Optimal Workflow Scheduling in Critical Infrastructure Systems with Neural Networks

S. Vukmirović*, A. Erdeljan, L. Imre, D. Čapko

Faculty of Technical Sciences.
University of Novi Sad. Novi Sad, Serbia
*srdjanvu@uns.ac.rs

ABSTRACT
Critical infrastructure systems (CISs), such as power grids, transportation systems, communication networks and water systems are the backbone of a country's national security and industrial prosperity. These CISs execute large numbers of workflows with very high resource requirements that can span through different systems and last for a long time. The proper functioning and synchronization of these workflows is essential since humanity's well-being is connected to it. Because of this, the challenge of ensuring availability and reliability of these services in the face of a broad range of operating conditions is very complicated. This paper proposes an architecture which dynamically executes a scheduling algorithm using feedback about the current status of CIS nodes. Different artificial neural networks (ANNs) were created in order to solve the scheduling problem. Their performances were compared and as the main result of this paper, an optimal ANN architecture for workflow scheduling in CISs is proposed. A case study is shown for a meter data management system with measurements from a power distribution management system in Serbia. Performance tests show that significant improvement of the overall execution time can be achieved by ANNs.

Keywords: Critical infrastructure system, neural network, grid computing.

1. Introduction

Modern societies have reached a point where everyday life relies heavily on the reliable operation and intelligent management of critical infrastructures, such as electric power systems, telecommunication networks, water distribution networks, and transportation systems. Monitoring and controlling such systems is becoming more challenging as their size, complexity and interactions are steadily growing [1, 2]. Moreover, these critical infrastructures are susceptible to natural disasters, frequent failures and malicious attacks. Characterized by direct and transitive interdependencies, these systems have become so interconnected that disruption of one may lead to disruptions in all [3].

Supervisory control and data acquisition (SCADA) systems are becoming more and more resource-demanding because their scope is becoming wider. This trend is especially visible in critical infrastructure systems (CISs), e.g., power, gas and water distribution systems. Processing and storage of measured data becomes a problem since more and more measured values are introduced in the controlled system. This paper proposes a system architecture which can solve this problem.

CISs have some special requirements, mainly because they are working in real-time environment and have to communicate with end devices (sensors and actuators). Additionally, they have to store very large volumes of time-series data about variable values. This makes workflow schedule control for CISs a special problem.

The development of the proposed workflow scheduling system was driven by requirement to develop a scheduling system for a commercial meter data management (MDM) system. An MDM system has to manipulate a large number of workflows in a distributed environment hence the proposed architecture could be used to reduce hardware requirements by optimizing resource usage [4].

A workflow is loosely defined as the automation of a process to coordinate people, data and tasks. Business workflows have been researched and utilized over many years; recently, the need for workflows has been also recognized in science thus several specialized workflow engines have been developed. [5]

Scheduling is the decision process that assigns application components to available resources to optimize various performance metrics. Grid workload management and scheduling subsystems enable the efficient distribution of tasks in grid systems and allow their transparent execution by hiding the complexity of grid infrastructure.

This paper proposes a novel hierarchical neural network model. Its goal is to solve the problem of workflow scheduling in large scale critical infrastructure systems. CIS clients initiate a request for data processing by calling appropriate server functions. The workflow manager running at the server decomposes these functions into tasks and sets an execution schedule (see Figure 1). As system resources are limited, an efficient task mapping (i.e., execution scheduling) becomes a fundamental concern. The aim of this work is to develop a novel system architecture and combine it with enhanced algorithms in order to boost the efficiency of the solution.

The rest of the article is organized as follows: in Section 2, we overview related work; Section 3 describes specific characteristics of the critical infrastructure system (CIS). In Section 4, we describe the architecture of the proposed system; Section 5 shows the basics of ANNs and describes the architecture of the network used in this paper. Section 6 describes the test results and future work, and Section 7 concludes the article.

2. Related work

Civil infrastructure provides the range of essential services generally necessary to support a nation's economy and quality of life – arguably, entire economies rely on the ability to move goods, people, and information safely and reliably. Consequently, it is of the utmost importance to government, business, and the public at large that the flow of services provided by a nation's infrastructure continues unimpeded in the face of a broad range of natural and manmade hazards [6, 7].

Some researchers argue that grid computing is the future computing paradigm for enterprise applications. Large scale grids are complex systems, composed of thousands of components belonging to distributed domains [8]. With the

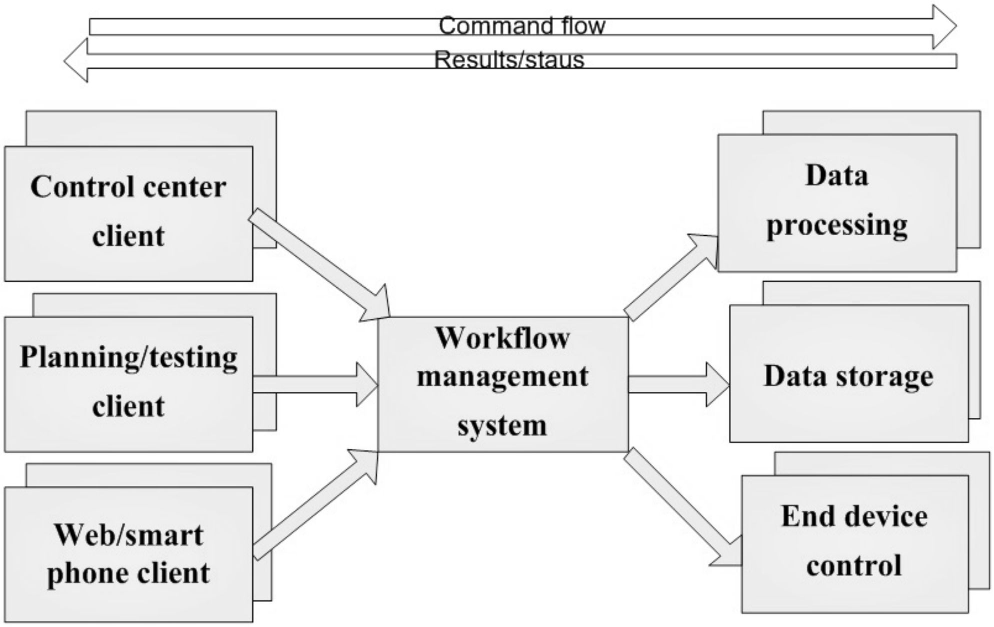

Figure 1. Overall system architecture.

development of large scale high-speed networks, the grid has become an attractive computational platform for high-performance parallel and distributed applications [9].

Grid infrastructure is used both to share expensive and centralized resources among many scientists as well as to integrate experimental data sources with the simulation codes necessary to analyze them [10].

SCADA systems have come a long way from simple process visualization, and distributed SCADA systems are very well-described in scientific papers [11]. Process data found their way to the Internet [12] or even mobile phones [13].

Meter data management (MDM) systems are responsible for controlling smart meters for individual power consumers. The latest customer requirements show that a few millions of these smart meters will be implemented in complex MDM systems and this number will have to be multiplied by dozens of variables per meter [14]. This kind of requirements calls for a new breed of CISs with a new architectural approach, and the grid approach seems like the right approach that promises good performance.

3. Critical infrastructure system architecture

In most general scientific workflow scheduling approaches only two parameters are considered for the workflow schedule: the computing power of the node and the network bandwidth between nodes. In a large CIS workflow there is an extended set of grid parameters that should be considered. We have analyzed the architecture and requirements for large scale distributed CISs (electric energy distribution systems, gas and water distribution systems, etc.) We have found that grid nodes in this type of CISs could be one of the following types:

Processing node – used for business calculation and data preprocessing, mainly for reporting and offline analysis of the system.

Objects database node – This node is used for storing static data from the distributed CIS. Most commonly, it hosts some kind of relational database for better search performances.

Time-series node – This node hosts data about process variables. Since the value of the process variable might change quite often, the number of variables can be extremely large.

Communication node – This computer node is responsible for communication with end devices. In many cases, this node communicates by a wireless network which means that the current status of communication media could significantly impact the performance of the whole grid.

This means that for developing an optimal strategy for a workflow scheduling system feedback, signals will have to be used for all four kinds of performance indicators. In this way the workflow scheduler will be able to create an optimal strategy for the dynamic state of the system.

Some other variables that could be used for developing an optimal strategy are the average execution time for previous instances of the same workflow, the priority of the workflow (for alarms purposes), and cancellation workflows – if some new event in the system could cancel the execution of a previously stated workflow. These new optimization constraints could be easily introduced in the proposed architecture.

Some characteristic CIS workflows follow: data processing, storing and querying data about devices and about process variable values, communication with end devices in order to get current values or send commands, etc. Workflows chosen for this paper are derived from real life CIS use cases. The following conditions are implied:

1. All workflows are independent from each other.
2. All workflows have the same priority.
3. Every node processes only one workflow at a time.
4. Every workflow is processed on one node at a time.
5. Workflow of the same type has the same execution time on each node.

For this paper we have implemented five workflows which use different types of grid nodes. [15] All of them have two phases which are executed on different nodes so there is a constant migration of workflows between nodes. Figure 2 shows an example execution process of five types of nodes on grid nodes.

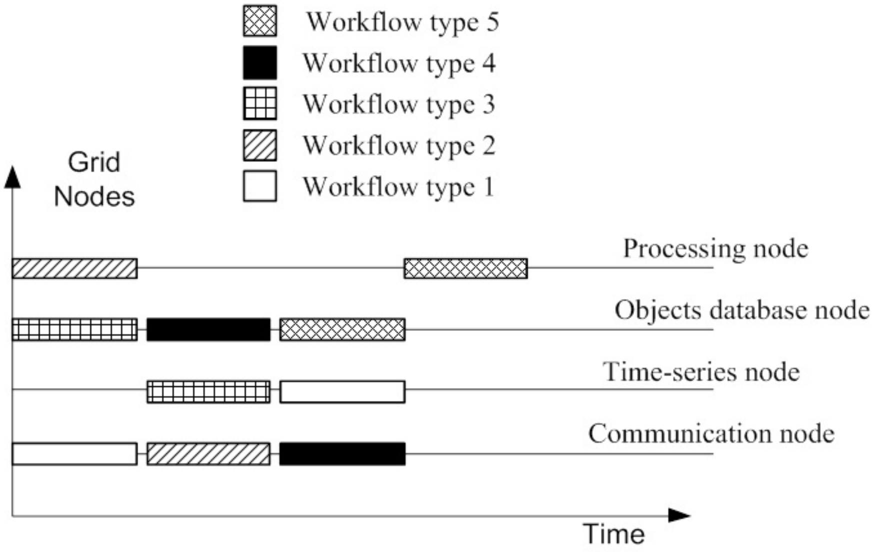

Figure 2. Workflow execution migration between nodes.

All the workflows in the described systems have two migration stages. In more complex systems some workflows could have more migration steps. In some cases, one workflow could be migrated back to the same node (if, for example, processing is done prior to and after communication with end devices). We anticipate that in these cases the use of soft computing methods will generate even more performance improvement.

4. Proposed architecture

The proposed architecture takes into account the dynamic nature of a real-world CIS and uses the grid environment approach for detecting and responding to the changing environment in CISs. The proposed framework is shown in Figure 3 and provides the required support for feedback from the scheduling process [16].

As Figure 3 shows, the framework consists of three major components: the grid manager, the workflow scheduler, and the application manager.

The workflow scheduler is responsible for deciding which queued workflow will be executed next. This decision is based on control variables from the monitored grid. This component is of the most interest in our research, since decision making is done there. We anticipate that significant performance improvements could be achieved by using soft computing optimization methods in this component.

The **grid manager** is responsible for monitoring the state of the grid and controlling the execution of workflows in it. It collects values of control variables that are used in the workflow scheduler.

The **application manager** is responsible for receiving workflows in execution time. It then queues them and works with the workflow scheduler to determine the right time to send workflows for execution in the grid. After receiving instructions from the workflow scheduler, it sends workflows to the grid manager and finally the results of the workflow execution are written into a database and sent to a client which initializes the workflow execution. The application manager is the only component visible from outside the proposed system – thereby, it hosts service interfaces for the initialization of the workflow execution by the client.

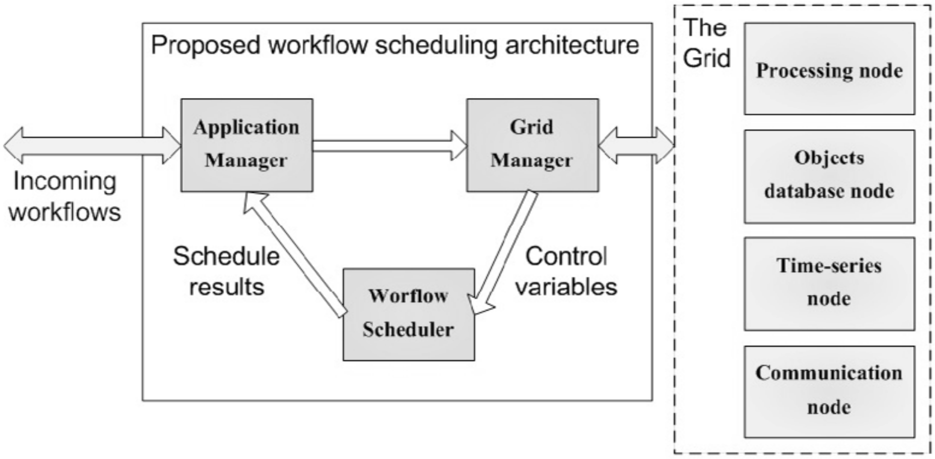

Figure 3. Proposed architecture of the system.

5. Neural network architecture

The classical methods for forecasting include regression and state-space methods. The most modern methods include expert systems, fuzzy systems, evolutionary programming, artificial neural networks (ANNs) and various combinations of these tools. Among the many existing tools, ANNs have received much attention because of their clear model, easy implementation and good performance [9]. ANNs have become popular in various real-world applications including prediction and forecasting, function approximation, clustering, speech recognition and synthesis, pattern recognition and classification, and many others. [10]

The multi-layer feed forward network will learn to associate the given output vectors with input vectors by adjusting their weights, which are based on the error at the output. The weight modification algorithm is the steepest descent algorithm (often called the delta rule) to minimize a nonlinear function [17]. The algorithm is called back-error propagation or back propagation because errors are propagated back through the hidden layers.

In workflow scheduling applications, the main function of ANNs is to predict the duration of the execution of a specified type of workflow based on the current state of the grid.

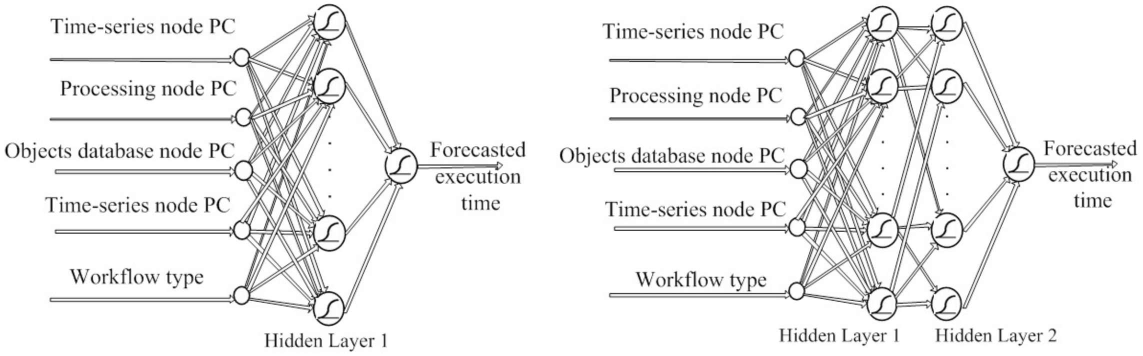

Figure 4. ANN architecture (with one and two hidden layers).

Figure 4 represents two types of ANNs that are used for workflow scheduling in CISs. The ANN predicts the execution time of the specified workflow type based on the current state of the grid nodes. Two different architectures are taken into consideration: the first one with one hidden layer (with 5 and 10 neurons in it) and the second one with two hidden layers (with 5 and 10 neurons in each).

5.1 Proposed neural network architectures

Five inputs for artificial neural network are selected for the ANN that proved to be the most important for optimal scheduling. The output of the ANN is chosen in a way that will uniquely identify the type of workflow that should be started in the current grid state. Standard Windows performance counters are used for monitoring appropriate features of the grid node.

The selected ANN inputs are

- Current CPU usage from the **processing node** [%]
- Current query execution time for the **objects database node** [%]
- Current read/write duration for the **time-series node** [%]
- Network bandwidth – **communication node** [%]
- **Workflow type -** provided by an application [one of the five provided]

The selected ANN output is

- **Forecasted execution** time [s]

All proposed ANN architectures used *the logarithmic sigmoid transfer function* for activation in all layers.

$$\mathrm{logsig(x)} = \frac{1}{(1+e^{-x})}$$

6. Results and discussion

The test environment was developed based on the proposed architecture. One computer node was dedicated for each type of node specific for CIS systems. To achieve full experiment credibility we have pre-defined the set of applications (i.e., workflows) which were input in the considered system under test.

For testing purposes, we scheduled five types of CIS workflow schedules. The number of workflows was from ten to one thousand. We measured the total execution times of all workflows with the following scheduling reasoning:

No optimization – the workflows were scheduled in the order in which they arrived in the input queue. This scenario was used for comparison.

Hard-coded optimization – in this scenario, we followed the simple reasoning that the workflows would execute more efficiently if they were of different types. Hence, when the time came for the new workflow to be executed, the workflow scheduler examined which workflow types were currently executing. Then the workflow scheduler tried to find the workflow of the type which was least deployed and tried to start executing that workflow.

One hidden layer ANN optimization – in this scenario, the system used the output of an ANN with one hidden layer to select which type of workflow should be scheduled in the next step.

Two hidden layers ANN optimization – in this scenario, the system used the output of an ANN with two hidden layers to select which type of workflow should be scheduled in the next step.

The most important part of any neural network is the learning set. The training data for the ANN consisted of 1000 sets of values, 90 percent of which was used for training while 10 percent was used for verification.

A case study was created for a meter data management system with real measurements collected from smart meters in Serbia. The workflows were executed on the grid in the order in which they arrived in the system (without any optimization of the scheduling process). At this stage, the execution time of the workflows, their type and the state of the grid nodes at the start of the workflow execution were recorded.

The test results are shown in Table 1. The test results clearly show two things: first, the use of ANNs in the workflow scheduling process for large scale CISs can improve the overall performance.

Number of workflows	Time of execution –simple ANN optimization [s]	Time of execution hard-coded Optimization [s]	ANN with 1 hidden layer		ANN with 2 hidden layer	
			Time of execution (5 -5-1)ANN optimization [s]	Time of execution (5 -10-1)ANN optimization [s]	Time of execution (5 -5-5-1)ANN Optimization[s]	Time of execution (5 -10-10-1)ANN optimization [s]
10	17	15	14	14	14	14
50	99	90	82	77	79	79
100	199	176	172	163	166	168
250	501	422	415	412	425	430
500	1007	868	838	827	846	866
1000	2018	1762	1702	1671	1744	1730

Table 1. Test results for the speed of the workflow execution.

7. Conclusions

This paper proposes an architecture for optimal workflow scheduling in critical infrastructure systems (CISs). The architecture is based on the feedback from the grid which consists of performance monitoring indicators. This way the workflow schedule can optimize the execution time of the workflows.

This paper also explains specific features of large scale CISs which make the proposed architecture different from standard grid scheduling systems. A set of control variables is also proposed, which is an additional contribution of this work.

The workflow scheduling component, which is responsible for starting appropriate workflows, uses an artificial neural network for choosing the optimal scheduling strategy. The performance analysis shows that this approach significantly boosts the performance of the whole system. Different ANN architectures were tried experimentally and the configuration with one hidden layer and ten neurons proved to be adequate for workflow optimization in CISs.

This scheduling mechanism can significantly reduce investments in hardware as the same results could be achieved with less grid nodes.

Future work on this topic could include testing of other types of ANN training algorithms and the application of other soft algorithms (genetic algorithm, ant colony optimization, particular swarm optimization, etc.). Future research could deal with workflows which have more than two steps in their execution and have different time of execution on each node. Test cases when migration of workflows between nodes is not predictable could also be an interesting topic for future research.

References

[1] Chandra J., Robust and resilient Critical Infrastructure Systems, Proceedings of the 38th Annual Hawaii, 2005, pp 62-66

[2] Polycarpou, M., Ellinas, G., Kyriakides, E., Panayiotou, C., Intelligent Health Monitoring of Critical Infrastructure Systems, Complexity in Engineering, 2010. COMPENG '10, March 2010, pp. 18-20

[3] Han C., Liu L., Rong M. et al, Addressing criticality levels in critical infrastructure system, Systems, Man and Cybernetics, October 2009, pp. 3965 – 3970

[4] Vukmirović S., Erdeljan A., Lendak I., Čapko D. : A novel software architecture for Smart Metering systems, Journal of Scientific and Industrial Research (JSIR), 2010, Vol. 2010, No 12, pp. 937-941

[5] Erdeljan A., Trninić N., Čapko D., An OPC Data Access server designed for large number of items, 6th International Symposium Interdisciplinary Regional Research (Hungary, Romania, Yugoslavia) ISIRR 2002, Novi Sad, 2002.

[6] Little R., Toward more robust infrastructure: observations on improving the resilience and reliability of critical systems, System Sciences, 2003. Proceedings of the 36th Annual Hawaii, 9 pp.

[7] Linda, O.,Vollmer, T.,Manic, M., Neural Network based Intrusion Detection System for critical infrastructures, International Joint Conference on Neural Networks, 2009., pp. 1827 – 1834

[8] Otavio A.S. Carpinteiro, Rafael C. Lemeb, Antonio C. Zambroni de Souza, Carlos A.M. Pinheiro, Edmilson M. Moreira, “Long-term load forecasting via a hierarchical neural model with time integrators”, Electric Power Systems Research 77 (2007) 371–378

[9] Yamin HY, Shahidehpour SM, Li Z. “Adaptive short-term electricity price forecasting using artificial neural networks in the restructured power markets”, Electrical Power Energy Systems 2004; 26:571–81.

[10] Paras Mandal, Tomonobu Senjyu, Toshihisa Funabashi, “Neural networks approach to forecast several hour ahead electricity prices and loads in deregulated market”, Energy Conversion and Management 47 (2006) 2128–2142

[11] Agrawal R. K., Merh B., Fatnani P., Yadav R., Gangopadhyay S., SCADA functionality for control operations of INDUS-2, 10th ICALEPCS Int. Conf. on Accelerator & Large Expt. Physics Control Systems. Geneva, 10 - 14 Oct 2005, PO1.098-8 (2005)

[12] Qiu B., Gooi H. B., Web-Based SCADA Display Systems (WSDS) for Access via Internet , Power Systems, IEEE Transactions on, May 2000, Issue 2 , pages 681-686

[13] Ozdemir E., Karacor M., Mobile phone based SCADA for industrial automation, ISA Transactions, Volume 45, Number 1, January 2006, pages 67–75

[14] Vukmirovic S., Lukovic S., Erdeljan A., Kulic F., Software architecture for Smart Metering systems with Virtual Power Plant, The 15th IEEE Mediterranean Electrotechnical Conference, Valletta, Malta, April 2010

[15] Vukmirović S., Erdeljan A., Čapko D., Lendak I., Nedić N. : Optimization of workflow scheduling in Utility Management System with hierarchical neural network, 2011Vol. 4, pp. 672-679

[16] Vukmirović S., Nedić N., Erdeljan A., Lendak I., Čapko D. : A Genetic Algorithm Approach for Utility Management System Workflow Scheduling, Information technology and control, 2010, Vol. 39, No 4, pp. 310-316

[17] Rumelhart D., Hinton G., Williams R. Learning internal representations by error propagation, Parallel Distributed Processing, Vol. 1, Chapter 8, MIT Press, 1986.