreover, back to the end goal of predicting the fouling in the evaporation plant, we already encounter an additional issue: the long-term loss of efficiency, only reflected on a single output (the increase of live-steam consumption) is masked with the cooling system performance and the plant operation conditions (spinbath flow). Hence, the fouling effect is hardly identifiable by a direct ML approach with the available measurements.

To overcome this issue, we also recalled dynamic DR [5], including the energy balances in the plant model, to *estimate* the lumped heat-transfer coefficient $UA(t)$ in the exchangers over time [7]. ML techniques can be now applied to "discover" models upon these coherent estimations, also called *virtual measurements* in the soft-sensors related literature. Details provided in the next section.

3 Prediction Models and Constrained Regression

Once reliable values for all process variables (states x, outputs y and inputs u) are available, including coherent estimates of time-varying parameters and / or process un-

known inputs z, any ML approach (e.g., artificial neural networks [4], canonical partial least squares [13], support vector machines [14], etc.) can be, in principle, a good candidate to build plant surrogate models in the general form

$$y = f(\alpha; x, u, z), \alpha \in \mathbb{R}^n \text{ regression parameters,} \quad (1)$$

or submodels (equations being part of a larger model) relating some variables $z^* \in z$

$$z^* = g(\beta; x, u, z), \beta \in \mathbb{R}^m \text{ regression parameters,} \quad (2)$$

At this point, there is a fundamental question to discuss: Even having reliable datasets for regression, are "standard" ML approaches enough to guarantee black-box models whose response is coherent with the process physics? Thinking on it, the answer to that question is in general NO, and the reason is given next. If one outlooks the training methods used by common ML tools, you will find that most of them rely exclusively on data, and that the performance of the black-box model to train is basically defined by the fitness to such data (plus suitable regularization to avoid overfitting, of course). In this way, although the data are coherent with the process physics (passing the tests in Section 3) and the model achieves a perfect fit to such data, there is no guarantee that its response (even with regularized smooth models) takes values that do not violate basic physical principles at input values not contained in the training dataset. Indeed, a model can show good statistics (R2, RMSE, etc.) in validation datasets, but it may still "predict" negative flows out of the training region (extrapolation issues) or a non-monotonic response between consecutive inputs (interpolation issues).

As the end purpose of surrogate or grey-box models is to be used for decision support in (economic) control and optimization routines (hence, mainly for interpolation and extrapolation), the data-driven parts must be in accordance with the process physics [6], [15]. Therefore, some properties on the model response, such as bounds on the outputs and/or in their derivatives (monotony, curvature, convexity, etc.) would like to be ensured, not only over the regression data but in the entire expected region of operation. Therefore, ML in the PSE framework needs to be extended to include additional constraints on the model. Constraints which ideally need to be enforced on infinitely many points belonging to the (usually local) plant operating region. Here is where the concept of *constrained regression* plays a key role.

3.1 Constrained Regression

Assume that a dataset of N samples over time for some outputs y (or, equivalently, estimations of those z^* in (2)) and some inputs (x, u, z) is available. Then, a candidate model for regression $f(\cdot)$ is sought such that a p-measure of the error (e.g., L_1-regularized or least squares) w.r.t. the data is minimized over a set of constraints $c(\cdot)$:

$$\begin{aligned} &\min_{\alpha} \textstyle\sum_{t=1}^{N} \left\| y_{[t]} - f\left(\alpha; x_{[t]}, u_{[t]}, z_{[t]}\right) \right\|_p \\ &\text{s.t.}: c(\alpha; x, u, z) \leq 0 \; \forall \, x \in \mathcal{X}, u \in \mathcal{U}, z \in \mathcal{Z} \\ &\quad \alpha \in \mathcal{A} \end{aligned} \quad (3)$$

Note that the additional constraints $c(\cdot)$ specifying some desired features on the model response are *locally* enforced in a compact region $\Omega := \mathcal{X} \cup \mathcal{U} \cup \mathcal{Z}$ of the input space variables. These constraints may range from the simpler bounds on y ensuring, for instance, non-negativity, to the more complex bounds on the model derivatives (slope, curvature, convexity, etc.). Defined this way, (3) is a semi-infinite constrained nonlinear optimization problem, but it can be computationally tractable under some assumptions [16]. Next, the authors briefly present two approaches and software available to handle (3), jointly with a discussion on their advantages and limitations.

Symbolic regression. In this approach, the functional form of the candidate model is assumed to be unknown a priori. Instead, the algorithm seeks to construct it from a set of predefined basis functions $\mathcal{B}$, e.g. $\mathcal{B} := \left\{1, x, x^2, \frac{1}{x}, \log(x), e^{\tau x}\right\}$. Once this set is specified, the lowest complexity function $f(\cdot)$ that accurately fits the data is found from the selection of the more suitable basis in $\mathcal{B}$ via mixed-integer programming (MIP). The idea is to split the resolution of (3) in two stages: first, solving a data-driven constrained regression (i.e., $c(\cdot)$ is only checked on the points in the dataset) and, subsequently, testing the fulfillment of constraints $c(\cdot)$ by solving a maximum-violation problem with the model already fixed from stage 1. Hence, if a point on the input space is found to violate $c(\cdot)$ with the initially proposed model, such point is virtually added to the inputs dataset and the procedure repeats until no violation of the constraints is found in stage 2 [6].

If the *basis functions* are chosen such that they *are affine in decision variables*; note that this is a strong limitation for the selection of some nonlinear basis functions in practice, like $e^{\tau x}$ (its time constant τ needs to be fixed a priori, i.e., cannot be identified by the fitting algorithm); typically they are coefficients of a linear combination.

In this case, the problem to solve in stage 1 is computationally tractable (MIQP or MILP depending on the chosen norm for the regression error, i.e., $p = 1$ or $p = 2$). For example, for input variables x, problem (3) may become:

$$\begin{aligned}\min_{\alpha,\eta} \sum_{t=1}^{N} (y_{[t]} - (\alpha_0 + \alpha_1 x_{[t]} + \alpha_2 x_{[t]}^2 + \frac{\alpha_3}{x_{[t]}} + \\ \alpha_4 \log(x_{[t]}) + \alpha_5 e^{x_{[t]}}))^2 \\ \text{s.t.}: -\alpha_0 - \alpha_1 x_{[t]} - \alpha_2 x_{[t]}^2 - \frac{\alpha_3}{x_{[t]}} - \\ \alpha_4 \log(x_{[t]}) - \alpha_5 e^{x_{[t]}} \le 0 \;\; t = 1, \dots, N \\ \underline{\mathbf{a_i}} \eta_i \le \alpha_i \le \overline{\mathbf{a_i}} \eta_i \;\; i = 0, \dots, 5; \;\; \alpha_i \in \mathbb{R} \\ \eta_0 + \eta_1 + \eta_2 + \eta_3 + \eta_4 + \eta_5 \le T; \;\; \eta_i \in \{0,1\}\end{aligned} \tag{4}$$

In this way, the basis functions are active when the corresponding binary variable $\eta_i = 1$ and inactive otherwise. Model complexity is specified by a parameter T that is increased until a goodness-of-fit measure worsens. Afterwards, in step 2 an adaptive sampling methodology based on *derivative-free global optimization* techniques is used to identify points where the model is inaccurate and/or does not fulfill constraints - for the above case:

$$\begin{aligned}\max_{x} \;\; & \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \frac{\alpha_3}{x} + \alpha_4 \log(x) + \alpha_5 e^x \\ \text{s.t.}: \;\; & x \in \mathcal{X}\end{aligned} \tag{5}$$

Note importantly that this problem is in general nonlinear and nonconvex.

This procedure is what the software ALAMO implements [18]. Although this approach involves iterations between MIP and NLP problems to global optimality (time consuming).

Sum-Of-Squares (SOS) regression. An alternative approach is casting problem (3) as a polynomial SOS optimization one [20] under mild assumptions. Of course, the main limitation of this approach is that the candidate models $f(\cdot)$ need to be polynomial in their arguments, i.e., the "potential set of basis functions" would be formed only by monomials in the input variables up to a predefined degree (the approach is recently extended in [19] to allow including some "smooth" non-polynomial basis via polytopic bounding). Nonetheless, paying this price worth it, because the resulting (single) optimization problem is *convex*, and the extra *constraints on the model* response and/or in its *derivatives are naturally enforced* (either globally, or locally in a region Ω defined by polynomial boundaries) with full guarantee of satisfaction, no matter how many samples are to be fitted, or which region was covered by the experiments. In this way, high-order polynomial regressors can be used with guarantees of well-behaved resulting function approximators, compared to most options in prior literature. For instance, a SOS version of the above (4)-(5) could be:

$$\begin{aligned}\min_{\alpha,\beta,\phi} \; & \textstyle\sum_{t=1}^{N} \phi_t \quad \text{s.t.}: \\ & \begin{bmatrix} \phi_t & \alpha_0 + \alpha_1 x_{[t]} + \alpha_2 x_{[t]}^2 + \alpha_3 x_{[t]}^3 + \alpha_4 x_{[t]}^4 \\ (*) & 1 \end{bmatrix} \succcurlyeq 0 \\ & t = 1, \dots, N; \phi_t \in \mathbb{R} \\ & \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \alpha_3 x^3 + \alpha_4 x^4 \\ & \quad - s(\beta; x) \cdot (5^2 - x^2) \text{ is SOS } \forall\, x \in \mathbb{R} \\ & \underline{\mathbf{a_i}} \le \alpha_i \le \overline{\mathbf{a_i}} \;\; i = 0, \dots, 5; \;\; \alpha_i \in \mathbb{R} \\ & (\beta; x) \coloneqq \beta_0 + \beta_1 x + \beta_2 x^2, \beta_j \in \mathbb{R}, \text{is SOS } \forall\, x \in \mathbb{R}\end{aligned} \tag{6}$$

Here, well-known Schur complement and Positivstellensatz results (see [7] for details) have been used to cast the quadratic objective function (with extra decision variables ϕ) and the local enforcement of the constraint in a region $\Omega \coloneqq \{x: |x| \le 5\}$ (with extra decision variables β), respectively. Note that the highest degree of the polynomial SOS multipliers $s(\cdot)$ is chosen such that $\deg(s(\beta; x) \cdot (5^2 - x^2)) \ge d$, being d the degree of the candidate polynomial model to fit.

In this case, although no automatic selection of the suitable monomials among a potential set is done via MIP, note that standard regularization on the model coefficients α can be trivially included in the objective function, for instance with a metaparameter Γ that progressively weights the coefficients corresponding to high-degree monomials.

3.2 Application to the Case Study

Recall from Section 2 that the aim is to get data-driven prediction models of limited complexity for the cooling power and the fouling evolution in the plant.

Modeling the cooling power provided by the surface condensers. The actual cooling power can be computed from the data collected by the temperature sensors at the water inlet (T_{in}) and outlet (T_{out}) of the SC, and by the flowmeter measuring the volumetric water flow (F_{w}) send through the SC, as follows:

$$C_{\text{pow}} = \frac{4.18}{3600} F_{\text{w}} \cdot (T_{\text{out}} - T_{\text{in}}) \tag{7}$$

Thus, what is missing to fully predict the cooling power is a model that relates the outlet temperature T_{out} with the water flow F_{w} and the inlet temperature T_{in}.

To model that, a polynomial candidate function up to degree 3 in F_{w} was proposed to experimentally fit the recorded temperature difference $\Delta T \coloneqq T_{\text{out}} - T_{\text{in}}$ [21]:

$$\Delta T = \alpha_0 + \alpha_1 F_{\text{w}} + \alpha_2 F_{\text{w}}^2 + \alpha_3 F_{\text{w}}^3 \tag{8}$$

The fitting of (8) to the experimental data was done first by standard LS unconstrained regression, obtaining the resulting blue curves depicted in Figure 4. As it can be seen, when computing the cooling power with the obtained model, it shows a behaviour incoherent with the physics at high flows (region highlighted in the dashed box), i.e. the cooling power cannot decrease at higher flows. However, the model fitted the measured outlet temperature quite well (T_{in} was nearly constant during the experiments), with a monotonic response in fact, but this didn't avoid the wrong response in C_{pow}.

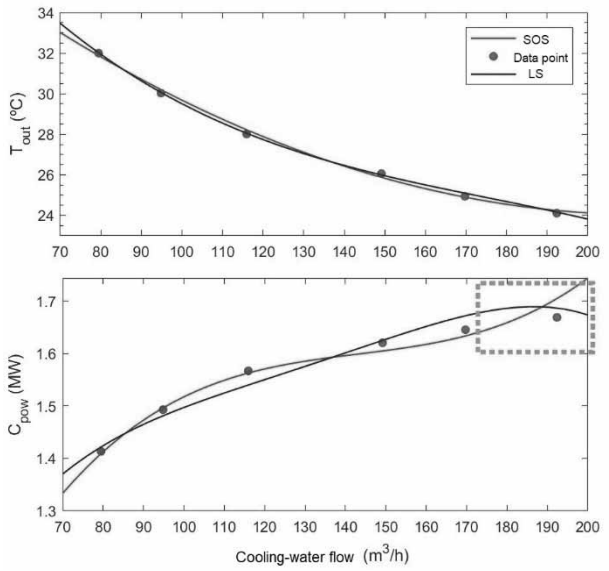


Figure 4: Fitting the cooling power developed by the SC at different water flows.

Then, SOS constrained regression was recalled in a second attempt, adding the constraint $\mathrm{d}C_{\text{pow}}/\mathrm{d}F_{\text{w}} > 0$ to enforce the known physical knowledge on the response. Note that derivatives of polynomials are also polynomials that can be directly checked for SOS. Now, the obtained model (red curves in Figure 4) behaves as expected, without showing any significant fitting degradation w.r.t the obtained by standard LS.

Modeling the heat transfer in exchangers. The goal here is to build up a model to predict both the influence of the spinbath flow F_{SB} and the operation time since last cleaning task t_{op} on the lumped heat-transfer coefficient UA (i.e. the fouling effect). Here the authors made use of the UA estimations provided by DR, already mentioned in Section 3 (omitted for brevity, see [7]).

The first issue arose when selecting sets for training and validation: although the recorded dataset looked huge (plant historian of 7-months length at 5-min. sampling time), the plant was usually running at high flows. Therefore, significant information of the convection and fouling behaviours at medium/low flows was missing.

In order to palliate this issue, a few experiments were executed on purpose when possible (normally it is not possible to "play" with an industrial plant in continuous production). Consequently, as often happens in the process industry, the authors thought that they will be facing "big-data stuff" in principle, but they ended up working with subsets of 22 samples for training plus 20 for validation, depicted in the figures below. This is nearly all the information available in the region of operation.

With this material, if no additional information about the process physics is included in the fitting problem, standard ML techniques fail in obtaining reliable black-box models in the regions where there is a lack of data to fit. See for instance problems of overfitting with standard LS in Fig. 5a, and problems of abrupt-falling responses (even going negative) where data is missing in Fig. 5b, despite using regularization techniques.

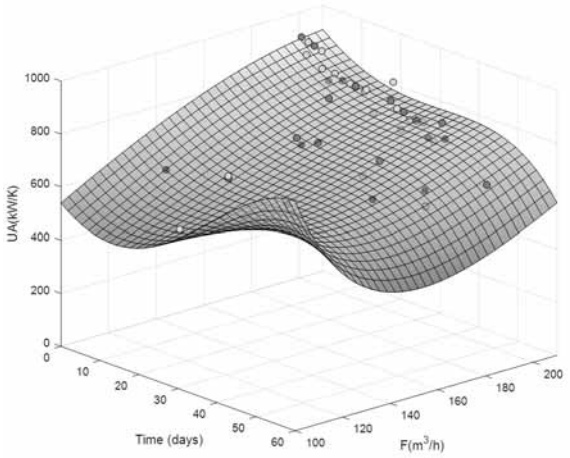
On the contrary, constrained regression in Section 4.1 fixed these modeling issues. We tested symbolic regression using the software ALAMO, with a large set of basis functions including monomials up to degree 4, rational powers, square roots, logarithms and exponentials. We also set up the additional constraint $f(\alpha; F_{\text{SB}}, t_{\text{op}}) > 200$ in the local-input region Ω. Thus, choosing the Akaike's criterion to avoid overfitting, we got the model (Fig. 6a):

$$UA = 2.27F_{\text{SB}} - 0.9095t_{\text{op}} + 84.978\log(F_{\text{SB}}) - 42.525\sqrt[3]{t_{\text{op}}} \tag{9}$$

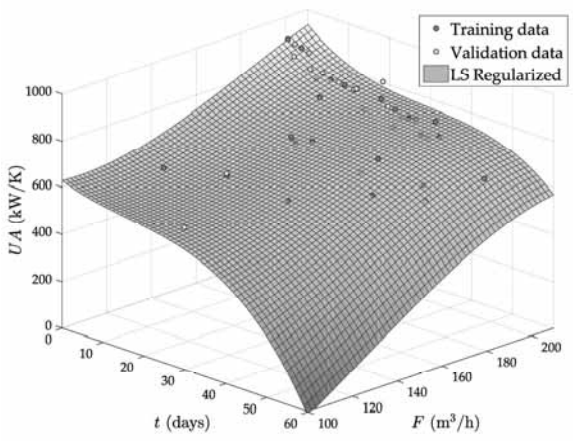
Going by the way of SOS constrained regression, proposing a candidate polynomial model of degree $d = 4$ and setting (local) bounds on its partial derivatives

$$0 < \frac{\mathrm{d}f(\alpha;F_{\text{SB}},t_{\text{op}})}{\mathrm{d}F_{\text{SB}}} < \lambda_{\text{F}}, \quad -\lambda_{\text{t}} < \frac{\mathrm{d}f(\alpha;F_{\text{SB}},t_{\text{op}})}{\mathrm{d}F_{\text{SB}}} < 0 \;\forall\, F_{\text{SB}}, t_{\text{op}} \in \Omega \tag{10}$$

to enforce a smooth and physically-coherent response, the model of Figure 6b is got [7].



a) Least-squares fitting



b) Least squares with regularization

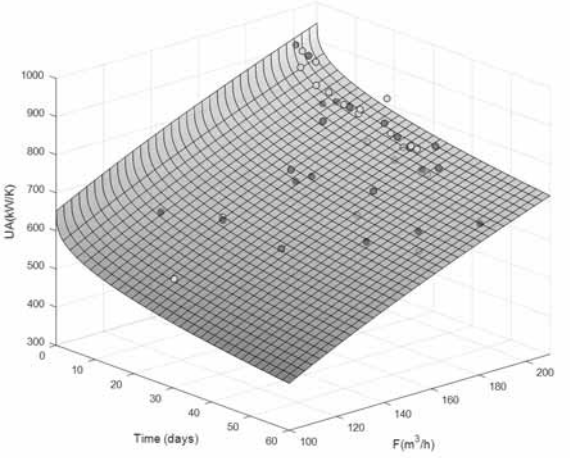
Figure 5: Fitting the heat-transmission coefficient by standard procedures.

Note that such model (11), got by SOS-constrained regression, keeps the desired physical features without incurring in significant fitness deterioration w.r.t. "the best" obtained by unconstrained LS regression with regularization, see the Table 1.

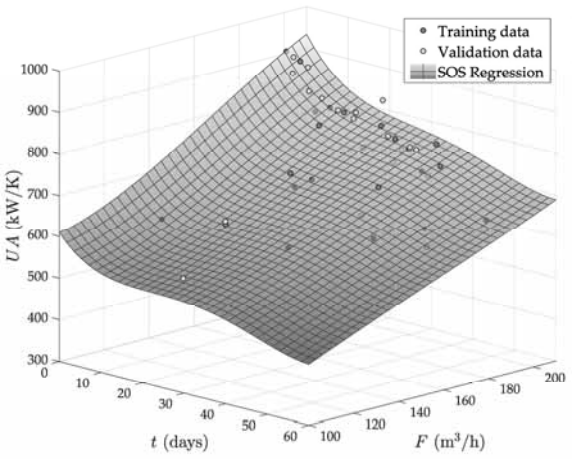
$$\begin{aligned} UA = {} & 7.06e^{-8}F_{SB}^4 + 2.95e^{-6}F_{SB}^3 t_{op} + \\ & 1.63e^{-6}F_{SB}^2 t_{op}^2 - 2.42e^{-6}F_{SB} t_{op}^3 + 1e^{-4} t_{op}^4 - \\ & 2e^{-4}F_{SB}^3 - 1.585e^{-3}F_{SB}^2 t_{op} + 5.1e^{-5}F_{SB} t_{op}^2 - \\ & 0.0138 t_{op}^3 + 0.089 F_{SB}^2 + 0.232 F_{SB} t_{op} + 0.627 t_{op}^2 - \\ & 10.87 F_{SB} - 22.78 t_{op} + 1000 \end{aligned} \quad (11)$$

4 Final Remarks

Digitalization in industrial sites is not just smart sensors, huge databases, and nice monitoring tools. In the authors' opinion, the step beyond current practice is to really extract and combine all the available process information to take better decisions in real time.



a) Fitting using software ALAMO



b) Fitting by SOS regression

Figure 6: Fitting the heat-transmission coefficient by constrained regression methods.

Method	Train. Err.	Val. Err.	Total	Decay
LS	14.452	15.226	29.719	7.17%
LS reg.	13.448	14.282	27.730	-
ALAMO	18.061	18.402	36.463	31.5%
SOS CR	14.751	13.362	28.113	1.38%

Table 1: Absolute least squared error to data accumulated by the presented models.

In this paper, the authors discussed how essential is incorporating process knowledge with sampled data in order to really extract sensible information, which can be later use for decision support in the process industry. For this task, model-based tests to detect (and improve) the data quality (robust DR methods in particular) as well as

constrained-regression approaches proven to be quite effective in our case study.

Constrained regression is especially relevant/useful when data is scarce, or when there are lots of samples but containing nearly the same information about the process. It is also worth to remark that incoherent model responses could be detected (and corrected ad-hoc perhaps) in two- or three-dimensional models, but this would be impossible in larger multidimensional systems.

Acknowledgement. This research received funding from the EU Horizon 2020 research and innovation programme under Grant No. 723575 (CoPro), and from the Spanish MICINN with FEDER funds via the research project InCO4In (PGC2018-099312-B-C31).

References

[1] Krämer S, Engell S (editors). *Resource efficiency of processing plants: monitoring and improvement*. Weinheim: Wiley-VCH, 2018, 504 p.

[2] Grossmann IE, Harjunkoski I. Process systems Engineering: Academic and industrial perspectives. *Computers & Chemical Eng.* 2019; 126: 474–484. doi: 10.1016/j.compchemeng.2019.04.028.

[3] Witten IH, Frank E, Hall MA. *Data mining: practical machine learning tools and techniques.* 3rd ed. Burlington, MA: Morgan Kaufmann, 2011, 629 p.

[4] Zorzetto LFM, Filho RM, Wolf-Maciel MR. Processing modelling development through artificial neural networks and hybrid models. *Computers. & Chemical Eng.* 2000; 24(2-7): 1355–1360. doi: 10.1016/S0098-1354(00)00419-1.

[5] de Prada C, Sarabia D. Data Pre-treatment. In: Krämer S., Engell S., editors. *Resource efficiency of processing plants: monitoring and improvement*. Weinheim: Wiley-VCH, 2018, p 181-210.

[6] Cozad A, Sahinidis V, Miller DC A combined first-principles and data-driven approach to model building. *Computers. & Chemical Eng.* 2015; 73: 116–127. doi: 10.1016/j.compchemeng.2014.11.010.

[7] Pitarch JL, Sala S, de Prada C. A Systematic Grey-Box Modeling Methodology via Data Reconciliation and SOS Constrained Regression. *Processes.* 2019; 7(3): 170. doi:10.3390/pr7030170.

[8] Marcos MP., Pitarch JL, de Prada C. Integrated Process Re-Design with Operation in the Digital Era: Illustration through an Industrial Case Study. *Processes* 2021; 9(7):1203. doi:10.3390/pr9071203.

[9] AENOR. *Automatic weather stations networks: Guidance for the validation of the weather data from the station networks. Real time validation*. UNE 500540:2004.

[10] Blanch J, Puig V, Saludes J, Quevedo J. ARIMA Models for Data Consistency of Flowmeters in Water Distribution Networks. *IFAC Proc. Vol.* 2009; 42(8): 480–485. doi: 10.3182/20090630-4-ES-2003.00080.

[11] Last M, Kandel A. Automated Detection of Outliers in Real-World Data. In *Proc. of the second inter. conf on intelligent technologies*. 2001 Nov; Bangkok. 292-301.

[12] Kujanpää M, Hakala J, Pajula T, Beisheim B, Krämer S, Ackerschott D, Kalliski M, Engell S, Enste U, Pitarch JL. *Successful Resource Efficiency Indicators for process industries: Step-by-step guidebook.* Espoo: VTT Technical Research Centre of Finland, 2017, 78 p.

[13] Indahl UG, Liland KH, Naes T. Canonical partial least squares-a unified PLS approach to classification and regression problems. *J. of Chemometrics*. 2009; 23(9): 495-504. doi:10.1002/cem.1243.

[14] Yan W, Shao H, Wang X. Soft sensing modeling based on support vector machine and Bayesian model selection. *Computers. & Chemical Eng.* 2004; 28(8): 1489–1498. doi: 0.1016/j.compchemeng.2003.11.004.

[15] Tulleken HJAF. Grey-box modelling and identification using physical knowledge and bayesian techniques. *Automatica*. 1993; 29(2): 285-308. doi: 10.1016/0005-1098(93)90124-C.

[16] Stein O. *Bi-Level Strategies in Semi-Infinite Programming*. In: Pardalos P., editor, Nonconvex Optimization and Its Applications, vol. 71. Boston, MA: Springer US, 2003, 202 p.

[17] Hurvich CM, Tsai C. A corrected Akaike information criterion for vector autoregressive model selection. *J. of Time Series Analysis.* 1993; 14(3): 271-279. doi: 10.1111/j.1467-9892.1993.tb00144.x.

[18] Wilson ZT, Sahinidis NV. The ALAMO approach to machine learning. *Computers. & Chemical Eng*. 2017; 106: 785–795. doi: 10.1016/j.compchemeng.2017.02.010

[19] Pitarch JL, Montes DA, de Prada C, Sala A. Application of SOS-constrained regression to model unknown reaction kinetics. *IFAC-PapersOnLine*. 2021; 54(3): 395-400. doi: 10.1016/j.ifacol.2021.08.274.

[20] Papachristodoulou A, Anderson J, Valmorbida G, Prajna S, Seiler P, Parrilo P, Peet MM, Jagt D. *SOSTOOLS: Sum of squares optimization toolbox for MATLAB.* 2013. http://arxiv.org/abs/1310.4716

[21] Marcos MP, Pitarch JL, de Prada C, Jasch C. Modelling and real-time optimisation of an industrial cooling-water network. In *22nd Inter. Conf. on System Theory, Control and Computing.* IEEE: 2018 Oct; Sinaia. 591–596. doi: 10.1109/ICSTCC.2018.8540655.

Extendable Hybrid Approach to Detect Conscious States in a CLIS Patient Using Machine Learning

Sophie Adama[1], Shang-Ju Wu[1*], Nicoletta Nicolaou[2,3], Martin Bogdan[1]

[1]Department of Neuromorphe Information Processing, Leipzig University, Augustusplatz 10, 04109 Leipzig, Germany; *adama* | **shanglu* | *bogdan@informatik.uni-leipzig.de*

[2]Medical School, University of Nicosia, Cyprus; *nicolaou.nic@unic.ac.cy*

[3]Centre for Neuroscience and Integrative Brain Research (CENIBRE), University of Nicosia Medical School, Nicosia, Cyprus

SNE 32(1), 2022, 37-45, DOI: 10.11128/sne.32.tn.10596
Received: 2020-11-10 (Selected EUROSIM 2019 Postconference Publication); Revised: 2021-09-08; Accepted: 2021-10-05
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna, ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. In this study a method for uncovering consciousness in complete locked-in syndrome (CLIS) patients is proposed. The main characteristic of CLIS patients is sufficiently intact cognition, but complete paralysis. It is, thus, vital to develop alternative means of communicating with CLIS patients, and brain-computer interfaces offer a possible platform to do so. A major issue in the study of consciousness in CLIS patients is that there is no certitude regarding their actual state of consciousness. Existing methods provide only a probability of what the states of the patients might be at each moment. This paper proposes a hybrid system based on the combination of complex coherence, sample entropy and Granger causality to uncover the underlying state of consciousness in a CLIS patient from electrocorticography signals. The contribution of each method to the system is determined using machine learning techniques. The aim of the research is to increase the probability of correctly detecting the patients' consciousness states and, ultimately, use that to develop a reliable brain-computer interface-based communication tool.

Introduction

Locked-in syndrome (LIS) is a state where patients are fully conscious but are unable to produce any speech or perform any muscle movements. Although it is not a disorder of consciousness, LIS is frequently misdiagnosed as one. One such a case was a patient who was considered in an unresponsive wakefulness syndrome (UWS) for 20 years [1]. Patients in a LIS state may still be able to move their eye muscles and can, thus, communicate using eye movements. However, even this limited communication becomes impossible when patients enter a complete locked-in state (CLIS) [2], during which it is thought that cognitive function and consciousness are maintained, but all muscle control is lost. Even though no means of communication is available to interact with CLIS patients, some attempts have been made using electroencephalography (EEG) and Near Infrared Spectroscopy (NIRS) [3], but not without controversy [4]. The most important limitation in any such approach is the lack of "ground truth". It is not possible to ascertain the "true" level of consciousness as the patients cannot express their will or answer in any manner.

We present here a general approach that combines different approaches to uncover the state of consciousness in a CLIS patient from continuously recorded electrocorticography (ECoG). The key advantage of this study is that the all-important "ground truth" is accessible and can provide objective means of detecting the presence of consciousness in such patients. We were able to obtain such a "ground truth" from a CLIS patient, who successfully communicated and answered patient-specific questions asked by an investigator using a brain-computer interface system. To the best of our knowledge, there is no other such dataset in existence. Our ultimate goal is to develop a method in order to detect consciousness in CLIS patients to re-establish communication during the time they are conscious.

The paper is organized as follows. The idea behind the general approach is first presented using a modus operandi. The methods that have initially been investigated as part of this approach, namely complex co-

herence, multiscale entropy and Granger causality, are then described. Preliminary results are presented and discussed, with considerations regarding future development of the proposed approach.

1 Modus Operandi

To uncover the state of consciousness in a CLIS patient from continuously recorded electrocorticography (ECoG) and with a priori knowledge of the "ground truth", we propose the use of a hybrid system incorporating a combination of feature extraction methodologies (in these initial investigations we use complex coherence, multiscale entropy and Granger causality) and machine learning (e.g. reinforcement learning to capture inter-subject variability) to detect the level of consciousness (Figure 1). The proposed system is modular and can, thus, incorporate additional combinations of consciousness detection algorithms to augment detection accuracy.

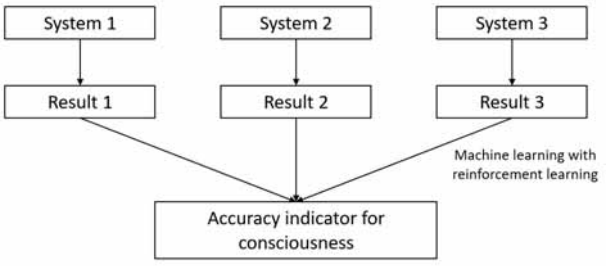
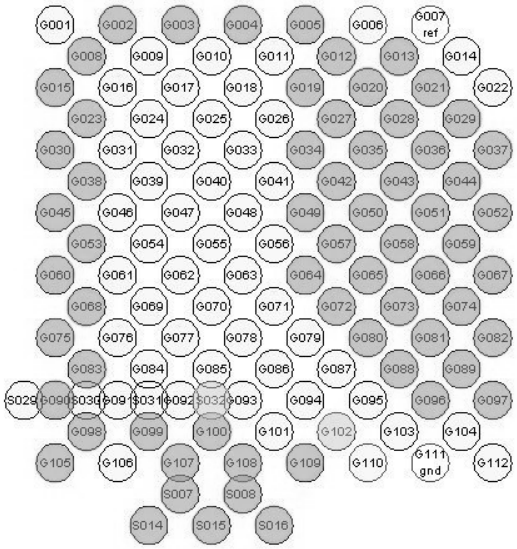


Figure 1: Scheme for the hybrid approach using three different processing systems.

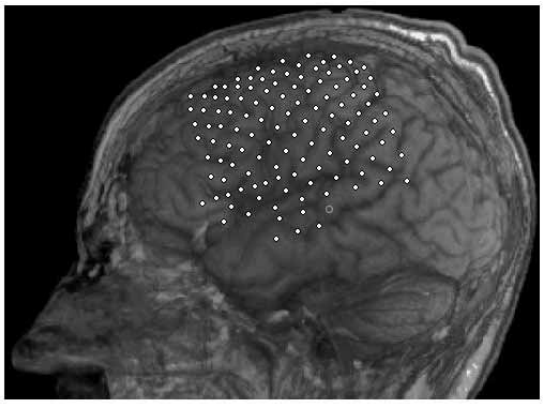
2 Methods

2.1 Dataset

The data was obtained from a 40-year-old male in a complete locked-in state (CLIS). The patient was first diagnosed with amyotrophic lateral sclerosis (ALS) in 1997 and entered CLIS 11 years later [5]. The dataset comprises 24 consecutive one-hour recordings (i.e. 24 hours) of the patient's intracranial brain activity (electrocorticogram - ECoG), acquired with a 64-channel amplifier (BrainAmp from Brainproducts GmbH, Munich, Germany) at a sampling rate of 500 Hz. The ECoG grid electrodes were surgically placed on the patient's left frontal and parietal lobes [1, 6], as shown in Figure 2. The specific channel locations, as well as the locations of the ground and reference electrodes, are also shown in Figure 2.



(a)



(b)

Figure 2: Channel positions. **(a)** Channel names, with functional recording channels shown in green, and ground and references channels in yellow. **(b)** Spatial location of surgically implanted ECoG grid electrodes.

An auditory paradigm similar to [3] was performed from 14:50 to 17:00, in which the patient was asked open questions requiring a "yes" or "no" answer. The questions covered a range of topics such as his mood and feelings and his physiological status, for example: "You feel good today?"/"You feel bad today?" or "Are you German?"/"Are you Dutch?". During the session, a pre-trained classifier was used to give feedback on the predicted answer.

2.2 Feature extraction: Complex coherence

Complex coherence, C_{xy}, at frequency f, of two signals, x and y, is defined as the ratio [7]:

$$C_{xy}(f) = \frac{S_{xy}(f)}{\sqrt{S_{xx}(f) \cdot S_{yy}(f)}} \quad (1)$$

where $S_{xy}(f)$ is the cross power spectral density of the signals, and $S_{xx}(f)$ and $S_{yy}(f)$ is the auto power spectral density of x and y respectively. Coherence is defined as $|C_{xy}(f)|^2$ and is typically used to measure the degree of association between two time series at a specific frequency f. Coherence ranges between 0 and 1.

Coherence has many applications in neuroscience, such as measuring functional relationships between pairs of brain regions. An increased functional interaction between the underlying neuronal networks leads to a higher value of coherence. The complex coherency is to reduce the effects of volume conduction in the brain [8]. Previous researches suggest that the brain waves of locked-in syndrome patients are nearly similar as those of healthy subjects [1]. For that reason, we hypothesize that patients brain rhythms would to some extent behave like the healthy subject's ones.

The data was analysed using MATLAB R2018b (The MathWorks, Natick MA, USA) and custom written codes. Prior to any other processing, the data was re-referenced to the mean and band pass filtered at frequencies 0.5 to 50 Hz using a third order Butterworth filter [9]. The signals were down-sampled to 100 Hz afterwards to reduce the computation time. The signal was subsequently partitioned into segments of 1-second length. Finally, for each frequency band (delta: 0.5-4 Hz, theta: 4-8 Hz, alpha: 8-12 Hz, beta: 12-30 Hz and gamma: 30-50 Hz) and for each time segment, coherency values were computed according to equation (1). A series of coherence matrices were, thus, obtained, each matrix representing the coherence between all pairs of electrodes at each time point. For each of the frequency bands of interest, these sequences were combined to produce a video file, thus facilitating visual inspection of any changes in the coherence over time.

2.3 Feature extraction: Entropy

Entropy is a physical concept, which is related to the total amount of disorder within a system. It can be used for nonlinear dynamic analysis in both the time and frequency domain to quantify the regularity (predictability) of a time series. The family of entropy-based methods is frequently used in neuroscience applications [10, 11, 12].

Sample entropy. Sample Entropy (*SampEn*) is a modification of Approximate Entropy (*ApEn*) proposed by Richman and Moorman [13] to address some of the limitations of *ApEn*. Specifically, the advantages of *SampEn* over *ApEn* are data length independence and non-inclusion of self-matches in the estimation (inclusion of self-matches in *ApEn* result in an interpretation of the signals as more regular than they are). *SampEn* has been used in a number of neuroscience applications, including applications relating to consciousness, such as evaluation of the patient's depth of anaesthesia (DOA) in surgery.

Consider a time series $X = [x(1), x(2), \ldots, x(N)]$, with a total of N samples. To estimate the *SampEn*, the time series is divided into a group of m-dimensional vectors (where m is the embedding dimension), $u_m(1), \ldots, u_m(N-m)$, with $u_m(i) = [x(i), x(i+1), \ldots, x(i+m-1)]$, $i = 1 \ldots N-m+1$. Define the distance, $d[u_m(i), u_m(j)]$, between $u_m(i)$ and $u_m(j)$:

$$d[u_m(i), u_m(j)] = \max |x(i+k) - x(j+k)| : 0 \leq k \leq m-1 \quad (2)$$

A threshold, $R = r * SD$, where SD is the standard deviation of the time series X and r is the tolerance, is set for the distance. $SampEn(N, m, r)$ is then estimated as

$$SampEn(N, m, r) = -\log \frac{A^m(r)}{B^m(r)} \quad (3)$$

where:

$$B^m(r) = (N-m)^{-1} \sum_{i=1}^{N-m} B_i^m(r) \quad (4)$$

$$A^m(r) = (N-m-1)^{-1} \sum_{i=1}^{N-m} A_i^m(r) \quad (5)$$

B_i^m be the number of vectors for which $d[u_m(i), u_m(j)] < R$, and A_i^m the number of vectors for which $d[u_{m+1}(i), u_{m+1}(j)] < R$. Larger values of *SampEn* indicate reduced self-similarity of the series and increased time series complexity. In contrast, smaller *SampEn* values indicate increased self-similarity and lower complexity. Thus, we are expecting that during periods of increased conscious-

ness the *SampEn* values will be higher. The variables m and r need to be set in advance. Commonly used values for neuroscience applications are $m = 1, ..., 3$ and $r = 0.1, 0.2$ [13, 14].

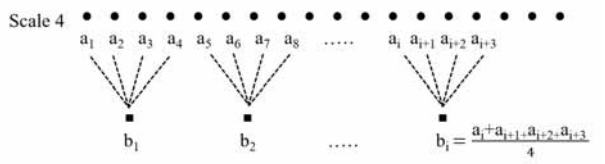


Figure 3: Multiscale Entropy: the coarse-graining procedure for scale 4.

Multiscale entropy. The method of Multiscale Entropy (MSE) analysis is useful for investigating complexity, in contrast to regularity, in signals that have correlations at multiple (time) scales. MSE is an extension of *SampEn* that was proposed by Costa et al. as a way of reducing the effect of white noise that is present in *SampEn* estimations [15, 16]. This is achieved by, first, obtaining a coarse-graining of the data by averaging the data points in non-overlapping windows of increasing length (scale). *SampEn* is then estimated for each coarse-grained series to obtain an index of complexity over multiple time scales. An example of the coarse graining for a scale of 4 is shown in Figure 3.

For the specific dataset, in order to reduce the computation time the original ECoG signals were, first, down-sampled to 125 Hz. Secondly, the down-sampled signals were band pass filtered by a sixth order Butterworth Filter in 1-45 Hz. Finally, the multiscale entropy algorithm is applied to obtain a level of consciousness. For *SampEn* we have set $m = 3$ and $r = 0.2$, and for MSE we have set the scale to 4. Similarly to *SampEn*, higher (lower) values of MSE indicate higher (lower) time series complexity. Thus, we are expecting that periods of consciousness will be characterised by higher MSE values.

We applied sample entropy algorithm and according to the relationship between parameters N and m, the value of N between minimum 10^m and maximum 30^m (when $m = 3$) is chosen [13, 14].

2.4 Granger causality

The concept of causality was first introduced by Wiener [20], as a means of quantifying cause-effect interactions between variables through modelling, prediction and assessment of the goodness-of-fit of models that incorporate the past information from one variable (cause) into the prediction of another variable (effect). Wiener's definition of causality is as follows: *for two simultaneously measured signals, if one can predict the first signal better by incorporating the past information from the second signal than using only information from the first one, then the second signal can be called causal to the first one*. Causality was given a formal mathematical framework by Granger, whereby the goodness-of-fit was assessed through the variance of the residual error of the fitted univariate and bivariate models, i.e. the smaller the residual error variance, the better the fit [21]. The common models of choice for Granger causality are Autoregressive models (AR). For a time series, $X_j = [x_1, x_2, \ldots, x_T]$, a univariate AR model is described by

$$x_j(t) = \sum_{i=1}^{p} a_{ix_j} x_j(t-i) + e_{x_j}(t) \qquad (6)$$

where a_{ix_j} are the estimated univariate AR coefficients for an AR model of order p, and e_{x_j} is the residual (prediction error) of the AR process. Similarly, a bivariate

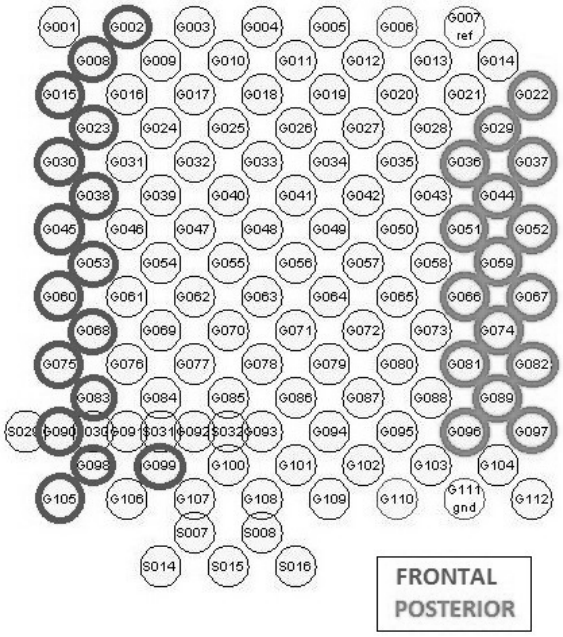


Figure 4: Channels for frontal (blue) and posterior (maroon) aggregate areas.

AR model is given by

$$x_j(t) = \sum_{i=1}^{p} a_{ix_jx_k} x_j(t-i) + \sum_{i=1}^{p} b_{ix_jx_k} x_k(t-i) + e_{x_jx_k}(t) \tag{7}$$

where $a_{ix_jx_k}$, $b_{ix_jx_k}$, and $e_{x_jx_k}$ are the corresponding bivariate AR coefficients and residuals for the bivariate AR model of order p.

By comparing the error variances of the univariate and bivariate AR model residuals, one can then deduce the causality as follows:

$$GC\,(X_j \to X_k) = \ln \frac{\sigma^2_{X_j/X_j}}{\sigma_{X_j/(X_j,X_k)}} \tag{8}$$

where $GC(X_j \to X_k)$ is the Granger causality from X_j to X_k, while $\sigma^2_{X_j/X_j}$ and $\sigma^2_{X_j/(X_j,X_k)}$ are the residual error variances from the univariate and bivariate AR models respectively. By definition, $GC = 0$ when the time series are independent, and $GC > 0$ otherwise. From equation (8) it can be seen that if the univariate AR model is a better fit, then GC will be close to zero. In contrast, if the bivariate model is a better fit, then GC will increase. If GC is high in one direction, this signifies a unidirectional causal relationship; if, however, GC is high in both directions, then a bidirectional causal relationship is inferred.

Causality has been widely applied in the field of neuroscience [22], with a number of applications relating to the study of consciousness. More specifically, it was found that the direction and strength of causal relationships displays distinct changes between wakefulness and lack thereof, induced by either physiological or pharmacological interventions and as captured by EEG activity, with fronto-posterior causal interactions identified as being of paramount importance [22, 23, 24]. The coupling between frontal and posterior areas appears to be an important mechanism for loss of consciousness, as a number of additional functional measures, such as transfer entropy (TE) [25], coherence and cross-dependence also indicate the breakdown of functional connectivity between frontal and posterior structures [26]. Coupling between anterior and posterior brain regions and propagation of EEG activity from fronto-central to posterior regions was also found during deep sleep [27], and breakdown of effective connectivity among specialized thalamocortical modules may underlie the fading of consciousness in deep sleep [28].

Even though GC is traditionally defined for pairwise relationships, there are some limitations that must be considered (see Bressler and Seth for a detailed review [29]). For example, it is not possible to distinguish between direct and indirect causal relationships when performing pairwise GC analysis. This is related to the issue of spurious causality that can appear between two processes when both are influenced by external sources that are not taken into account [30]. In order to infer a more precise structural causality, in theory one must include all sources of influence into the estimation. However, in practice this is unfeasible and even though multivariate versions of causality exist, these minimise the effects rather than eliminate them completely. Wang and colleagues have shown that both pairwise and blockwise approaches to GC estimation give consistent results [23]. As such, pairwise time-domain GC analysis still remains a valid methodology, particularly when a blockwise approach is taken (i.e. the two time series are aggregate activity from a number of individual time series or the GC is itself an aggregate of a number of pairwise GC estimates). An additional consideration relating to AR modelling is the issue of stationarity. Given that AR models assume a stationary process, nonstationary EEG signals must be analyzed in windows of short duration. It is widely accepted that EEG exhibits stationary properties for segments less than 20 seconds [31]. Therefore, common practice involves EEG analysis in short segments.

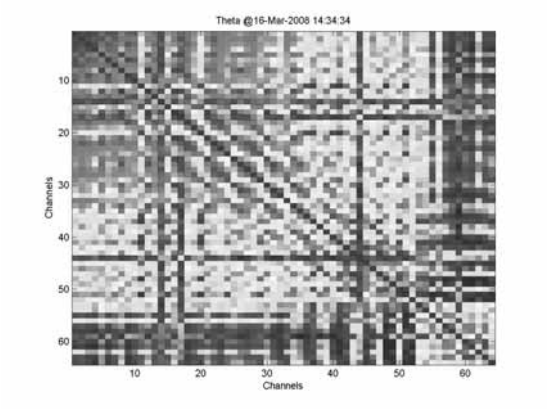
Taking into account the above considerations and related studies, the EEG was analysed in 4-s windows, overlapping by 2-s. For each 4-s segment, GC was estimated for aggregate activity from “frontal” and “posterior” areas, consisting of the channels indicated in Figure 4.

3 Results

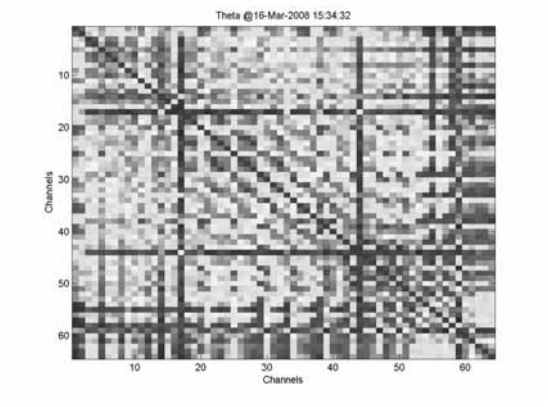
The main purpose of the investigations was to compare and contrast the ability of different methods, which have commonly been applied in neuroscience applications, to correctly indicate periods of consciousness in a CLIS patient corresponding to the "ground truth" known *a priori*.

3.1 Imaginary coherence

Figure 5 shows an example coherence matrix obtained for all combinations of the 64 ECoG channels. A visual inspection of the resulting coherence matrices videos



(a)



(b)

Figure 5: Coherence matrices of theta rhythms **(a)** during unconsciousness; **(b)** during consciousness.

indicates changes in coherence in distinct directions. To assess these direction changes further analysis was performed using artificial neural networks [32]. These changes across time are shown in Figure 6. On the other hand, analysis on the coherence value between each pair of channels across time revealed interesting variations in the higher frequencies in some channels (cf. Figure 6). The combination of the obtained results suggests an interesting change of state around 15:15-15:30 to 16:00-16:10. This corresponds to the time window during which the CLIS patient was consciously responding to the investigator's questions, as reported by the investigator.

3.2 Sample entropy

Multiscale sample entropy applied to analyze the cognition state in time domain. The higher multiscale sample

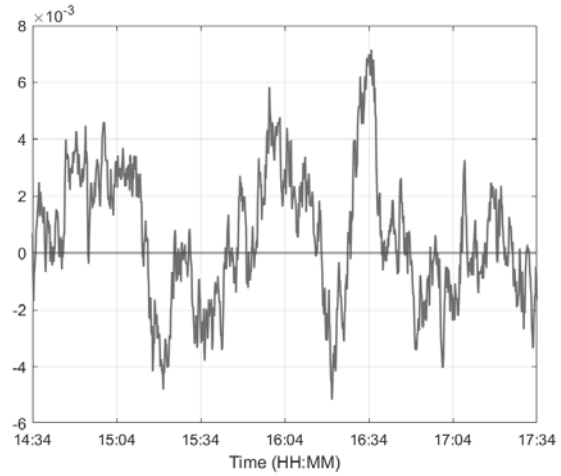


Figure 6: Motion direction changes in the theta bands.The x-axis represents the time and the y-axis represent the direction changes.

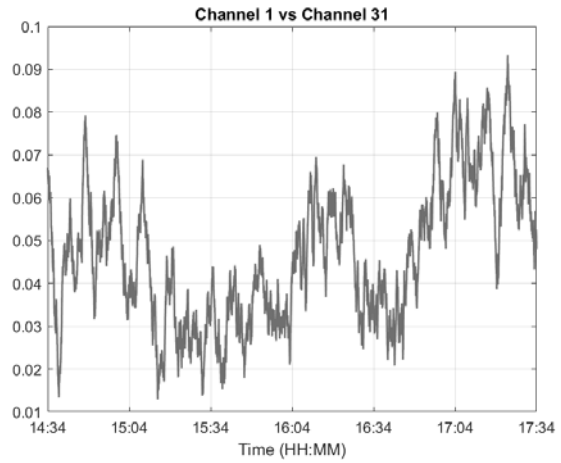


Figure 7: Coherence between one frontal and one parietal channel. There is a distinguished decrease of value between 15:10 and 16:10, and from 16:30 to around 16:45.

entropy means more indicated for consciousness. Figure 8 shows the result is the average from all usable channels, shows that the value of multiscale sample entropy relative high in the period between 15:24-16:14. This period coincides with the time window during the experiment in which the investigators receive feedback from CLIS patients.

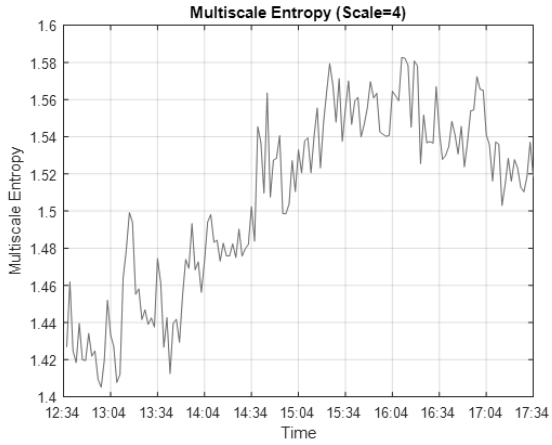


Figure 8: The result of Multiscale Entropy.

3.3 Granger causality

Figure 9 shows the estimated fronto-posterior GC in the evaluation of the state of the locked-in patient based on the recorded brain activity. Findings reported in the literature regarding the patterns of GC during wakefulness and unconsciousness suggest that a unidirectional increase in fronto-posterior GC is indicative of loss of consciousness. Based on these findings, the GC pattern observed during the period of 15:34 – 16:14 pm suggests that the patient is awake during this period. This period matches the period during which the CLIS patient was reported to have been communicating with the investigator. Similar patterns of reduced fronto-posterior GC can also be seen at various time points in Figure 9, which seems suggestive of additional transient periods of awareness.

4 Discussion

The methods presented here suggest a measurable change of the consciousness level occurring approximately between 15:15-16:45. This coincides with the period during which the CLIS patient was reported to have been communicating with the investigator via a brain-computer interface [5]. The probability of detecting changes in the consciousness level of the CLIS patient can, thus, be increased by combining the results of these methods. Such combination will reduce the uncertainty that is inherent to characterisation of the level of consciousness of CLIS patients, for which the "ground truth" is rarely available.

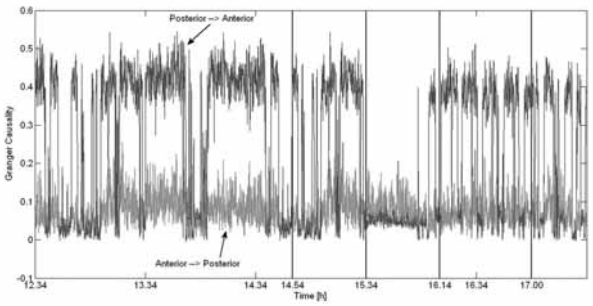


Figure 9: Fronto-posterior GC. A pattern of increased unidirectional fronto-posterior GC suggests unconsciousness, while a reduced bidirectional fronto-posterior GC suggests wakefulness. A long period of wakefulness, as well as several periods of transient wakefulness, can be identified.

5 Conclusion

In this paper, three approaches to detect conscious state in a complete locked-in patient were presented. These approaches will be combined in a hybrid approach with machine learning in a brain-computer interface system to ultimate establish a means of communication with CLIS patients. The combination of the different approaches should increase the probability of correctly detecting the patient's state. Based on the data set used in this paper, it has not only been shown even though different methods reporting different time slice each, but all around the same time slice the experimenter confirmed consciousness of the patient, the certainty of correctness of consciousness can be augmented by combing these three systems into one answer leading to the whole time slice reported by the experimenter judging the CLIS patient conscious.

Acknowledgement

Data was kindly provided by Prof. Dr. Dr. hc. mult. Niels Bierbaumer from the Institute for Medical Psychology and Behavioural Neurobiology, University of Tübingen.

References

[1] Vanhaudenhuyse A, Charland-Verville V, Thibaut A, Chatelle C, Tshibanda JFL, Maudoux A. Conscious While Being Considered in an Unresponsive Wakefulness Syndrome for 20 Years. *Front. Neurol.*. 2018; 9: 671. doi: 10.3389/fneur.2018.00671.

[2] Laureys S, Tononi G. *The neurology of consciousness. Cognitive neuroscience and neuropathology*. 2nd edn. Amsterdam, London: Academic; 2009. 440 p.

[3] Chaudhary U, Xia B, Silvoni S, Cohen LG, Birbaumer N. Brain–Computer Interface–Based Communication in the Completely Locked-In State. *PLoS Biol*. 2017; 15(1): page–page. doi: 10.1371/journal.pbio.1002593.

[4] Spueler M. Questioning the evidence for BCI-based communication in the complete locked-in state. *PLOS Biol*. 2019; doi: 10.1371/journal.pbio.2004750.

[5] Murguialday AR, Hill J, Bensch M, Martens S, Halder S, Nijboer F. Transition from the locked in to the completely locked-in state: A physiological analysis. *Clinical Neurophysiology*. 2011; 122(5): 925–933. doi: 10.1016/j.clinph.2010.08.019.

[6] Soekadar SR, Born J, Birbaumer N, Bensch M, Halder S, Murguialday AR. Fragmentation of slow wave sleep after onset of complete locked-in state. *Journal of clinical sleep medicine (JCSM)*. 2013; 9(9): 951–953. doi: 10.5664/jcsm.3002.

[7] Priestley, MB. *Spectral analysis and time series*. Probability and mathematical statistics. Academic Press; 1989. 890 p.

[8] Nolte G, Bai O, Wheaton L, Mari Z, Vorbach S, Hallett M. Identifying true brain interaction from EEG data using the imaginary part of coherency. *Clinical Neurophysiology*. 2004; 115(10): 2292–2307. doi: 10.1016/j.clinph.2004.04.029.

[9] Cohen, MX. *Analyzing neural time series data. Theory and practice*. Issues in clinical and cognitive neuropsychology. Cambridge, Massachusetts: The MIT Press; 2014. 600 p.

[10] Yeragani V. K., Pohi R., Mallavarapu M., Balon R. Approximate entropy of symptoms of mood: an effective technique to quantify regularity of mood. *Bipolar Disord*. 2003; 5: 279–286. doi: 10.1034/j.1399-5618.2003.00012.x .

[11] Diambra L, de Figueiredo JC, Malta CP. Epileptic activity recognition in EEG recording. *Phys A*. 1999; 273: 495–505. doi: 10.1016/S0378-4371(99)00368-4.

[12] Courtiol J, Perdikis D, Petkoski S, Muller V, Huys R, Sleimen-Malkoun R, Jirsa VK. The multiscale entropy: Guidelines for use and interpretation in brain signal analysis. *Journal of Neuroscience Methods*. 2016; 273: 175–190. doi: 10.1016/j.jneumeth.2016.09.004.

[13] Richman JS, Moorman JR. Physiological time-series analysis using approximate entropy and sample entropy. *Am J Physiol Heart Circ Physiol*. 2000; 278: H2039–H2049. doi: 10.1152/ajpheart.2000.278.6.H2039.

[14] Pincus SM, Goldberger AL. Physiological time-series analysis: what does regularity quantify?. *Am J Physiol Heart Circ Physiol*. 1994; 266: H1643–H1656. doi: 10.1152/ajpheart.1994.266.4.H1643.

[15] Costa M, Goldberger AL, Peng CK. Multiscale entropy analysis of biological signals. *Phys Rev E*. 2005; 71.

[16] Costa M, Goldberger AL, Peng CK. Multiscale entropy analysis of physiologic time series. *Phys Rev Lett*. 2002; 89.

[17] Wu SJ, Chen NT, Jen KK, Fan SZ. Analysis of the Level of Consciousness with Sample Entropy a comparative study with Bispectral Index. *European Journal of Anaesthesiology*. 2015; 32: 11.

[18] Wu SJ, Chen NT, Jen KK. Application of Improving Sample Entropy to Measure the Depth of Anesthesia. In *2014 International Conference on Advanced Manufacturing (ICAM)*; 2014; Chiayi, Taiwan.

[19] Wu SJ, Chen NT, Jen KK, Shieh JS, Fan SZ. The Physiological Signals EEG, ECG, and SpO2 are Applied to Analyze the Consciousness and Anesthesia Depth. *AMPT*. 2013; Taipei.

[20] Wiener, N. The theory of prediction. In: E. Beckenbach (eds). *Modern Mathematics for the Engineer*. New York, NY: McGraw-Hill; 1956. p 165–190.

[21] Granger CWJ. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*. 1969; 37: 424–438. doi: 10.1017/CBO9780511753978.002.

[22] Nicolaou N, Georgiou J. Spatial analytic phase difference of EEG activity during anesthetic-induced unconsciousness. *Clin Neurophysiol*. 2014; 125: 2122–2131. doi: 10.1016/j.clinph.2014.02.011.

[23] Nicolaou N, Georgiou J. Neural Network based classification of anesthesia / awareness using Granger Causality features. *Clinical EEG and Neuroscience*. 2014; 45(2): 77–88. doi: 10.1177/1550059413486271.

[24] Ku SW, Lee U, Noh GJ, Jun IG, Mashour GA. Preferential Inhibition of Frontal-to-Parietal Feedback Connectivity Is a Neurophysiologic Correlate of General Anesthesia in Surgical Patients. *PLOS ONE*. 2011; 6(10): e25155. doi: 10.1371/journal.pone.0025155.

[25] Imas OA, Ropella KM, Douglas Ward B, Wood JD, Hudetz AG. Volatile anesthetics disrupt frontal-posterior recurrent information transfer at gamma frequencies in rat. *Neuroscience Letters*. 2005; 387: 145–150. doi: 10.1016/j.neulet.2005.06.018.

[26] Lee U, Kim S, Noh GJ, Choi BM, Hwang E, Mashour GA. The directionality and functional organization of frontoparietal connectivity during consciousness and anesthesia in humans. *Conscious Cogn*. 2009; 18: 1069–1078. doi: 10.1016/j.concog.2009.04.004.

[27] Kamiński M, Blinowska K, Szelenberger W. Topographic analysis of coherence and propagation of EEG activity during sleep and wakefulness. *Electroen Clin Neuro*. 1997; 102: 216–227. doi: 10.1016/S0013-4694(96)95721-5.

[28] Massimini M, Ferrarelli F, Huber R, Esser SK, Singh H, Tononi G. Breakdown of Cortical Effective Connectivity During Sleep. *Science*. 2005; 309: 2228–2232. doi: 10.1126/science.1117256.

[29] Bressler SL, Seth AK. Wiener-Granger causality: a well established methodology. *Neuroimage*. 2011; 58: 323–329. doi: 10.1016/j.neuroimage.2010.02.059.

[30] Granger CWJ. Testing for causality: a personal viewpoint. *J Econ Dyn Control*. 1980; 2: 329–352. doi: 10.1016/0165-1889(80)90069-X.

[31] da Silva FL. EEG analysis: theory and practice. In: Niedermeyer E, da Silva FL, editors. *Electroencephalography: Basic Principles, Clinical Applications, and Related Fields*. Philadelphia, PA: LWW; 2005. p 1199–1232.

[32] Adama VS, , Blankenburg A., Ernst C., Kummer R., Murugaboopathy S., Bogdan M. Motion Detection in Videos of Coherence Matrices in order to detect Consciousness States in CLIS-patients – an Approach. In *10th EUROSIM Congress on Modelling and Simulation*; 2019 July; Logrono, Spain.

Genetic Algorithms in the Domain of Personalized Nutrition

Petri Heinonen[1*], Esko K. Juuso[2]

[1]Nutri-Flow Oy, Villiperäntie 5, FI-90410 Oulu, Finland,; *petri.heinonen@nutri-flow.fi*

[2]Control Engineering, Environmental and Chemical Engineering,
Faculty of Technology, P.O. BOX 4300, FI-90014 University of Oulu, Finland

SNE 32(1), 2022, 47-54, DOI: 10.11128/sne.32.tn.10597
Received: 2020-11-10 (Selected EUROSIM 2019 Postconference Publication); Revised: 2022-02-15; Accepted: 2022-02-25
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna, ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. Lifestyle related public health problems are common around the world. Personal nutrient guidance is a tool for promoting healthier lifestyles. Most of the applications available on the market are based on energy only, and a reliable individual assessment and guidance is given by licensed nutritionists. Nutri-Flow has a novel approach into personalized nutrition guidance with Fuzzy Expert System (FES) enhanced with Genetic Algorithms (GA) optimization. While FES assesses the foods and beverages added into a search space, GA is used to find the level of intake for them. The optimization problem is to minimize the distance to ideal nutrient intake levels, and to keep the level of change in a feasible level and take into account other nutrition variables. In this study, the suitability of GA was assessed. Also, the performance the GA was evaluated and evolved. The objective function is presented, and the overall results were evaluated numerically if the system was feasible in the domain of nutrition. The nutritional aspect is not in the scope of this study.

Introduction

According to FinHealth 2017 study [1], obesity is still one of the key public health problems among all age groups in Finland. It is also stated in the study that healthy diet is one solution to prevent many key public health problems. It was found that there is a need to promote healthy lifestyle.

There are available national nutrition and food recommendations. Food pyramids and plate models are given as a general guidance towards a balanced diet. General guidelines and recommended intake levels for each nutrient are introduced nationally, too [2]. In this study, all nutritional values are achieved from Finnish nutrition and food recommendations, which are based on Nordic recommendations [2]. To find the nutrient composition in a diet, there is a need also for a food composition database. National food composition databases are used for this purpose, such as Fineli [3] in Finland, Livsmedeldatabasen [4] in Sweden, USDA [5] in the USA, etc. These databases are collections of foods, beverages, and recipes with their averaged nutrient composition.

A lot of data is available for healthy eating habits, therefore, experts are needed to put all together for assessment and guidance for individuals. In the Internet, there are available thousands services and applications to monitor and balancing the diet. Those are too simplified approaches, since they take into account only the energy in a very complex problem domain. If only the energy is taken into account, the rest of the variables, protective nutrient levels, will deteriorate and might lead even to malnutrition. Typical diets are shown in Figure 1:

- **Diet 1**: The sufficient intake of protective nutrients is accompanied by an excess intake of energy;
- **Diet 2**: The insufficient intake of protective nutrients is combined with the excess intake of energy;
- **Diet 3**: Insufficient intake of protective nutrients with sufficient intake of energy;
- **Diet 4**: Nutrient dense diet with ideal proportion of intake of protective nutrients and energy. [6]

The eating habits have widely moved to excess energy (Diet 2) and the energy-only approach has neglected the importance of the protective nutrients (Diet 3). Diet 4 is needed to balance these problems.

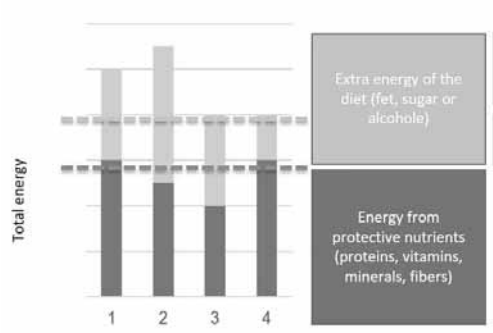


Figure 1: Differences between typical diets [6].

For individual nutritional guidance, there are four main steps to take into account when developing an algorithm for the problem domain. First, there should be a food record or a meal diary which is used to assess nutrient intake levels. This step includes imprecision and uncertainty due to human error while filling in a record with kitchen units. Second step is to assess a personal nutrition recommendation based on the current national recommendation. Third step is to assess the needed level of change to balance the diet. Fourth step is to generate the guidance as foods and beverages with their portion sizes.

In the previous research [7, 8], a Fuzzy Expert System was developed to handle the imprecision and uncertainty present in the system. It was also discussed in the previous publications that an optimization algorithm is needed to assess the portion sizes. GA was selected initially for testing. GA was selected due to the complexity of the problem. GA is widely used with complex real-world problems [9, 10, 11].

This research combines computational intelligence (Section 1) for personalized nutrition guidance with focus on genetic algorithms (Section 2). The operation is demonstrated in a test case (Section 3) and the results are presented in Section 4. Conclusions and future research are discussed in Section 5.

1 Background

1.1 Data Acquisition

Nutrition experts have generated test cases with typical personal data and a meal diary example for the study. In this study, a test case called Sum Mikko, is used. The test case is 41 years old, thus national recommendations for males of ages between 31 and 60 are used to calculate the personal recommendation. The personal data are acquired from Nutri-Flow directly. The meal record of one day represents one week average including seven meals with 16 different foodstuffs. Nutrient intake levels are calculated based on the food record and Fineli Food Composition Database [3]. No personal data from Nutri-Flow database is used in this study.

1.2 Fuzzy Expert System

Fuzzy Expert System (FES) has the key role to handle the imprecision and assesses the personalized dietary guidance. FES is adapted to personal recommendation, by applying the personal nutrition recommendation values. Nutrient intake levels are converted into the fuzzy domain with membership functions. Three linguistic fuzzy variables are used to define the intake levels: too little, ideal, too much. Personal nutrient recommendation values are used to tune the membership functions. Most of the 30 nutrients to take into account have three recommendation values: Lower Intake Level (LI), Recommended In-take Level (RI), and Upper Intake Level (UI). The fuzzy variables are defined with values A,B and C (Figure 2) are tuned with LI, RI, and UI, respectively.

The linguistic variables are applied to form the knowledge base for the fuzzy inference machine. Knowledge of nutrition experts was acquired and coded into the knowledge base as rules, for example

IF vitamin C is too little AND fiber is too little
THEN fruits and berries group IS add

The rules are mapping the guidance directly into foodstuff level from the nutrient level. The inference is Mamdani type, therefore, the output is also fuzzy with similar sets of linguistic fuzzy variables. The used variables are “reduce”, “no action” and “add”. The desired output for each food is a crisp number representing the direction and importance of recommended change.

1.3 Optimization problem

The output of FES is a list of foods and beverages with the defuzzified value expressing the direction and importance of change. The objective is to find the best way to balance a diet. To achieve this objective, portion sizes of the foods and beverages for the guidance should be assessed. As discussed in [7], the problem domain is complex. There are 30 nutrient variables and energy to be taken in account. Also, foodstuffs and recipes present a complex composition of sources of each nutrient.

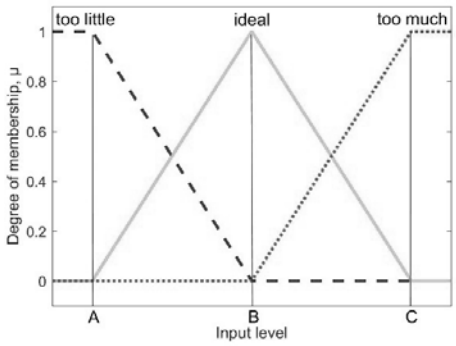


Figure 2: Guidance mapping represents recommendations as fuzzy numbers.

A diet can be altered in various ways to reach similar composition in the nutrient domain. Therefore, there can be solutions which have the same fitness value or the fitness values are very close to each other. The system must be feasible in the field of the nutrition domain, which brings more constrains to be taken into account. E.g. the difference between recommended diet and initial diet cannot be too drastic. The diet should be altered with smaller steps if it was originally far away from the recommendations. Also, other nutrition related variables should be taken into account. The main objective of the optimization algorithm is to minimize distance from ideal nutritional composition, keep the recommended step in desired magnitude, and take into account the other tuning variables.

As discussed in [7], traditional optimization methods are not suitable due to the nature of the problem. The initial approach was to study if GA is a suitable solution.

2 Genetic Algorithms

Genetic Algorithms belong to the group of Evolutionary Algorithms. The roots and ideology of GA are based on Darwin's theory of natural selection. The theory was introduced in 1975, and the terminology was closely adapted from natural genetics. [12]. A single solution is called a chromosome, which has locus bind variables, genes. When the theory was published, a binary approach for coding the results were applied. It was discussed that binary coding is not always the best way with real world problems where the high precision makes the chromosomes long and the algorithm gets inefficient [13].

A population is a set of chromosomes, which is set in a competitive environment to find the global optimum. Each chromosome is evaluated through an objective function. The calculated fitness value is used to rank the solutions. The population is evolved by applying GA operators in every iteration round. The optimization starts with generating the population, where the selection and crossover operators are applied to combine genetic material for the new population. With a mutation operator, random variations are introduced into the population. This helps to prevent stopping on local minimas. Crossover and mutation operations might lead to the loss of the best solutions, the elitism operator can be used to copy the best solutions into the new population directly [12, 13].

There has been a lot of development with the GA since the theory was published. In the literature, there is a large number of problem specific approaches for the operators [14, 15]. The variables for the operators have been traditionally found by trial and error. There are also studies which provide tools for finding the best configuration [15].

2.1 Configuration

The configuration of GA has a strong effect on convergence and on finding a global optimum. Convergence time and feasible solutions are key factors when applying GA in an online service.

Population Population is a set of possible solutions. The size of the population is an important variable. Too small a population might lead to insufficient divergence between the solutions and the optimum is not reached. Too large a population might lead to slow convergence, since the evolving needs more iterations and the objective function must evaluate more solutions. It is discussed in [16] that, the longer the chromosome is, the larger the number of individuals in population is needed.

The initial population is usually generated randomly within the given constraints. There are also statistical methods available. If the population is generated randomly, it is recommended to run the GA with different initial populations, since the initial guess might not always lead to the global optimum [17].

The size of the population can be static or vary between the iterations [18].

Coding The results can be coded into chromosomes in binary or real-valued domain. The size of search space and accuracy level of the results should be used when selecting the coding. Real-value coding is used

widely in real world optimization when the size of search space is big and higher accuracy is required on the results. With real-value coding there is no need for result mapping which reduces need for computational resources. However, it has been discussed that real-value coding has problems to yield good results always.

Crossover Crossover is the core operator to evolve population towards better solutions. The good genetic material is distributed between generations. The good genetic material from the population is found with a selection method. The selection is done usually by the roulette wheel method or by the tournament selection method.

The tournament selection method selects randomly chromosomes from the population and the fitness values are compared. The better chromosome is selected into a mating pool for the crossover method. The tournament selection method has a configuration parameter k, which defines the number of selected chromosomes from the population. Typical value for k is two.

The crossover operator is applied to mating population to form offspring. Crossover probability, P_c, is a design parameter for the crossover operator. The parameter is used to determine if current mating population chromosomes are combined with the crossover operator or directly to the offspring population.

The selection of the crossover operator depends on the coding and the problem. For real-value coded chromosomes, non-uniform and uniform crossover operators are applied. The operation of non-uniform crossover operators depends on the age of the population, and the uniform crossover operators operate in the same way in every generation. Arithmetic crossover operator combines two parent chromosomes to two offspring chromosomes as shown in

$$y_i^1 = \alpha_i x_i^1 + (1-\alpha_i^1)x_i^2, \quad (1)$$
$$y_i^2 = \alpha_i x_i^2 + (1-\alpha_i^1)x_i^1, \quad (2)$$

where α_i are uniform random numbers. In the non-uniform crossover, the parameter α_i can vary between the iterations, and in uniform crossover the value is constant. There are several studies presenting different approaches and new development on crossover operators.

Mutation Optimization can stop at local minimums if the diversity of the population is low. With random variation in the population, new solutions are found and the diversity will grow. Mutation operation is applied to prevent from stopping at local minimums. With correct design parameters, the mutation operator can be used efficiently. Mutation probability P_m controls how strong effect the mutation brings into the population. A too low value does have only a very little or no effect and too big value could lead to the loss of good genetic material and slowing down the convergence rate. Mutation operator with real-coded chromosomes can be uniform or non-uniform.

Elitism GA operators alter the genetic data of a population towards to better solutions. The best solution is possible to be lost during the iterations. The elitism operator is used to prevent this to happen. The elitism saves the one or several best chromosomes and transfers them directly into new population. If the population size is static, usually the best chromosome replaces the worst chromosome in the new population.

3 Genetic Algorithms in Personalized Nutrition Guidance

The main objective is to find a feasible solution to balance a diet. Initial values for the test case are acquired from Nutri-Flow software. The test environment is developed in Matlab and Genetic Algorithms solver is applied.

3.1 Configuration

Configuration used in [7] is applied in this study, too. Configuration parameters are presented in Table 1. Real-value coding is selected due to search space. Intake levels vary between the foodstuffs in a great level; e.g. cinnamon one teaspoon vs. 1000 g water.

Parameter	Value
Population size	100
P_c	0.8
k	2
P_m	0.01
Elite individuals	5% of the population
Maximum iterations	500

Table 1: GA configuration parameters.

Name	Lower limit	Upper limit
Bread, graham	0	0
Potato, peeled, cooked	**180**	**540**
Minced meat, beef 17% brown sauce, no fat	0	0
Salad, buffet, no dressing	**50**	**150**
Rye bread	0	0
Macaroni casserole, beef-pork, milk 1.5% fat	0	0
Bread, wheat	0	0
Coffee drink, brewed	600	770
Margarine 40%, industrial average	0	0
Cheese, hard cheese, fat 24-27%	**0**	**52**
Margarine 60%	0	0
Fat-free milk, vitamin-D 1 μg	**0**	**510**
Water	**170**	**3000**
Ketchup	0	0
Cider, sweet, 4,7 vol% alcohol	**0**	**1000**
Cookie, oatmeal, industrial	0	0
Bell pepper	**0**	**200**
Orange, peeled	**0**	**800**
Tangerine, peeled	**0**	**450**
Kiwi fruit, peeled	**0**	**300**
Apple, peeled	**0**	**800**
Wok vegetables	**0**	**500**

Table 2: Search space in the test case.

3.2 Initial state

Sum Mikko has a meal diary for one day filled in NutriFlow database. The state of the diet can be presented as membership degrees for each 30 nutrients. The search space has 22 foods, and nine of them should be added and three reduced according to FES output.

3.3 Search space

The search space is generated following output of FES and meal diary of test case Sum Mikko. The search space is presented in Table 2. Foodstuffs to add or reduce are shown in bold. The direction of change is inherited from FES output. The constrains for the intake levels are defined for foods to add with original input M_i, and upper intake recommendation M_u as $[M_i, M_u]$, and foods to reduce with zero and M_i as $[0, M_i]$. Foods with no action needed, constrains are exactly at M_i, $[M_i, M_i]$.

3.4 Objective function

The objective is to balance a diet. In this study, the fitness is evaluated applying membership grades. The distance to ideal state is minimized when membership grades for too little μ_l and too much μ_l are minimized. Calculation of fitness value, F_μ, for nutrients is presented in

$$F_\mu = \sum_{1}^{n} (b_{l,n}\mu_{l,n} + b_{u,n}\mu_{u,n}) \tag{3}$$

where n is nutrient Id, $b_{l,n}$ is a weight factor for too low intake level for nutrient n, $\mu_{u,n}$ is a membership grade for too little for nutrient n, $b_{u,n}$ is a weight factor for too much intake level for nutrient n, and $\mu_{u,n}$ is a membership grade for too little for nutrient n.

According to experts in nutrition, too drastic changes are difficult to follow. The balancing should be done in smaller steps. Minimizing fitness value F_d for food intakes take into account the step size for the guidance. Calculation of F_d is presented in

$$F_d = \sum_{1}^{n} |d_m|, \tag{4}$$

where m is Id for a foodstuff and d_m is distance from the initial diet. Other nutrition related factors, such as

the amount of vegetables, or even the carbon footprint if available, should be taken in account when assessing guidance. F_o is reserved for other variables to be calculated in the fitness value as presented in

$$F_o = \sum_{1}^{n} |k_p|, \tag{5}$$

The objective function minimizes all the fitness values F_μ, F_d, and F_o as presented in

$$MIN(aF_\mu + bF_d + bF_o), \tag{6}$$

where a, b and c are tuning factors for each component. Tuning factors are applied to enhance the importance of the fitness values. The main factor is naturally the nutrient state, and the two other fitness values are used to guide the optimization towards to desired and feasible result.

3.5 Matlab model

This study applies Matlab Optimization Toolbox and Genetic Algorithms solver for testing the system. All the data used in the optimization are first acquired from Nutri-Flow software into Matlab workspace. The results are saved numerically and graphically for further evaluation. Custom objective function is applied according to equations presented in Section 3.4.

4 Results

Convergence of calculations and effects on foodstuff level are used for assessing the operation.

4.1 GA performance

Convergence was recorded on all the test rounds. However, the best result was not reached on all rounds within 500 iteration rounds. Most of the test rounds were stopped by Matlab algorithm after no improvement during last 50 iterations. The stopping point was recorded between 100 and 200 iterations except two tests where the maximum number of iterations was the stopping criteria.

The population size was kept at 100 which was sufficient for the current test case which had 22 genes in each chromosome. It is recognized that the chromosome size varies between the test cases and between the evaluation periods. The population size needs further testing with a larger test case set.

The convergence speed is the fastest at the beginning of the iterations and drops fast as shown in Figure 3.

GA design variables have an effect on the convergence speed and finding the global optimum. For further testing, different values for design variables should be tested.

Id	Initial	GA	Difference
1	0.36	0.53	0.17
2	0.50	0.52	0.02
3	0.35	0.47	0.12
4	0.97	0.61	-0.36
5	0.32	0.39	0.08
6	N/A	N/A	N/A
7	0.00	0.83	0.83
8	0.00	0.69	0.69
9	0.00	0.78	0.78
10	0.92	0.98	0.05
11	0.93	0.84	-0.09
12	0.77	0.87	0.10
13	1.00	0.98	-0.02
14	0.31	0.63	0.32
15	0.68	1.00	0.32
16	1.00	0.94	-0.06
17	0.93	0.93	0.00
18	1.00	0.99	-0.01
19	1.00	1.00	0.00
20	0.64	0.89	0.24
21	N/A	N/A	N/A
22	0.57	0.93	0.37
23	0.81	0.75	-0.06
24	0.54	0.68	0.14
25	0.86	0.92	0.07
26	0.72	0.80	0.08
27	0.44	0.58	0.14
28	1.00	1.00	0.00
29	0.73	0.90	0.16
30	0.99	0.99	0.00

Table 3: Membership grade for ideal input for initial and GA recommendation with difference.

4.2 GA output

The overall assessment is done by evaluating the output of GA on the nutrient level and on the foodstuff level. Only a numerical evaluation is carried in this study. The nutrient level analysis is done by comparing the membership grades for each nutrient. Table 3 presents the results of one GA optimization run. The status of most nutrients has improved, but one value has dropped significantly and the status of five nutrients has

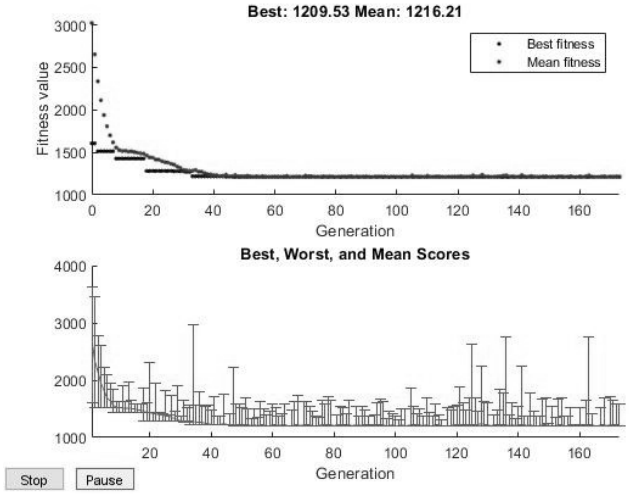


Figure 3: GA optimization with test case Sum Mikko.

moved slightly away from the original ideal status. Two values, added sugar and salt intake are not taken in account in this study directly.

Default feasible range for intake levels for nutrients is set at $[0.5, 1]$ for ideal level. Conclusion of nutrient level evaluation is that the result is at the feasible level. Actual guidance is given as foods and beverages as presented in Table 4. All the recommended values are in the feasible level and the direction of the change is correct. Assessment of level of change should be done as portions. A weekly food plan should be able to generate from the recommendation.

5 Conclusions and Discussion

The test was carried out for one test case. The results are promising, and the nutritional status was improved with the GA optimization. It is important to keep the nutrient intake levels in the feasible range. More studies are needed with a larger test set.

References

[1] Koponen P, Borodulin K, Lundqvist A, Sääksjärvi K, Koskinen S, editors. *Health, functional capacity and welfare in Finland – FinHealth 2017 study, Report 4/2018 (abstract in English)*. 8th ed. Helsinki: THL; 2018. 247 p. http://urn.fi/URN:ISBN:978-952-343-105-8.

[2] Nordic Council of Ministers. *Nordic Nutrition Recommendations 2012*. 5th ed. Copenhagen: Nordisk Ministerråd; 2014. 627 p. http://dx.doi.org/10.6027/Nord2014-002

[3] National Institute for Health and Welfare. *Nutrition Unit: Fineli. Finnish food composition data-base. Release 19*. Helsinki: THL; 2018. www.fineli.fi

[4] Livsmedelsverket. *The National Food Agency food database. Version 2017-12-15*. http://www.slv.se/SokNaringsinnehall/

[5] US Department of Agriculture, Agricultural Research Service, Nutrient Data Laboratory. *USDA National Nutrient Database for Standard Reference. Legacy. Version Current*. USDA; April 2018. https://www.ars.usda.gov/nea/bhnrc/ndl

[6] Nutrition experts team. *Nutrient Density*. Oulu: Nutri-flow; 2018.

[7] Heinonen P, Mannelin M, Iskala H, Sorsa A, Juuso E. Development of a Fuzzy Expert System for a Nutri-tional Guidance Application. In Carvalho P, Dubois D, Kaymak K, Sousa JCM, editors. *Proceedings of 2009 IFSA World Congress / 2009 EUSFLAT Conference*; 2009; Lisboa, Portugal. 1685-1690. ISBN: 978-989-95079-6-8.

[8] Heinonen P, Juuso EK. Development of a Genetic Algorithms Optimization Algorithm for a Nutritional Guidance Application. In Juuso E, Dahlquist E, Leiviskä K, editors. *The 9th EUROSIM Congress on Modelling and Simulation, EUROSIM 2016, The 57th SIMS Conference on Simulation and Modelling SIMS 2016, number 142*; 2018; Oulu, Finland. Linköping: Linköping University Electronic Press. 755–761. doi: 10.3384/ecp1714255.

Name	Recommendation	Level of change
Bread, graham	100	0
Potato, peeled, cooked	**287**	**107**
Minced meat, beef 17% brown sauce, no fat	180	0
Salad, buffet, no dressing	**76**	**26**
Rye bread	60	0
Macaroni casserole, beef-pork, milk 1,5% fat	350	0
Bread, wheat	60	0
Coffee drink, brewed	770	0
Margarine 40%, industrial average	62	0
Cheese, hard cheese, fat 24-27%	**6**	**-46**
Margarine 60%	12	0
Fat-free milk, vitamin-D 1 μg	**203**	**307**
Water	**718**	**548**
Ketchup	30	0
Cider, sweet, 4,7 vol% alcohol	**114**	**-886**
Cookie, oatmeal, industrial	25	0
Bell pepper	**41**	**41**
Orange, peeled	**151**	**151**
Tangerine, peeled	**207**	**207**
Kiwi fruit, peeled	**87**	**87**
Apple, peeled	**269**	**269**
Wok vegetables	**32**	**32**

Table 4: Personalized nutritional guidance in foodstuff level.

[9] Al-Obaidi MA, Li J-P, Kara-Zaïtri C, Mujtaba IM. Optimisation of reverse osmosis based wastewater treatment system for the removal of chlorophenol using genetic algorithms. *Chemical Engineering Journal*. 2017; 316: 91-100. doi: 10.1016/j.cej.2016.12.096.

[10] Abdelaziz M. Distribution network reconfiguration using an geletic algorithm with varying population size. *Electric Power Systems Research*. 2017; 142: 9-11. doi: 10.1016/j.epsr.2016.08.026.

[11] Francescomarino CD, Dumas M, Federici M, Ghidini C, Maggi FM, Rizzi W, Simonetto L. Genetic algorithms for hyperparameter optimization in predictive business process monitoring. *Information Systems*. 2018; 74: 67-83. doi: 10.1016/j.is.2018.01.003.

[12] Holland JH. *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. Ann Arbor: University of Michigan Press; 1975.

[13] Goldberg DE. *Genetic Algorithms in Search, Optimization and Machine Learning*. Boston: Addison-Wesley; 1989.

[14] Singh RK, Panchal VK, Singh BK. A review on Genetic Algorithm and Its Applications. *2018 Second International Conference on Green Computing and Internet of Things (ICGCIoT)*; 2018. 376-380. doi: 10.1109/ICGCIoT.2018.8753030.

[15] Yuan B, Gallagher M. A hybrid approach to parameter tuning in genetic algorithms. *2005 IEEE Congress on Evolutionary Computation*; 2005. 1096-1103. doi: 10.1109/CEC.2005.1554813.

[16] Goldberg DE. Sizing Populations for Serial and Parallel Genetic Algorithms. *3rd International Conference on Genetic Algorithms*; 1989. 70-79.

[17] Diaz-Gomez P, Hougen DF. Initial Population for Genetic Algorithms: A Metric Approach. *2007 International Conference on Genetic and Evolutionary Methods, GEM 2007*; 2007. 8 p.

[18] Davis L. *Handbook of Genetic Algorithms*. New York: Van Nostrand Reinhold; 1991. 385 p.

SNE Simulation News

EUROSIM Data and Quick Info

VESS – Virtual EUROSIM Seminar
Virtual Simulation Presentations, since June 2020 www.eurosim2023.eu

MATHMOD Vienna 2022
July 27-29, 2022, Vienna, Austria www.mathmod.at
ASIM 2022 - 26.Symposium Simulation Technique
July 25-27, 2022, Vienna, Austria www.asim-gi.org/asim2022

EUROSIM CONGRESS 2023
June 28-30, 2023, Amsterdam, The Netherlands www.eurosim2023.eu

Contents

Simulation Notes Europe SNE is the official membership journal of **EUROSIM** and distributed / available to members of the **EUROSIM Societies** as part of the membership benefits.

If you have any information, announcement, etc. you want to see published, please contact a member of the editorial board in your country or the editorial office. For scientific publications, please contact the EiC.

This *EUROSIM Data & Quick Info* compiles data from **EUROSIM** societies and groups: addresses, weblinks, and officers of societies with function and email, to be published regularly in **SNE** issues. This information is also published at **EUROSIM**'s website *www.eurosim.info*.

SNE Editorial Office /ARGESIM

→ *www.sne-journal.org, www.eurosim.info*
office@sne-journal.org, eic@sne-journal.org
✉ SNE Editorial Office
Johannes Tanzler (Layout, Organisation)
Irmgard Husinsky (Web, Electronic Publishing)
Felix Breitenecker EiC (Organisation, Authors)
ARGESIM/Math. Modelling & Simulation Group,
Inst. of Analysis and Scientific Computing, TU Wien
Wiedner Hauptstrasse 8-10, 1040 Vienna, Austria

EUROSIM Federation of European Simulation Societies

General Information. **EUROSIM**, the Federation of European Simulation Societies, was set up in 1989. The purpose of **EUROSIM** is to provide a European forum for simulation societies and groups to promote modelling and simulation in industry, research, and development – by publication and conferences. → *www.eurosim.info*

Member Societies. **EUROSIM** members may be national simulation societies and regional or international societies and groups dealing with modelling and simulation. At present **EUROSIM** has *Full Members* and *Observer Members* (*), and *Member Candidates* (**).

ASIM	Arbeitsgemeinschaft Simulation *Austria, Germany, Switzerland*
CEA-SMSG	Spanish Modelling and Simulation Group; *Spain*
CSSS	Czech and Slovak Simulation Society *Czech Republic, Slovak Republic*
DBSS	Dutch Benelux Simulation Society *Belgium, Netherlands*
KA-SIM	Kosovo Simulation Society, *Kosovo*
LIOPHANT	LIOPHANT Simulation Club; *Italy & International*
LSS	Latvian Simulation Society; *Latvia*
PSCS	Polish Society for Computer Simulation; *Poland*
NSSM	Russian National Simulation Society *Russian Federation*
SIMS	Simulation Society of Scandinavia *Denmark, Finland, Norway, Sweden*
SLOSIM	Slovenian Simulation Society; *Slovenia*
UKSIM	United Kingdom Simulation Society *UK, Ireland*
ALBSIM	Albanian Simulation Society*; *Albania*
ROMSIM	Romanian Society for Modelling and Simulation*; *Romania*
Societies in Re-Organisation:	
CROSSIM	*Croatian Society f. Simulation Modeling;* Croatia
FRANCOSIM	*Société Francophone de Simulation* *Belgium, France*
HSS	*Hungarian Simulation Society;* Hungary
ISCS	*Italian Society for Computer Simulation, Italy*

EUROSIM Board / Officers. **EUROSIM** is governed by a board consisting of one representative of each member society, and president, past president, and **SNE** representative. The President is nominated by the society organising the next **EUROSIM** Congress. Secretary, and Treasurer are elected out of members of the board.

President	M. Mujica Mota (DBSS), *m.mujica.mota@hva.nl*
Past President	Emilio Jiménez (CAE-SMSG), *emilio.jimenez@unirioja.es*
Secretary	Niki Popper, *niki.popper@dwh.at*
Treasurer	Felix Breitenecker (ASIM) *felix.breitenecker@tuwien.ac.at*
Webmaster	Irmgard Husinsky, *irmgard.husinsky@tuwien.ac.at*
SNE Editor	F. Breitenecker, *eic@sne-journal.org*

SNE – Simulation Notes Europe. **SNE** is **EUROSIM**'s scientific journal with peer reviewed contributions as well as a membership journal for **EUROSIM** with information from the societies. **EUROSIM** societies distribute **SNE** (electronic or printed) to their members as official membership journal. **SNE** Publishers are **EUROSIM**, **ARGESIM** and **ASIM**.

SNE Editor-in-Chief	Felix Breitenecker *eic@sne-journal.org*

→ *www.sne-journal.org*, *office@sne-journal.org*

EUROSIM Congress and Conferences.

Each year a major **EUROSIM** event takes place, the **EUROSIM CONGRESS** organised by a member society, **SIMS EUROSIM Conference**, and **MATHMOD Vienna Conference (ASIM)**.

EUROSIM Congress 2019, the *10th EUROSIM Congress*, was organised by **CEA-SMSG**, the Spanish Simulation Society, in La Rioja, Logroño, Spain, July 1-5, 2019;

Due to Covid-19 virus some **EUROSIM** events had to be cancelled in 2020 or 2021, resp. To bridge this gap, **EUROSIM** is organising the series **VESS - Virtual EUROSIM Simulation Seminar** – seminars by simulation professionalists (2 hours via web), in preparation for upcoming **EUROSIM** events. → ***www.eurosim2023.eu***

Next main event is **MATHMOD Vienna**. This triennial **EUROSIM Conference** is mainly organized by **ASIM**, the German simulation society, and **ARGESIM,** with main co-sponsor **IFAC**.

MATHMOD 2022, the *10th MATHMOD Vienna Conference on Mathematical Modelling* will take place in Vienna, July 27-29, 2022. → ***www.mathmod.at***

EUROSIM Congress 2023, the *11th EUROSIM Congress*, will be organised by **DBSS**, the Dutch Benelux simulation society, in Amsterdam, June 28-30, 2023.

→ ***www.eurosim2023.eu***

Furthermore, **EUROSIM Societies** organize also local conferences, and **EUROSIM** co-operates with the organizers of the **I3M Conference Series.**

→ ***www.liophant.org/conferences/***

EUROSIM Member Societies

ASIM
German Simulation Society
Arbeitsgemeinschaft Simulation

ASIM (Arbeitsgemeinschaft Simulation) is the association for simulation in the German speaking area, servicing mainly Germany, Switzerland and Austria. **ASIM** was founded in 1981 and has now about 400 individual members (including associated), and 90 institutional or industrial members.

→ *www.asim-gi.org* with members' area

info@asim-gi.org, admin@asim-gi.org

✉ ASIM – Inst. of Analysis and Scientific Computing
Vienna University of Technology (TU Wien)
Wiedner Hauptstraße 8-10, 1040 Vienna, Austria

ASIM Officers	
President	Felix Breitenecker *felix.breitenecker@tuwien.ac.at*
Vice presidents	Sigrid Wenzel, *s.wenzel@uni-kassel.de*
	T. Pawletta, *thorsten.pawletta@hs-wismar.de*
	A. Körner, *andreas.koerner@tuwien.ac.at*
Secretary	Ch. Deatcu, *christina.deatcu@hs-wismar.de*
	I. Husinsky, *Irmgard.husinsky@tuwien.ac.at*
Membership Affairs	S. Wenzel, *s.wenzel@uni-kassel.de*
	Ch. Deatcu, *christina.deatcu@hs-wismar.de*
	F. Breitenecker, *felix.breitenecker@tuwien.ac.at*
Repr. EUROSIM	F. Breitenecker, *felix.breitenecker@tuwien.ac.at*
	A. Körner, *andreas.koerner@tuwien.ac.at*
Internat. Affairs – GI Contact	O. Rose, *Oliver.Rose@tu-dresden.de*
	N. Popper, *niki.popper@dwh.at*
Editorial Board SNE	T. Pawletta, *thorsten.pawletta@hs-wismar.de*
	Ch. Deatcu, *christina.deatcu@hs-wismar.de*
Web EUROSIM	I. Husinsky, *Irmgard.husinsky@tuwien.ac.at*

Last data update *April 2020*

ASIM is organising / co-organising the following international conferences:

- **ASIM SPL** Int. Conference 'Simulation in Production and Logistics' – biannual
- **ASIM SST** 'Symposium Simulation Technique' – biannual
  ASIM SST 2022: TU Vienna, July 25-27, 2022
  www.asim-gi.org/asim2022
- **MATHMOD** Int. Vienna Conference on Mathmatical Modelling – triennial

Furthermore, **ASIM** is co-sponsor of *WSC* - Winter Simulation Conference, of SCS conferences *SpringSim* and *SummerSim*, and of *I3M* and *Simutech* conference series.

ASIM Working Committees	
GMMS	Methods in Modelling and Simulation Th. Pawletta, *thorsten.pawletta@hs-wismar.de*
SUG	Simulation in Environmental Systems Jochen Wittmann, *wittmann@informatik.uni-hamburg.de*
STS	Simulation of Technical Systems Walter Commerell, *commerell@hs-ulm.de*
SPL	Simulation in Production and Logistics Sigrid Wenzel, *s.wenzel@uni-kassel.de*
EDU	Simulation in Education/Education in Simulation A. Körner, *andreas.koerner@tuwien.ac.at*
BIG DATA	Working Group Data-driven Simulation in Life Sciences; *niki.popper@dwh.at*
WORKING GROUPS	Simulation in Business Administration, in Traffic Systems, for Standardisation, etc.

CEA-SMSG – Spanish Modelling and Simulation Group

CEA is the Spanish Society on Automation and Control and it is the national member of IFAC (International Federation of Automatic Control) in Spain. Since 1968 **CEA-IFAC** looks after the development of the Automation in Spain, in its different issues: automatic control, robotics, *SIMULATION*, etc. The association is divided into national thematic groups, one of which is centered on Modeling, Simulation and Optimization, constituting the **CEA** Spanish Modeling and Simulation Group (**CEA-SMSG**). It looks after the development of the Modelling and Simulation (M&S) in Spain, working basically on all the issues concerning the use of M&S techniques as essential engineering tools for decision-making and optimization.

→ *http://www.ceautomatica.es/grupos/*

→ *emilio.jimenez@unirioja.es*
simulacion@cea-ifac.es

✉ CEA-SMSG / Emilio Jiménez, Department of Electrical Engineering, University of La Rioja, San José de Calasanz 31, 26004 Logroño (La Rioja), SPAIN

CEA - SMSG Officers	
President	Emilio Jiménez, *emilio.jimenez@unirioja.es*
Vice president	Juan Ignacio Latorre, *juanignacio.latorre@unavarra.es*
Repr. EUROSIM	Emilio Jiménez, *emilio.jimenez@unirioja.es*
Edit. Board SNE	Juan Ignacio Latorre, *juanignacio.latorre@unavarra.es*
Web EUROSIM	Mercedes Perez *mercedes.perez@unirioja.es*

Last data update February 2018

CSSS – Czech and Slovak Simulation Society

CSSS -The *Czech and Slovak Simulation Society* has about 150 members working in Czech and Slovak national scientific and technical societies (*Czech Society for Applied Cybernetics and Informatics*, *Slovak Society for Applied Cybernetics and Informatics*). **CSSS** main objectives are: development of education and training in the field of modelling and simulation, organising professional workshops and conferences, disseminating information about modelling and simulation activities in Europe. Since 1992, **CSSS** is full member of **EUROSIM**.

→ *www.fit.vutbr.cz/CSSS*

snorek@fel.cvut.cz

✉ CSSS / Miroslav Šnorek, CTU Prague
FEE, Dept. Computer Science and Engineering,
Karlovo nam. 13, 121 35 Praha 2, Czech Republic

CSSS Officers	
President	Miroslav Šnorek, *snorek@fel.cvut.cz*
Vice president	Mikuláš Alexík, *alexik@frtk.fri.utc.sk*
Scientific Secr.	A. Kavička, *Antonin.Kavicka@upce.cz*
Repr. EUROSIM	Miroslav Šnorek, *snorek@fel.cvut.cz*
Edit. Board SNE	Mikuláš Alexík, *alexik@frtk.fri.utc.sk*
Web EUROSIM	Petr Peringer, peringer@fit.vutbr.cz

Last data update December 2012

DBSS – Dutch Benelux Simulation Society

The *Dutch Benelux Simulation Society* (**DBSS**) was founded in July 1986 in order to create an organisation of simulation professionals within the Dutch language area. **DBSS** has actively promoted creation of similar organisations in other language areas. **DBSS** is a member of **EUROSIM** and works in close cooperation with its members and with affiliated societies.

→ *www.DutchBSS.org*

a.w.heemink@its.tudelft.nl

✉ DBSS / A. W. Heemink
Delft University of Technology, ITS - twi,
Mekelweg 4, 2628 CD Delft, The Netherlands

DBSS Officers	
President	M. Mujica Mota, *m.mujica.mota@hva.nl*
Vice president	A. Heemink, *a.w.heemink@its.tudelft.nl*
Treasurer	A. Heemink, *a.w.heemink@its.tudelft.nl*
Secretary	P. M. Scala, p.m.scala@hva.nl
Repr. EUROSIM	M. Mujica Mota, *m.mujica.mota@hva.nl*
Edit. SNE/Web	M. Mujica Mota, *m.mujica.mota@hva.nl*

Last data update June 2016

LIOPHANT Simulation

Liophant Simulation is a non-profit association born in order to be a trait-d'union among simulation developers and users; Liophant is devoted to promote and diffuse the simulation techniques and methodologies; the Association promotes exchange of students, sabbatical years, organization of International Conferences, courses and internships focused on M&S applications.

→ *www.liophant.org*

info@liophant.org

✉ LIOPHANT Simulation, c/o Agostino G. Bruzzone,
DIME, University of Genoa, Savona Campus
via Molinero 1, 17100 Savona (SV), Italy

LIOPHANT Officers	
President	A.G. Bruzzone, *agostino@itim.unige.it*
Director	E. Bocca, *enrico.bocca@liophant.org*
Secretary	A. Devoti, *devoti.a@iveco.com*
Treasurer	Marina Massei, *massei@itim.unige.it*
Repr. EUROSIM	A.G. Bruzzone, *agostino@itim.unige.it*
Deputy	F. Longo, *f.longo@unical.it*
Edit. Board SNE	F. Longo, *f.longo@unical.it*
Web EUROSIM	F. Longo, *f.longo@unical.it*

Last data update June 2016

LSS – Latvian Simulation Society

The Latvian Simulation Society (LSS) has been founded in 1990 as the first professional simulation organisation in the field of Modelling and simulation in the post-Soviet area. Its members represent the main simulation centres in Latvia, including both academic and industrial sectors.

→ *www.itl.rtu.lv/imb/*

Egils.Ginters@rtu.lv

✉ Prof. Egils Ginters, Kirshu Str.13A, Cesis LV-4101,
Latvia

LSS Officers	
President	Yuri Merkuryev, *merkur@itl.rtu.lv*
Vice President	Egils Ginters, *egils.ginters@rtu.lv*
Secretary	Artis Teilans, *artis.teilans@rta.lv*
Repr. EUROSIM	Egils Ginters, egils.ginters@rtu.lv
Deputy	Artis Teilans, *artis.teilans@rta.lv*
Edit. Board SNE	Juri Tolujew, *Juri.Tolujew@iff.fraunhofer.de*
Web EUROSIM	Vitaly Bolshakov, *vitalijs.bolsakovs@rtu.lv*

Last data update November 2020

KA-SIM Kosovo Simulation Society

Kosova Association for Modeling and Simulation (KA-SIM, founded in 2009), is part of Kosova Association of Control, Automation and Systems Engineering (KA-CASE). KA-CASE was registered in 2006 as non Profit Organization and since 2009 is National Member of IFAC – International Federation of Automatic Control. KA-SIM joined EUROSIM as Observer Member in 2011. In 2016, KA-SIM became full member.

KA-SIM has about 50 members, and is organizing the international conference series International Conference in Business, Technology and Innovation, in November, in Durrhes, Albania, and IFAC Simulation Workshops in Pristina.

→ *www.ubt-uni.net/ka-case*

ehajrizi@ubt-uni.net

✉ MOD&SIM KA-CASE; Att. Dr. Edmond Hajrizi
Univ. for Business and Technology (UBT)
Lagjja Kalabria p.n., 10000 Prishtina, Kosovo

KA-SIM Officers	
President	Edmond Hajrizi, *ehajrizi@ubt-uni.net*
Vice president	Muzafer Shala, *info@ka-sim.com*
Secretary	Lulzim Beqiri, *info@ka-sim.com*
Treasurer	Selman Berisha, *info@ka-sim.com*
Repr. EuroSim	Edmond Hajrizi, *ehajrizi@ubt-uni.net*
Deputy	Muzafer Shala, *info@ka-sim.com*
Edit. Board SNE	Edmond Hajrizi, *ehajrizi@ubt-uni.net*
Web EuroSim	Betim Gashi, *info@ka-sim.com*

Last data update December 2016

NSSM – National Society for Simulation Modelling (Russia)

NSSM - The Russian National Simulation Society (Национальное Общество Имитационного Моделирования – НОИМ) was officially registered in Russian Federation on February 11, 2011. In February 2012 NSS has been accepted as an observer member of **EUROSIM**, and in 2015 NSSM has become full member.

→ *www.simulation.su*

yusupov@iias.spb.su

✉ NSSM / R. M. Yusupov,
St. Petersburg Institute of Informatics and Automation RAS, 199178, St. Petersburg, 14th lin. V.O, 39

NSSM Officers	
President	R. M. Yusupov, *yusupov@iias.spb.su*
Chair Man. Board	A. Plotnikov, *plotnikov@sstc.spb.ru*
Secretary	M. Dolmatov, *dolmatov@simulation.su*
Repr. EuroSim	R.M. Yusupov, *yusupov@iias.spb.su* Y. Senichenkov, *senyb@dcn.icc.spbstu.ru*
Deputy	B. Sokolov, *sokol@iias.spb.su*
Edit. Board SNE	Y. Senichenkov, *senyb@mail.ru*, *senyb@dcn.icc.spbstu.ru*,

Last data update February 2018

PSCS – Polish Society for Computer Simulation

PSCS was founded in 1993 in Warsaw. PSCS is a scientific, non-profit association of members from universities, research institutes and industry in Poland with common interests in variety of methods of computer simulations and its applications. At present PSCS counts 257 members.

→ *www.eurosim.info*, www.ptsk.pl/

leon@ibib.waw.pl

✉ PSCS / Leon Bobrowski, c/o IBIB PAN,
ul. Trojdena 4 (p.416), 02-109 Warszawa, Poland

PSCS Officers	
President	Leon Bobrowski, *leon@ibib.waw.pl*
Vice president	Tadeusz Nowicki, *Tadeusz.Nowicki@wat.edu.pl*
Treasurer	Z. Sosnowski, *zenon@ii.pb.bialystok.pl*
Secretary	Zdzislaw Galkowski, *Zdzislaw.Galkowski@simr.pw.edu.pl*
Repr. EuroSim	Leon Bobrowski, *leon@ibib.waw.pl*
Deputy	Tadeusz Nowicki, *tadeusz.nowicki@wat.edu.pl*
Edit. Board SNE	Zenon Sosnowski, z.sosnowski@pb.ed.pl
Web EuroSim	Magdalena Topczewska *m.topczewska@pb.edu.pl*

Last data update December2013

SIMS – Scandinavian Simulation Society

SIMS is the *Scandinavian Simulation Society* with members from the five Nordic countries Denmark, Finland, Iceland, Norway and Sweden. The SIMS history goes back to 1959. SIMS practical matters are taken care of by the SIMS board consisting of two representatives from each Nordic country (Iceland one board member).

SIMS Structure. SIMS is organised as federation of regional societies. There are **FinSim** (Finnish Simulation Forum), **MoSis** (Society for Modelling and Simulation in Sweden), **DKSIM** (Dansk Simuleringsforening) and **NFA** (Norsk Forening for Automatisering).

→ *www.scansims.org*

bernt.lie@usn.no

✉ SIMS / Bernt Lie, Faculty of Technology, Univ.College of Southeast Norway, Department of Technology, Kjølnes ring 56, 3914 Porsgrunn, Norway

SIMS Officers	
President	Bernt Lie, *Bernt.Lie@usn.no*
Vice president	Erik Dahlquist, *erik.dahlquist@mdh.se*
Treasurer	Vadim Engelson, *vadime@mathcore.com*
Repr. EUROSIM	Esko Juuso, *esko.juuso@oulu.fi*
Edit. Board SNE	Esko Juuso, *esko.juuso@oulu.fi*
Web EUROSIM	Vadim Engelson, *vadime@mathcore.com*

Last data update February 2020

SLOSIM – Slovenian Society for Simulation and Modelling

SLOSIM - Slovenian Society for Simulation and Modelling was established in 1994 and became the full member of **EUROSIM** in 1996. Currently it has 90 members from both Slovenian universities, institutes, and industry. It promotes modelling and simulation approaches to problem solving in industrial as well as in academic environments by establishing communication and cooperation among corresponding teams.

→ www.slosim.si

slosim@fe.uni-lj.si

✉ SLOSIM / Vito Logar, Faculty of Electrical Engineering, University of Ljubljana, Tržaška 25, 1000 Ljubljana, Slovenia

SLOSIM Officers	
President	Vito Logar, *vito.logar@fe.uni-lj.si*
Vice president	Božidar Šarler, *bozidar.sarler@ung.si*
Secretary	Simon Tomažič, *simon.tomazic@fe.uni-lj.si*
Treasurer	Milan Simčič, *milan.simcic@fe.uni-lj.si*
Repr. EUROSIM	B. Zupančič, *borut.zupancic@fe.uni-lj.si*
Deputy	Vito Logar, *vito.logar@fe.uni-lj.si*
Edit. Board SNE	R. Karba, *rihard.karba@fe.uni-lj.si*
Web EUROSIM	Vito Logar, *vito.logar@fe.uni-lj.si*

Last data update December 2018

UKSIM - United Kingdom Simulation Society

The UK Simulation Society is very active in organizing conferences, meetings and workshops. UKSim holds its annual conference in the March-April period. In recent years the conference has always been held at Emmanuel College, Cambridge. The Asia Modelling and Simulation Section (AMSS) of UKSim holds 4-5 conferences per year including the EMS (European Modelling Symposium), an event mainly aimed at young researchers, organized each year by UKSim in different European cities. Membership of the UK Simulation Society is free to participants of any of our conferences and their co-authors.

→*uksim.info*

david.al-dabass@ntu.ac.uk

✉ **UKSIM / Prof. David Al-Dabass**
Computing & Informatics,
Nottingham Trent University
Clifton lane, Nottingham, NG11 8NS, United Kingdom

UKSIM Officers	
President	David Al-Dabass, *david.al-dabass@ntu.ac.uk*
Secretary	T. Bashford, *tim.bashford@uwtsd.ac.uk*
Treasurer	D. Al-Dabass, *david.al-dabass@ntu.ac.uk*
Membership chair	G. Jenkins, *glenn.l.jenkins@smu.ac.uk*
Local/Venue chair	Richard Cant, *richard.cant@ntu.ac.uk*
Repr. EUROSIM	Dr Taha Osman, taha.osman@ntu.ac.uk
Deputy	T. Bashford, *tim.bashford@uwtsd.ac.uk*
Edit. Board SNE	D. Al-Dabass, *david.al-dabass@ntu.ac.uk*

Last data update March 2020

EUROSIM Observer Members

ROMSIM – Romanian Modelling and Simulation Society

ROMSIM has been founded in 1990 as a non-profit society, devoted to theoretical and applied aspects of modelling and simulation of systems.

→ *www.eurosim.info/societies/romsim/*

florin_h2004@yahoo.com

ROMSIM / Florin Hartescu,
National Institute for Research in Informatics, Averescu Av. 8 – 10, 011455 Bucharest, Romania

ROMSIM Officers	
President	N. N.
Vice president	Florin Hartescu, *florin_h2004@yahoo.com* Marius Radulescu, *mradulescu.csmro@yahoo.com*
Repr. EUROSIM	Marius Radulescu
Deputy	Florin Hartescu
Edit. Board SNE	Constanta Zoe Radulescu, *zoe@ici.ro*
Web EUROSIM	Florin Hartescu

Last data update June 2019

ALBSIM – Albanian Simulation Society

The Albanian Simulation Society has been initiated at the Department of Statistics and Applied Informatics, Faculty of Economy at the University of Tirana, by Prof. Dr. Kozeta Sevrani. The society is involved in different international and local simulation projects, and is engaged in the organisation of the conference series ISTI - Information Systems and Technology. In July 2019 the society was accepted as EUROSIM Observer Member.

→ *www.eurosim.info/societies/albsim/*

kozeta.sevrani@unitir.edu.al

Albanian Simulation Goup, attn. Kozeta Sevrani
University of Tirana, Faculty of Economy
rr. Elbasanit, Tirana 355 Albania

Albanian Simulation Society- Officers	
Chairt	Kozeta Sevrani, *kozeta.sevrani@unitir.edu.al*
Repr. EUROSIM	Kozeta Sevrani
Edit. Board SNE	Albana Gorishti, *albana.gorishti@unitir.edu.al* Majlinda Godolja, *majlinda.godolja@feut.edu.al*

Last data update July 2019

Societies in Re-organisation / Former Societies

The following societies are at present inactive or under re-organisation:

- CROSSIM – *Croatian Society for Simulation Modelling*
  Contact: Tarzan Legović, *Tarzan.Legovic@irb.hr*
- FRANCOSIM – Société Francophone de Simulation
- HSS – Hungarian Simulation Society
- ISCS – Italian Society for Computer Simulation

The following societies have been formally terminated:

- MIMOS –Italian Modeling & Simulation Association; terminated end of 2020.

HSS – Hungarian Simulation Society

There are plans to reactivate Hungarian Simulation Society. M. Mujica Mota EUROSIM President, is in contact with Andrási Gábor, Head of the Dean's office at the Faculty of International Management and Business of Budapest Business School University of Applied Sciences (BBS). We ask interested people to contact Mr. Gábor, *andrasi.gabor@uni-bge.hu*.

Association Simulation News

ARGESIM is a non-profit association generally aiming for dissemination of information on system simulation – from research via development to applications of system simulation. **ARGESIM** is closely co-operating with **EUROSIM**, the Federation of European Simulation Societies, and with **ASIM**, the German Simulation Society. **ARGESIM** is an 'outsourced' activity from the *Mathematical Modelling and Simulation Group* of TU Wien, there is also close co-operation with TU Wien (organisationally and personally).

→ *www.argesim.org*

→ *office@argesim.org*

→ ARGESIM/Math. Modelling & Simulation Group,
Inst. of Analysis and Scientific Computing, TU Wien
Wiedner Hauptstrasse 8-10, 1040 Vienna, Austria
Attn. Prof. Dr. Felix Breitenecker

ARGESIM is following its aims and scope by the following activities and projects:

- Publication of the scientific journal **SNE** – *Simulation Notes Europe* (membership journal of **EUROSIM**, the *Federation of European Simulation Societies*) – *www.sne-journal.org*
- Organisation and Publication of the **ARGESIM Benchmarks** for *Modelling Approaches and Simulation Implementations*
- Publication of the series **ARGESIM Reports** for monographs in system simulation, and proceedings of simulation conferences and workshops
- Publication of the special series **FBS Simulation** – Advances in Simulation / Fortschrittsberichte Simulation - monographs in co-operation with ASIM, the German Simulation Society
- Support of the Conference Series **MATHMOD Vienna** (triennial, in co-operation with **EUROSIM**, **ASIM**, and **TU Wien**) – www.mathmod.at
- Administration of **ASIM** (German Simulation Society) and administrative support for **EUROSIM** *www.eurosim.info*
- Simulation activities for TU Wien

ARGESIM is a registered non-profit association and a registered publisher: **ARGESIM Publisher Vienna**, root ISBN 978-3-901608-xx-y, root DOI 10.11128/z…zz.zz. Publication is open for **ASIM** and for **EUROSIM Member Societies**.

SNE – Simulation Notes Europe

The scientific journal **SNE** – *Simulation Notes Europe* provides an international, high-quality forum for presentation of new ideas and approaches in simulation – from modelling to experiment analysis, from implementation to verification, from validation to identification, from numerics to visualisation – in context of the simulation process. **SNE** puts special emphasis on the overall view in simulation, and on comparative investigations.
Furthermore, **SNE** welcomes contributions on education in/for/with simulation.

SNE is also the forum for the **ARGESIM Benchmarks** on *Modelling Approaches and Simulation Implemen-tations* publishing benchmarks definitions, solutions, reports and studies – including model sources via web.

→ *www.sne-journal.org*,

→ *office@sne-journal.org*, *eic@sne-journal.org*

→ SNE Editorial Office
ARGESIM/Math. Modelling & Simulation Group,
Inst. of Analysis and Scientific Computing, TU Wien
Wiedner Hauptstrasse 8-10, 1040 Vienna, Austria
EiC Prof. Dr. Felix Breitenecker

SNE, primarily an electronic journal, follows an open access strategy, with free download in basic layout. **SNE** is the official membership journal of **EUROSIM**, the *Federation of European Simulation Societies*. Members of **EUROSIM Societies** are entitled to download **SNE** in high-quality, and to access additional sources of benchmark publications, model sources, etc. On the other hand, **SNE** offers **EUROSIM Societies** a publication forum for post-conference publication of the society's international conferences, and the possibility to compile thematic or event-based **SNE Special Issues**.

Simulationists are invited to submit contributions of any type – *Technical Note*, *Short Note*, *Project Note*, *Educational Note*, *Benchmark Note*, etc. via **SNE**'s website:

→ *www.sne-journal.org*,

Schedule for EUROSIM Conferences and Congress

EUROSIM societies organise the following virtual and in-person events in 2022 and 2023:

The **EUROSIM Board** and **DBSS** organise **VESS** – the **Virtual EUROSIM Seminar**, a series of online presentations discussing trends in modelling and simulation. These international online simulation seminars – monthly or bi-monthly – are open to everybody, via Zoom, lasting 60 minutes (45 minutes presentations, 15 minutes Q & A).
Information and informal registration via website **www.eurosim2023.eu**

MATHMOD organizers continue the conference series one year later, with **10th MATHMOD 2022**, July 27-29, 2022, as *in-person* event. **MATHMOD 2022**, one of **EUROSIM**'s main events, provides a forum for professionals, researchers, and experts in the field of theoretic and applied aspects of mathematical modelling for systems of dynamic nature.
The scope of the **MATHMOD 2022** conference covers theoretic and applied aspects of various types of mathematical modelling (equations of various types, automata, Petri nets, bond graphs, qualitative and fuzzy models) for systems of dynamic nature (deterministic, stochastic, continuous, discrete or hybrid) – info and details **www.mathmod.at**

ASIM - the German / Austrian / Swiss simulation society – is organising the **26th Symposium Simulation Technique** – **ASIM 2022** at TU Vienna, July 25-27, just before **MATHMOD 2022**. ASIM hopes for a German/English-based event as it used to be before – with personal contacts, and in synergy with **MATHMOD 2022**. – info **www.asim-gi.org/asim2022**

EUROSIM 2023, the **11th EUROSIM Congress**, will take place in Amsterdam, The Netherlands, June 28-30, 2023. It will be organized by the Dutch Benelux Simulation Society (www.dutchbss.org) supported mainly by their corporate members like TU Delft, Amsterdam University of Applied Sciences, EUROCONTROL and IGAMT (www.igamt.eu).
Due to the growth of Simulation and its relationship with other analytical techniques like Big Data, AI, Machine Learning, Large Scale Simulation and others, the event will be structured, for the first time, in dedicated tracks focused on different areas and applications of Simulation ranging from aviation to health care and humanitarian activities.
Please follow the news and activities towards the **EUROSIM 2023** at **www.eurosim2023.eu**

New Schedule for Vienna **ASIM 2022** and **MATHMOD 2022**

Due to the critical pandemic situation in Europe it is not possible to hold these combined conferences as on-site events in February 2022. Since the simulation community strongly desires in-person events, it was decided to postpone **ASIM 2022** and **MATHMOD 2022** to **July 2022**.

The scope of the *ASIM Symposium Simulationstechnik* – also including the *workshop of the working groups GMMS and STS* – covers basics, methods, and tools of modeling and simulation as well as all areas of application (from engineering sciences to computer science, production and logistics, bio-, environmental and geosciences, climate and ecosystem, up to training and education in modeling and simulation.

Conference languages are German and English.

Submission of Full Contributions and Short Contributions is now possible until April 1, 2022.

Website: **www.asim-gi.org/asim2022**

Contact: *asim2022@asim-gi.org*

10th Vienna International Conference on Mathematical Modelling

July 27 - 29, 2022, Vienna, Austria

The scope of *MATHMOD 2022* covers theoretic and applied aspects of various types of mathematical modelling (e.g., equations of various types, automata, Petri nets, bond graphs, qualitative and fuzzy models, machine learning)for systems of dynamic nature (deterministic, stochastic, continuous, discrete or hybrid with respect to time).

The reviewing of already submitted papers is already finished, and authors will be informed about acceptance until January 14, 2022. But it is possible to submit late papers: late full contributions until February 1, 2022, and late discussion contributions papers until March 15, 2022. See www.mathmod.at for details.

Website: **www.mathmod.at**

Contact: *mathmod@acin.tuwien.ac.at*

www.tuverlag.at
ISBN 978-3-903311-30-5