2010 24th IEEE International Conference on Advanced Information Networking and Applications

An Ant-Colony-Optimization based Approach for Determination of Parameter Significance of Scientific Workflows

Fakhri Alam Khan, Yuzhang Han, Sabri Pllana, and Peter Brezany
Department of Scientific Computing, Faculty of Computer Science, University of Vienna
Nordbergstrasse 15/C/3, A-1090, Vienna Austria
{khan|han|pllana|brezany}@par.univie.ac.at

***Abstract*— In the process of a scientific experiment a workflow is executed multiple times using various values of the parameters of activities. For real-world workflows that may contain hundreds of activities, each having several parameters, it is practically not feasible to conduct a parameter sensitivity study by simply following a ”brute-force approach” (that is experimental evaluation of all possible cases). We believe that a heuristic-guided approach enables to find a near-optimal solution using a reasonable amount of resources without the need for the evaluation of all possibilities. In this paper we present a novel methodology for determination of parameter significance of scientific workflows that is based on Ant Colony Optimization (ACO). We refer to our methodology, which is a customization of ACO for Parameter Significance determination, as ACO4PS. We use ACO4PS to identify (1) which parameter strongly affects the overall result of the workflow and (2) for which combination of parameter values we obtain the expected result. ACO4PS generates a list of all workflow parameters sorted by significance as well as is capable of generating a subset of significant parameters. We empirically evaluate our methodology using a real-world scientific workflow that deals with the Non-Invasive Glucose Measurement.**

I. Introduction

The emergence of service-oriented Grid infrastructures, which enables the use of heterogeneous and geographically distributed computational and data resources, has paved the road for the development of e-Science. Usually e-Science tasks that are to be executed on Grid infrastructures are expressed as *workflows*. e-Science workflows [1][2] are used to conduct scientific/technical experiments in various domains including medicine, meteorology, or astronomy.

Workflows in the scientific domain usually have exploratory nature, where a scientific phenomenon is studied or an answer to a scientific question is sought. In the process of a scientific experiment a workflow is executed multiple times using various values of the parameters of activities. Different combination of parameter values produce different workflow results. For a scientist it is relevant to know, (1) which parameters strongly affect the overall result of the workflow, and (2) for which combination of parameter values we obtain the expected and/or optimum results. Typically, for a scientist to get the desired result through brute force method (i.e. by trying all different value combinations of parameters to get the desired result) is a tedious and time consuming process. Consider the scenario that we have a workflow of N activities (the number of activities be small or large, depending on the complexity of workflow), and workflow has k number of parameters s i.e. $S_k = \{s_1, s_2, s_3, ..., s_k\}$, whereas each parameter has a value range V (value range can be continuous or discrete). In such a case where we want to come to the most significant parameter via a brute-force method, the number of times the experiment would need to be executed will be:

$$\begin{aligned}&(|V_{(s_1)}| \times |V_{(s_2)}| \times |V_{(s_3)}| \times ... |V_{(s_k)}|)\\ =&\prod_{i=1}^{k} |V_{(s_i)}| \end{aligned} \tag{1}$$

where, $V_{(s_i)}$ represents value range of i^{th} parameter. It is obvious from the equation (1) that applying a brute-force method for finding most significant parameters for complex workflows is impractical as the number of experiments grows very fast with the number of parameters and their corresponding value ranges. Problems, where the solution time increases significantly with the problem size, are known as NP-hard problems [3]. NP-hard problems are dealt by a class of heuristic algorithms that give near-optimal solutions within a reasonable time. In our case it is appropriate to search for a subset of most significant parameters using a heuristic method. It is of paramount importance to collect and provide information on parameters significance in order to enable the scientist to focus on these significant parameters and ignore the least significant parameters during future scientific experiments.

Ant Colony Optimization (ACO) was introduced in 1991 by Morco Dorigo [4] and in the meantime has been successfully applied to a number of NP-hard problems, such as Traveling Salesman Problem (TSP) [5], scheduling problems [6], or vehicle routing [7]. But, we have not found any evidence that ACO has been used for problems such as the determination of parameter significance of scientific workflows. In the context of scientific workflows, the estimation of parameter significance will streamline the experimental learning process and will enable the scientist to obtain his/her results efficiently.

In this paper we present our methodology for customizing

1550-445X/10 $26.00 © 2010 IEEE
DOI 10.1109/AINA.2010.24

ACO to enable the determination of the significance of all workflow parameters as well as to estimate the most significant parameter of the workflow. The major contributions of this paper include:

- Customization of ACO heuristic for determining parameters sensitivity in the context of scientific workflows.
- An ACO-based approach for exploration of the space of workflow parameters and their corresponding values.
- Implementation of our ACO based approach.
- Demonstration of usefulness of our approach using a real-world workflow for Non-Invasive Glucose Measurement [8].

The rest of the paper is organized as follows. In Section II related work is briefly described. In Section III we present in detail our modification of ACO and describe our approach, whereas in Section IV we give details of our methodology applied to a real time NIGM workflow. Finally, Section V concludes the paper and discusses merits, de-merits, and the future work.

II. Background and Related Work

In sub section. II-A we give details of ACO and related work, whereas in sub section. II-B we give details of why we have chosen to use ACO for determining workflows parameter significance.

A. ACO related work and background

Genetic Algorithms (GA) [9], Simulated Annealing (SA) [10], and Ant Colony Optimization (ACO) [4] all are search algorithms used in computational techniques to find optimal or near optimal results. Especially, genetic algorithms have been extensively used for scheduling of resources in grid workflows. S. Garg et al. [11] have proposed and used Linear Programming driven Genetic Algorithm (LPGA) for scheduling of multiple resources, Prodan et al. [12] used genetic algorithms to address recursive loop handling in static scheduling problems for scientific workflows, and Brandic et al. [13] used integer programing (IP) techniques for scheduling for QoS-aware scientific workflows.

Ant Colony Optimization (ACO) is a class of meta-heuristic algorithms, whose first flavor called Ant System, was proposed by Morco Dorigo and Co. [4] in early 1990s. ACO is modeled from the behavior of real ants, where ants independently search for food source and communicate via depositing a volatile chemical substance called pheromone on their path. Initially the ants travel on random path but over time they merge on some near optimal path and follow it. In ACO, ants are intelligent agents with extra capabilities of memory, determining quality of solution, pheromone deposition, and best node selection. ACO has been successfully applied to a number of combinational optimization problems such as Traveling Salesman Problem (TSP), job scheduling, vehicle routing, and network routing etc. In TSP [14], the ants are randomly located on cities. These ants then iteratively move to the next city in parallel. The next city is selected probabilistically and the selection of next city by an ant depends on the pheromone value (which is also known as attractiveness), as well as on desirability (heuristic information). Ant k probabilistically moves from city i to j via criteria shown below:

$$p_{ij}^{k} = \begin{cases} \dfrac{\tau_{ij}^{\alpha} + \eta_{ij}^{\beta}}{\sum_{cl \in tabu-list} \tau_{cl}^{\alpha} + \eta_{cl}^{\beta}} \\ 0 \end{cases} \tag{2}$$

where τ_{ij} represent the pheromone value on path ij, and η_{ij} means the heuristic information, whereas the parameters α and β control the relative importance of pheromone and heuristic information. After an ant moves from city i to city j, it stores the newly visited city in its memory and updates the pheromone value on the path ij as per following rule, and this process is known as local updating:

$$\tau_{ij} \leftarrow (1-\rho)\tau_{ij} + \rho\tau_0 \tag{3}$$

where ρ represents the evaporation constant of pheromone and τ_0 represents the initial pheromone value of a city. This process of selecting next city and updating the pheromone value on path between cities is done iteratively unless an ant completes visiting all the cities and returns to the start node. After ant completes the tour the newly path tour distance is compared to the existing best solution and if it is shorter than the existing solution then the pheromone value on this path is updated as per following criteria, this process is known as global updating:

$$\tau_{ij} \leftarrow (1-\rho)\tau_{ij} + \rho\Delta\tau_{ij} \tag{4}$$

where $\Delta\tau_{ij} = 1/L_{best}$, and L_{best} means best tour length.

Gambardella et al. [7] modified and applied the ACO to address the vehicle routing problems. In their approach they designed many colonies of ants, where one colony reduced the number of vehicles and other colonies reduced the traveled distances, whereas the communication between these colonies was handled through pheromone value. In [15], Gambardella et al. solve the sequential ordering problem with the help of modified ACO combined to 3-opt search procedure. Whereas Gutjahr [16] predicted the convergence speed of ACO using different construction graphs for subset problems.

To the best of our knowledge ACO has not been used for problems such as the determination of parameter significance of scientific workflows.

B. Comparison of ACO to other heuristic algorithms

Several heuristic algorithms such as genetic algorithms (GA) [9], tabu search (TS) [17], ant colony optimization (ACO)[4], simulated annealing (SA) [10], local search (LS) [18], branch and bound (B&B) exists and the decision to chose one depends on computational times, solution quality, complexity of algorithms, parameters, parameter interactions, runtime growth rate, and behavior when larger problems are used.

We have chosen ACO for determination of parameter significance of scientific workflows because Gagne et al. [19] have shown that ACO has competitive and advantageous behavior for lager problems as compared to the GA, SA, LS, and branch-and-bound algorithm. Gagne et al. [19] also concludes that for larger problems ACO has equal or better solution quality and computational times are appreciably lower.

Milena et al. [20] compared the efficiency of parallel computational models for ACO, SA, and GA for finding near-optimal solution in TSP. Milena et al. [20] concluded that in terms of speedup and solution quality ACO performed better.

III. Proposed Methodology

e-Science scientific workflows include many activities and every activity has certain parameters. The activities effect the result of the workflow and the effectiveness of an activity depends on parameter values. In complex e-Science workflows the parameters may grow from few to hundreds and it is important for a scientist to know which parameters effect the final output more significantly than others. For this purpose we modify and adapt the ACO and propose ACO for Parameter Significance (ACO4PS) to (i) find the significance of all parameters, (ii) find the most significant parameter, and (iii) find a subset of most significant parameters, to save the scientist efforts and time to fine-tune his/her workflow. To get the significance of workflow parameters we follow a three step approach:

1) Pre-ACO4PS preparation
2) Determine *cost* and $profit$ of all parameters
3) Apply ACO for parameter significance

These steps are explained in the following subsections.

A. Pre-ACO4PS preparation

As a preparation we represent our workflow in form of nodes and transitions. Every parameter of a workflow represents a node and the selection of next node represents transition. Pheromone value means attractiveness, that is greater the pheromone value on a parameter, greater are its chances to be selected, so initially all the parameters are assigned same amount of pheromone τ_0. ACO has two heuristic parameters α and β, and the best values for these heuristic preferences have been proved to be one and two respectively [21]. In the pre-preparation step we also tune and set the ACO parameters like number of ants, numbers of iterations, and number of repetitions. Furthermore, the nodes are designed such that the pheromone value is stored on nodes.

B. Determine cost and $profit$ of all parameters

For parameters significance we define two new ACO parameters, $cost$ and $profit$. The $cost$ and $profit$ have significant effect on our ACO4PS and are critical in guiding the ACO4PS to find the workflow parameters significance. Every parameter (node) has its own $cost$ and $profit$ associated with it.

Algorithm 1 Pseudo Code of our methodology

```
Input:
    $Workflow\ parameters\ with\ heuristic\ information$
Output:
    $List\ of\ most\ significant\ parameters,$
    $sorted\ by\ significance$
$Represent\ the\ parameters\ in\ form\ of\ nodes$
$Define\ and\ set\ \alpha\ and\ \beta$
$Set\ the\ ACO4PS\ parameters$
$Calculate\ Cost, \forall\ parameters$
$Calculate\ Profit,\ \forall\ parameters$
$Apply\ ACO4PS$
    for $every\ ant$ do
        $Place\ ants\ randomly$
        $Update\ ant\ memory$
        While($Item.In.Memory\ \leq\ entered\ subset$) do
            $Probabilistically\ select\ next\ node$
            $Do\ local\ pheromone\ update$
            $Update\ memory$
        $repeat$
        $Compare\ Solution$
        $Do\ global\ pheromone\ update$
    $End\ for$
    $Print\ best\ solution$
```

1) Cost of a parameter: The $cost$ of a parameter can be defined as "The computation time and resources a parameter takes". We believe that various factors have effect on a parameter's cost, but the one factor that has significant effect on parameter's cost and can be numerically quantified is the parameter's value range. A parameter with greater value range has greater cost as it will consume more time of scientist to try different combination of values and to get to the optimal workflow output and vice versa. To make the $costs$ of parameters relevant, we represent them as a distribution of 100, and use the formula below:

$$cost_i = \frac{range_i}{\sum_{k=1}^{n} range_k} \times 100 \tag{5}$$

where, $range_i$ is the value range of parameter i, and $\sum_{k=1}^{n} range_k$ is the sum of all parameters value range.

2) Profit of a parameter: Parameters are associated with workflow activities and these activities have impact on workflow output, and the $profit$ of a parameter can be realized if we know to which processes they are attached and how much critical those processes are to the overall output of workflow. Like $cost$, the $profits$ of all parameters is also represented as a distribution of 100 and is calculated as:

$$profit_i = \frac{profit_i}{\sum_{k=1}^{n} profit_k} \times 100 \tag{6}$$

where, $profit_i$ is the profit of parameter i, and $\sum_{k=1}^{n} range_k$ is the sum of all parameters profits. In cases where the $profit$ and $cost$ of parameters cannot be determined or numerically

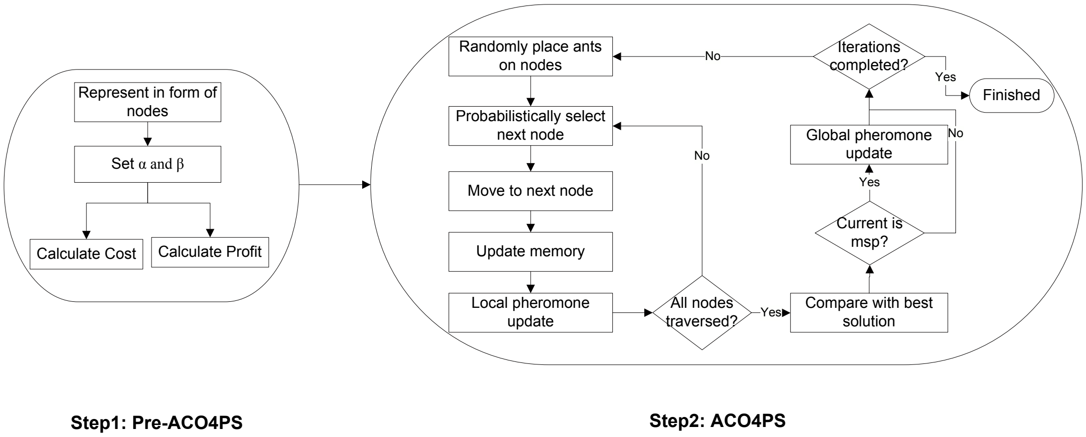

Fig. 1. Ant Colony Optimization for Parameter Significance (ACO4PS). Step1 shows the pre-ACO4PS parameters setup process, where a researcher determines and sets the parameters like α, β, $cost$, and $profit$. Parameters α and β control the relative importance of pheromone and heuristic information. Step2 shows ACO4PS, which is the procedure for obtaining a list of significant parameters sorted by significance.

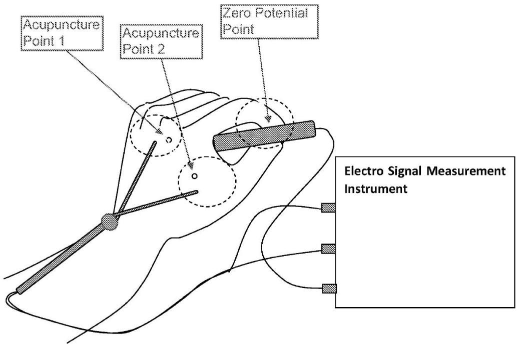

Fig. 2. Non-Invasive Glucose Measurement Technique

quantified, they are then taken as homogeneous, that is all the parameters are assigned same amount of $cost$ and $profit$ values.

C. Apply ACO for parameter significance

After properly representing the workflow in the form of nodes, tuning the ACO parameters, and determining the $cost$ and $profit$, we then apply our modified ACO4PS. Initially ants are randomly placed on nodes and then ants selects the next node (parameter) probabilistically according to following definition:

$$p_j^k = \frac{(\tau_j + profit_j)^\alpha + cost_j^\beta}{\sum_{c \notin mem}(\tau_c + profit)^\alpha + cost_c^\beta} \quad \forall\ c \notin mem \quad (7)$$

Equation (7) shows that the probability of parameter j being selected by ant k depends on the pheromone value (τ_j), associated $profit_j$, and $cost$ of parameter j. It is evident from Equation (7) that the probability of parameter j being selected is directly proportional to profit of parameter j ($profit_j$) and pheromone value (τ_j), and is inversely proportional to the cost of parameter j ($cost_j$). After an ant selects a parameter j, the ant performs local pheromone update, that is the pheromone value of parameter j is updated as per following rule:

$$\tau_j = (1-\rho)\tau_j + \rho\tau_0 \quad (8)$$

where ρ is the pheromone evaporation constant and τ_0 is the initial pheromone value of parameter j. When ant selects a parameter and updates the pheromone value of selected parameter then it stores the selected parameter in its memory and iteratively repeats this process until all the parameters are visited. On completion of a tour the cost of the completed tour is compared to the best existing tour, and if the new tour cost is found to be lower than the existing then it is set as best tour and the pheromone values on this tour are updated (global updating is performed) as:

$$\tau_j = (1-\rho)\tau_j + \rho\Delta\tau_j \quad (9)$$

where $\Delta\tau_j = 1/L_{best}$, and L_{best} means best tour cost. After all the ants complete their solutions, the best tour gives a sorted list of parameters, that is the parameter are sorted by their significance and the parameter with most significance comes first in the list and the least significant parameter appears last on this list.

The list obtained in this way can then be used by scientists or researchers to know:

- Which parameter is the most significant (that is by fetching the first parameter from the list)
- Which parameters to consider for changes to optimize the workflow result (scientist can select a subset of most significant parameters)
- Which parameters are least significant and can be ignored (that is the parameters belonging to the bottom half of the sorted list)

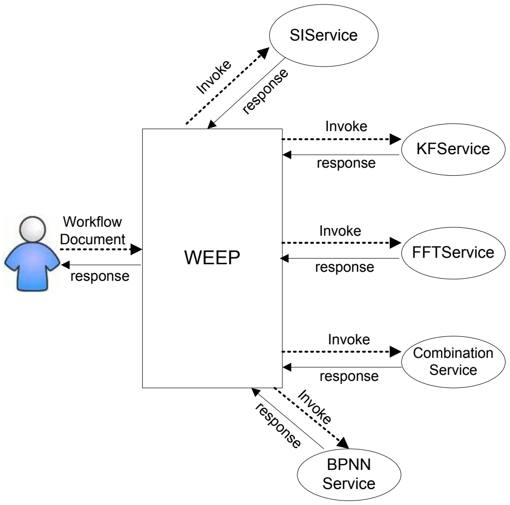

Fig. 3. NIGM Workflow execution by WEEP

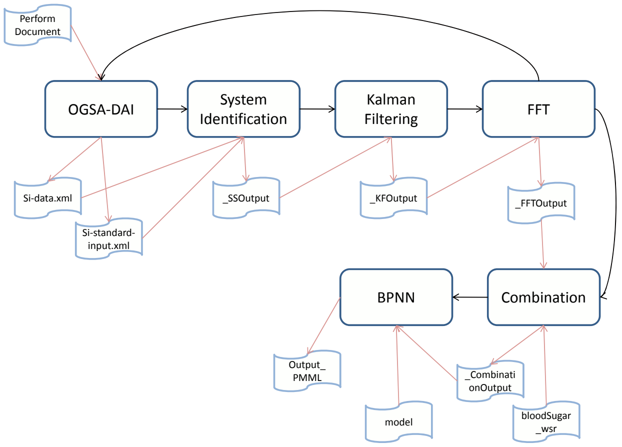

Fig. 4. NIGM Workflow

The above listed items mean that now the scientist is fully aware of his/her parameters affect on the workflow result and hence can modify them as per his/her preferences, which ultimately saves the time, efforts, and enhances the knowledge about the workflow being executed. Our proposed methodology has the advantage that it is sufficiently generalized and can be applied to any computational model, as well as to any number of scientific workflow parameters. Figure 1 visualizes our methodology, whereas the pseudo code of our methodology is shown in Algorithm. 1.

IV. Application to real time workflow

Before proceeding to experimental details, we give a brief introduction to the constraints and workflow domain in Subsection IV-A. In Subsection IV-B, we apply our proposed methodology on the selected workflow and present their results.

A. Non-Invasive Glucose Measurement Workflow (NIGM)

The non-invasive method for measuring human glucose values used in the NIGM workflow is based on the meridian theory, which is an important part of the Traditional Chinese Medicine (TCM) [22], according to which the human body has fourteen acupuncture meridians. Each of these longitudinally distributed lines on our human body have 24 main points, called source points. In order to prove the meridian theory with modern methods a number of special meridian measurement instruments have been developed. Analyzing meridian measurement data collected via these instruments with advanced data mining techniques and models can lead to important information about human illness state and other health relevant knowledge. The electro signal measurement instrument sends an electric signal (white noise) into one meridian source point and measures the corresponding signal output at another source point either on the same meridian or on another meridian. In particular a random electro signal with the maximal voltage less than 2.0 V is produced by the instrument. This process is illustrated in Figure 2. The measurements obtained in this process can, if analyzed by the meridian electro information transmission model, derive diabetic patients blood glucose values. The support with an enhanced Grid infrastructures allowing collaborative research with advanced data mining services, efficient data and workflow management services, and visualization services contribute to the progress in this domain.

The NIGM workflow as depicted by Figure 3 consists of five WS-I [23] and WSRF-compliant [24] Web services executed by WEEP[1] [25]. The NIGM workflow takes data obtained by measurements (see Figure 2) on patient meridians as input and then performs several activities, like *System Identification* [26], *Kalman Filtering* [27], and *Fast Fourier Transformation* [28] to remove noise from data and prepare it for accurately predicting blood glucose values.

NIGM efficiently computes the glucose value in patient blood by a method developed with the help of a Traditional Chinese Medicine (TCM) theory. The first three services compute eigenvalues for a given set of input value pairs (meridian measurements) as shown in Figure 4 along with all associated input(s) and output(s). The combination service combines the results given by first three services and the neural network service accurately predicts the blood glucose value. Computational models used in NIGM have several parameters. So it is very important for the user of the model to know about the significance of these parameters, if he/she wants to fine tune the model. To address this issue, we empirically validate our proposed methodology via estimating the most significant parameter of the NIGM workflow. Moreover, our methodology is sufficiently generic and be applied to any workflow and

[1] Workflow Enactment Engine Project (WEEP) is developed by Department of Scientific Computing at the University of Vienna. The source code of the engine is freely available at http://weep.gridminer.org for download under the terms and conditions of the Apache License, Version 2.0.

TABLE I
NIGM WORKFLOW PARAMETERS

Activity Name	Parameter Name	Type	Value range
SI	SI-input samples	discrete	$\leq$ no. of samples
KF	KF-input samples	discrete	$\leq$ no. of samples
FFT	FFT-input samples	discrete	$\leq$ no. of samples
Combination	Comb-input samples	discrete	$\geq$0
	BPNN-input samples	discrete	$\geq$1
BPNN	Learning rate	continuous	[0,1]
	Momentum	continuous	[0,1]

computational models.

B. The Experiment

As described in Section. III, we first prepare the data to be able to execute the ACO4PS on it. NIGM workflow contain five WSRF-compliant [24] services and each of these activities has parameter(s) associated with it. In Table. I we have shown the activities names along with parameters names, type, and allowed interval. The first four activities SI, KF, FFT, and Combination, each have one parameter whereas, neural network has three parameters. These seven parameter make up for seven nodes, each of which will contain $cost$, $profit$, and pheromone value (τ). Initially all parameters are assigned same amount of pheromone, whereas their $cost$ and $profit$ determination methods are described in the following Subsections IV-B.1 and IV-B.2. We also set the number of ants to ten, the number of iterations to ten, and the number of repetitions to four. The heuristic preferences α and β have been proved to best effective when α and β are set to one and two respectively [21].

1) Cost of parameters: As described in the methodology Section III, for $cost$ of a parameter we take its allowed value range into account and then represent their share over hundred to be relative. Table II shows the parameters along with their value ranges that we used to execute the NIGM workflow. By observing the value ranges of all parameters we noticed that $learning\ rate$ and $momentum$ are of continuous type, and hence their value range is practically infinite. To make $learning\ rate$ and $momentum$ discrete, we assumed their accuracy to be three decimal points and calculated their discrete value range as:

$$DVR = 10^{precision} \Rightarrow 10^3 = 1000 \tag{10}$$

where, DVR represents discrete value range. Now after having discrete value range for all parameters, we apply cost estimation formula:

$$\begin{aligned} cost_{SI} &= \frac{range_{SI}}{\sum_{k=1}^{7} range_k} \times 100 \\ &= \frac{1000}{26192} \times 100 \\ &= 3.818 \end{aligned}$$

We repeat the same approach to calculate costs for all parameters and are shown in Table. II.

TABLE II
NIGM WORKFLOW PARAMETERS *cost*

Activity Name	Parameter Name	Value range	Cost
SI	SI-input samples	1000	3.818
KF	KF-input samples	5000	19.090
FFT	FFT-input samples	8192	31.277
Combination	Comb-input samples	5000	19.090
	BPNN-input samples	5000	19.090
BPNN	Learning rate	1000	3.818
	Momentum	1000	3.818
		Total Cost	**100**

2) Profit of parameters: To find the $profit$ of a parameter, that is to know the criticalness of this parameter and associated activity to the workflow output, we follow the approach from our preliminary research work [29]. For every parameter we randomly select two values and execute the NIGM workflow, while keeping the rest of parameters unchanged during this process. In this way, we observe the change in workflow output, and calculate the estimated value of parameter i (PC_i). The PC_i is defined as "the ratio of percent change in workflow result to the percent change in parameter value" [29]:

$$PC_i = \frac{r_i}{p_i}$$

where, r_i is the percent change in result and p_i represents percent change in parameter. In this way we calculate the PC_i of all parameters. In the next step we calculate the normalization factor (NF) as:

$$\begin{aligned} NF &= \frac{\sum_{i=1}^{n} PC_i}{n} \\ &= \frac{3.516}{7} \\ NF &= 0.529 \end{aligned}$$

Finally, we calculate the $profit$ of a parameter as below:

$$\begin{aligned} profit_{SI} &= \frac{PC_{SI}}{NF} \\ &= \frac{0.238}{0.529} \\ profit_{SI} &= 0.45 \end{aligned}$$

Finally, we convert parameters profit values as a share of hundred. In Table III we show the profits of all NIGM workflow parameters. For further details of $profit$ calculation methodology please refer to our previous research work [29]. In Table III, we show the profits of all workflow activities parameters.

3) Apply ACO for parameter significance: Now that the NIGM workflow is properly represented in form of nodes and transitions, the heuristic preferences have been set, and we have determined the $cost$ and $profit$ of all parameters; we feed this data into our ACO4PS algorithm to get a sorted list

TABLE III

NIGM Workflow parameters $profit$

Parameter Name	Input Size	Change in result	Abs. Diff	r_i	p_i	PC_i	NF	Profit	Profit in 100
SI-input samples	1000 5000	2.218 2.181	0.037	1.19	5	0.238	0.529	0.45	6.77
KF-input samples	5000 10000	2.244 1.639	0.605	2.20	5	0.44	0.529	0.832	12.52
FFT-input samples	8192 16384	2.218 2.269	0.051	1.26	5	0.252	0.529	0.476	7.16
Comb-input samples	1000 5000	2.25 2.76	0.51	1.70	5	0.34	0.529	0.643	9.67
BPNN-input cycles	25 50	1.639 2.218	0.579	1.93	5	0.386	0.529	0.729	10.97
Learning rate	0.30 0.35	3.58 3.89	0.31	6.20	5	1.24	0.529	2.344	35.27
Momentum	0.30 0.35	2.28 2.44	0.16	3.10	5	0.62	0.529	1.172	17.63
Total						**3.516**			**100**

of parameters by significance. ACO4PS is able to generate two kinds of lists:

- Set of all parameters sorted by significance: This information is important for a scientist to see the most significant and least significant parameters.
- A subset of significant parameters: In this case the scientist specifies the size of the subset (that is the number of significant parameters he wants to deal with). The benefit of a subset of parameters is that the problem space reduces greatly and thus it enables the scientist to ignore least significant parameters straight away.

We executed the ACO4PS five times to get a subset of three most significant parameters using ten ants, seven nodes, ten iterations, and four repetitions. The results generated by ACO4PS were very close to the brute force method. As it can be seen from Table IV, that four out of 5 times, ACO4PS results in $learning\ rate$ and $momentum$ being the most significant parameters and once gave SI-input samples as second most significant parameter. It is noteworthy here that we did not considered the first node, as it is randomly selected when ACO4PS starts.

The result of ACO means that the $learning\ rate$ and $momentum$ are most significant parameters and a scientist has to pay attention to them if he/she would like to get optimal results from the NIGM workflow.

The reason that $learning\ rate$ and $momentum$ are most significant because they are associated with neural network model, which actually trains and then predicts the blood glucose value.

These results are in line with the NIGM workflow, as from our experiences we know that the first three models of the NIGM workflow remove noise from data, the combination service combines the value pairs, and the back propagation neural network activity is left with the core functionality of predicting blood glucose value.

TABLE IV

ACO4PS results for five tests

Test No.	Randomly selected parameter	Most Significant Parameter (MSP)	2nd MSP	3rd MSP
1	BPNN-input samples	Learning rate	Momentum	SI-input samples
2	Learning rate	Momentum	SI-input samples	BPNN-input samples
3	KF-input samples	Learning rate	Momentum	SI-input samples
4	FFT-input samples	Momentum	Learning rate	SI-input samples
5	Comb-input samples	Learning rate	Momentum	SI-input samples

V. Conclusions

Scientific workflows are used by researchers to understand a phenomenon or to answer scientific-relevant questions. The workflow results depend on values of parameters of workflow activities. To get optimal or near-optimal results, scientists use hit and trial process (that is they vary values of different parameters). Unfortunately, in complex workflows with large number of activities and parameters the chance to produce near optimal results decrease significantly and hence require huge amount of time, efforts, experiences, and luck to get optimal results. It is obvious from this scenario that it is of paramount importance to collect and provide information on parameters significance, so that the search space can be reduced significantly and to enable the scientist to focus on these significant parameters and ignore the least significant parameters.

In this paper we have proposed and implemented a novel ACO-based method for determination of parameter significance of scientific workflows. Our approach has the advantage of taking into account the heuristic information as well as experimental experiences of a scientist. Moreover, our approach is sufficiently generalized and can be applied to any workflow with any number of parameters. Our Ant Colony Optimization for Parameter Significance (ACO4PS) technique is capable of producing, (i) a list of estimated significance values of all parameters sorted by significance and, (ii) a subset of most significant parameters from workflow.

An advantage of having a sorted list of significant parameters is that it gives an insight on workflows parameter significance and enables a scientist to decide which parameters to consider and which to ignore. On the other hand, if a scientist is interested in some subset of most significant parameters, ACO4PS can generate this subset depending on the subset size provided by a scientist. It is relevant to clarify that our technique can be applied if the expected or desired results of workflows are known.

We have empirically validated our methodology using a real-world workflow for Non-Invasive Glucose Measurement (NIGM). We experimentally demonstrated that in cases where we know the expected overall workflow result (such as NIGM workflow in our case study), we can reason about the quality of results of a certain activity by studying how this activity affects the overall workflow result (that is, which parameter values of an activity leads to the overall workflow result that is close the expected result).

Data mining and scientific workflows are executed multiple times so it is very important from implementation point of view to store the parameters significance information in a place where it can easily be accessed by researchers. For this reason, we store the parameters significance information along side our provenance information [30], so that these information can be used when the workflow is re-executed and/or re-enacted.

In future, we will extend our technique to estimate parameters significance for workflows where the expected outputs are not known in advance. We will also focus to devise a methodology to reduce the value ranges of each significant parameter effectively, so that the search space can be further reduced, that is first reduce the set of relevant parameters (by establishing the most significant parameters out of the whole lot) and then reduce the value ranges of these selected most significant parameters.

ACKNOWLEDGMENT

This research is done in the context of our workflow provenance research [30] and ADMIRE Project [31]. As a starting point it has been implemented and tested on the NIGM workflow [8] of CADGrid Project [32].

REFERENCES

[1] e Science, "UK e-Science," http://www.rcuk.ac.uk/escience.

[2] I. J. Taylor, E. Deelman, D. B. Gannon, and M. S. (Eds.), *Workflows for e-Science: Scientific Workflows for Grid*. Springer, 2006.

[3] D. S. Hochbaum, Ed., *Approximation algorithms for NP-hard problems*. Boston, MA, USA: PWS Publishing Co., 1997.

[4] A. Colorni, M. Dorigo, and V. Maniezzo, "Distributed Optimization by Ant Colonies," in *Towards a Practice of Autonomous Systems: Proceedings of the First European Conference on Artificial Life*, F. J. Varela and P. Bourgine, Eds. Cambridge, MA: MIT Press, 1992, pp. 134–142.

[5] L. M. Gambardella and M. Dorigo, "Solving Symmetric and Asymmetric TSPs by Ant Colonies." IEEE Press, 1996, pp. 622–627.

[6] S. van der Zwaan, A. R. Pais, and C. Marques, "Ant Colony Optimisation for Job Shop Scheduling," 1999.

[7] L. M. Gambardella, ric Taillard, and G. Agazzi, "MACS-VRPTW: A Multiple Colony System For Vehicle Routing Problems With Time Windows," in *New Ideas in Optimization*. McGraw-Hill, 1999, pp. 63–76.

[8] I. Elsayed, J. Han, T. Liu, A. Wohrer, F. A. Khan, and P. Brezany, "Grid-Enabled Non-Invasive Blood Glucose Measurement," in *Computational Science - ICCS 2008*. Krakow, Poland: LNCS 5101, June 2008, pp. 76–85.

[9] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1989.

[10] S. Kirkpatrick, C. D. Gelatt, Jr., and M. P. Vecchi, "Optimization by Simulated Annealing," *Science*, vol. 220, pp. 671–680, 1983.

[11] S. K. Garg, P. Konugurthi, and R. Buyya, "A Linear Programming Driven Genetic Algorithm for Meta-Scheduling on Utility Grids," *CoRR*, vol. abs/0903.1389, 2009, informal publication.

[12] R. Prodan and T. Fahringer, "Dynamic scheduling of scientific workflow applications on the grid: a case study," in *SAC*, 2005, pp. 687–694.

[13] I. Brandic, S. Pllana, and S. Benkner, *Specification, Planning, and Execution of QoS-aware Grid Workflows*. Wiley Inc., 2009.

[14] E. Lawler, Ed., *The traveling salesman problem: a guided tour of combinatorial optimization*. John Wiley and Sons Inc, 1985.

[15] L. M. Gambardella and M. Dorigo, "HAS-SOP: Hybrid Ant System for the Sequential Ordering Problem," Tech. Rep., 1997.

[16] W. J. Gutjahr, "On the finite-time dynamics of ant colony optimization," *Methodology and Computing in Applied Probability*, vol. 8:105–133, 2006.

[17] F. Glover and F. Laguna, *Tabu Search*. Norwell, MA, USA: Kluwer Academic Publishers, 1997.

[18] P.-C. Kanellakis and C. H. Papadimitriou, "Local Search for the Asymmetric Traveling Salesman Problem," *OPERATIONS RESEARCH*, vol. 28, no. 5, pp. 1086–1099, 1980.

[19] C. Gagne, W. L. Price, and M. Gravel, "Comparing an ACO Algorithm with Other Heuristics for the Single Machine Scheduling Problem with Sequence-Dependent Setup Times," *The Journal of the Operational Research Society*, vol. 53, no. 8, pp. 895–906, 2002. [Online]. Available: http://www.jstor.org/stable/822917

[20] M. Lazarova and P. Borovska, "Comparison of parallel metaheuristics for solving the TSP," in *CompSysTech '08*. New York, NY, USA: ACM, 2008, pp. II.12–1.

[21] M. Dorigo and T. Sttzle, *Ant Colony Optimization*. Cambridge, MA: MIT Press, 2004.

[22] G. Maciocia, *The Foundations of Chinese Medicine: A Comprehensive Text for Acupuncturists and Herbalists*. Elsevier Churchill: Livingstone, 2005.

[23] K. Ballinger, D. Ehnebuske, C. Ferris, M. Gudgin, C. K., Liu, M. Nottingham, and P. Yendluri, "WS-I Basic Profile Version 1.1," http://www.ws-i.org/Profiles/BasicProfile-1.1.html, 2006.

[24] I. Foster, J. Frey, S. Graham, S. Tuecke, K. Czajkowski, D. Ferguson, F. Leymann, M. Nally, T. Storey, and S. Weerawaranna, "Modeling Stateful Resources with Web Services," v1.1. Technical report, Globus Alliance, 2004.

[25] I. Janciak and P. Brezany, "Workflow Enactment Engine for WSRF-compliant services orchestration," in *The 9th IEEE/ACM International Conference on Grid Computing (Grid 2008)*. Tsukuba, Japan: IEE/ACM, September 29 - October 1 2008.

[26] L. Ljung, *System Identification - Theory For the Use*. Upper Saddle River, N.J.: PTR Prentice Hall, 1999.

[27] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," ASME - Journal of Basic Engineering, 1960.

[28] S. Bochner and K. Chandrasekharan, *Fourier Transforms*. Princeton Book Comp., 2001.

[29] F. A. Khan, Y. Han, S. Pllana, and P. Brezany, "Estimation of Parameters Sensitivity for Scientific Workflows," in *Proceedings of International Workshop on Advanced Distributed and Parallel Network Applications at ICPP 2009, Vienna, Austria*. IEEE Computer Society, 2009.

[30] F. A. Khan, Y. Han, S. Pllana, and P. Brezany, "Provenance Support for Grid-Enabled Scientific Workflows," in *Proceedings of Fourth International Conference on Semantics, Knowledge and Grid*. Beijing, China: IEEE, December 3-5 2008, pp. 173–180.

[31] ADMIRE, "Advanced Data Mining and Integration Research for Europe," http://www.admire-project.eu/.

[32] CADGrid, "China Austria Data Grid Project," http://www.par.univie.ac.at/project/cadgrid/.