INFOCOMP 2022

The Twelfth International Conference on Advanced Communications and Computation

ISBN: 978-1-61208-961-4

June 26th –30th, 2022

Porto, Portugal

INFOCOMP 2021 Editors

Claus-Peter Rückemann, Westfälische Wilhelms-Universität Münster (WWU);
DIMF, Germany, LUH, Germany

INFOCOMP 2022

Forward

The Twelfth International Conference on Advanced Communications and Computation (INFOCOMP 2022), held between June 26th and June 30th, 2022, continued a series of events dedicated to advanced communications and computing aspects, covering academic and industrial achievements and visions.

The diversity of semantics of data, context gathering and processing led to complex mechanisms for applications requiring special communication and computation support in terms of volume of data, processing speed, context variety, etc. The new computation paradigms and communications technologies are now driven by the needs for fast processing and requirements from data-intensive applications and domain-oriented applications (medicine, geoinformatics, climatology, remote learning, education, large scale digital libraries, social networks, etc.). Mobility, ubiquity, multicast, multi-access networks, data centers, cloud computing are now forming the spectrum of de factor approaches in response to the diversity of user demands and applications. In parallel, measurements control and management (self-management) of such environments evolved to deal with new complex situations.

We take here the opportunity to warmly thank all the members of the INFOCOMP 2022 technical program committee, as well as all the reviewers. The creation of such a high-quality conference program would not have been possible without their involvement. We also kindly thank all the authors who dedicated much of their time and effort to contribute to INFOCOMP 2022. We truly believe that, thanks to all these efforts, the final conference program consisted of top-quality contributions. We also thank the members of the INFOCOMP 2022 organizing committee for their help in handling the logistics of this event.

We hope that IMFOCOMP 2022 was a successful international forum for the exchange of ideas and results between academia and industry and for the promotion of progress in the field of Advanced Communications and Computation.

INFOCOMP 2022 Chairs

INFOCOMP 2022 Steering Committee

Claus-Peter Rückemann, Westfälische Wilhelms-Universität Münster (WWU) / DIMF / Leibniz Universität Hannover, Germany
Nicola Calabretta, Eindhoven University of Technology, Netherlands

INFOCOMP 2022 Publicity Chairs

José Miguel Jiménez, Universitat Politècnica de València (UPV), Spain
Laura Garcia, Universitat Politècnica de València (UPV), Spain

INFOCOMP 2022
Committee

INFOCOMP 2022 Steering Committee

Claus-Peter Rückemann, Westfälische Wilhelms-Universität Münster (WWU) / DIMF / Leibniz Universität Hannover, Germany
Nicola Calabretta, Eindhoven University of Technology, Netherlands

INFOCOMP 2022 Publicity Chairs

José Miguel Jiménez, Universitat Politècnica de València (UPV), Spain
Laura Garcia, Universitat Politècnica de València (UPV), Spain

INFOCOMP 2022 Technical Program Committee

Vicki H. Allan, Utah State University, USA
Daniel Andresen, Kansas State University, USA
Vijayan K. Asari, University of Dayton, USA
Marc Baaden, CNRS, France
Jacob Balma, Hewlett Packard Enterprise Company, USA
Bernhard Bandow, GWDG - Göttingen, Germany
Christine Bassem, Wellesley College, USA
Raoudha Ben Djemaa, MIRACL, Sfax, Tunisia
Tekin Bicer, Argonne National Laboratory, USA
Julien Bigot, Maison de la Simulation / CEA, France
David Boehme, Lawrence Livermore National Laboratory, USA
Radouan Boukharfane, KAUST, Saudi Arabia / MSDA | UM6P, Morocco
Abbas Bradai, University of Poitiers, France
Stephanie Brink, Lawrence Livermore National Laboratory, USA
Paolo Burgio, University of Modena and Reggio Emilia, Italy
Xiao-Chuan Cai, University of Colorado Boulder, USA
Nicola Calabretta, Eindhoven University of Technology, Netherlands
Enrico Casella, University of Kentucky, USA
Jian Chang, Bournemouth University, UK
Jieyang Chen, Oak Ridge National Laboratory, USA
Albert M. K. Cheng, University of Houston, USA
Enrique Chirivella Pérez, Universitat de Valencia, Spain
Noelia Correia, Center for Electronics Opto-Electronics and Telecommunications (CEOT) | University of Algarve, Portugal
Bruce D'Amora, IBM T. J. Watson Research Center, USA
Tiziano De Matteis, ETH Zurich, Switzerland
Daniele De Sensi, ETH Zurich, Switzerland
Iman Faraji, Nvidia Inc., Canada
Josué Feliu, Universitat Politècnica de València, Spain
Francesco Fraternali, University of California, San Diego, USA
Hans-Hermann Frese, Gesellschaft für Informatik e.V., Germany

Copyright Information

For your reference, this is the text governing the copyright release for material published by IARIA.

The copyright release is a transfer of publication rights, which allows IARIA and its partners to drive the dissemination of the published material. This allows IARIA to give articles increased visibility via distribution, inclusion in libraries, and arrangements for submission to indexes.

I, the undersigned, declare that the article is original, and that I represent the authors of this article in the copyright release matters. If this work has been done as work-for-hire, I have obtained all necessary clearances to execute a copyright release. I hereby irrevocably transfer exclusive copyright for this material to IARIA. I give IARIA permission or reproduce the work in any media format such as, but not limited to, print, digital, or electronic. I give IARIA permission to distribute the materials without restriction to any institutions or individuals. I give IARIA permission to submit the work for inclusion in article repositories as IARIA sees fit.

I, the undersigned, declare that to the best of my knowledge, the article is does not contain libelous or otherwise unlawful contents or invading the right of privacy or infringing on a proprietary right.

Following the copyright release, any circulated version of the article must bear the copyright notice and any header and footer information that IARIA applies to the published article.

IARIA grants royalty-free permission to the authors to disseminate the work, under the above provisions, for any academic, commercial, or industrial use. IARIA grants royalty-free permission to any individuals or institutions to make the article available electronically, online, or in print.

IARIA acknowledges that rights to any algorithm, process, procedure, apparatus, or articles of manufacture remain with the authors and their employers.

I, the undersigned, understand that IARIA will not be liable, in contract, tort (including, without limitation, negligence), pre-contract or other representations (other than fraudulent misrepresentations) or otherwise in connection with the publication of my work.

Exception to the above is made for work-for-hire performed while employed by the government. In that case, copyright to the material remains with the said government. The rightful owners (authors and government entity) grant unlimited and unrestricted permission to IARIA, IARIA's contractors, and IARIA's partners to further distribute the work.

Table of Contents

Solving Stationary Gas Transport Problems with Compressors of Piston and Generic Type

Anton Baldin
Fraunhofer Institute for Algorithms and Scientific Computing
Sankt Augustin, Germany
email: Anton.Baldin@scai.fraunhofer.de

Kläre Cassirer
Fraunhofer Institute for Algorithms and Scientific Computing
Sankt Augustin, Germany
email: Klaere.Cassirer@scai.fraunhofer.de

Tanja Clees
University of Applied Sciences Bonn-Rhein-Sieg and Fraunhofer Institute for Algorithms and Scientific Computing
Sankt Augustin, Germany
email: Tanja.Clees@scai.fraunhofer.de

Bernhard Klaassen
Fraunhofer Institute for Algorithms and Scientific Computing
Sankt Augustin, Germany
email: Bernhard.Klaassen@scai.fraunhofer.de

Igor Nikitin
Fraunhofer Institute for Algorithms and Scientific Computing
Sankt Augustin, Germany
email: Igor.Nikitin@scai.fraunhofer.de

Lialia Nikitina
Fraunhofer Institute for Algorithms and Scientific Computing
Sankt Augustin, Germany
email: Lialia.Nikitina@scai.fraunhofer.de

Sabine Pott
Fraunhofer Institute for Algorithms and Scientific Computing
Sankt Augustin, Germany
email: Sabine.Pott@scai.fraunhofer.de

***Abstract*—In this paper, modeling of piston and generic type gas compressors for a globally convergent algorithm for solving stationary gas transport problems is carried out. A theoretical analysis of the simulation stability, its practical implementation and verification of convergence on a realistic gas network have been carried out. The relevance of the paper for the topics of the conference is defined by a significance of gas transport networks as an advanced application of simulation and modeling, including the development of novel mathematical and numerical algorithms and methods.**

***Index Terms*—simulation and modeling; mathematical and numerical algorithms and methods; advanced applications; gas transport networks**

I. Introduction

In this paper, we will continue the study of globally converging methods for solving stationary network problems on the example of gas transport networks. Differently from our previous works, where the gas compressors of the most common turbine type were considered, in this paper, we investigate compressors of piston and generic type. In our work [1], we introduced the concept of generalized resistivity of network elements and formulated stability conditions for the algorithm solving the corresponding network problems. In the works [2] [3] [4] we have considered in detail the modeling of gas compressors of the turbine type. For these compressors, individually calibrated characteristics and data resampling on a regular grid were used. Now we consider compressors of piston and generic type, which are characterized by the existence of analytical solutions and a simpler representation of control equations. This simulation extends our system MYNTS (Multi-phYsics NeTwork Simulation) [5].

Globally convergent methods in applications to electric networks were formulated in [6], as well as in a more general form for piecewise linear systems in [7] [8] and for general smooth systems in [9]. Modeling of gas networks is described in detail in [10] [11]. This modeling is based on the nonlinear friction law in pipes [12] [13] and empirical approximations for the equation of state of a real gas [14] [15] [16].

In this paper, in Section II, we recall the general concepts of element resistivity and describe their physical meaning in more detail. In Section III, we will look at compressors of piston type and in Section IV – of generic type. In Section V, we will carry out a numerical solution of a realistic network problem with compressors of the described types.

II. Transport variables in stationary network problems

Network problems of a stationary type are described by a system of equations that includes linear Kirchhoff equations of the form $\sum Q_i = 0$, which describe the conservation of flows in network nodes, and equations of elements of the form $f(P_{in}, P_{out}, Q) = 0$, in the general case, nonlinear, introduced on each edge of the network graph. Here the transport variables $P_{in/out}$ are used – nodal variables for the input and output of the element, for gas networks – pressure values, Q – the flow through the element. In gas problems, flows are considered in different normalizations, which is indicated by the index: Q_m – mass flow, Q_ν –

 ISBN: 978-1-61208-961-4

molar flow, Q_N – volumetric flow under normal conditions, $Q_{vol,in/out}$ – volumetric flow in input or output conditions (by default, input conditions are taken), etc. An element is called generalized resistive if its equation has derivatives of the following signature:

$$\partial f/\partial P_{in} > 0,\ \partial f/\partial P_{out} < 0,\ \partial f/\partial Q < 0. \qquad (1)$$

The work [1] shows that stationary network problems in which all elements have a given signature have a unique solution that can be found by the standard stabilized Newton algorithm with an arbitrary choice of starting point. Technically, it also requires a supply with a set pressure P_{set} in each disconnected component of the graph, as well as a proper condition for the behavior of functions at infinity, which can be satisfied if there are linear continuations of the equations of elements outside the working region that have the signature (1). Also, the completely inverse signature is formally admissible, since the sign change of $f \to -f$ is admissible for stationary problems. To eliminate this trivial ambiguity, one can choose the sign of f, postulating the fulfillment of one of the conditions (1), for example, the first one.

The physical meaning of these conditions is illustrated in Figure 1. It shows the serial connection of the tested element (in this case the compressor, a circle) and a linear resistor (a rectangle). Pressure $P_{set1,2}$ is set at the free ends. The intermediate node must satisfy the equation

$$P_{out}(P_{set1}, Q) = P_{set2} + RQ, \qquad (2)$$

graphically depicted in the central and lower parts of the figure. Here $R > 0$ is the resistance value, the corresponding line on the figure increases monotonically. If the tested element has the signature (1), then the function $P_{out}(P_{set1}, Q)$ decreases monotonically in Q, which corresponds to the central part of the figure. In this case, the intersection of lines exists and is unique. It can also occur outside this graph, when the above condition is met at infinity (continuation of the element's characteristic by a linearly strictly decreasing function outside the working region). In the case, if the signature (1) would be violated and the function $P_{out}(P_{set1}, Q)$ would increase in Q, then by choosing the parameters P_{set2} and R it is possible to achieve that the lines will have several intersections or no intersection. Even if the function $P_{out}(P_{set1}, Q)$ increases in Q only locally, a linear resistor can be fitted to it, which will give several solutions to the problem under consideration. It is also clear that a non-linear resistor can also be used for this purpose, as long as its characteristic increases and has enough parameters for tuning.

Similarly, by connecting elements in reverse order, as well as considering their parallel connection, it can be shown that any violation of the condition (1) leads to a violation of the uniqueness of solution. If the signature is violated, then the tested element can be connected to an elementary resistive element in such a way that the equation will have several solutions or none. The case when the signature is satisfied for all elements and the system has a unique solution is, of course, more preferable in practical applications.

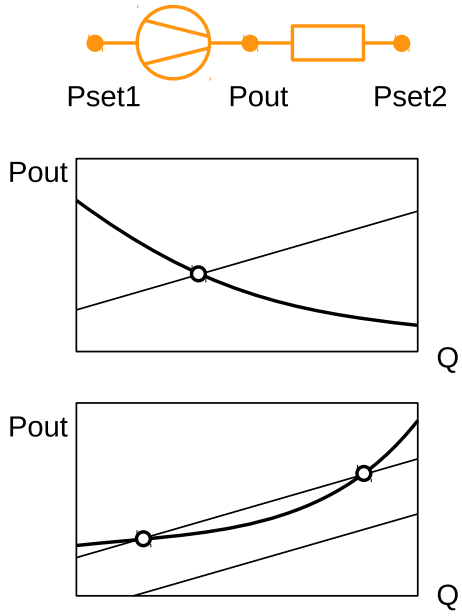


Fig. 1. On the top: a serial connection of compressor (circle) and resistor (rectangle); in the center: decreasing compressor $P_{out}(Q)$ characteristics (thick line) and increasing resistor $P_{out}(Q)$ characteristics (thin line) have a single intersection (stable case); at the bottom: increasing compressor $P_{out}(Q)$ characteristics (thick line) and increasing resistor $P_{out}(Q)$ characteristics (thin lines) can have multiple intersections or no intersection (unstable case).

Compressors are the most complex elements in gas problems; several levels of modeling are used to represent them. The main purpose of introduction of these levels is the gradual sophistication of modeling, where the solution of a simple model is used as a starting point for the more complex one. Also, it allows to separate effects dependent on individual calibration of compressors from their basic representation.

Free model: is the simplest, formulated only in terms of transport variables, and is described by a piecewise linear formula of the form

$$\max(\min(\ P_{in} - P_L, -P_{out} + P_H, -Q + Q_H), \qquad (3)$$
$$P_{in} - P_{out}, -Q) + \epsilon(P_{in} - P_{out} - Q) = 0, \qquad (4)$$

where parameters P_L, P_H, Q_H define target values, for example, $P_H = SPO$ for specified output pressure, or upper and lower limits for other controlled values. This formula defines a polyhedral surface in the space of transport variables in the so-called *maxmin* representation [8]. Particular attention should be paid to the last term in the equation, which is controlled by a small positive parameter ϵ. The reason for its introduction is that the exact equation satisfies the signature condition (1) only marginally, some derivatives vanish. The geometric interpretation of this is that the normals to the faces of the polyhedron described by the equation are directed strictly along the axes, although they should be directed inside the octant described by the condition (1). Such marginality leads to degeneracy of the Jacobi matrix, ambiguity of solutions, bad condition numbers, and other troubles for the numerical solution procedure. The introduction of a regularizing ϵ term

 ISBN: 978-1-61208-961-4

formally eliminates this problem by making the condition (1) strictly satisfied. At the same time, adjusting this parameter represents a compromise between the physical accuracy and the numerical stability of the solution procedure. In practice, the values $\epsilon = 10^{-6}...10^{-3}$ are tolerable, meaning the relative violation of, e.g., SPO-condition, up to 0.1%, simultaneously keeping the convergence rate near 100%.

Advanced model: introduces additional internal variables for compressors: revolution number rev, adiabatic enthalpy increase H_{ad}, performance $Perf$, efficiency η, torque M_t, and additional equations:

$$P = \rho RTz/\mu,\ Q_m = Q_{vol}\rho_{in}, \tag{5}$$

$$H_{ad} = P_{in}/(\rho_{in}\alpha)\cdot((P_{out}/P_{in})^{\alpha} - 1), \tag{6}$$

$$Perf = Q_m H_{ad}/\eta,\ M_t = Perf/(2\pi\cdot rev), \tag{7}$$

$$\alpha = (\kappa - 1)/\kappa,\ 0 < \alpha < 1,\ 0 < \eta < 1, \tag{8}$$

where the equation of state is written first with its parameters: density ρ, universal gas constant R, absolute temperature T, compressibility factor z, molar mass μ; the second is the relationship between the mass flow and the volumetric flow in the input conditions; the following are definitions of internal variables in terms of transport variables; $\kappa > 1$ is the adiabatic exponent.

For the turbocompressors considered in [2] [3] [4], additional relationships between internal variables are introduced based on the calibration procedure. We will now consider piston and generic type compressors, for which there is a simpler model that allows an analytical solution. The general strategy is to resolve all internal variables from the corresponding equations, obtain a formula in terms of transport variables, check its signature, and use it in the standard solution algorithm.

III. Piston compressors

Compressors of piston types are modeled by direct proportionality

$$Q_{vol} = V\cdot rev \tag{9}$$

with given constants η and V – compressor chamber volume. The control equation has the following patches:

$$f_1 = rev_{max} - rev \geq 0, \tag{10}$$

$$f_2 = M_{t,max} - M_t \geq 0, \tag{11}$$

$$f_3 = Perf_{max} - Perf \geq 0, \tag{12}$$

$$f_4 = rel_{max} - P_{out}/P_{in} \geq 0, \tag{13}$$

$$f_5 = \Delta P_{max} - (P_{out} - P_{in}) \geq 0, \tag{14}$$

with given constants rev_{max}, $M_{t,max}$, rel_{max}, ΔP_{max} and the function $Perf_{max}(rev)$ determined by the characteristics of the compressor drive.

Stability analysis: calculating the derivatives of f_i with respect to (P_{in}, P_{out}, Q_m) in the working region $0 < P_{in} \leq P_{out}$, $Q_m > 0$, $rev > 0$, we get the signatures given in Table I. In this case, the above formulas are used, as well as the stability of the equation of state: $\rho > 0$, $\partial\rho/\partial P > 0$.

TABLE I
Patch signatures of piston compressor

patch	sgn	condition
f_1	$(+\ 0\ -)$	
f_2	$(+ - 0)$	$P_{out}/P_{in} < \beta$
f_3	$(+ - -)$	$P_{out}/P_{in} < \beta$, $\partial M_{t,drv}/\partial\, rev < 0$
f_4	$(+ - 0)$	
f_5	$(+ - 0)$	

TABLE II
Patch signatures of generic compressor

patch	sgn	condition
f_1	$(+\ 0\ -)$	
f_2	$(+ - 0)$	$\partial z_{in}/\partial P_{in} < 0$ or small
f_3	$(+ - -)$	$\partial z_{in}/\partial P_{in} < 0$ or small

In particular, $rev = Q_m/(\rho_{in}V)$ has signature $(-0+)$, which implies the signature of f_1 in the table. $M_t = H_{ad}\rho_{in}V/(2\pi\eta)$ has signature $(*+0)$, where $* = \partial(H_{ad}\rho_{in})/\partial P_{in} < 0$ for $P_{out}/P_{in} < (1-\alpha)^{(-1/\alpha)} = \beta$. Thus, the signature f_2 is correct only if the compressor raises the pressure by no more than the factor β, with the value $\kappa = 1.29$ typical for natural gas, we get $\beta = 3.10408$. To eliminate the fold in the equation, f_2 should be replaced with $H_{ad}\rho_{in}|P_{in} \to \max(P_{in}, P_{out}/\beta)$. It is convenient to divide the expression f_3 by $(2\pi\, rev)$ and consider the signature $\tilde{f}_3 = M_{t,drv}(rev) - M_t$. As noted in [4], for drive equations to be stable it is necessary that $M_{t,drv}$ decrease with rev. Therefore, the first term in $\tilde{f}_3$ has the signature $(+0-)$, and the second already calculated $(+-0)$ in the region $P_{out}/P_{in} < \beta$, which gives the complete signature $(+--)$. Calculation of other derivatives is trivial. We also note that the presence of zeros in the signatures means that the rule (1) is satisfied marginally, which is corrected by adding a regularizing ϵ-term to the element equation. Also, for the practical implementation of these formulas, it is necessary to introduce clamps, which force all variables to the working region: $Q_m \to \max(Q_m, 0)$, $P_{out}/P_{in} \to \max(P_{out}/P_{in}, 1)$, etc.

IV. Generic compressors

Compressors of generic type can also be considered as an intermediate level of modeling (generic model). In this model, the variable rev is not introduced, and restrictions are introduced on other variables

$$f_1 = Q_{vol,max} - Q_{vol} \geq 0, \tag{15}$$

$$f_2 = H_{ad,max} - H_{ad} \geq 0, \tag{16}$$

$$f_3 = Perf_{max} - Perf \geq 0, \tag{17}$$

with constant $Q_{vol,max}$, $H_{ad,max}$ and $Perf_{max}$.

Stability analysis: Calculating derivatives similarly, for $Q_{vol} = Q_m/\rho_{in}$ we have signature $(-0+)$, hence $(+0-)$ for f_1. For $H_{ad} = RT_{in}z_{in}/(\mu_{in}\alpha)((P_{out}/P_{in})^{\alpha} - 1)$ we get $(*+0)$, where $* = \partial(z_{in}((P_{out}/P_{in})^{\alpha} - 1))/\partial P_{in} < 0$. For an ideal gas $z = 1$, hence, obviously, $* = -$. For natural gas z is a decreasing function of P, in this case also $* = -$. For

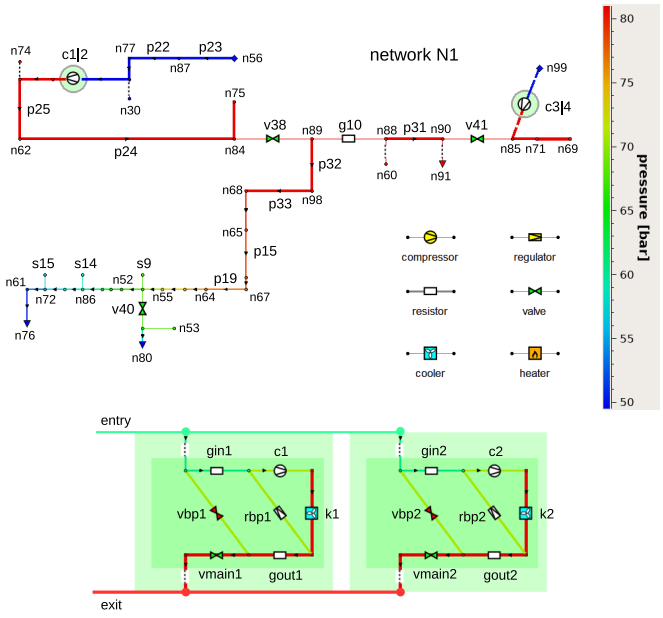


Fig. 2. On the top: test network N1; at the bottom: the structure of parallel compressor station. Images from [1].

some gases, such as hydrogen, z may increase with P, but it remains close to 1 and changes so slowly that the remaining decreasing dependence of H_{ad} on P_{in} dominates. Under these conditions, f_2 has signature $(+-0)$. For $Perf = Q_m H_{ad}/\eta$ the signature $(-++)$ under the same conditions on z_{in}, thus f_3 has the signature $(+--)$.

V. Numerical tests

The described patches are inserted into the free formula as follows:

$$\max(\min(\ P_{in} - P_L, -P_{out} + P_H, -Q + Q_H, \quad (18)$$

$$\underline{f_1, ..., f_n}\ ), \quad (19)$$

$$P_{in} - P_{out}, -Q) + \epsilon(P_{in} - P_{out} - Q) = 0, \quad (20)$$

after that the stabilized Newton algorithm described in [1] can be used to solve the system. The tests were carried out on the network N1 shown in Figure 2 on the top. This network has 100 nodes and 111 edges, of which 4 compressors are organized into two compressor stations c1|2 and c3|4 with individual compressors connected in parallel, as shown in Figure 2 at the bottom. Compressors in station c1|2 are configured as piston ones, in station c3|4 as generic ones. Values P_H, Q_H are set to unreachable high values, thereby activating the f_i patches described above. Note that the stations also include other elements, but they have trivial equations and are eliminated by the topological cleaning filter used in the solution procedure. The procedure consists of several phases with a gradual increase in the modeling level. First (init) the compressors are set to fulfill the main target values, e.g., $P = P_H$, then (free) the modeling level (3)-(4) is used, taking into account additional conditions, then (adv) the modeling level (18)-(20) is taken. The solution procedure described in [5] consists of the translation phase of the system from the network description language to the language understood by the numerical solver, and the actual numerical solution phase. The corresponding timing is given in Table III; approximately the same results are obtained if turbocompressors are used instead of piston/generic ones. The performed numerical experiment shows that the inclusion of piston and generic compressors in the system does not lead to any divergences or slowdown of the solution procedure, which is a direct consequence of the implementation of the stability criteria described above.

TABLE III
Timing for different phases of the solution procedure*

phase	translate	solve
init	15	8
free	15	7
adv	17	20
total	47	35

* in milliseconds, for 2.6 GHz Intel i7 CPU 16 GB RAM computer.

We also performed numerical experiments with test networks from work [4]. The test set contains 85 networks with complexity up to four thousand nodes and up to 42 compressors. Among them are multiple piston and generic compressors, in parallel and series connections. We have found that the presence and placement of such compressors does not affect performance in any way, and this is consistent with the convergence conditions we developed. The extension of the convergence theory to the dynamic case is the subject of our further work.

VI. Conclusion

In this work, modeling of piston and generic type gas compressors was carried out. The signatures of the derivatives of the control equation are analyzed, the ranges of parameter values are identified, under which the conditions for the stable operation of the algorithm for solving stationary network problems are satisfied. After the practical implementation of the modeling, in a numerical experiment on a realistic gas network, the convergence of the solution algorithm is shown.

Our future plans include extending the described methods to dynamic problems.

Acknowledgment

The work has been supported by the project TransHyDE-Sys, grant 03HY201M.

References

[1] T. Clees, I. Nikitin, and L. Nikitina, “Making Network Solvers Globally Convergent”, Advances in Intelligent Systems and Computing, vol. 676, 2018, pp. 140-153.

[2] T. Clees, I. Nikitin, L. Nikitina, and L. Segiet, “Modeling of Gas Compressors and Hierarchical Reduction for Globally Convergent Stationary Network Solvers”, International Journal On Advances in Systems and Measurements, vol. 11, 2018, pp. 61-71.

[3] A. Baldin, T. Clees, B. Klaassen, I. Nikitin, and L. Nikitina, “Topological Reduction of Stationary Network Problems: Example of Gas Transport”, International Journal On Advances in Systems and Measurements, vol. 13, 2020, pp. 83-93.

 ISBN: 978-1-61208-961-4

[4] A. Baldin et al., “AdvWarp: A Transformation Algorithm for Advanced Modeling of Gas Compressors and Drives”, in Proc. of SIMULTECH 2021, International Conference on Simulation and Modeling Methodologies, Technologies and Applications, pp. 231-238, SciTePress, 2021.
[5] T. Clees et al., “MYNTS: Multi-phYsics NeTwork Simulator”, in Proc. of SIMULTECH 2016, International Conference on Simulation and Modeling Methodologies, Technologies and Applications, pp. 179-186, SciTePress, 2016.
[6] J. Katzenelson, “An algorithm for solving nonlinear resistor networks”, Bell System Technical Journal, vol. 44, 1965, pp. 1605-1620.
[7] M. J. Chien and E. S. Kuh, “Solving piecewise-linear equations for resistive networks”, International Journal of Circuit Theory and Applications, vol. 4, 1976, pp. 1-24.
[8] A. Griewank, J.-U. Bernt, M. Radons, and T. Streubel, “Solving piecewise linear systems in abs-normal form”, Linear Algebra and its Applications, vol. 471, 2015, pp. 500-530.
[9] C. T. Kelley, Iterative Methods for Linear and Nonlinear Equations, SIAM, Philadelphia, 1995.
[10] J. Mischner, H. G. Fasold, and K. Kadner, System-planning basics of gas supply, Oldenbourg Industrieverlag GmbH, 2011 (in German).
[11] M. Schmidt, M. C. Steinbach, and B. M. Willert, “High detail stationary optimization models for gas networks”, Optimization and Engineering, vol. 16, 2015, pp. 131-164.
[12] J. Nikuradse, “Laws of flow in rough pipes”, NACA Technical Memorandum 1292, Washington, 1950.
[13] C. F. Colebrook and C. M. White, “Experiments with Fluid Friction in Roughened Pipes”, in Proc. of the Royal Society of London, Series A, Mathematical and Physical Sciences, vol. 161, num. 906, 1937, pp. 367-381.
[14] J. Saleh, Fluid Flow Handbook, McGraw-Hill 2002.
[15] DIN EN ISO 12213-2: Natural gas - Calculation of compression factor, European Committee for Standardization, 2010.
[16] O. Kunz and W. Wagner, “The GERG-2008 wide-range equation of state for natural gases and other mixtures: An expansion of GERG-2004”, J. Chem. Eng. Data, vol. 57, 2012, pp. 3032-3091.

 ISBN: 978-1-61208-961-4

High Entropy Quantum Communication Framework for Secure Key Distribution and Secure Messaging

Rohit De
Del Norte High School
San Diego, California 92127, USA
Email: de.rohit01@gmail.com

***Abstract*—This work explores quantum computing and quantum communication with a focus on cybersecurity. A high entropy quantum communication framework is set up for secure Quantum Key Distribution (QKD) and secure short messaging, based on the Deutsch-Jozsa (DJ) algorithm. QKD allows Alice and Bob to securely share a secret key and improves over the Public Key Infrastructure (PKI), which can become vulnerable as quantum computing matures. However, QKD itself can be compromised by sophisticated Man In The Middle (MITM) quantum attacks, such as intercept-resend and quantum cloning. Recent research on QKD improved the entropy by reordering the qubits within a DJ-packet and by hopping to a different size for each run of the DJ-algorithm. This paper further increases the entropy by additionally using multiple orthogonal bases for the different qubits in a DJ-packet, called the HRB (Hopping Reorder Basis) scheme. Furthermore, the HRB scheme does not require any pre-sharing to establish the protocol. Functionality of the HRB scheme is tested on Google's Cirq quantum simulator. Simulations show that an attacker's interception success drops 200-times in the HRB scheme when using two orthogonal bases vs. 12-times in the previous work. When three orthogonal bases are used, the attacker's interception success drops more than 1000-times, improving the secrecy of the communication.**

***Keywords*—PKI; QKD; Deutsch-Jozsa; MITM; Qubit.**

I. INTRODUCTION

Quantum technology has a great potential to advance computing and communication. While it can help strengthen internet security through means like Quantum Key Distribution (QKD) [6] [8], its computation power can also be exploited to break classical security schemes such as Public Key Infrastructure (PKI) [4]. In future, resourceful quantum computers utilizing Shor's algorithm can make asymmetric encryption algorithms like RSA [11], which is used in PKI, vulnerable to attacks. This puts sensitive information such as bank transactions, login credentials, and any encrypted communications at risk. To overcome this threat, QKD supports next generation key distribution when quantum networks and quantum computers become prevalent [4]. QKD is used for generating and sharing a secret key between two parties, Alice and Bob, using quantum mechanical properties of qubits. The secret key is then used to set up an encrypted data communication channel between them, as shown in Figure 1.

In quantum technology, information is encoded in elements called qubits. Qubits can exist as superposition of two states but can collapse to either $zero$ or one (i.e., $|0\rangle$ or $|1\rangle$) states when measured or copied. This is called the 'no-cloning' property [12] and is utilized in most of the QKD methods. The 'no-cloning' property lets the receiver, Alice, detect

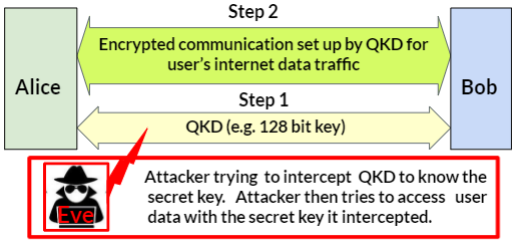


Figure 1. Attack on the QKD step to intercept the shared secret key.

an eavesdropper or a Man-In-The-Middle (MITM) attacker, Eve. This is unlike classical communication system where an eavesdropper can stealthily read, copy and store the bits transmitted, and then do offline brute force analysis. While the no-cloning property benefits QKD, it may still be possible for a very resourceful attacker to timely replace the collapsed qubits with fresh initialized qubits [13], e.g., initialization to $|0\rangle$ or $|1\rangle$ followed by superposition to replace the collapsed qubits. This threat and other attacks, like intercept/resend (faked-state) and quantum cloning [14] [15], can compromise QKD. This paper provides a high entropy quantum communication framework based on the Deustch-Jozsa (DJ) algorithm [1] by leveraging the original work by Nagata and Nakamura [2], and recent work by De et al. [3]. The DJ-algorithm allowed easy addition of new methods to increase entropy. The specific way the DJ-algorithm is leveraged illustrates an unique integration of quantum computing and quantum communication. This work also serves as a case-study on employing quantum technology for security and privacy.

This paper is organized as follows. Section II gives a brief survey of some of the QKD approaches and the previous research using the DJ-algorithm. Section III describes the new HRB mechanism and its entropy improvements, which decreases Eve's chance of successful interception versus previous research. Section IV shows the simulation results of the HRB scheme. Section V describes the end-to-end communication framework based on the HRB scheme, and how it can be used not only for secure QKD but also for secure short messages directly on a quantum communication channel. Finally, Section VI concludes the paper.

II. REVIEW OF LITERATURE

In BB84 [6] [7] QKD protocol, the qubits use two conjugate pairs of states, where the two states within each pair use orthogonal basis (the states of 0° and 90° form the rectilinear basis, while the states of 45° and 135° form the diagonal

 ISBN: 978-1-61208-961-4

basis). Alice creates a random bit (0 or 1) and randomly selects one of the two bases, rectilinear or diagonal, to transmit photons to Bob. Bob does not know the specific basis Alice picked; he also randomly selects either the rectilinear or the diagonal basis, and uses it to measure the photons he receives. When Alice and Bob share the bases they each used for each of the photons, they discard the photons for which they used mismatched bases. An interception by an eavesdropper would introduce errors due to the no-cloning property of qubits. The Six-State protocol [9] is the version of BB84 using a six-state polarization scheme on three orthogonal bases. The decoy-state technique [10] uses multiple intensity levels that are randomly chosen at the transmitter's source. Only one of the intensity levels is the signal state, while the others are the decoy states. Alice at the end publicly announces the intensity level that was used in the transmission of each qubit. The E91 [8] scheme uses perfectly correlated entangled photon pairs. Alice and Bob would get the same result if they measured the polarization of their photons. Any attempt by Eve to eavesdrop destroys these correlations in a way that Alice and Bob can detect. For some of these approaches, particularly those using few qubits with a few states for a secret bit, sophisticated QKD attackers [15] may be able to successfully intercept and replace the collapsed qubits with fresh initialized qubits and stay undetected.

A. Earlier work on QKD using the DJ-algorithm

The quantum DJ-algorithm [1] categorizes an n-qubit function U_f (called an oracle) in a single iteration, making it exponentially faster than the classical counterpart. The oracle U_f is determined to be *balanced* if for half of the inputs the output is $|0\rangle$ and for the other half the output is $|1\rangle$. The oracle U_f is *constant* if for all possible inputs the output is either always $|0\rangle$ or always $|1\rangle$. Each run of the DJ-algorithm requires a set of input qubits and a helper target qubit, which together form the DJ-packet.

The work by Nagata and Nakamura [2] for QKD using the DJ-algorithm is shown in Figure 2. Alice sends a sequence of DJ-packets, e.g., DJ-Packet1, and DJ-Packet2 as requests to Bob with the input qubits in them set to $|0\rangle$, the helper target qubit set to $|1\rangle$, and superposed by the *Hadamard* transform. Bob applies a balanced or a constant oracle U_f for each DJ-packet request and sends them back to Alice. Alice measures the qubits in the received DJ-packets and computes to determine if the oracle U_f Bob applied on each of them was balanced or constant. Bob's choices of { *constant*, *balanced* } map to { 0, 1 } bits of a key which now becomes a secret shared binary information (bit) between Bob and Alice. Figure 2 shows two DJ-packets, each carrying one binary bit information. To share a 128 bit secret key at least 128 DJ-packet communication is needed. The DJ-packets in Figure 2 are of fixed size, each with four qubits. The solid box marked (T) is the helper target qubit, the others are input qubits. With the throughput $= 1/4$ secret bit per qubit, the secrecy is very low due to predictable qubit positions. Similar to some existing QKD approaches, Nagata and Nakamura's [2]

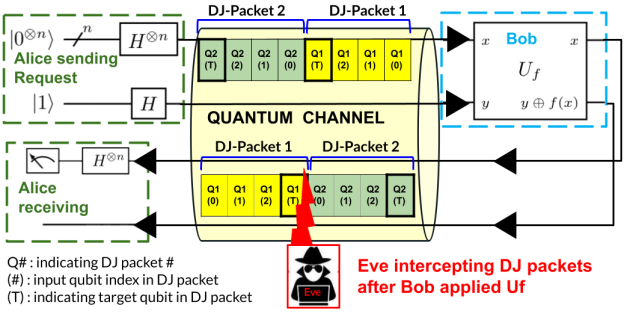


Figure 2. The DJ-algorithm for QKD with fixed-sized DJ-packets [2]

approach is also prone to MITM and eavesdropping attacks. The attacker (Eve) can predictably intercept the fixed size DJ-packets, seen as high as 25% in our simulations. A resourceful attacker can even replace the collapsed qubits with fresh qubits all initialized to $|0\rangle$ state [13] if the oracle U_f is constant, enabling Eve to stay undetected. Resourceful attackers can also do intercept/resend (faked-state) and quantum cloning attacks [14] [15]. An improvement in secrecy is achieved by De et al. [3], by changing the sizes of the consecutive DJ-packets based on a hopping (H) pattern and reordering (R) the position of the qubits within the DJ-packet, called the HR scheme. Hopping and Reordering make it hard for the attacker to identify all the required qubits and their type (input or target (T)), thereby increasing the difficulty of determining if the oracle U_f is constant or balanced. Figure 3 shows a sequence of DJ-packets with size hopping from 3 qubits to 2 qubits, and then to 4 qubits. It also shows the helper target qubit (solid box with (T) in the figure) can be at any position in the DJ-packet. The throughput $= 3/(3+2+4) = 1/3$ secret bit per qubit.

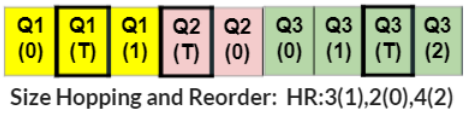


Figure 3. Hopping and reordering (HR) scheme for DJ-packet communication [3]. In the scheme "HR:N1(M1),N2(M2),N3(M3)", 'N1','N2' and 'N3' denote the number of qubits; 'M1', 'M2' and 'M3' denote the target qubit indices, for three consecutive DJ-packets.

However, the secrecy increase in the HR scheme is still not enough and there is a noticeable opportunity of successful interception, seen as much as 2% in our simulations. Furthermore, the HR method needs the specific HR scheme to be pre-shared between Alice and Bob. Hence, there is a need to develop a mechanism with much higher secrecy and that does not require pre-sharing. The next section provides the new HRB scheme that satisfies these requirements.

III. THE NEW HRB SCHEME

The HRB scheme provides a very high entropy quantum communication framework by using multiple orthogonal bases for the qubits in the DJ-packets. This increases the secrecy when compared to the HR mechanism [3] and the BB84-based schemes. Computations in DJ-algorithm still operate with qubits in the standard Z-basis, but during transmission certain selected qubits are transformed into a distinct value such that they are in a different set of orthogonal basis, e.g., the

 ISBN: 978-1-61208-961-4

X-basis or the Y-basis. Alternatively, the selected qubits can be rotated by distinct angle values that are orthogonal to each other. The HRB scheme thus harnesses both the computation and communication benefits of quantum technology.

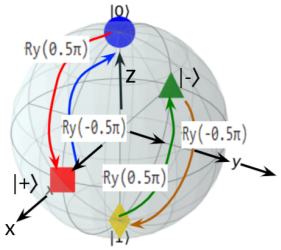


Figure 4. Qubit rotation about the Y-axis to change the values between the Z-basis and the X-basis.

The bloch-sphere in Figure 4 shows that $|0\rangle$ and $|1\rangle$ in Z-basis upon rotation about the Y-axis by 0.5π radians (90°) become the $|+\rangle$ and the $|-\rangle$ in the X-basis, respectively. To recover the qubits into Z-basis values, a rotation of the qubits by -0.5π radians ($-90°$) about the Y-axis is needed. Alternatively, different qubits can be rotated by different orthogonal angular values (e.g., $\theta 1$ and $\theta 2$, where $\theta 1$ and $\theta 2$ are orthogonal to each other) by the sender and rotated in the reverse direction (by $-\theta 1$ and $-\theta 2$) by the receiver before processing. The values for $\theta 1$ and $\theta 2$ are selected such that practical quantum hardware implementation with error correction and fault-tolerance are feasible. It is possible using a combination of Hadamard (H) and T gates [16]. The T-gate is a rotation around the z-axis by $\pi/4$ radians. With a sequence of H and T gates in specific orders as shown in Figure 5, a single-qubit gate rotation of various angle values can be set-up around an arbitrary axis in the Bloch sphere [16]. However, cost and decoherence problems are a potential limiting factor for expansive use of the T gates. The HRB scheme is described using basis, however, qubit rotation by a specific angle (e.g., $\theta 1$ and $\theta 2$) can be alternatively used instead of basis.

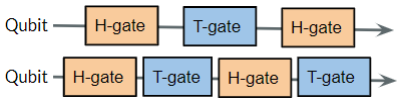


Figure 5. Various qubit rotations using H and T gates.

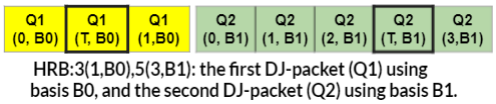
Figure 6 shows two approaches for applying multiple orthogonal bases (or qubit rotations by orthogonal angles) as:

(i) All qubits in a DJ-packet use the same basis, but different DJ-packets in a hopping sequence can use different bases. Example HRB: 3(1, B0), 5(3, B1)

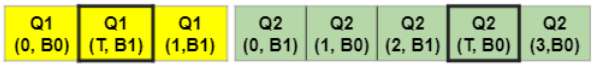
(ii) Qubits within a DJ-packet use different basis. Example HRB: 3(1, B0, B1, B1), 5(3, B1, B0, B1, B0, B0)

For illustration, two orthogonal bases (B0, B1) are used, e.g., B0=Z-basis, and B1=X-basis. The hopping sequence shows DJ-packets of two sizes with target qubit reordering. The scheme description is extended to include the orthogonal basis (or $\theta 1$, $\theta 2$) information for each qubit after the qubit index field. When orthogonal angle rotations are used, B0 and B1 represent two angles $\theta 1$ and $\theta 2$ orthogonal to each other.

Figure 7 shows the Cirq circuits for the HRB scheme that is shown in Figure 6 with two DJ-packets of sizes 3 and 5



(i) Different orthogonal basis only across DJ-packets.



(ii) Different orthogonal basis within each DJ-packet.

Figure 6. The two options, (i) and (ii), for the HRB scheme.

qubits, and using Z-basis(=B0) and the X-basis(=B1). Note the Cirq 'Ry' operator for rotation about the Y-axis by 0.5π or -0.5π radians, which is required for the value changes of the qubits to be in the different orthogonal basis (B0/B1). Alternatively, instead of changing to different orthogonal basis, different orthogonal angular rotations $\theta 1$ and $\theta 2$ can be used.

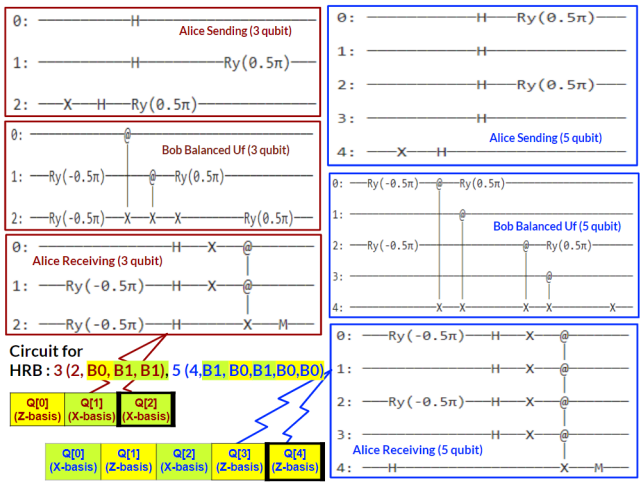


Figure 7. HRB Cirq circuits for 3-qubit and 5-qubit DJ-packets

Unlike BB84 [6], the HRB scheme does not suffer from the problem of basis mismatch between Alice and Bob. The bases (or, rotation angles) are predefined in the HRB scheme and are communicated using the mechanism described in Section V.A. The use of multiple orthogonal bases together with size-hopping and reordering makes the HRB scheme a more secure QKD mechanism than some of the BB84 based approaches [6] [7], and will be discussed more in Section IV B.

A. DJ-packet buffering and transmission

The qubits in the DJ-packets are momentarily buffered before sending to convert from the parallel order as in the DJ-algorithm, into a linear sequence for transmission on the quantum channel. The change from the Z-basis is done just before buffering, while target qubit reordering is done when the buffered qubits are serialized. Figure 8 shows qubits Q[0] and Q[2] are received in X-basis by Bob, then changed to Z-basis to apply the oracle, and then sent out again in X-basis. The reordered qubits are ordered back during buffering as they were in the original parallel form, and then changed back to Z-basis, as illustrated in Figure 8 using qubits Q[0] and Q[2]. Then, the next part of the DJ-algorithm is applied.

 ISBN: 978-1-61208-961-4

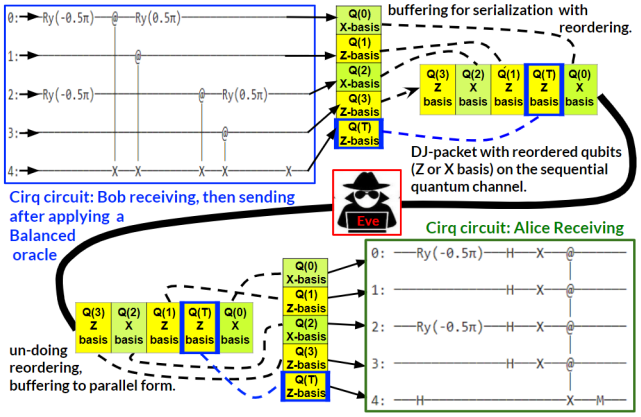


Figure 8. Reordering of DJ-packet qubits with buffering, and Ry rotation for using Z-basis or X-basis, on the quantum channel.

The DJ-algorithm computation always occurs with the qubits in Z-basis and in their natural order, while the quantum communication channel transmits the qubits in different orthogonal bases and positions reordered, leading to a large increase in entropy. The buffering time is determined by the time taken to transmit all the DJ-packet qubits in the quantum channel.

IV. SIMULATION AND TEST RESULTS

The HRB scheme is implemented in Python using Cirq [5] quantum operations and run on the Cirq quantum simulator. Cirq simulates the behavior of quantum hardware using stochastic models while running on classical computers. The Python code using Cirq operations sets up a quantum circuit for the HRB scheme and takes the DJ-packets as input. The Cirq implementation for Eve intercepts the DJ-packet qubits in transit according to the attack models. Being independent entities, Alice, Bob and Eve use three different Cirq simulator instances. The stream of DJ-packets is set up with a specific HRB scheme and sent between the simulator instances of Alice and Bob.

A. Experiments and the results

The attack models use a fixed-size scan window M (e.g., $M = 4$ qubits) with the target qubit (T) at the last index (i.e., $M - 1 = 3$). The models scan the continuous stream of DJ-packets between Bob and Alice with starting qubit offsets between 0 to 'scan window size -1' (i.e., $M - 1$). Attacker Eve expects that one of the offsets will match a DJ-packet boundary with size M qubits. Eve also assumes that up to three orthogonal bases can be randomly selected by Alice and Bob.

Figure 9 shows the simulation results for the attacker's successful interception rate. Figure 10 shows the secrecy, which is defined as "(100% - the successful interception%)", assuming there are no other compromises. Eve's interception is successful only if all the qubits of a DJ-packet are correctly identified. Partial interception of DJ-packet fails to determine the type of oracle U_f Bob applied, and leads to the attacker replacing incorrect collapsed qubits with fresh qubits that get detected by Alice, thereby exposing the attacker. The first bar (F:4) in Figure 9 represents the work by Nagata and Nakamura [2] with fixed size DJ-packets where the attacker's interception success is as high as 25%. The second and the third bars show the HR schemes, which are representative of the prior work by De et al. [3]. HR:2,4,3 uses qubit reordering and has three consecutive DJ packets of 2, 4, and 3 qubits, where the attacker's interception dropped to 2.77%. HR:2,6,4,5,3 has more variable sized DJ-packets and hence more entropy, where the attacker's successful interception dropped to 2.0%, which is a 12.5-times drop compared to F:4. The fourth, fifth

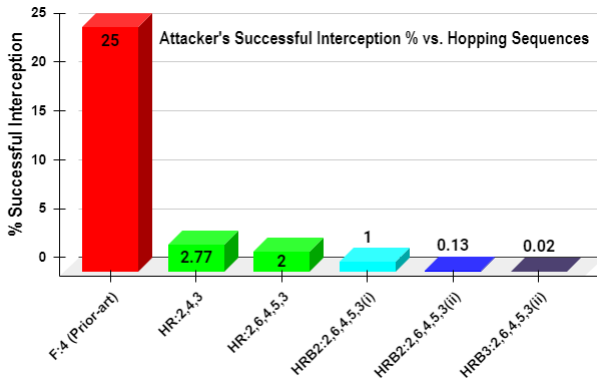


Figure 9. Bar chart comparing attacker's successful interception rates for Fixed, HR, and HRB schemes. Lower is better.

and the sixth bars show the new HRB scheme that combines size hopping, reordering, and multi-basis. The fourth bar uses two orthogonal bases and option-1 (HRB2:2,6,4,5,3(i)) where Eve's successful interception is 1%. The fifth bar also uses two orthogonal bases but is with option-2 (HRB2:2,6,4,5,3(ii)), where Eve's interception success drops further to 0.13%, which is 200-times lower than F:4. Hence, option-2 for multi-basis is much more effective than option-1. The sixth bar uses three orthogonal bases with option-2 (HRB3:2,6,4,5,3(ii)) and has the highest entropy. Eve's successful interception rate drastically drops to 0.02%, which is more than 1000-times lower than F:4.

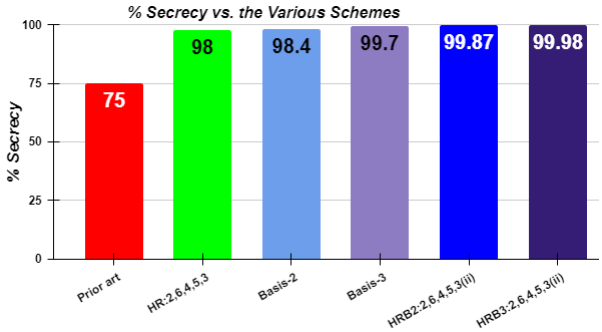


Figure 10. Bar chart comparing secrecy for Fixed (prior art), HR, Fixed-Basis, and HRB schemes. Higher is better.

Figure 10 shows that secrecy increases as the entropy through DJ-packet size-diversity, reordering and multi-basis increase. Basis-2 (secrecy=98.4%) and Basis-3 (secrecy=99.7%) are with fixed sized packets without reordering, but they use two and three different orthogonal bases, respectively. They show secrecy can be better from just using multiple orthogonal bases, rather than hopping/reordering as in HR scheme (98%) [3]. The best secrecy is achieved for

 ISBN: 978-1-61208-961-4

HRB3:2,6,4,5,3(ii) (99.98%) that uses three different bases, the next best is HRB2:2,6,4,5,3(ii) (99.87%) that uses two different orthogonal bases. These tests compared results among the use of one, two and three bases, where the secrecy improves from 75% for one basis, to 98.4% for two bases, and to 99.7% for three bases. This shows a gradually flattening increase in secrecy with further increase in the number of bases. However, the complexity of the hardware implementation increases with the number of bases due to increase in quantum gates needed. Hence, a trade-off should be done for the secrecy needed to thwart the existing interception threat on the quantum communication channel versus the number of bases needed to attain the secrecy requirements.

B. Secrecy and efficiency comparison with BB84

A probabilistic comparison is provided between BB84 and the HRB scheme. Each qubit in BB84 can be in one of the four different states and maps to a binary bit, which results in 25% successful interception probability assuming equally likely presence of the four states. For comparison, we select the HRB scheme HRB2:2,6,4,5,3(ii) that has five DJ-packets of sizes 2, 6, 4, 5, and 3 qubits, with a total of 20 ($H = 20$) qubits in the hopping sequence. There is only one DJ-packet ($P = 1$) that matches the attacker scan window size ($M = 4$ qubits). In HRB2:2,6,4,5,3(ii) each qubit can be using one of the two orthogonal bases ($B = 2$). The probability for the attacker matching the specific orthogonal basis, out of 'B' possible orthogonal bases, for all the M qubits in the DJ-packet is $(1/B)^M$. The probability of matching the reordered qubits is $1/M$. The probability of matching the DJ-packet boundary is (P/H). Hence, the total probability of successful interception is $= (P/H) * (1/M) * (1/B)^M = (1/20) * (1/4) * (1/2)^4 * 100\% = 0.078\%$, and is much lower than the BB84 protocol. If three orthogonal bases (or distinct rotation angles) are used, the probability of the attacker's successful interception is theoretically reduced to 0.015%. Unlike BB84 [6] [7], the HRB scheme does not suffer from the problem of orthogonal basis mismatch between Alice and Bob. This fact, together with the high entropy, makes the HRB scheme more secure than some of the BB84 based QKD.

V. THE COMMUNICATION FRAMEWORK

Alice, Bob and all participants using this technology have the list of all possible HRB schemes as a part of the system software available with their computing system equipped to perform QKD. As may be occasionally needed, the list can be updated as a software update to their computing system.

A. Setting up the HRB scheme before communication

Whenever QKD or secure messaging is needed, the specific HRB scheme to be used is first communicated, shown as 'step 0' in Figure 11. It can be done by a few possible approaches. One such approach is to use the BB84 to randomly select and communicate an N-bit (e.g., N=8,12,16) value that will indicate the index of the HRB scheme within the list (e.g., 8-bits when list size $\leq$ 256, 12-bits when $\leq$ 4,096, 16-bits when $\leq$ 65,536) of all predefined HRB schemes. BB84 is only used for sharing the index of the HRB scheme secretly. Once the HRB scheme for this session is communicated, both Bob and Alice loads the relevant portions of the quantum circuitry (as shown in Figure 7) for the particular HRB scheme into the quantum hardware. The actual 128 bit key is then shared securely by the HRB scheme, as shown in 'step 1' in Figure 11. This strategy is used since the secrecy achieved by HRB DJ-algorithm is much more than BB84, as discussed in Section IV.D. Furthermore, BB84 has a statistical rejection rate of 50% for the shared secret due to Alice and Bob's random mismatch in basis. This problem of BB84 is now limited to only sharing the list index with small number of bits (between 8 to 16) instead of all the 128 bits for the secret key. The combination of BB84 for the initial step (step 0) to communicate the specific HRB scheme index and then using the HRB scheme for the actual QKD or secure messaging ('step 1') as in Figure 11 leads to an overall improved secrecy and effectiveness than using just BB84.

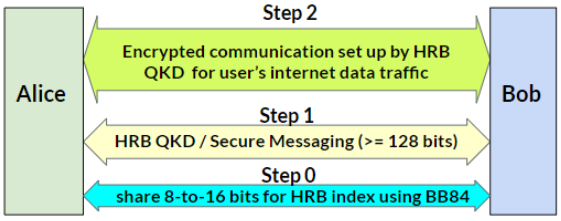


Figure 11. The HRB QKD framework showing 'step 0' for communicating the HRB scheme index. The 'step 1' can do QKD or secure messaging

B. Secure short messages over quantum channel

With a specific HRB scheme set up in 'step 0' as in Figure 11, it is possible to directly send secure short messages in 'step 1' over the quantum channel by encoding the message as a specific sequence of constant or balanced oracles. This is not possible by most other QKD mechanisms as the secret bits are randomly generated with high chances of rejection due to mismatch at the two end points (e.g., basis mismatch between Bob and Alice in BB84). Step 2 is unused in this case.

C. Detecting attacker's interceptions and actions thereafter

Any DJ-packet for which Alice detected interception is discarded. Alice notifies Bob of the sequence numbers of the DJ-packets to discard over the classical communication channel. Bob reconstructs the 128 bit shared secret key by throwing away the discarded DJ-packet sequence numbers sent by Alice. The attacker can still intercept these sequence numbers, but it is irrelevant since those indicate discarded DJ-packets by Alice and Bob. Thus, the actual shared M-bit (e.g., $M = 128$) key stays secret between Alice and Bob. Alice measures the input qubits in the DJ-packet received from Bob, computes to determine if Bob used a constant or balanced U_f, and updates the result in the target qubit, forming the output DJ-packet. The qubit states of the output DJ-packet are then compared with the expected qubit states for the same sized DJ-packets. Alice detects an interception if the states of one or more qubits in the output DJ-packet do not match

 ISBN: 978-1-61208-961-4

the states in any of the expected output DJ-packets. Figure 12 shows the flowchart for detecting interception. Table I shows the Cirq code for some constant and balanced oracles. The

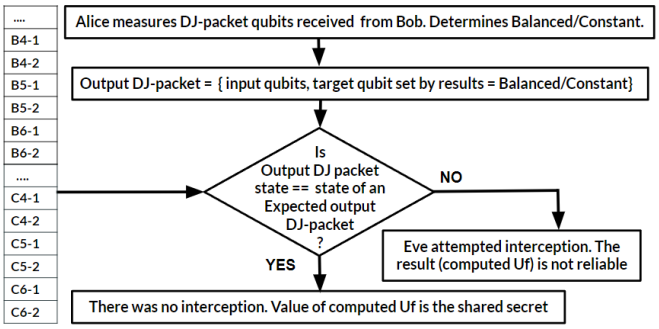


Figure 12. The flowchart for detecting interception by Alice

TABLE I. EXPECTED OUTPUT DJ-PACKETS FOR 4-QUBIT ORACLES

U_f: Example 4-qubit oracle (q0, q1, q2, q3) forms	Expected output DJ-packet
`[ ]`	Constant `C4-1`
`[cirq.X(q3)]`	Constant `C4-2`
`[cirq.CNOT(q0, q3), cirq.CNOT(q1, q3), cirq.CNOT(q2, q3)]`	Balanced `B4-1`
`[cirq.CNOT(q0, q3), cirq.CNOT(q1, q3), cirq.CNOT(q2, q3), cirq.X(q3)]`	Balanced `B4-2`

suffix "-1" or "-2" indicate the different expected output DJ-packets for all possible forms of the constant and the balanced oracles. Alice maintains a repository of expected output DJ-packets for different sized constant and balanced oracle forms. It is shown on the left side of Figure 12 as the list (.., B4-1,.., B6-2,.., C4-1,..,C6-2). As an example, prefixes C4 and B4 are for expected output DJ-packets with 4-qubits when Bob applied the constant and the balanced oracles, respectively. The number of expected output DJ-packets per DJ-packet size is a small finite number after the effects of reordering and multi-basis are removed.

VI. CONCLUSION

A novel way to tremendously increase the entropy of the DJ-packets communicated over the quantum channel is developed by employing different orthogonal basis for the qubits in the DJ-packets. Simulations showed that attacker's successful interception rate drops 200-times when using two orthogonal bases, and more than 1000-times with three orthogonal bases vs. prior work. This framework can be used for QKD and also for secure messages due to the very high secrecy ($\geq 99.98\%$) it provides, and also because it sets up a new HRB scheme for every new session. Hence, this work enhanced communication secrecy and broadened the scope compared to the earlier published works [3] [6]- [10].

Future work needs to evaluate different HRB schemes on real quantum hardware and perform trade-offs on HRB quantum circuit size vs. sustainability to decoherence effects and noisy channels. These studies can also help determine the need for dynamic selection of the HRB schemes of different secrecy levels depending on the existing level of threat on the quantum communication channel from a MITM attacker or an interceptor. If an increased interception rate is detected by Bob or Alice, they can decide to select a HRB scheme with an even higher secrecy, but at the cost of increased quantum circuit size and qubit requirements. Alternatively, if Alice or Bob finds zero interception, then they can decide to use a HRB scheme of reduced secrecy level so as to reduce the quantum circuit size and the number of qubits required. Finally, this research can also provide a foundation for interested readers to learn more about how quantum computing and quantum communications impact cybersecurity.

ACKNOWLEDGMENT

The author would like to immensely thank Mr. Jeremy Juybari, Mr. Colton Beery, and late Dr. Raymond Moberly of Faster Logic LLC; Dr. Kyle Sundqvist of the Physics Department at San Diego State University; for their support on the early research foundations for this project. Many thanks to Ms. Jo Buehler, Rohit's High School Calculus teacher, for her encouragement and support.

REFERENCES

[1] D. Deutsch and R. Jozsa, "Rapid Solutions of Problems by Quantum Computation," Proceedings of the Royal Society of London A.439, pp.553–558, Dec. 1992. doi:10.1098/rspa.1992.0167.

[2] K. Nagata and T. Nakamura, "The Deutsch-Jozsa Algorithm Can Be Used for Quantum Key Distribution," Open Access Library Journal Vol.2, No.8, Aug. 2015. doi:10.4236/oalib.1101798

[3] R. De, R. Moberly, C. Beery, J. Juybari and K. Sundqvist, "Multi-Qubit Size-Hopping Deutsch-Jozsa Algorithm with Qubit Reordering for Secure Quantum Key Distribution," in 2021 IEEE International Conference on Quantum Computing and Engineering (QCE), Broomfield, CO, USA, pp. 473-474, 2021. doi:10.1109/QCE52317.2021.00084

[4] A. Ananthaswamy, "The Quantum Internet Is Emerging, One Experiment at a Time," Scientific American, June 2019.

[5] A. Ho and D. Bacon, "Announcing Cirq: An Open Source Framework for NISQ Algorithms," Google AI Blog, July 2018.

[6] C. H. Bennett, G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," Theoretical Computer Science, Volume 560, Part 1, 2014, pp.7-11, ISSN 0304-3975, doi:10.1016/j.tcs.2014.05.025.

[7] V. Scarani, H. Bechmann-Pasquinucci, N. J. Cerf, M. Dušek, N. Lütkenhaus, M. Peev, "The security of practical quantum key distribution," Rev. Mod. Phys. 81 (3): pp.1301–1350, 2009.

[8] A. Ekert, "Quantum cryptography based on Bell's theorem," Physical Review Letters. American Physical Society. 67 (6): pp.661–663, 1991.

[9] H. Bechmann-Pasquinucci and N. Gisin, "Incoherent and coherent eavesdropping in the six-state protocol of quantum cryptography," Physical Review A 59: pp.4238-4248, 1999.

[10] H. Lo, X. Ma, and K. Chen, "Decoy State Quantum Key Distribution," Physical Review Letters. American Physical Society (APS). 94 (23): 230504, June 2005. doi:10.1103/PhysRevLett.94.230504.

[11] A. P. Bhatt and A. Sharma, "Quantum Cryptography for Internet of Things Security," Journal of Electronic Science and Technology, Volume 17, Issue 3,2019, pp 213-220, ISSN 1674-862X.

[12] W. K. Wootters and W. H. Zurek, "A single quantum cannot be cloned," Nature, vol. 299, no. 5886, pp. 802–803, 1982.

[13] G. L. Long and Y. Sun, "Efficient scheme for initializing a quantum register with an arbitrary superposed state," Physical Review A, vol. 64, no. 1, 2001. doi:10.1103/PhysRevA.64.014303.

[14] Y. Fei, X. Meng, M. Gao, H. Wang, and Z. Ma, "Quantum man-in-the-middle attack on the calibration process of quantum key distribution," Sci Rep 8, 4283, 2018.

[15] D. R. Kuhn, "Vulnerabilities in Quantum Key Distribution Protocols," NISTIR 6977, May 2003, NIST.

[16] Learn Quantum Computation using Qiskit, https://qiskit.org/textbook/ch-gates/more-circuit-identities.html: May, 2022

 ISBN: 978-1-61208-961-4

Secure Authorization for RESTful HPC Access

Mohammad Hossein Biniaz
Computing
Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen
Göttingen, Germany
E-Mail: mohammad-hossein.biniaz@gwdg.de

Sven Bingert
eScience
Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen
Göttingen, Germany
E-Mail: sven.bingert@gwdg.de

Christian Köhler
Computing
Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen
Göttingen, Germany
E-Mail: christian.koehler@gwdg.de

Hendrik Nolte
Computing
Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen
Göttingen, Germany
E-Mail: hendrik.nolte@gwdg.de

Julian Kunkel
Computing
Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen/Universität Göttingen
Göttingen, Germany
E-Mail: julian.kunkel@gwdg.de

***Abstract*—The integration of external services, such as workflow management systems, with High-Performance Computing (HPC) systems and cloud resources requires flexible interaction methods that go beyond the classical remote interactive shell session. In a previous work, we proposed the architecture and prototypical implementation of an Application Programming Interface (API) which exposes a Representational State Transfer (REST) interface, which clients can use to manage their HPC environment, transfer data, as well as submit and track batch jobs. In the present article, we expand on this foundation by integrating a fine-grained role-based authorization and authentication system, which facilitates the initial setup and increases the user's control over the jobs that services intend to submit on their behalf. The developed *HPCSerA* service provides secure means across multiple sites and systems and can be utilized for one-off code execution and repetitive automated tasks.**

***Index Terms*—HPC, automation, RESTful API, OAuth, authorization, web interface.**

I. Introduction

Due to the increasing demand on computing power driven by the success of resource-intensive methods in various scientific domains, there is an equally increasing requirement by researchers to utilize HPC resources to satisfy their demand in a cost-effective manner. This has led to the creation of different services, which for instance expose a RESTful API, with which users can remotely interact with an HPC system. There are numerous different use cases for such a requirement. One motivating example can be the ability to manage complex and compute intensive workflows with a graphical user interface to improve usability for inexperienced users [1].

While, one one hand, there are these efforts to ease and open up the use of HPC systems, there is, on the other hand, a constant threat by hackers. Since users typically interact with the host operating system of an HPC system directly, local vulnerabilities can be immediately exploited. Two of the most favored attacks by outsiders are brute-force attacks against a password system [2] and probe-based login attacks [3]. These attacks, of course, become obsolete if attackers can find easier access to user credentials. Therefore, it is of utmost importance to keep access, and access credentials, to HPC systems safe.

In this context, services easing the use of and the access to HPC systems should be treated with caution. For example, if access via Secure Shell (SSH) [4] to an HPC system is only possible using *SSH keys* due to security concerns, these measures are rendered ineffective if users re-establish a password-based authentication mechanism by deploying a RESTful service on the HPC system that is exposed on the Internet. Observing these developments, it becomes obvious that there is a requirement to offer a RESTful service to manage data and processes on HPC systems remotely which is comfortable enough in its usage to discourage concocted and insecure solutions built by inexperienced users with the main objective of "getting it to work", but which adheres to the **highest security standards**.

The *key contributions* of this article are:

1) analysis of possible attack scenarios based on a RESTful service running on an HPC system;
2) presentation of a state-of-the-art REST API design, called *HPCSerA*, to secure the RESTful service;
3) discussion of the usability utilizing explicit use cases.

A *REST Service*, i.e., a web application, is typically deployed in a suitable cloud environment. User requests for code execution on the HPC system are generated manually or automatically and then sent by a *Client* to this *REST Service*. In order to execute the requested task, an *Agent* is deployed on an HPC system that retrieves the tasks and executes them, for instance by submitting a job on the cluster via the batch system.

The remainder of this paper is structured as follows: In Section II, the related work is presented, including state-of-the-art mechanisms to solve this issue. In Section III, existing security issues preventing a wide-spread application of *HPCSerA* are being discussed and an improved architecture

 ISBN: 978-1-61208-961-4

with a security-based scope definition is presented. In the following Section IV, our implementation is presented. At the end, a diverse set of use cases are presented in Section V, as well as a concluding discussion, which is provided in Section VI.

II. Related Work

There is without question a general trend towards remote access for HPC systems, for instance in order to use web portals instead of terminals [5]. These applications actually have a long standing history with the first example of a web page remotely accessing an HPC system via a graphical user interface dating back to 1998 [6].

Newer approaches are the *NEWT* platform [7], which offers a RESTFul API in front of an HPC system and is designed to be extensible: It uses a pluggable authentication model, where different mechanisms like Open-Authorization (OAuth), Leightweight Directory Access Protocol (LDAP) or *Shibboleth* can be used. After authentication via the */auth* endpoint, a user gets a cookie which is then used for further access. With this mechanism *NEWT* forwards the security responsibility to external services and does not guarantee a secure deployment on its own. This has the disadvantage, that *NEWT* is not intrinsically safe, therefore providers of an HPC-system need to trust the provider of a *NEWT* service, that it is configured in a secure manner. Additionally, no security taxonomy is provided, which is key when balancing security concerns and usability.

Similarly, *FirecREST* [8] aims to provide a REST API interface for HPC systems. Here, the Identity and Access Management is outsourced as well, in this case to *Keycloak*, which offers different security measures. In order to grant access to the actual HPC resources after successful authentication and authorization, a *SSH certificate* is created and stored at a the *FirecREST* microservice. Although this is a sophisticated mechanism, there seem to be a few drawbacks. First of all, the *sshd* server must be accordingly configured to support this workflow, secondly it remains unclear how reliable status updates about the jobs can be continuously queried when using short-lived certificates, and lastly these certificates needs to be stored at a remote location, which might conflict with the terms of service of the data center of the user. Additionally, HPC systems are often configured to allow logins from a trusted network only, which means that the *FirecREST* microservice can not serve multiple HPC systems at a time.

While the *Slurm Workload Manager* provides a REST interface that exposes the cluster state and in particular allows the submission of batch jobs, the responsible daemon is explicitly designed to not be internet-facing [9] and instead is intended for integration with a trusted client. Its ability to generate JSON Web Token (JWT) tokens for authentication provides an interesting alternative route for interaction with our architecture, provided both services are hosted in conjunction. Clients that shall execute Slurm jobs authenticate the trusted Slurm controller via the *MUNGE* service [10] that relies on a shared secret between client and server. If either of these is compromised, then it is assumed that the whole cluster is insecure. Slurm can be deployed across multiple systems and administrative sites and there are various options for Slurm to support a meta-scheduling scenario or federation. However, if the Slurm controller is compromised, it can dispatch arbitrary jobs to any of the connected compute systems. In addition, decoupling the API implementation from the choice of the job scheduler, as we propose, allows interoperation of multiple sites, possibly using different schedulers.

An alternative execution model popular with public cloud systems is Function-as-a-Service (FaaS). In this model, a platform for execution of functions is provided, i.e., code can be submitted by the user and execution of the function with parameters are triggered via an exposed endpoint. A runtime system executes the function in an isolated container and automatically scales up the number of containers according to the response time and number of incoming requests. Customers are billed for the execution time of the function. The core assumption is that the function is a sensible unit of work, e.g., running for 100ms, running on a single core, side-effect free, and thus only suitable for embarrassingly parallel workloads. Authentication and security is of high importance for these systems as well. For example, OpenFaaS is a Kubernetes-based FaaS system that utilizes, e.g., OAuth to authorize users and to generate tokens that are verified upon function deployment or execution. While this mechanism has similarities to our approach, FaaS is for short-running (subsecond to several second) single node jobs, we provide different, security-derived authorization processes for the different available operations, while mitigating user impact via push notifications and solve the issue for long-running HPC systems including parallel jobs.

III. Architecture

We first analyze the potential security issues from our initial architecture and describe an approach to address them via an updated authorization and authentication process. Finally, each step of the revised workflow is discussed individually.

A. Problem statement

In the original architecture, static *bearer tokens* were used for user authentication. There was one *bearer token* per user, which means that each client, as well as each agent authenticated towards *HPCSerA* with the same token, compare [11, III. B.]. Although considered state-of-the-art, this approach has different security flaws, which prevented a public deployment. These security problems become apparent, when particularly taking into account that an access mechanism for an HPC system is provided. One problem is that this single *bearer token* can be used to access all endpoints, which means that it can be used to perform any possible operation. This can be maliciously exploited in two different ways:

- If that token is not properly guarded, an attacker can use it to post a malicious job, to gain direct access to the HPC system.

 ISBN: 978-1-61208-961-4

- If an attacker has escalated their privileges, the token used by the agent is left vulnerable. If the user has authorized that token to get access to more than one HPC system, the attacker has immediately gained access to another cluster.

There are two different conclusions one can deduce from these observations: First, it is a highly vulnerable step to allow code ingestion via a RESTful service into an HPC system and one has to take the chance of a token loss into account, when designing such a system. Second, the agent sometimes only needs the permissions to read queued jobs and to update the state of a job, e.g., from *queued* to *running*. It is, therefore, an unnecessary risk to allow a job ingress from the token of an agent.

B. Improved Architecture

The separation of access tokens by the user who created them and the services (clients and HPC agents) to which they are deployed, as described in [11], already enables revoking trust in a setup with multiple services and multiple backend HPC systems easily. However, during operation, there is global access to the entire state, i.e., in-flight jobs, to all parties involved. In order to segment trust between groups of services and HPC backends, our revised architecture (cf. Figure 1) resolves this issue by introducing a dedicated tag field into the design of the database for access tokens. Based on this information, client services and HPC agents can be authorized individually. Moreover, each token can be assigned one or multiple roles that restrict the combination of Hypertext Transfer Protocol (HTTP) endpoints and verbs which can be used for all entities that have been created using the same tag. The token's individual lifetime is implied by the granted role.

User control over each individual task and job that is allowed to be run or submitted, respectively, is enforced by introducing an intermediate authentication step that requires user interaction via an external application. This could be run on a mobile device or hardware token, like the ones being used for two-factor authentication or integrated into the web-based user interface used for token and device management for fast iterations on the workflow configuration. Metadata about the action to be authorized is included in the user prompt in order to allow an informed decision. However, the measure is restricted to this most critical step of the process, while non-critical endpoints, such as retrieving the state of pending jobs, can continue to respond immediately. For submitting a new job, the necessity of individual user confirmation is also determined by whether new code is ingested or an already existing job is merely triggered to run on new input data.

From the user's perspective, setting up the workflow would start with logging into the web interface and creating tokens for each service to be connected to the API and configuring them in each client and agent, respectively. In order to acquire a minimal working setup, at least one token for the client service and one for the agent communicating to the batch system on the HPC backend system would be required. OAuth compatible clients could initiate this step externally, thereby sidestepping the need for the user to manually transfer the token to each client configuration. As soon as each client has acquired the credentials either way, HPC jobs can be relayed between each service and the HPC agent.

While the OAuth 2.0 terminology [12] allows a distinction between an authorization server which is responsible for granting authorization and creating access tokens, and a resource server which represents control over the entities exposed by the API, in our case the tasks and batch jobs to be run, both roles are assumed by our architecture, so the design can be as simple as possible and deployed in a single step. However, since the endpoints for acquiring access tokens and the original endpoints that require these access tokens are distinct, a separation into microservices (which again need to be authenticated against each other) would also be compatible with the presented design.

The steps necessary for code execution are illustrated in Figure 1. As a preliminary, we assume that the HPC agent is set up and configured with the REST service as an endpoint. The arrows indicate the interactions and the initiator. The individual steps are as follows:

1) The workflow starts by a user logging into the web interface. The Single sign-on (SSO) authentication used for this purpose has to be trusted, since forging the user's identity could allow an attacker to subsequently authorize a malicious client to ingest arbitrary jobs.
2) The user can create tokens for the the REST service in the WebUI.
3) The tokens are stored in the Token database (DB), along with the granted role, project tag and token lifetime.
4) The retrieved tokens can then be used by a client, e.g., to run some code on the HPC system or have an automatic process in place, provided the code is already present on the system, rendering manual authentication unnecessary.
5) The request is forwarded to the REST Service, which verifies the information in the Token DB. On success, the code to execute is forwarded to the HPC agent.
6) If the client chooses to use the OAuth flow instead in order to avoid manual token creation, the authorization request is forwarded to the Auth app instead.
7) The user can choose to confirm or deny the authorization request. In the former case, the generated token is stored (cf. 3) in the Token DB. Again, further requests can then in general proceed via step 5 without further user interaction.
8) Like any other client, the HPC agent uses a predefined token or alternatively initiates the OAuth flow in order to get access to the submitted jobs.
9) For the most critical task of executing code on the HPC frontend or submitting batch jobs, the agent can be configured to get consent from the user by using the Auth app for authentication.
   This request is accompanied by metadata about the job to be executed, such as a hash of the job script, allowing an informed decision by the user. This step also avoids the need for trust in a shared infrastructure,

since the authentication part can be hosted by each site individually.

10) Once the user confirmed execution, the HPC agent executes the code, e.g., by submitting it via the batch system. In this case, information about the internal job status is reported back to HPCSerA.

We assume that the HPC agent is secure as otherwise the system and user account it runs on are compromised and, hence, could execute arbitrary code via the batch system anyway. The Web-based User Interface (WebUI), HPC agent, HPCSerA Service and Client are all independent components. For example, a compromised REST Service could try to provide arbitrary code to the HPC agent anytime or manipulate the user's instructions submitted via the client. However, as the user will be presented with the code via the authenticator app and can verify it similarly to a 2 Factor Authentication (2FA), the risk is minimized.

There are multiple approaches to deploy HPCSerA across multiple clusters and administrative domains:

a) Replication: Each center could deploy the whole HPCSerA infrastructure which we develop (cf. Figure 1) independently maximizing security and trust. By adjusting the endpoint URL, a user could connect via the identical client to either the REST service at one or another data center – this is identical to the URL endpoints in S3. Although the user now has two independent WebUIs for confirming code execution on the respective data center, the authenticator and the identity manager behind it could be shared. An additional advantage of this setup would be that the versions of HPCSerA deployed at each center could differ.

b) Shared infrastructure: The maximum shared configuration would be that for each HPC system a user has to deploy a dedicated HPC agent on an accessible node but all the other components are only deployed once.As the HPC agents register themselves with the REST service, now the user can decide at which center they would like to execute any submitted code. While using a single WebUI for many centers and cloud deployments maximizes usability, it requires the highest level of trust in the core infrastructure: If two of these components are compromised, arbitrary code can be executed on a large number of systems.

IV. Implementation

In the following, more details about the technologies chosen for our implementation are provided. Due to the conceptualized architecture in Section III, this section has a focus on the current scope definition and the authentication/authorization scheme employed. Generally, the OpenAPI 3.0 specification [13] was used to define the RESTful API, which is a language-agnostic API-first standard used for documenting and describing an API along with its endpoints, operations, request- and response-definitions as well as their security schemes and scopes for each endpoint in *YAML* format. This API is backed by a *FLASK*-based web application written in *Python*. The token database is in a SQL-compatible format, thus SQLite can be used for development and, e.g., PostgreSQL for the production deployment. The database schema contains only the user (`user_id`) and project (`project_id`) that the token belongs to as well as the individual permission-level (`token_scope`).

A. Definition of Access Roles

In order to give granular permissions for accessing each of the endpoints, OpenAPI 3.0 allows to define multiple security schemes providing different scopes to define a token matching to the security level of each of the endpoints. Eight different roles have been identified, which are listed and described in Table I.

These roles are entirely orthogonal, which means they can be combined as necessary. If, for instance, on one HPC system only parameterized jobs needs to be submitted, the *agent* can be provided with a token which has only the permissions of role 2 and 3, thus lacking role 5, which is required to fetch new files. Similarly, if a token is provided to a client which is not 100% trustworthy, one can choose to only provide a token with the role 6, i.e., to only allow to trigger a predefined job. Important to understand is the difference in mistrust between the role 3, 4, and 5. The security mistrust in role 4 comes from the admins of the *HPCSerA*, which want to ensure that a code ingestion is indeed done by the legitimate user. Therefore, in order to allow code ingestion, the possession of a token with the corresponding permission is not enough, the user has to confirm the code ingestion via a 2FA. The mistrust in role 3 and 5 comes, however, from the user, who wants to ensure that only jobs s/he confirmed are being executed. This is, again, completely orthogonal, to the enforced 2FA in role 4 and can be optionally used by the user. This fine-grained differentiation between the different security implications of the discussed endpoints, minimize user interference while providing a high level of trust.

B. Providing Tokens via Decoupled OAuth

The introduction of OAuth-compatible API endpoints has several advantages: Access tokens can be created on demand in a workflow initiated by a client or HPC agent, respectively. In addition, while there is a default API client provided, a standard-compliant API enables users to easily develop drop-in replacements.

It is important to note here that we modified the usual OAuth flow, where a client gets redirected to the corresponding login page to authorize the client. This "redirect approach" has two problems:

- The client is a weak link, where the Transport Layer Security (TLS) encryption is terminated and therefore becomes susceptible to attacks and manipulation.
- It does not support a headless application, like the HPC agent, which is not able to properly forward the redirect to the user.

Due to these shortcomings, a modified OAuth flow was developed to enable the usage of headless apps and improve security. This modified version decouples the user confirmation from the client, which means that the client is not

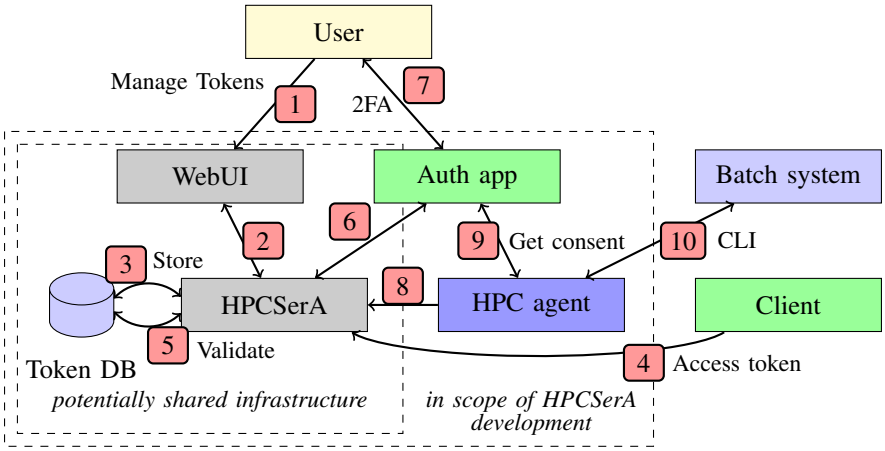


Fig. 1. A sketch of the proposed token-based authorization flow. The following parts are shown: 1) WebUI login 2) Connection to the HPCSerA service 3) Storage of access tokens 4) Client connecting to the API 5) Validation of access tokens 6) Authorization request 7) User interaction with the Auth app 8) HPC agent connecting to the API 9) Authentication request for code execution 10) Interaction with the HPC batch system

TABLE I
DEFINITION OF THE EIGHT ROLES. OPERATIONS MARKED IN RED HAVE TO BE CONSIDERED SECURITY CRITICAL FROM THE ADMIN POINT OF VIEW, WHEREAS THE ORANGE MARKED OPERATIONS FROM A USER POINT OF VIEW.

Role Number	Role	Description
1	GET_JobStatus	Client can retrieve information about a submitted job
2	UPDATE_JobStatus	Used by client/agent to update the job status
3	GET_Job	Endpoint used by the agent to retrieve job information
4	POST_Code	Client to ingest new code to the HPC sytsem
5	GET_Code	Agent pulls new code. Might be necessary to run new job
6	POST_Job	Client triggers parameterized job
7	UPDATE_Job	Client updates already triggered job
8	DELETE_Job	Client deletes already triggered job

being redirected but that the confirmation request is being sent out-of-band, e.g., via the WebUI or via notification on a smartphone device.

Starting with the case that the script does not already come equipped with a token, analogous to the usual OAuth flow, the generation of a token is requested. Since our use case was initially built as an instance of machine-to-machine interaction, i.e., headless, the issue of a lack of user interface is encountered; the usual OAuth flow - implemented in the browser - would redirect the user to an authorization server where the user could actively provide their username and password to the authorization server. The authorization server would then return a code, in the case of the authorization code flow, in the redirect URI which would be posted in a backchannel along with a client secret assigned at the time of registering the client to attain an access token.

In order to circumvent this headless-app problem, this work has implemented a synchronous push notification system analogous to the Google prompt where a notification is pushed to a user's device awaiting a confirmation to proceed. In the Minimum Viable Product (MVP), we have implemented this in the SSO-secured WebUI in order to have a more integrated interface. Eventually, the final product will see an Android and iOS app that receives such notifications. This flow then grants the permission to execute a security critical operation, compare Table I.

This confirmation via push notification cannot solely rely on time-synchronicity since it would be susceptible to an attacker requesting tokens and/or 2FA confirmation for carrying out a security-critical operation in the same approximate time frame. Therefore, a sender constraint has to be implemented. This is done in a similar way to the original authorization code flow: The access code is signed with a client secret, which was configured with *HPCSerA* prior to the execution of this workflow, and then sent to *HPCSerA*. *HPCSerA* verifies the secret and only then sends the actual token. This secret is implemented using public-private key pairs, where the public key is uploaded to *HPCSerA* in the initial setup to register a new client (or agent).

Alternatively, in the case that a token is supplied along with the software or script that is submitting a job to the HPCSerA API, the permissions are validated against a token database. In the case that the token provided contains permissions for accessing a sensitive endpoint, the second factor check is triggered through the WebUI and the notification / confirmation process is once again undergone. It is important to note that this is not a hindrance since already-running jobs and non-sensitive endpoints proceed without user-intervention.

V. USE-CASES

Due to the previously stated changes in the architecture, there are certain adaptions in the previously presented use

 ISBN: 978-1-61208-961-4

cases [11]. These changes will be discussed in the following and serve as the basis for a broader user impact analysis.

A. GitLab CI/CD

Since the *GitLab* Runner can be configured to run arbitrary code without including secrets in the repository, thanks to GitLab's project Continuous Integration and Integration Development (CI/CD) variables [14], the required tokens can be made available to the CI/CD job so it can in turn access the API endpoints required to transmit the current repository state to an HPC system where the code can be tested using the HPC software environment or even multiple compute nodes.

A new commit might of course introduce arbitrary code to the HPC environment, therefore it is advisable to enforce the extra authentication step (cf. Section III-B), when code from a new commit is submitted to the HPC system. The corresponding hash, available by default via the `GIT_COMMIT_SHA` variable, would be a helpful piece of information to display to the user when asking to authorize the request.

B. Workflow Engine

In the workflow use case, HPC jobs should be fully automated without user interaction. Due to multiple repetitions and time dependencies, interactions severely limit the functionality and practicability of the workflow. One possibility is to prepare the workflow in such a way that only parameterized jobs are called and thus only safe endpoints of HPCSerA are used. Another possibility is to use dedicated (legacy) endpoints that are only accessible through firewall regulations and fixed network areas. The latter can also be regulated via an additional proxy server, such as a *nginx*.

C. Data Lake

In order to provide high performance computing capabilities to a data lake [15], *HPCSerA* is used to submit jobs on behalf of the data lake users. A user sends a so-called *Job Manifest* to the data lake, where the software, the compute command, the environment, and the input data are unambiguously specified. By transferring the responsibility of scheduling the job from the user to the data lake, it has the control about it. This allows to reliably capture the data lineage and to foster reproducibility. The added benefit of the newly implemented security measures in *HPCSerA* is that, before, users had to trust the data lake, and hereby the admins, with their *bearer tokens*. By introducing *OAuth* and enforcing a 2FA for code ingestion, this is not necessary anymore, since users now need to confirm each submission. Since users submit jobs actively, for instance via a *Jupyter Notebook* using a PythonSDK, the requirement to confirm each submission does interrupt the workflow too much.

VI. Conclusion

In the paper presented here, we have examined the issue of security in accessing HPC resources via a RESTful API. The initial situation with a very simplified token model does not meet the requirements. Therefore, a fine-granular token model, coupled with interactive user consent and OAuth flows, was proposed. With this new model, particularly critical interactions, such as code transfer, can be secured. User consent is requested in a prototype via a WebUI, which in turn uses a central Identity Management (IDM) for authentication. This means that no critical user-specific data needs to be managed.

In future work, the possibilities for obtaining user consent will be further analyzed. The development of mobile apps is planned, which will greatly simplify the consent workflow for the user. So far, the focus has been on the transmission and execution of code. However, there is also a requirement to transmit data objects that are necessary for execution. Therefore, it is examined to what extent the current implementation is suitable for such tasks and where possible limits are reached in terms of data quantity and transmission speed.

Acknowledgments

We gratefully acknowledge funding by the "Niedersachsisches Vorab" funding line of the Volkswagen Foundation and "Nationales Hochleistungsrechnen" (NHR).

References

[1] Z. Wang et al., "RS-YABI: A workflow system for Remote Sensing Processing in AusCover," in *Proceedings of the 19th International Congress on Modelling and Simulation*. MODSIM 2011 - 19th International Congress on Modelling and Simulation - Sustaining Our Future: Understanding and Living with Uncertainty, 2011, pp. 1167–1173.

[2] A. K. Singh and S. D. Sharma, "High Performance Computing (HPC) Data Center for Information as a Service (IaaS) Security Checklist: Cloud Data Governance." *Webology*, vol. 16, no. 2, pp. 83–96, 2019.

[3] J.-K. Lee, S.-J. Kim, and T. Hong, "Brute-force Attacks Analysis against SSH in HPC Multi-user Service Environment," *Indian Journal of Science and Technology*, vol. 9, no. 24, pp. 1–4, 2016.

[4] T. Ylonen, "SSH - Secure Login Connections Over the Internet," in *Proceedings of the 6th USENIX Security Symposium (USENIX Security 96)*. San Jose, CA: USENIX Association, Jul. 1996, pp. 37–42, [accessed: 2022-03-21]. [Online]. Available: https://www.usenix.org/conference/6th-usenix-security-symposium/ssh-secure-login-connections-over-internet

[5] P. Calegari, M. Levrier, and P. Balczyński, "Web portals for high-performance computing: a survey," *ACM Transactions on the Web (TWEB)*, vol. 13, no. 1, pp. 1–36, 2019.

[6] R. Menolascino *et al.*, "A realistic UMTS planning exercise," in *Proc. 3 ACTS Mobile Communications Summit 98*, 1998.

[7] S. Cholia and T. Sun, "The newt platform: an extensible plugin framework for creating restful hpc apis," *Concurrency and Computation: Practice and Experience*, vol. 27, no. 16, pp. 4304–4317, 2015.

[8] F. A. Cruz *et al.*, "FirecREST: a RESTful API to HPC systems," in *2020 IEEE/ACM International Workshop on Interoperability of Supercomputing and Cloud Technologies (SuperCompCloud)*, 2020, pp. 21–26.

[9] SchedMD. (2022) Slurm REST API. [accessed: 2022-03-18]. [Online]. Available: https://slurm.schedmd.com/rest.html

[10] Chris Dunlap. (2022) MUNGE Uid 'N' Gid Emporium. [accessed: 2022-03-21]. [Online]. Available: https://dun.github.io/munge/

[11] S. Bingert, C. Köhler, H. Nolte, and W. Alamgir, "An API to Include HPC Resources in Workflow Systems," in *INFOCOMP 2021, The Eleventh International Conference on Advanced Communications and Computation*, C.-P. Rückemann, Ed., 2021, pp. 15–20.

[12] D. Hardt, "The OAuth 2.0 Authorization Framework," RFC 6749, Oct. 2012, [accessed: 2022-03-21]. [Online]. Available: https://www.rfc-editor.org/info/rfc6749

[13] OpenAPI Initiative. (2017) OpenAPI Specification v3.0.0. [accessed: 2022-03-21]. [Online]. Available: https://spec.openapis.org/oas/v3.0.0

[14] GitLab. (2022) GitLab CI/CD variables. [accessed: 2022-03-18]. [Online]. Available: https://docs.gitlab.com/ee/ci/variables/

[15] H. Nolte and P. Wieder, "Realising Data-Centric Scientific Workflows with Provenance-Capturing on Data Lakes," *Data Intelligence*, pp. 1–13, 03 2022. [Online]. Available: https://doi.org/10.1162/dint_a_00141

 ISBN: 978-1-61208-961-4

Procedural Component Framework Implementation and Realisation for Creation of a Coherent Multi-disciplinary Conceptual Knowledge-based Holocene-prehistoric Inventory of Volcanological Features Groups

Claus-Peter Rückemann
Westfälische Wilhelms-Universität Münster (WWU), Germany;
Unabhängiges Deutsches Institut für Multi-disziplinäre Forschung (DIMF), Germany;
Leibniz Universität Hannover, Germany
Email: ruckema@uni-muenster.de

***Abstract*—This paper presents the results of the procedural component framework implementation and realisation for creation of a coherent multi-disciplinary conceptual knowledge-based Holocene-prehistoric inventory of worldwide volcanological features groups. The goal is the creation of a sustainable framework of components, which can be employed for multi-disciplinary integration of knowledge contexts, especially from prehistory and archaeology. The component framework has to enable further coherent conceptual knowledge contextualisation and georeferenced symbolic representation. This paper provides the results on experiences of sustainable component integration and practical procedural implementations and realisations. Future research will address the creation of a component framework for a Holocene-prehistoric inventory of worldwide volcanological features, which enables coherent conceptual knowledge integration and contextualisation with prehistorical and archaeological knowledge resources.**

Keywords–Prehistory; Holocene; Knowledge-based Component Integration; CRI Framework; CKRI.

I. INTRODUCTION

Coherent conceptual knowledge resources are results of often complex and long-term multi-disciplinary creation processes. Coherent conceptual knowledge resources may have to achieve an advanced level of implementation before procedural components can be created for sustainably employing these resources. The conceptual knowledge implementation for this inventory is in focus of multi-disciplinary research groups and matter to be reported in separate publications. Motivation is the creation of a sustainable and practical component framework based on coherent multi-disciplinary conceptual knowledge.

This paper presents the results of the procedural component framework implementation and realisation for creation of a coherent multi-disciplinary conceptual knowledge-based Holocene-prehistoric inventory of worldwide volcanological features groups, which are employing respective coherent knowledge resources. The goal of this research is the creation of a sustainable framework of components, which can be employed for multi-disciplinary integration of knowledge contexts, especially from prehistory and archaeology, too. The component framework further has to enable a coherent conceptual knowledge contextualisation and georeferenced symbolic representation. The coherent knowledge resources and the practical realisation are fully based on the Component Reference Implementations (CRI) framework [1], which is employing the main implementations of the prehistory-protohistory and archaeology Conceptual Knowledge Reference Implementation (CKRI) [2]. CRI provides the required component groups and components for the implementation and realisation of all the procedural modules. CKRI provides the knowledge framework, including multi-disciplinary contexts of natural sciences and humanities [3]. Both provide sustainable fundaments for highest levels of reproducibility and standardisation and allow continuous and consistent further development of discipline-centric and multi-discipline development of knowledge resources. Both reference implementations are in continuous further development. The approach conforms with information science fundaments and universal knowledge and enables an integration of the required components from methodologies to realisations for knowledge representations of realia and abstract contexts [4], namely the Conceptual Knowledge Pattern Matching (CKPM) methodology, considering that many facets of knowledge, including prehistory, need to be continuously acquired and reviewed [5].

The rest of this paper is organised as follows. Section II presents the methodological implementation and realisation, workflow procedure, respective component reference implementation and integration and coherent conceptual knowledge implementation for the new inventory. Section III discusses the procedural potential regarding integration of components, parallelisation, and implementation features. Section IV summarises lessons learned, conclusions, and future work.

II. METHODOLOGICAL IMPLEMENTATION AND REALISATION

Implementation and realisation are based on the CKRI [2]. Components outside the core scope of this geoscientific, prehistoric, and archaeological research are employed and can be extended via the CRI frame [1]. The following implementation and realisation start with a description of a workflow procedure for creation of a coherent multi-disciplinary conceptual knowledge-based Holocene-prehistoric inventory of worldwide volcanological features groups, followed by the component implementation and realisation based on the general coherent multi-disciplinary conceptual knowledge implementation.

A. Methodological workflow procedure

A workflow procedure for the creation closely integrates the component framework and the coherent knowledge implementation of the Knowledge Resources (KR):

- (KR/components selection, continuous development.)
- Component implementation and realisation.
  - Scientific parametrisation of components (including algorithms, in each discipline).
  - Workflow decision making.
  - Country identification algorithm.
  - Country representation algorithm.
  - Area of Interest (AoI) representation algorithm.

 ISBN: 978-1-61208-961-4

    - Symbolic representation of country
    - Symbolic representation of AoI.
    - Knowledge and discipline depending algorithm creation.
    - Knowledge Resources processing.
    - Chorological assignment and processing, e.g., spatial calculations, e.g., countries and areas.
    - Chronological assignment and processing, e.g., time related calculations, e.g., geological and prehistoric.
- Coherent conceptual knowledge implementation.
    - Coherent conceptual knowledge references, main tables.
    - Coherent conceptual knowledge references, auxiliary tables.
- Symbolic representation, generation.
    - Context area views.
    - Symbolic representation of features groups, integrated visualisation.
    - (Further symbolic representation of narratives.)
    - (Multitude of further contextualisation and narratives.)
    - ...

After understanding the selected task-related algorithms and the fundamentals of knowledge complements many different realisations can be done straightforward, deploying the CKRI and CRI framework components.

The symbolic representation of features groups and the integrated visualisation will provide manifold ways of contextualisation. We can only demonstrate a single group of examples here.

Nevertheless, the realisation of the implemented workflow procedure may depend on the capacities the participating disciplines want to invest in their education, scientific research and contextualisation. It should not be uncommon with today's scientific research to invest increasing resources, 25 to over 50 percent of overall project resources, of each participating discipline into multi-disciplinary knowledge integration and contextualisation.

The CKRI and CRI framework can create coherent multi-disciplinary conceptual knowledge references effectively and efficiently and focus on core tasks within available capacities of time and other resources available for a workflow procedure.

B. Component implementation and realisation

The following passages give a compact overview of the major component framework integrated with this research. All the components and references are given, which were employed for the implementation and realisation and which are in a continuous further development process towards even closer integration and standards. More detailed, comprehensive discussion and examples regarding fundaments are available with the references on knowledge representations, methodology, contextualisation, and conceptual knowledge.

a) Conceptual knowledge frameworks: The created and further developed reference implementations of conceptual knowledge frameworks (this research major references in Tables I and II) are used with the implementation and realisation *KR* [6]. CKRI can be created by any disciplines and for multi-disciplinary scenarios and coherently integrated, e.g., in contextualisation for prehistorical and archaeological narratives.

b) Conceptual knowledge base: Conceptual knowledge base is The *Universal Decimal Classification (UDC)* [7], a general plan for knowledge classification, providing an analytico-synthetic and *faceted* classification, designed for subject description and indexing of content of information resources *irrespective of the carrier, form, format, and language*. UDC-based references for demonstration are taken from the multi-lingual UDC summary [7] released by the UDC Consortium, Creative Commons license [8].

c) Integration of scientific reference frameworks: Relevant scientific practices, frameworks, and standards from disciplines and contexts are integrated with the Knowledge Resources, e.g., here details regarding volcanological features, chronologies, spatial information, and Volcanic Explosivity Index (VEI) [9], [10].

d) Formalisation: All integration components, for all disciplines, require an *explicit and continuous formalisation* [11] *process*. The formalisation includes computation model support, e.g., *parallelisation standards, OpenMP* [12], [13], Reg Exp patterns, e.g., *Perl Compatible Regular Expressions (PCRE)* [14], and common standard methods, algorithms, and frameworks.

e) Methodologies and workflows integration: Methodologies for creating and utilising methods include model processing, remote sensing, spatial mapping, high information densities, and visualisation. Respective contextualisation of (prehistoric) scenarios should each be done under specific (prehistoric) conditions, especially supported by state-of-the-art methods, e.g., spatial operations, triangulation, gradient computation, and projection. The symbolic representation of the contextualisation can be done with a wide range of methods, algorithms, and available components, e.g., via LX Professional Scientific Content-Context-Suite (LX PSCC Suite) deploying the Generic Mapping Tools (GMT) and integrated modules [15] for visualisation.

f) Prehistory Knowledge Resources: Prehistoric objects and contexts are taken from *The Prehistory and Archaeology Knowledge Archive (PAKA)*, in continuous development for more than three decades [16] and is released by DIMF [17]. The KR support seamless coherent multi-disciplinary conceptual knowledge integration for workflow procedures.

g) Natural Sciences Knowledge Resources: Several coherent systems of major natural sciences' context object groups from *KR realisations* have been implemented, especially Knowledge Resources focussing on volcanological features [9] deployed with in depth contextualisation [10] and with a wide range of contexts [6], [7], [18]. The KR support seamless coherent multi-disciplinary conceptual knowledge integration for workflow procedures.

h) Inherent representation groups: The contextualisation for the inventory can employ state-of-the-art results from many disciplines, e.g., context from the natural sciences resources, integrating their inherent representation and common utilisation, e.g., *points*, *polygons*, *lines*, *Digital Elevation Model (DEM), Digital Terrain Model (DTM), and Digital Surface Model (DSM) representations* sources, e.g., from *satellites*, Unmanned Aerial Vehicles (UAV), *z-value representations*, *distance representations*, *area representations*, *raster*, *vector*, *binary*, and *non-binary data*. Employed resources are High Resolution (HR) (Space) Shuttle Radar Topography Mission (SRTM) [19], [20], HR Digital Chart of the World (DCW) [21], and Global Self-consistent Hierarchical High-resolution

 ISBN: 978-1-61208-961-4

Geography (GSHHG) [22]. SRTM was produced under the National Aeronautics and Space Administration (NASA) Making Earth System Data Records for Use in Research Environments (MEaSUREs) program. The Land Processed Distributed Active Archive Center (LPDAAC), USA [23], operates as a partnership between the U.S. Geological Survey (USGS) and the National Aeronautics and Space Administration (NASA), USA, and is a component of NASA's Earth Observing System Data and Information System (EOSDIS). Resources are released by NASA and JPL Jet Propulsion Laboratory (JPL), USA, [24], [25]. SRTM15 Plus [19], [20] is continuously updated and improved.

i) Scientific context parametrisation: Scientific *context parametrisation of prehistoric targets* can use the overall insight from all disciplines, e.g., parametrising algorithms and creating palaeolandscapes. Parametrisation is supported for all contexts and can consider views of participated disciplines. For the new inventory, parametrisation ranges from contexts, methods, representation of heights, illumination, symbol design, symbolic consistency to data locality and parallelisation.

j) Structures and symbolic representation: Structure is an organisation of interrelated entities in a material or non-material object or system [18]. Structure is essential in logic as it carries unique information. Structure means features and facilities. There are merely higher and lower facility levels of how structures can be addressed, which result from structure levels. Structure can, for example, be addressed by logic, names, references, address labels, pointers, fuzzy methods, phonetic methods. The deployment of long-term universal structure and data standards is essential. Relevant examples of sustainable implementations are *NetCDF* [26] based standards, including advanced features, hybrid structure integration, and parallel computing support (*PnetCDF*) and generic multi-dimensional table data, standard xyz files, universal source and text based structure and code representations.

C. Resulting coherent conceptual knowledge implementation

The CKRI implementations provide the fundament for the coherent multi-disciplinary knowledge based integration and the realisations of the methodological component integration.

Universally consistent conceptual knowledge of CKRI references, based on UDC code references, for demonstration, spanning the main tables [27] shown in Table I. Table II shows an excerpt of universally consistent conceptual knowledge of CKRI references, based on UDC code references, spanning auxiliary tables [28].

The tables contain major UDC code references required for the implementation and realisation of the methodological workflow procedure, especially for place (countries and AoI), time (Holocene), and disciplines (volcanology and prehistory).

D. Resulting symbolic representation of features groups

The procedural component framework implementation and realisation enable the creation of numerous contextualisations and symbolic representations for the coherent multi-disciplinary conceptual knowledge-based Holocene-prehistoric inventory of volcanological features groups.

For this research, we choose the resulting symbolic representation of a volcanological features group, maars, based on the coherent conceptual knowledge integration. The sequence of procedural steps enables contextualisation for flexible larger

TABLE I. CKRI IMPLEMENTATION OF COHERENT CONCEPTUAL KNOWLEDGE CONTEXTUALISATION; MAIN TABLES (EXCERPT).

Code / Sign Ref.	*Verbal Description (EN)*
UDC:0	Science and Knowledge. Organization. Computer Science. Information. Documentation. Librarianship. Institutions. Publications
UDC:1	Philosophy. Psychology
UDC:2	Religion. Theology
UDC:3	Social Sciences
UDC:5	Mathematics. Natural Sciences
UDC:52	Astronomy. Astrophysics. Space research. Geodesy
UDC:53	Physics
UDC:539	Physical nature of matter
UDC:54	Chemistry. Crystallography. Mineralogy
UDC:55	Earth Sciences. Geological sciences
UDC:550.3	Geophysics
UDC:551	General geology. Meteorology. Climatology. Historical geology. Stratigraphy. Palaeogeography
UDC:551.21	Vulcanicity. Vulcanism. Volcanoes. Eruptive phenomena. Eruptions
UDC:551.2...	Fumaroles. Solfataras. Geysers. Hot springs. Mofettes. Carbon dioxide vents. Soffioni
UDC:551.44	Speleology. Caves. Fissures. Underground waters
UDC:551.46	Physical oceanography. Submarine topography. Ocean floor
UDC:551.7	Historical geology. Stratigraphy
UDC:551.8	Palaeogeography
UDC:56	Palaeontology
UDC:6	Applied Sciences. Medicine, Technology
UDC:7	The Arts. Entertainment. Sport
UDC:8	Linguistics. Literature
UDC:9	Geography. Biography. History
UDC:902	Archaeology
UDC:903	Prehistory. Prehistoric remains, artefacts, antiquities
UDC:904	Cultural remains of historical times

TABLE II. CKRI IMPLEMENTATION OF COHERENT CONCEPTUAL KNOWLEDGE CONTEXTUALISATION; AUXILIARY TABLES (EXCERPT).

Code / Sign Ref.	*Verbal Description (EN)*
UDC (1/9)	Common auxiliaries of place
UDC:(23)	Above sea level. Surface relief. Above ground generally. Mountains
UDC:(3/9)	Individual places of the ancient and modern world
UDC:(3)	Places of the ancient and mediaeval world
UDC:(32)	Ancient Egypt
UDC:(35)	Medo-Persia
UDC:(36)	Regions of the so-called barbarians
UDC:(37)	Italia. Ancient Rome and Italy
UDC:(38)	Ancient Greece
UDC:(399)	Other regions. Ancient geographical divisions other than those of classical antiquity
UDC:(4/9)	Countries and places of the modern world
UDC:(4)	Europe
UDC:(5)	Asia
UDC:(6)	Africa
UDC:(7/8)	America, North and South. The Americas
UDC:(7)	North and Central America
UDC:(8)	South America
UDC:(9)	States and regions of the South Pacific and Australia. Arctic. Antarctic
UDC:"..."	Common auxiliaries of time.
UDC:"6"	Geological, archaeological and cultural time divisions
UDC:"62"	Cenozoic (Cainozoic). Neozoic (70 MYBP - present)
UDC:"63"	Archaeological, prehistoric, protohistoric periods and ages

 ISBN: 978-1-61208-961-4

and smaller context scales, e.g., generated symbolic representation (Figure 1) of country identification contexts (Figure 1(a)) and generated symbolic representation of AoI contexts for respective object entities (Figure 1(b)). Generated representations include integrated CKRI references, projection of topographic and bathymetric results, and further knowledge for respective areas, based on the coherent conceptual knowledge.

III. Discussion of procedural potential

Logic is a general limit to many overblown claims, from universal parallelisation to 'Artificial Instruments'.

Therefore, parallelisation can only deliver feasible approaches for simple, formalised cases of contextualisation and small parts of much more complex contexts of knowledge. The goals and complexity of conceptual knowledge-centric tasks and procedural tasks require the insight of eminently suitable structures and resources.

The resources, which provide highest potential for the realisation based on the inventory model are huge, based on quantity and resulting from quality of the contextualisation resources. Models are even continuously growing when considering ongoing state-of-the-art research. In consequence, these scenarios require a high level of scalability. A realistic conceptual-procedural environment for the coherent multi-disciplinary conceptual knowledge-based Holocene-prehistoric inventory of volcanological features groups includes:

- Different object groups, objects and views, e.g., for over 500 volcanological object entities and features.
- Multi-dimensional views, e.g., focus dependent views per objects, e.g., via OpenMP [12] / specifications [13].
- Embarrassingly parallel procedures (e.g., knowledge dimensional computation), e.g., via OpenMP [12] and specifications [13].
- Job parallel procedures (e.g., knowledge objects and resources localities).

Table III shows the inherent representation groups used by the disciplines for the formalised representation of knowledge integrated for the implementation and realisation (serial, parallel, not applicable, n.a.).

The respective locality-license and parallelisation aspects refer to the realisation resources, primarily depending on the respective knowledge and organisation. Therefore, precondition for implementation is a deep understanding of the knowledge complexity within a discipline, which is represented by the task as well as the required formalisations for all the components.

OpenMP is a mature and portable industry standard, which can be efficiently implemented directly by scientists of any discipline in their contextualisation, methodological workflow logic, and for their workflow procedure implementations and realisations.

Organisation of data structure and formalisation of knowledge are core tasks of a discipline itself and not at all a technical task. Nevertheless, the organisation of knowledge also defines feasible data locality concepts. Parallelisation of workflows with plain-dimension and multi-dimension targets can differ regarding their contextualisation results.

For example, a plain-dimension workflow can deliver different contextualisation contexts of an AoI. A multi-dimension workflow can deliver a certain contextualisation context of an AoI, depending on further dimensions, views or chorologies. Therefore, plain- and multi-dimension workflows can complement in chorological and chronological contextualisation while sharing resources, structural and procedural fundaments.

Table III. Inherent representation groups with integrated components inventory object entity example workflow.

Inherent Representation Group	*Locality-License*	*Parallelisation*	
	Model	*Plain*	*Multi*
KR preprocessing, conceptual	On-premise	Ser./par.	Ser./par.
Context preprocessing	Restricted	Ser./par.	Ser./par.
KR processing, conceptual	On-premise	OpenMP	OpenMP
Conceptual knowledge processing	On-premise	OpenMP	OpenMP
PoI	On-premise	OpenMP	OpenMP
Point spatial operations	Restricted	OpenMP	OpenMP
Line operations	Restricted	OpenMP	OpenMP
Polygon operations	Restricted	OpenMP	OpenMP
DEM	Restricted	OpenMP	OpenMP
PCRE	Restricted	OpenMP	OpenMP
Editing	Restricted	OpenMP	OpenMP
Projecting	Restricted	OpenMP	OpenMP
Cutting	Restricted	OpenMP	OpenMP
Sampling	Restricted	OpenMP	OpenMP
Filtering	Restricted	OpenMP	OpenMP
Illumination	Restricted	OpenMP	OpenMP
Triangulation	Restricted	OpenMP	OpenMP
Projection	Restricted	OpenMP	OpenMP
Filter operations	On-premise	OpenMP	OpenMP
View/frame computation	Restricted	OpenMP	OpenMP
Model reduction (frames)	n.a.	OpenMP	OpenMP
Model reduction (animation)	n.a.	n.a.	Serial
...	...	...	...

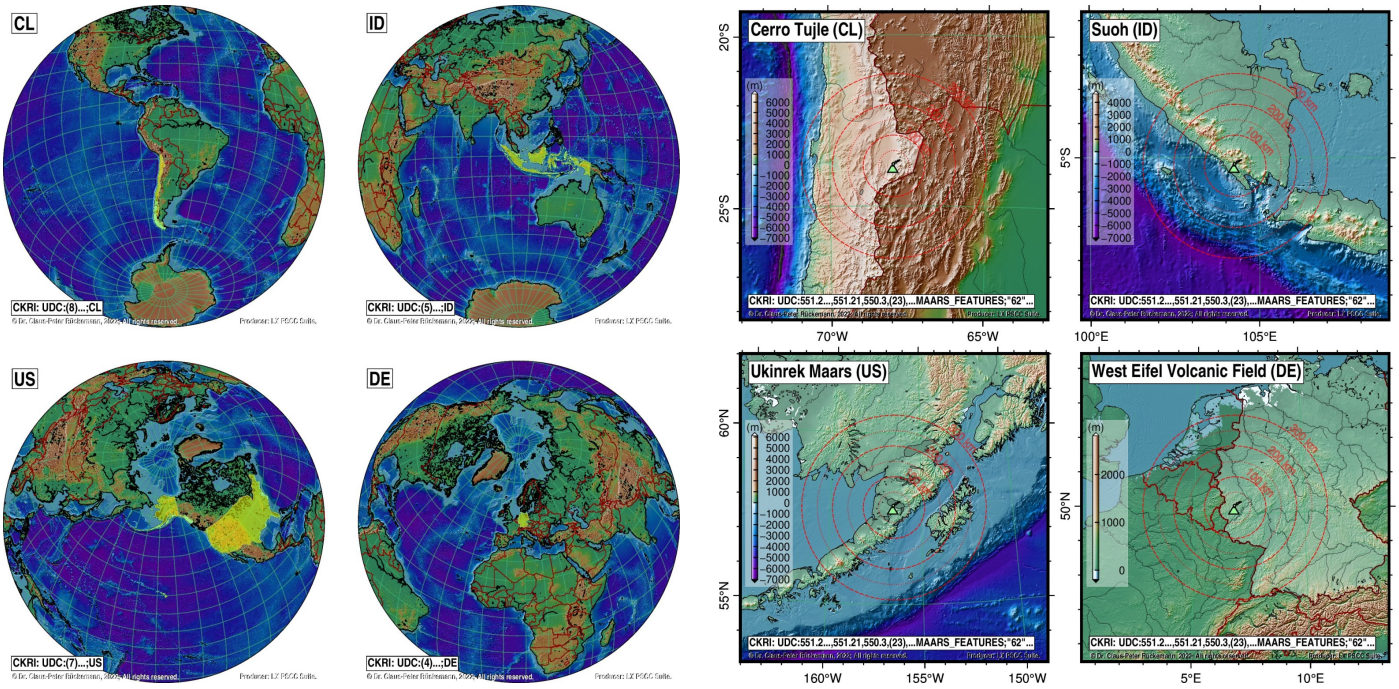
An example for a model reduction in plain-dimension is, e.g., generation of multiple context views. An example for a model reduction in multi-dimension is, e.g., generation of a video of geospherical satellite view frames with moving observer position. It is obvious that the workflow logic of the examples also differ in their ways of parallelisation. The use of on-premise (e.g., in-house) and restricted (distributed) resources is attributable to the licenses of the core assets, the knowledge resources. Inherent representation groups are major matter of scalable processing and conversion (two-dimensional/three-dimensional) and higher multi-dimensional workflow procedures. Table IV shows the scalability of the example workflow procedure for parallelised parts of the coherent multi-disciplinary conceptual knowledge-based Holocene-prehistoric inventory of volcanological features groups, based on mean requirements for an object entity, with numbers of objects entities, n_o, numbers of frames, n_f, and numbers of views, n_v, for $n_f = 1$ and $n_v = 2$ as in the above symbolic representation example of volcanological features groups.

Table IV. Parallelised processing of inventory workflow (parallelised Knowledge Resources and context resources).

Number of CPU Cores	*Wall Time Workflow (Plain)*	*Number of Object Entities* $n_o = 100$	$n_o = 500$
1	$n_o \cdot (n_v \cdot n_f \cdot 1,680/1)$ s	336,000 s	1,680,000 s
36	$n_o \cdot (n_v \cdot n_f \cdot 1,680/36)$ s	9,333 s	46,667 s

The architecture choosen for this realisation is an efficient 36-core-based Central Processing Unit (CPU) (Intel Xeon), which is taking into account that we commonly use 36 cores for many basic global approaches, e.g., considering the 360 degrees of a global model. Results on other architectures with same numbers of respective cores will be highly comparable.

 ISBN: 978-1-61208-961-4



(a) Generated symbolic representation of country identification contexts.

(b) Generated symbolic representation of object AoI contexts.

Figure 1. Resulting symbolic representation of a volcanological features group (maars) based on the coherent conceptual knowledge integration (excerpt). Sequence of procedural steps for larger scale and smaller scale contextualisation, including country identification contexts (a) and AoI contexts (b). Generated representations include integrated CKRI references, projection of topographic and bathymetric results, and further knowledge for respective areas.

Precondition for parallelisation is sufficient memory for parallel use of integrated resources. Considering the employed resources, e.g., SRTM, 128 GB for 36 parallel processes is comfortable when data limits are cut to the limits required for the algorithms with the range of a few hundred kilometres area per object entity.

Wall and compute times, especially of multi-dimensional workflow results, can greatly be reduced from the integrated parallelisation, which makes the procedural solution highly scalable. The wall times for numbers of objects entities, n_o, illustrate the high scalability when the same workflow is using higher numbers of CPU cores. Probably, most practical workflows may contain parts which cannot be reasonably parallelised. This is especially true for scientific tasks with a certain complexity. The percentage of non parallelised parts is very low here. For multi-dimension targets, e.g., animations with $n_f = 1000$ and $n_v = 1$, it may be considered to employ hundreds to thousands of CPU cores so parallelised wall times per object can be reduced from days to hours.

Serial and parallel compute times, e.g., for groups of object entities, are non-linear. For example, mean times for the same workflow realisation may greatly differ for different object entities. To significant extent, this is consequence of the inherent complexity of the knowledge complements, which have to be integrated and analysed. In practice, compute times for object entities may commonly vary to over several hundred percent. Component and knowledge contributions of different disciplines may have different weight in their contributions to the non-linearities. The resulting compute times may even deliver continuously new and non-linear compute times in a dynamical workflow realisation with knowledge resources and components, which are in continuous development.

IV. Conclusion

The new approaches for the creation of a sustainable procedural component framework for implementation and realisation of a coherent multi-disciplinary conceptual knowledge-based inventory proved efficient and sustainable. Based on the methodology of coherent conceptual knowledge classification, the CKRI, and the CRI frameworks, the procedural and conceptual implementation and realisation of a Holocene-prehistoric inventory of worldwide volcanological features groups showed very flexible and scalable, supported by many scenarios over the last years. The developed framework of components, can be employed for multi-disciplinary integration of knowledge contexts. Everything has been done to deploy standards and provide maximum flexibility so that context from prehistory, archaeology, natural sciences, and humanities can be be coherently integrated. The component framework showed to enable an effective and efficient coherent conceptual knowledge contextualisation and georeferenced symbolic representation. Researchers from all disciplines already practice the procedural component framework development, coherent conceptual knowledge, and procedure parallelisation in professional long-term research knowledge and data management.

Future research will address the creation of a component framework for a Holocene-prehistoric inventory of worldwide volcanological features, which enables procedural integration and coherent conceptual knowledge contextualisation with prehistorical and archaeological knowledge resources.

Acknowledgements

This ongoing research is supported by scientific organisations and individuals. We are grateful to the “Knowledge in Motion” (KiM) long-term project, Unabhängiges Deutsches

Institut für Multi-disziplinäre Forschung (DIMF), for partially funding this research, implementation, case studies, and publication under grants D2022F1P05312, D2022F1P05308, and D2020F1P05228. and to its senior scientific members and members of the permanent commission of the science council, especially to Dr. Friedrich Hülsmann, Gottfried Wilhelm Leibniz Bibliothek (GWLB) Hannover, to Dipl.-Biol. Birgit Gersbeck-Schierholz, Leibniz Universität Hannover, for fruitful discussion, inspiration, and practical multi-disciplinary contextualisation and case studies. We are grateful to Dipl.-Geogr. Burkhard Hentzschel and Dipl.-Ing. Eckhard Dunkhorst, Minden, Germany, for prolific discussion and exchange of practical spatial, UAV, and context scenarios. We are grateful to Dipl.-Ing. Hans-Günther Müller, Göttingen, Germany, for providing specialised, manufactured high end computation and storage solutions. We are grateful to The Science and High Performance Supercomputing Centre (SHPSC) for long-term support. / DIMF-PIID-DF98_007; URL: https://scienceparagon.de/cpr .

REFERENCES

[1] C.-P. Rückemann, "Towards a Component Reference Implementations Frame for Achieving Multi-disciplinary Coherent Conceptual and Chorological Contextualisation in Prehistory and Prehistoric Archaeology," International Journal on Advances in Systems and Measurements, vol. 14, no. 1&2, 2021, pp. 103–112, ISSN: 1942-261x, LCCN: 2008212470 (Library of Congress), URL: http://www.iariajournals.org/systems_and_measurements [accessed: 2022-04-24].

[2] C.-P. Rückemann, "Towards Conceptual Knowledge Reference Implementations for Context Integration and Contextualisation of Prehistory's and Natural Sciences' Multi-disciplinary Contexts," International Journal on Advances in Systems and Measurements, vol. 14, no. 1&2, 2021, pp. 113–124, ISSN: 1942-261x, LCCN: 2008212470 (Library of Congress), URL: http://www.iariajournals.org/systems_and_measurements [accessed: 2022-04-24].

[3] C.-P. Rückemann, "The Information Science Paragon: Allow Knowledge to Prevail, from Prehistory to Future – Approaches to Universality, Consistency, and Long-term Sustainability," The International Journal "Information Models and Analyses" (IJ IMA), vol. 9, no. 3, 2020, pp. 203–226, Markov, K. (ed.), ISSN: 1314-6416 (print), Submitted accepted article: November 18, 2020, Publication date: August 17, 2021, URL: http://www.foibg.com/ijima/vol09/ijima09-03-p01.pdf [accessed: 2022-04-24].

[4] C.-P. Rückemann, "From Knowledge and Meaning Towards Knowledge Pattern Matching: Processing and Developing Knowledge Objects Targeting Geoscientific Context and Georeferencing," in Proc. GEOProcessing 2020, November 21–25, 2020, Valencia, Spain, 2020, pp. 36–41, ISSN: 2308-393X, ISBN-13: 978-1-61208-762-7.

[5] R. Gleser, Zu den erkenntnistheoretischen Grundlagen der Prähistorischen Archäologie. Leiden, 2021, 2021, (title in English: On the Epistemological Fundaments of Prehistorical Archaeology), in: M. Renger, S.-M. Rothermund, S. Schreiber, and A. Veling (Eds.), Theorie, Archäologie, Reflexion. Kontroversen und Ansätze im deutschsprachigen Diskurs, (in print).

[6] C.-P. Rückemann, "Prehistory's and Natural Sciences' Multi-disciplinary Contexts: Contextualisation and Context Integration Based on Universal Conceptual Knowledge," in Proc. INFOCOMP 2020, May 30 – June 3, 2021, Valencia, Spain, 2021, pp. 8–14, ISSN: 2308-3484, ISBN: 978-1-61208-865-5.

[7] "Multilingual Universal Decimal Classification Summary," 2012, UDC Consortium, 2012, Web resource, v. 1.1. The Hague: UDC Consortium (UDCC Publication No. 088), URL: http://www.udcc.org/udcsummary/php/index.php [accessed: 2022-04-24].

[8] "Creative Commons Attribution Share Alike 3.0 license," 2012, URL: http://creativecommons.org/licenses/by-sa/3.0/ [accessed: 2022-04-24], (first release 2009, subsequent update 2012).

[9] C.-P. Rückemann, "Cognostics and Knowledge Used With Dynamical Processing," International Journal on Advances in Software, vol. 8, no. 3&4, 2015, pp. 361–376, ISSN: 1942-2628, LCCN: 2008212462 (Library of Congress), URL: http://www.iariajournals.org/software/ [accessed: 2022-04-24].

[10] C.-P. Rückemann, "Long-term Sustainable Knowledge Classification with Scientific Computing: The Multi-disciplinary View on Natural Sciences and Humanities," International Journal on Advances in Software, vol. 7, no. 1&2, 2014, pp. 302–317, ISSN: 1942-2628.

[11] C.-P. Rückemann, R. Pavani, B. Gersbeck-Schierholz, A. Tsitsipas, L. Schubert, F. Hülsmann, O. Lau, and M. Hofmeister, Best Practice and Definitions of Formalisation and Formalism. Post-Summit Results, Delegates' Summit: The Ninth Symp. on Adv. Comp. and Inf. in Natural and Applied Sciences (SACINAS), The 17th Int. Conf. of Num. Analysis and Appl. Math. (ICNAAM), Sept. 23–28, 2019, Rhodes, Greece, 2019, pp. 1–16, DOI: 10.15488/5241.

[12] L. Dagum and R. Menon, "OpenMP: an industry standard API for shared-memory programming," Computational Science & Engineering, (IEEE), vol. 5, no. 1, 1998, pp. 46–55.

[13] OpenMP Architecture Review Board, "OpenMP API 5.1 Specification," Nov. 2020, URL: https://www.openmp.org/wp-content/uploads/OpenMP-API-Specification-5-1.pdf [accessed: 2022-04-24].

[14] "Perl Compatible Regular Expressions (PCRE)," 2021, URL: https://www.pcre.org/ [accessed: 2022-04-24].

[15] P. Wessel, W. H. F. Smith, R. Scharroo, J. Luis, and F. Wobbe, "The Generic Mapping Tools (GMT)," 2020, URL: http://www.generic-mapping-tools.org/ [accessed: 2022-04-24], URL: http://gmt.soest.hawaii.edu/ [accessed: 2022-04-24].

[16] C.-P. Rückemann, "Information Science and Inter-disciplinary Long-term Strategies – Key to Insight, Consistency, and Sustainability: Conceptual Knowledge Reference Methodology Spanning Prehistory, Archaeology, Natural Sciences, and Humanities," International Tutorial, DataSys Congress 2020, Sept. 27 – Oct. 1, 2020, Lisbon, Portugal, 2020, pp. 113, URL: http://www.iaria.org/conferences2020/ProgramINFOCOMP20.html [accessed: 2022-04-24].

[17] "The Prehistory and Archaeology Knowledge Archive (PAKA) license," 2021, (release 2021), Unabhängiges Deutsches Institut für Multidisziplinäre Forschung (DIMF): All rights reserved. Rights retain to the contributing creators.

[18] C.-P. Rückemann, "The Impact of Information Science Accompanied Structural Information on Computation of Knowledge Pattern Matching and Processing: A Prehistory, Archaeology, Natural Sciences, and Humanities Conceptual Integration Perspective," in Proc. INFOCOMP 2020, Sept. 27 – Oct. 1, 2020, Lisbon, Portugal, 2020, pp. 1–6, ISBN: 978-1-61208-807-5, URL: http://www.thinkmind.org/index.php?view=article&articleid=infocomp_2020_1_10_60015 [accessed: 2022-04-24].

[19] C. L. Olson, J. J. Becker, and D. T. Sandwell, "SRTM15_PLUS: Data fusion of Shuttle Radar Topography Mission (SRTM) land topography with measured and estimated seafloor topography," (NCEI Accession 0150537), National Centers for Environmental Information (NCEI), NOAA, 2016.

[20] B. Tozer, D. T. Sandwell, W. H. F. Smith, C. Olson, J. R. Beale, and P. Wessel, "Global Bathymetry and Topography at 15 Arc Sec: SRTM15+," Earth and Space Science, vol. 6, no. 10, Oct. 2019, pp. 1847–1864, ISSN: 2333-5084, DOI: 10.1029/2019EA000658.

[21] P. Wessel, "DCW for GMT 6 or later," 2022, URL: http://www.soest.hawaii.edu/pwessel/dcw/ [accessed: 2022-04-24].

[22] P. Wessel, "GSHHG," 2017, URL: http://www.soest.hawaii.edu/pwessel/gshhg/ [accessed: 2022-04-24].

[23] "Land Processed Distributed Active Archive Center," 2022, LP DAAC, URL: https://lpdaac.usgs.gov/ [accessed: 2022-04-24].

[24] "U.S. Releases Enhanced Shuttle Land Elevation Data," 2014, JPL, September 23, 2014, URL: https://www.jpl.nasa.gov/news/us-releases-enhanced-shuttle-land-elevation-data [accessed: 2022-04-24].

[25] "U.S. Releases Enhanced Shuttle Land Elevation Data, Official NASA SRTM Site," 2014, Australia, September 23, 2014, URL: https://www2.jpl.nasa.gov/srtm/ [accessed: 2022-04-24].

[26] "Network Common Data Form (NetCDF)," 2021, DOI: 10.5065/D6H70CW6, URL: http://www.unidata.ucar.edu/software/netcdf/ [accessed: 2022-04-24].

[27] "UDC Summary Linked Data, Main Tables," 2022, Universal Decimal Classification (UDC), UDC Consortium, URL: https://udcdata.info/078887 [accessed: 2022-04-24].

[28] "UDC Summary Linked Data, Auxiliary Tables," 2022, Universal Decimal Classification (UDC), UDC Consortium, URL: https://udcdata.info/ [accessed: 2022-04-24].

 ISBN: 978-1-61208-961-4