Óbuda University

Thesis Booklet

Reproducibility analysis of the scientific workflows

Anna Bánáti

Supervisors:
Péter Kacsuk Phd. Prof.
Miklós Kozlovszky, Phd

Doctoral School of Applied Informatics

Budapest, 2016.

1 INTRODUCTION

1.1 Scientific experiments – In vivo, In vitro, In situ, In silico

During the last decade, scientific workflows have emerged as a widely accepted solution for performing *in silico* experiments for large computational challenges. The traditional scientific experiments are conducted on living organisms, called *in vivo* (Latin: "within the living"), in the nature, called *in situ* (Latin: locally, on site) or in laboratories, called *in vitro* (Latin: in glass) experiments. During *in vivo* experiments, the effects of various biological entities are tested in their original environment on whole living organisms, usually animals or humans. In situ observation is performed on site, typically in the habitat of the animal being studied and generally it is the environment that is modified in order to increase/improve the life conditions of a certain animal. The *in vitro* term refers to a controlled environment such as test tubes, flasks, petri dishes, etc. where the studied component is tested in an isolated way from their original, living surroundings. These experiments have fewer variables and simpler conditions than *in vivo* experiments and they can avoid the continuously changing impact and interactions of real life. This way/Thus they could allow a more fine-grained analysis of the studied phenomena. At the same time, correlating their results to real-world scenarios was not always straightforward, thus, generally *in vitro* results have to be verified in the original environment.
In contrast to the traditional methods, the *in silico* (Latin: in silicon, referring to semiconductor computer chips) experiments are performed on computer or via computer simulation, modelling the original components, variables and the studied effects. Thanks to the particularly fast growing of computer science technology these experiments become more and more complex, more data and compute intensive which requires parallel and distributed infrastructure (supercomputers, grids, clusters, clouds) to enact them. Generally, these in-silico experiments consist of a huge amount of activities (call jobs) – their number can reach hundreds or even thousands - which invoke particularly data and compute intensive programs. Tying the jobs to a single, multi thread chain provides a scientific workflow to model the *in silico* experiments which can be executed by the Scientific Workflow Management Systems.

1.2 Reproducibility

To be able to proof or verify a scientific claim, the repeatability or the reproducibility of any type of experiments is a crucial requirement in the scientist's community. The different users for different purposes may be interested in reproducing of the scientific workflow. The scientists have to prove its results, other scientists would like to reuse the results and reviewers intend to verify the correctness of

the results (Koop & al, 2011). A reproducible workflow can be shared in repositories and it can become useful building blocks that can be reused, combined or modified for developing new experiments.

In the traditional method, the scientists make notes about the steps of the experiments, the partial results and the environment to make the experiments reproducible. Additionally, during the history of the scientific research, different standards, metrics, measurements and conventions had been developed to allow to provide the exact descriptions, the repeatability and the possibility of reusing each other's results. After all, certain types of the scientific experiments are unable to be repeatable because of the continuously changing environment such as the living organisms or nature in which many factors can be interacts and, in this way influence the results. Similarly, in case of the *in silico* experiments, the same way has to be walked and has to develop tools to make them reproducible. On one hand, like the scientist make notes about the traditional experiments, provenance information has to be collected about the environment of the execution and the partial result of the scientific workflow. On the other hand the ontologies of these type of experiments also has to be developed to allow the knowledge sharing and the reusability on the so called scientific workflow repositories. However many researcher work in these fields the reproducibility of the scientific workflows is still a big challenge because of:

- The complexity and the ever changing nature of the parallel and distributed infrastructure: Computations on a parallel and distributed computer system arise particularly acute difficulties for reproducibility since, in typical parallel usage, the number of processors may vary from run to run. Even if the same number of processors is used, computations may be split differently between them or combined in a different order. Since computer arithmetic is not commutative, associative, or distributive, achieving the same results twice can be a matter of luck. Similar challenges arise when porting a code from one hardware or software platform to another (Stodden & al., 2013)
- The labyrinthine dependencies of the different applications and services: A scientific workflow inherently can interconnect hundred or even thousand jobs which can be based on different tools and applications which has to work together and deliver data to each other. In addition each job can depend on external inputs complicating the connections and dependencies.
- The complexity of the scientific workflows managing a huge amount of data.

1.3 Motivation

Zhao et al. (Zhao & al, 2012) and Hettne (Hettne & al, 2012) investigated the main purposes of the so-called *workflow decay*, which means that year by year the ability and success of the re-execution of any workflow significantly reduces. In their investigation they examined 92 Taverna workflows from myExperiment repository in 2007-2012 and re-execute them. This workflow selection had a large

coverage of domain according to 18 different scientific (such as life sciences, astronomy, or cheminformatics) and non-scientific domains (such as testing of Grid services). The analysis showed that nearly 80% of the tested workflows failed to be either executed or produce the same results. The causes of workflow decay can be classified into four categories:

1. Volatile third-party Resources
2. Missing example data
3. Missing execution environment
4. Insufficient descriptions about workflows

By incorporating these results we have deeply investigated the requirements of the reproducibility and I intended to find methods which make the scientific workflows reproducible.

To sum up our conclusions, in order to reproduce an in-silico experiment the scientist community and the system developers have to face three important challenges:

1. More and more meta-data has to be collected and stored about the infrastructure, the environment, the data dependencies and the partial results of an execution in order to make us capable of reconstructing the execution in a later time even in a different infrastructure. The collected data – called provenance data – help to store the actual parameters of the environments, the partial and final data product and system variables.
2. Descriptions and samples have to be stored together with the workflows which are provided by the user (scientist).
3. Some services or input data can change or become unavailable during the years. For example, third party services, special local services or continuously changing databases. Scientific workflows which are established on them can become instable and non-reproducible. In addition certain computations may base on random generated values (for example, in case of image processing) thus, its execution are not deterministic so these computations cannot be repeated to provide the same result in a later time. These factors – call dependencies of the execution - can especially influence the reproducibility of the scientific workflows, consequently, they have been eliminated or handled.

In this dissertation I deal with the third item.

The goal of computational reproducibility is to provide a solid foundation to computational science, much like a rigorous proof is the foundation of mathematics. Such a foundation permits the transfer of knowledge that can be understood, implemented, evaluated, and used by others. (Stodden & al., 2013) However, nowadays more and more workflow repositories (myExperiment; CrowdLabs etc.) can help the knowledge sharing and the reusability, the reproducibility cannot be guaranteed by the systems. The

ultimate goal of my research is to support the scientist by giving information about the reproducibility of the workflows found in the repositories. Investigating and analysing the change of the components (call descriptors) required to the re-execution I reveal their nature and I can identify the crucial descriptor which can prevent the reproducibility. In certain cases, based on the behavior of the crucial component an evaluation can be performed for the case of unavailability which can replace the missing component with a simulated one making the workflow reproducible. With help of this reproducibility analysis also the probability of reproducibility can be calculated or the reproducible part of the workflow can be determined. To make the workflow reproducible, extra computations, resources or time are required which impose an extra cost for the execution. This cost can be measured and it can qualify the workflow from the reproducibility perspective. Additionally, the analysis presented in this dissertation can support the scientist not only to find the most suitable and reliable workflow on the repository but also can help to design a reproducible scientific workflow. The process, from the first execution of a workflow to achieving a complete and reproducible workflow is very long and the jobs get over a lot of change.

1.4 Research methodology

As a starting point of my research I thoroughly investigated the related work in the theme of reproducibility and also provenance which is the most significant requirements of the reproducibility. According to the reviewed literature I gave a taxonomy about dependencies of the scientific workflows and about the most necessary datasets required to reproduce a scientific workflow.

Based on this investigation I formalized the problem and set out the mathematical model of the reproducibility analysis. First, I introduced the necessary terms and definitions according to reproducible jobs and workflows which serve as a building blocks to determine and prove the statements and the methods. With help of the mathematical statistics tool, I analyzed the nature of the descriptors based on a sample set originating from the previous executions of the workflow to find statistical approximation tools to describe the relation between the descriptors and the results. Additionally, I introduced two metrics of the reproducibility based on the probability theory, the Average Reproducibility Cost (ARC) and the Non-reproducibility Probability (NRP) and defined a calculation method to calculate them in polynomial time. The universal approximation capabilities of neural networks have been well documented by several papers (Hornik & al., 1989), (Hornik & al., 1990), (Hornik, 1991) and I applied the Radial Basis Function (RBF) networks to evaluate the ARC in case if the exact calculation is not possible. To evaluate the NRP the Chernoff's inequality (Bucklew & Sadowsky, 1993) was applied based on Large Deviation Theory which concerns the asymptotic behavior of remote tails of sequences of probability distributions.

To perform the statistical calculations and prove the assumptions and the results, I used the MatLab and Excel applications.

2 NOVEL SCIENTIFIC RESULTS (THESES)

Thesis group 1: I have defined and extended the mathematical definition of the reproducible job and reproducible scientific workflow and I have determined the empirical and theoretical decay-parameters of the descriptors.

Thesis 1.1

I have introduced the terms of the descriptor-space assigned to the jobs and the theoretical decay-parameter assigned to the descriptors, and I have determine with these two terms the definition of a reproducible job.

Related publications: 1-B, 2-B, 3-B, 4-B, 5-B

Thesis 1.2

I have extended the definition of the reproducible job for the scientific DAG (directed acyclic graph) type workflows and based on the definition I have proved that if and only if a job is reproducible, than the scientific workflow is also reproducible.

Related publications: 1-B, 4-B, 5-B

Thesis 1.3

Based on *s* previous executions of a deterministic job I have defined an empirical decay-parameter assigned to the descriptors of a given job in case of time-dependent and time-independent descriptors and I revealed the relationships between the behavior of the descriptors and the values of the decay-parameters.

Related publications: 2-B

Thesis group 2: Based on simulations and on the empirical decay-parameters I have investigated and determined the behavior, the coverage of the changing descriptors and the feasible approximation of the result deviation.

Thesis 2.1

Based on a sample set originated from *s* previous executions I have defined and realized a method to determine that subgraph of a given scientific (DAG) workflow, in which the job results are influenced by a given descriptor.

Related publications: 1-B,

Thesis 2.2

I have introduced the term reproducibility rate index (RRI) to calculate how big part of the scientific workflow is reproducible and I have developed a method to determine the reproducible sub-graph of a partially reproducible scientific workflow represented by a DAG.

Related publications: 1-B, 7-B

Thesis 2.3

I have defined the impact factor term of a changing descriptor set based on *s* previous executions, and I have determined the feasible approximation of the result deviation.

Related publications: 1-B, 2-B

Thesis 2.4

Based on the theoretical decay-parameter and the empirical probability calculated according to the *s* previous job executions, I have defined and proved the theoretical and the empirical probability of the reproducibility concerning to a given scientific workflow assuming that the descriptors and the jobs are independent.

Related publications: 2-B, 5-B

Thesis group 3: I defined two metrics of the reproducibility and I determined approximations to evaluate them in polynomial time if the exact calculation is not possible in real-time.

Thesis 3.1

I have introduced the term of the repairing cost-index assigned to the computational job descriptors, which gives the ability to determine the reproducibility metrics of the DAG type scientific workflow:, namely the Average Reproducibility Cost (ARC) and the Non-Reproducibility Probability (NRP) values.

Related publications: 3-B, 4-B, 5-B

Thesis 3.2

I have determined a real time computable method to evaluate in polynomial time the ARC of a DAG type scientific workflow in case the descriptors are independent.

Related publications: 4-B

Thesis 3.3

I have determined a real time computable method to calculate upper estimates in polynomial time the NRP value of a scientific workflow, when the descriptors and jobs are independent and the g(y) cost function is the linear function of the y_i binary variables.

Related publications: 3-B

Thesis 3.4

Based on the decay-parameters and the cost index I have categorized from the reproducibility perspective the scientific DAG-type workflows.

Related publications: 5-B

3 PRACTICAL APPLICABILITY OF THE RESULTS

Based on this research I designed two extra modules of the WSPGRADE/gUSE to reproduce an in other way non-reproducible SWf. It performs an pre-analysis phase before re-execute a SWf based on the descriptor space to determine in which way the SWf can be reproduced and which extra tools (evaluation tool, descriptor value capture or extra storage) is required. After the re-execution an post analysis phase perform an estimation (if necessary) and updates the provenance database with the appropriate parameters needed to evaluation.

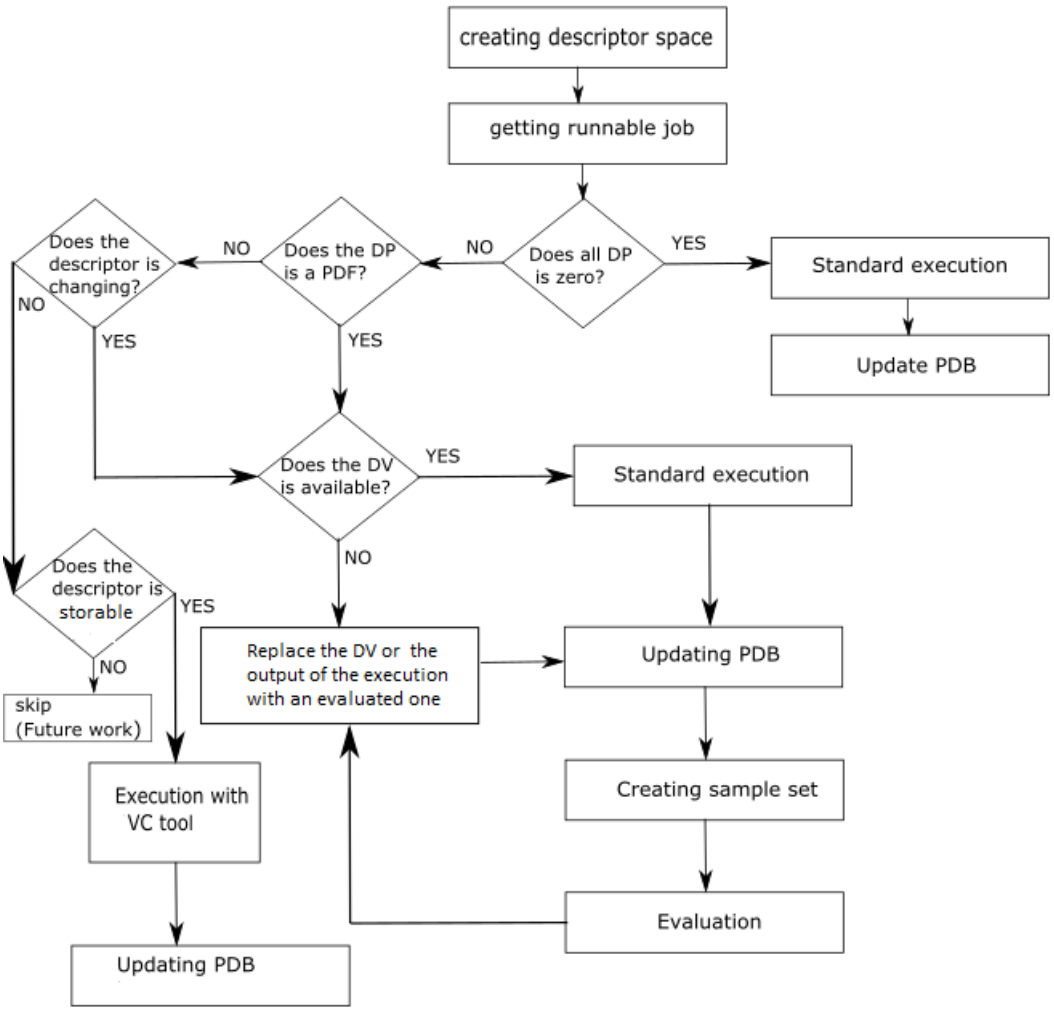
The process of reproducibility-analysis

Based on the descriptor's space the pre-analyzer performs a classification of the jobs of the given Wf. Depending on the classification, the job can be executed in three ways:

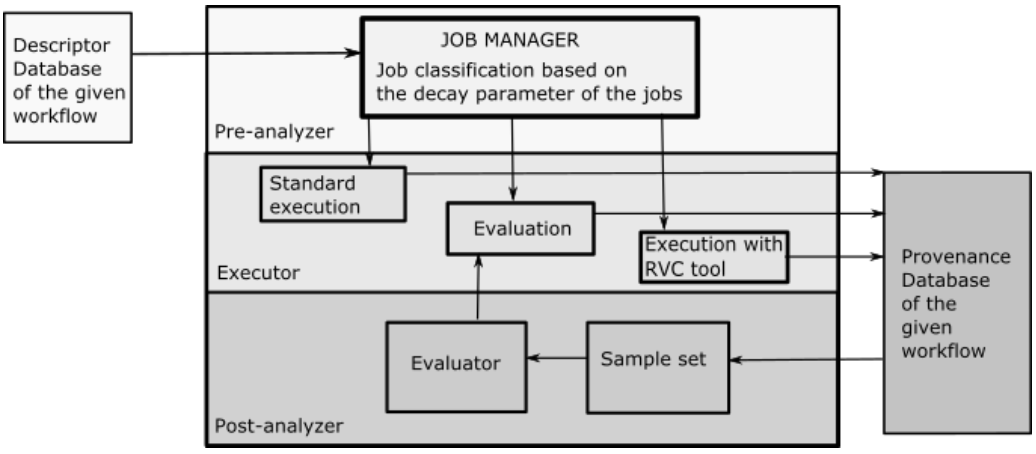
1. Standard execution, if all the decay parameters are zero.
2. Replacing the execution with evaluation, if there are changing descriptor values in the descriptor-space and their availabilities are changing in time.
3. Execution with descriptor value capture (VC) tool, if the execution of the job is based on operation related descriptor value or the value cannot be stored due to the

In all cases updating the Provenance Database (PDB) is performed occasionally by extra provenance information (for example a random value).

Based on the PDB the post-analyzer creates a sample set. The evaluator module computes the evaluated output of the given job (figure 1,2)



1. *Figure* The flowchart of the reproducing process



2. *Figure* The block diagram of the reproducing process

4 CONCLUSION

During the last decades the e-science widely gather ground among the scientific communities. Thanks to the high performance computing and to the parallel and distributed systems the classical analytical experiments conducted in the laboratories are taken over by the data and compute intensive in-silico experiments. The steps of these experiments are chained to a so called scientific workflow. An essential part of the scientific method is to repeat and reproduce the experiments of other scientists and to test the outcomes themselves even in a different execution environment. A scientific workflow is reproducible, if it can be re-executed without failures and gives the same result as the first time. In this approach the failures do not mean the failures of the Scientific Workflow Management System (SWfMS) but the correctness and the availability of the inputs, libraries, variables etc. The different users for different purposes may be interested in reproducing of the scientific workflow. The scientists have to prove its results, other scientists would like to reuse the results and reviewers intend to verify the correctness of the results. A reproducible workflow can be shared in repositories and it can become useful building blocks that can be reused, combined or modified for developing new experiments.

In this dissertation I investigated the requirements of the reproducibility and I set out methods which can handle and solve the problem of changing or missing descriptors to be able to reproduce a – in other way – non-reproducible scientific workflow. In order to achieve this goal I formalized the problem and based on provenance database I introduced the term of the descriptor-space which contains all the necessary component (call descriptor) to reproduce a job. Concerning to the descriptors I defined the theoretical and the empirical decay-parameter which describe the change of the descriptor in time-dependent and time-independent cases as well. Additionally, with the help of the decay parameters the crucial descriptors – which can influence or even prevent to reproduce a SWf – can be identified. Based on provenance database I created a sample set referred to a job which contains the descriptors of the job originated from the previous executions. Analyzing the empirical decay-parameter based on the sample set the relation can be determined between the change of the descriptor values and the empirical decay-parameter. Our goal was to find methods which can help to compensate the changing nature of the descriptors and which can help to perform evaluation to make the scientific workflow reproducible by replacing the missing values with simulated ones. In addition I determined the impact of a descriptor which says how the descriptor influences the result of a given job. The sample set also can help to determine the probability of the reproducibility and the reproducible part of a given SWf. Since the basis of our analysis is the decay-parameter, according to it I assigned to every descriptor a cost-index which means the “work” required to reproduce a given job or workflow. In this way I introduced two measures of the reproducibility: the Average Reproducibility Cost and the Non-reproducibility Probability. The first one determines the expected value of the cost to reproduce a – on other way – non-reproducible

SWf. The other measure is the Non-reproducibility Probability which gives how likely the reproducibility cost is greater than a predefined C threshold. The analyses was bounded on the special cases when the cost function is linear or can be approximated by a linear function. Finally I classified the scientific workflows from the reproducibility perspective and I determined the reproducible, partial reproducible, reproducible by substitution, reproducible with probability *p* and the non-reproducible scientific workflows.

During the design phase the results of this investigation can help the scientists to analyze the crucial descriptors of their workflow which can prevent to reproduce it. Additionally, storing this information, statistics and evaluation methods together with the workflows in the repositories, can provide a useful tool to support the reusability of the SWf making it reproducible and the scientists to find the most adequate (in sense of reproducibility) workflow to reuse.

5 FUTURE RESEARCH DIRECTINOS

As a further extension of my research I plan to investigate scientific workflows represented by non-DAGs. These cyclic graph may contain execution loops which results recursive workflows. Moreover, the evaluability of the two reproducibility metrics, ARC and NRP can be investigated without assuming the independency of the descriptors.
First and foremost an implementation of the extension (mentioned in section 9) should be carried out in WSPGRADE/gUSE scientific workflow management system developed by MTA SZTAKI.

References

Bucklew, J. A., & Sadowsky, J. S. (1993). A contribution to the theory of Chernoff bounds. *IEEE Transactions on Information Theory, 39*(1), 249-254.

Hettne, K., & al, e. (2012). Best Practices for Workflow Design: How to Prevent Workflow Decay. *SWAT4LS.*

Hornik, K. (1991). *Approximation Capabilities of Multilayer Feedforward Networks, Neural Networks, 4*, 251-257.

Hornik, K., & al., e. (1989). *Multilayer Feedforward Networks are Universal Approximators; Neural Networks, 2*, 359-366.

Hornik, K., & al., e. (1990). *Universal Approximation of an Unknown Mapping and Its Derivatives Using Multilayer Feedforward Networks, Neural Networks, 3*, 251-257.

Koop, D., & al, e. (2011). A Provenance-Based Infrastructure to Support the Life Cycle of Executable Papers. *Procedia Computer Science*, 648-657.

Stodden, V., & al., e. (2013). *Setting the default to reproducible. computational science research. SIAM News, 46, 4-6.* computational science research. SIAM News, 46, 4-6.

Zhao, J., & al, e. (2012). Why workflows break—Understanding and combating decay in Taverna workflows. *E-Science (e-Science), 2012 IEEE 8th International Conference on*, (old.: 1-9).

Own Publications Pertaining to Theses

1-B A. Bánáti, P. Kacsuk, M. Kozlovszky: Reproducibility analysis of scientific workflows; Acta Politechnica Hungarica, accepted, unpublished

2-B A. Bánáti, P. Kacsuk, M. Kozlovszky; Investigation of the Descriptors to make the Scientific Workflows reproducible; CINTI 2016 - 17th IEEE International Symposium on Computational Intelligence and Informatics. Budapest, Hungary, (IEEE Computational Intelligence Society), accepted, unpublished

3-B A. Bánáti, P. Kárász, P. Kacsuk, M. Kozlovszky: Evaluating the Average Reproducibility Cost of the Scientific Workflows, In: International Symposium on Intelligent Sytems and Informatics (SISY), 2016

4-B A. Bánáti, P. Kacsuk, M. Kozlovszky, M. Evaluating the Reproducibility cost of the scientific workflows Applied Computational Intelligence and Informatics (SACI), 2016 IEEE 11th Jubilee International Symposium on. IEEE, 2016

5-B A. Bánáti, P. Kacsuk, M. Kozlovszky; Classification of Scientific Workflows Based on Reproducibility Analysis; 39th International Convention on Information and Communication Technology, Electronics and Microelectronics: MIPRO 2016. Opatia, Rijeka: Croatian Society for Information and Communication Technology Electronics and Microelectronics (MIPRO'16),

6-B A. Bánáti, P. Kacsuk, M. Kozlovszky; Minimal sufficient information about the scientific workflows to create reproducible experiment; 19th IEEE International Conference on Intelligent Engineering Systems: INES 2015. Bratislava, 2015.09.03-2015.09.05. Bratislava: IEEE, 2015. pp. 189-194.

7-B A. Bánáti, P. Kacsuk, M. Kozlovszky; Four level provenance support to achieve portable reproducibility of scientific workflows; 38th International Convention on Information and Communication Technology, Electronics and Microelectronics: MIPRO 2015. Opatia, may, 2015. Rijeka: Croatian Society for Information and Communication Technology Electronics and Microelectronics (MIPRO'15), pp. 241-244.

8-B Eszter Kail, Anna Bánáti, Péter Kacsuk, Miklós Kozlovszky; Dynamic workflow support in gUSE; 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO'14). Opatija, Croitia, 2014.05.26-2014.05.30. Rijeka: IEEE, 2014. pp. 369-374.

9-B Eszter Kail, Anna Bánáti, Péter Kacsuk, Miklós Kozlovszky; Provenance based adaptive and dynamic workflows; CINTI 2014 - 15th IEEE International Symposium on Computational Intelligence and Informatics. Budapest, Hungary, 2014.11.19-2014.11.21. (IEEE Computational Intelligence Society) pp. 215-219.

10-B Eszter Kail, Anna Bánáti, Péter Kacsuk, Miklós Kozlovszky: Provenance Based Runtime Manipulation and Dynamic Execution Framework for Scientific Workflows, in Scientific Bulletin of The Politehnica University of Timisoara, 2016, Vol: 61(75) No: 1,