Intelligent Automatic generation control: multiagent Bayesian networks approach

H. Bevrani, F. Daneshfar, P. R. Daneshmand

Intelligent Automatic Generation Control: Multi-agent Bayesian Networks Approach

H. Bevrani, *Senior Member, IEEE*, F. Daneshfar, and P. R. Daneshmand

Abstract—**A new intelligent agent based control scheme, using Bayesian networks (BNs), to design automatic generation control (AGC) system in a multi-area power system is addressed. Model independency and flexibility in specifying the control objectives, make the proposed approach as an attractive solution for AGC design in a real-world power system.**

The proposed control scheme is tested in simulation on a three areas power system and shows desirable performance. The results are also compared with the multi-agent reinforcement learning based AGC design technique.

Index Terms—**AGC, Bayesian networks, Frequency deviation, Multi-agent system.**

I. INTRODUCTION

THE conventional automatic generation control (AGC) designs are usually suitable for specific operating points, it seems that these AGC synthesis methodologies are not more efficient for modern power systems, considering increasing size, changing structure, emerging renewable energy sources, microgrids, and new uncertainties. Most of conventional AGC design strategies provide model based controllers that are highly dependent to the considered specific models, and are not usable for large-scale power systems with nonlinearities, undefined and uncertain parameters [1]. In new environment, design of intelligent AGC schemes that are more adaptive and flexible than conventional ones is become an appealing approach.

Several intelligent techniques are used for the AGC design in the power systems; however there are just few reports on the AGC synthesis in a modern environment using of intelligent multi-agent systems [2-5].

Bayesian Network (BN) [6] within a multi-agent system can be considered as a powerful adaptive control technique for the purpose of AGC design. The BNs provide suitable tools for knowledge representation and inference under conditions of uncertainty, and they have been successfully applied in a variety of real-world engineering issues. It has been effectively used to incorporate expert knowledge and historical data for revising the prior belief in the light of new evidence in many fields. The main feature of the BN is that it is possible to include local conditional dependencies into the model, by directly specifying the causes that influence a given effect.

Since, the BNs are based on learning methods they are independent of environment conditions and can consider all kind of environment disturbances. Therefore, they are not model based and can be easily scalable for large scale systems, such as power systems. The BNs can also work well in nonlinear conditions and nonlinear systems. A major advantage of the BN over many other types of predictive and learning models, such as neural networks, is that its structure represents the inter-relationships among the data set attributes.

This paper addresses the AGC design using an agent based solution for a large interconnected power system concerning the integration of wind power units. Here, a BNs multi-agent control structure is proposed. It has one agent in each control area that provides an appropriate control signal according to load disturbances and tie-line power changes received from other areas. The results are compared with the authors' previous work on application of multi-agent reinforcement learning based AGC design method [3, 5].

The above technique has been applied to the AGC problem in three control areas power system. The organization of the rest of the paper is as follows. In Section 2, a brief introduction to BNs is given. In Section 3, the power system test example is introduced. In Section 4, the proposed intelligent AGC technique using BN and the structure of a network which the above architecture is implemented for are discussed. Simulation results are provided in Section 5 and the paper is concluded in Section 6.

II. PRELIMINARIES

A. Graphical Models

Graphical models provide a natural tool for dealing with two problems that occur throughout applied mathematics and engineering - uncertainty and complexity - and in particular they are playing a significant role in the design and analysis of machine learning algorithms. Fundamental idea of a graphical model is the notion of modularity - a complex system is built by combining simpler parts. Graphical models are a marriage between probability theory and graph theory. Probability theory provides the glue whereby the parts are combined, ensuring that the system as a whole is consistent, and providing ways to interface models to data. The graph theoretic side of graphical models provides both an intuitively appealing interface by which humans can model highly-interacting sets of variables as well as a data structure that

H. Bevrani, F. Daneshfar, and P. R. Daneshmand are with the Department of Electrical and Computer Engineering, University of Kurdistan, Sanandaj, Iran. Currently, H. Bevrani is a visiting professor at Kumamoto University, Japan (corresponding author e-mail: bevrani@ieee.org).

lends itself naturally to the design of efficient general-purpose algorithms [7].

The graphical model provides a suitable tool to view real-world systems as instances of a common underlying formalism. This view has many advantages - in particular, specialized techniques that have been developed in one field can be transferred between research communities and exploited more widely. Moreover, the graphical model formalism provides a natural structure for the design of new systems [7].

Probabilistic graphical models are graphs in which nodes represent random variables, and arcs represent conditional independence assumptions between variables [7]. If there is not any arc between two nodes, they are independent variables otherwise they are dependent variables. The arcs pattern presents the graph structure. Hence, it provides a compact representation of joint probability distributions. For example, for N binary random variables, an atomic representation of the joint $p(x_1,...,x_n)$, needs $O(2^n)$ parameters, whereas a graphical model may need exponentially fewer, depending on which conditional assumptions we make.

There are two main kinds of graphical models: undirected and directed [7]. Undirected graphical models, also known as Markov networks or Markov random fields (MRFs), are more popular with the physics and vision communities. Directed graphical models, also known as BNs, belief networks, generative models, causal models, etc. are more popular with the artificial intelligence (AI) and machine learning communities. In a directed graphical model (i.e., a BN), an arc from A to B can be informally interpreted as indicating that A "causes" B (or B is dependent to A), in this structure A is the parent node of B and B is the child node of A. Hence directed cycles are disallowed. A node in a BN is independent of all the other nodes in the graph given its Markov blanket. However, in the case of a BN, the Markov blanket of a node is the node's parents, children and children's parents [7].

B. Bayesian networks

In real learning problems, there is large number of variables with relationships. The BN is a representation suited to this task. It is a graphical model that efficiently encodes the joint probability distribution for a large set of variables.

They are a widely used formalism for representing uncertain knowledge in AI [6, 8]. They have become the standard methodology for the construction of systems relying on probabilistic knowledge and have been applied in a variety of real-worlds tasks.

A BN consists of (*i*) an acyclic graph S, (*ii*) a set of random variables $x=\{x_1,...,x_n\}$ (the graph nodes) and a set of arcs that determines the nodes (random variables) dependencies, and (*iii*) a conditional probability table (CPT) associated with each variable ($p(x_i|pa_i)$).

Together these components define the joint probability distribution for x. The nodes in S are in one-to-one correspondence with the variables x. In this structure, x_i denotes both the variables and its corresponding node, and pa_i to denote the parents of node x_i in S as well as the variables

corresponding to those parents. The lack of possible arcs in S encodes conditional indecencies. In particular given structure S, the joint probability distribution for x is given by

$$p(x_1,\cdots,x_n) = \prod_{i=1}^{n} p(x_i|pa_i) \qquad (1)$$

The probability encoded by a BN may be Bayesian or physical. When building BNs from prior knowledge alone, the probabilities will be Bayesian. When learning these networks from data, the probabilities will be physical.

The basic tasks related to the BNs are (*i*) structure learning phase: finding the graphical model structure, (*ii*) parameter learning phase: finding nodes probability distribution, and (*iii*) Bayesian network inference. The structure and parameter learning are based on the prior knowledge and prior data (training data) of the model.

The basic inference task of a BN consists of computing the posterior probability distribution on a set of query variables q, given the observation of another set of variables e called the evidence (i.e. $p(q|e)$). Different classes of algorithms have been developed that compute the marginal posterior probability $p(x|e)$ for each variable x, given the evidence e.

One of the important points in the BNs is that it doesn't need to learn the inference data. Inference is a probabilistic action that obtains the probability of the query using prior probability distribution.

III. 3-CONTROL AREA POWER SYSTEM EXAMPLE

To illustrate the effectiveness of the proposed control strategy described in Section 4, and to compare the results with multi-agent reinforcement learning (MARL) based controllers [3], a 3-control area power system, which is actually an updated version of the IEEE 10 generators, 39-bus system is considered as a test case study.

A single-line diagram of the system is given in Fig. 1. This system has 10 generators, 19 loads, 34 transmission lines, and 12 transformers. Here, the test system is updated by two wind farms in areas 1 and 3, as shown in Fig. 1. The 39 buses system is organized into 3 areas. Total system installed capacity are 841.2 MW of conventional generation and 45.34 MW of wind power generation. There are 198.96 MW of conventional generation, 22.67 MW of wind power generation and 265.25 MW load in Area 1. In Area 2, there are 232.83 MW of conventional generation, and 232.83 MW load. In Area 3, there are 160.05 MW of conventional generation, 22.67 MW of wind power generation and 124.78 MW of load.

The simulation parameters for the generators, loads, lines, and transformers of the test system are given in [3, 4]. All power plants in the power system are equipped with speed governor and power system stabilizer (PSS). However, only one generator in each area is responsible for the AGC task; G1 in Area 1, G9 in Area 2, and G4 in Area 3.

Fig. 1 shows the test system with three controllers. An intelligent controller is used in each area, which is responsible to provide an appropriate supplementary control action.
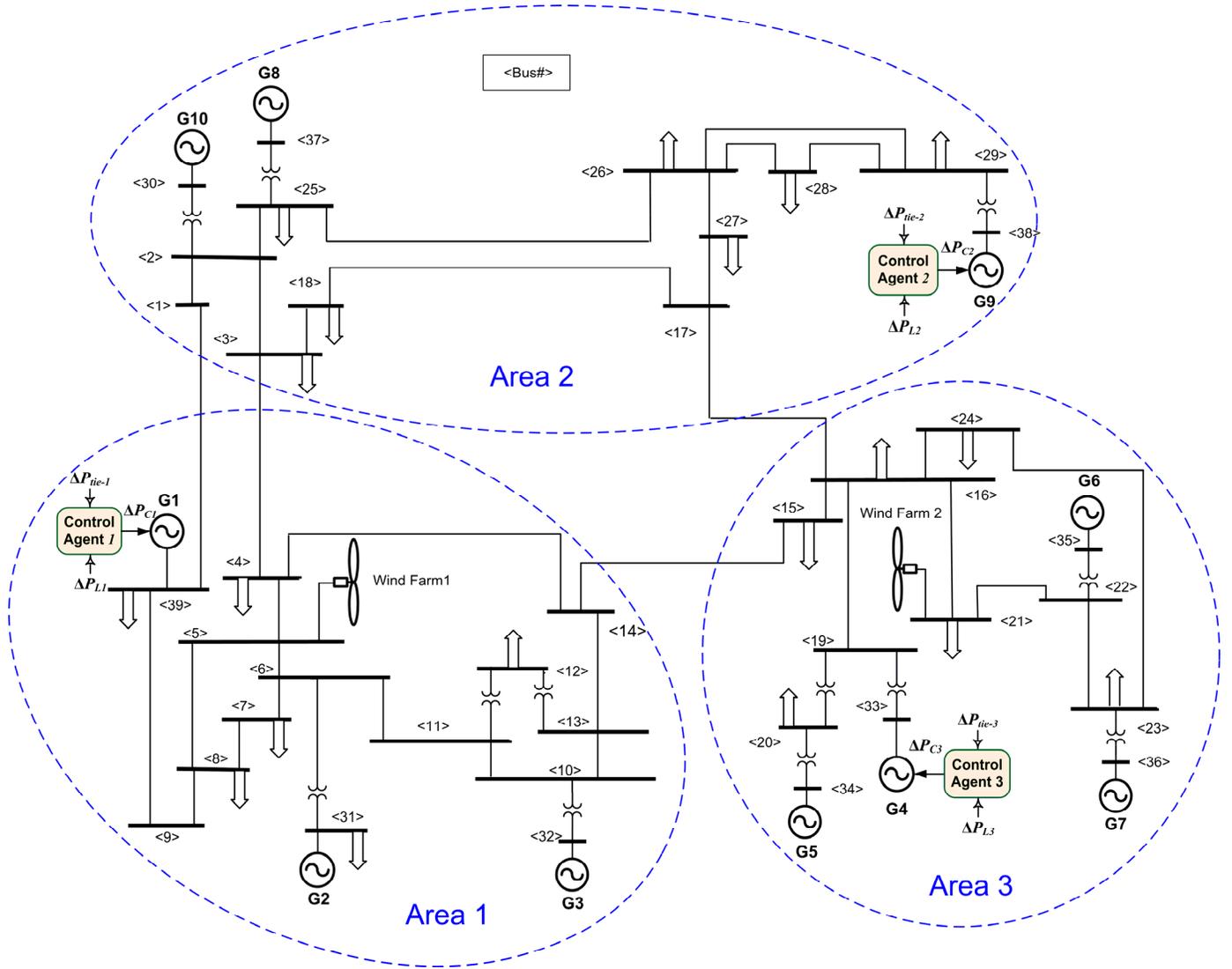
Fig.1. Single-line diagram of 39-bus test system

Following a load disturbance within the control area, the frequency of the area experiences a transient change and the feedback mechanism generates appropriate rise or lower signal to the participating generator units according to their participation factors to make generation follow the load. In the steady state, the generation is matched with the load, driving the tie-line power and frequency deviations to zero. As there are many conventional generators in each area, the control signal has to be distributed among them in proportion to their participation [10].

Because of range of use and specific dynamic characteristics such as a considerable amount of kinetic energy, the wind units are more important than the other renewable energy resources [1].

IV. PROPOSED CONTROL STRATEGY

The main advantages of the proposed BN model for the AGC problem can be summarized as: *i*) Simple and intuitive model building that is closely based on the physical power network topology, *ii*) Easy incorporation of uncertainty and dependency in the frequency response model, *iii*) Capability to monitor the probability of any variable in the whole system, *iv*) Propagation of probabilistic information that allows a wide range of what-if analysis, and *v*) Independent of power system parameter values (e.g. frequency bias factor (*B*), etc).

The main purpose at this step, is to clearly show the various steps in implementation and illustrate the method. The performance results presented here correspond to the performance of the controllers after the learning phase are completed. All the essential parameters for learning phase of the test system are ΔP_{tie}, ΔP_L, Δf, *ACE*, and ΔP_c.

To find related suitable set of training data, a 100 seconds simulation is provided for the described model with well-tuned PI controllers. After running simulation for each instance of 100 seconds simulation, one row of the training data matrix can be provided. Then the training data related to the variables of each area, are given to that area's agent. As the BNT does not work with continuous values, agent must digitize the input data and provides them for the BNT. The BNT is responsible to find the conditional probability

distribution values related to the graphical model variables of each area.

After completing the learning phase, the inference phase is done as follows: At each simulation time step, corresponding controller agents of each area, get the input parameters (ΔP_{tie}, ΔP_L) of the model, and computing the posterior probability distribution $p(\Delta P_c | \Delta P_{tie}, \Delta P_L)$ using the BNT [9], then the controller agents given the observation of the evidence ΔP_c. Using this change to the governors setting and the current values of the load disturbances, the tie-line power deviation is integrated for the next time.

A. Controller Structure

In practice, the AGC system is traditionally using a proportional-integral (PI)-type controller. In this section, an intelligent control design algorithm using BNs technique for such a controller integrated with wind turbine is presented. The objective of the proposed design is to regulate the frequency in power system concerning the integration of wind power units with various load disturbances and achieve a desirable control performance. The results are compared with the results from applying the proposed multi-agent reinforcement learning (MARL) control design given in [3, 5].

Fig. 2 shows the proposed model for area *i*. An intelligent controller is used in this area, which is responsible to provide an appropriate supplementary control action.

B. BN Construction

To illustrate the process of a BN construction, it is better to start by determining of the necessary variables for modeling. This initial task is not always straightforward. As part of this task we must (*i*) correctly identify the goal of modeling, (*ii*) identify many possible observations that may be relevant to the problem, (*iii*) determine what subset of those observations is worthwhile to model, and (*iv*) organize the observations into variables having mutually exclusive and collectively exhaustive states.

In this algorithm, the aim is to achieve the conventional LFC objective and keep the *ACE* signal within a small band around zero using the supplementary control action signal (Fig. 1). Then, the query variable in the posterior probability distribution is ΔP_c signal and the posterior probabilities according to possible observations relevant to the problem are as follows,
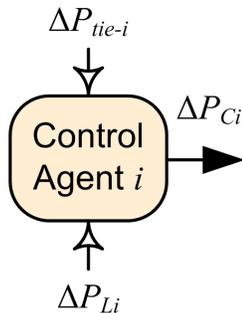

Fig. 2. The proposed model for area *i*

$$p(\Delta P_c | ACE, \Delta P_{tie}, \Delta P_L, \Delta f)$$
$$p(\Delta P_c | ACE, \Delta P_L, \Delta f)$$
$$p(\Delta P_c | \Delta P_{tie}, \Delta P_L, \Delta f)$$
$$p(\Delta P_c | ACE, \Delta P_{tie}, \Delta P_L)$$
$$p(\Delta P_c | ACE, \Delta f) \tag{2}$$
$$p(\Delta P_c | \Delta P_{tie}, \Delta P_L)$$
$$\vdots$$
$$p(\Delta P_c | \Delta P_{tie})$$
$$p(\Delta P_c | \Delta P_L)$$

According to (2), there are so many observations that are related to this problem, however the best one that has the least dependency to the model parameters (e.g. frequency bias factor, etc) and causes the maximum effect on the frequency deviation and consequently *ACE* signal changes, are load disturbance and tie-line power deviation signals. Then the appropriate posterior probability that should be found is $p(\Delta P_c | \Delta P_{tie}, \Delta P_L)$.

The ΔP_{tie} can be practically obtained. However, the ΔP_L is one of the input parameters that is not measurable directly, but it can be easily estimated using a numerical/analytical method. A simple method to estimate the amount of load change immediately following a serious fault (load disturbance) is discussed in [1]. This estimation method is initially based on the measured frequency gradient and the specified system characteristics. On the other hand, regarding the AGC duty cycle, the total consumed time needed for the estimation process is not important.

C. Learning

After determining the most worthwhile subset of the observations (ΔP_{tie}, ΔP_L), in the next phase of the BN construction, a directed acyclic graph that encodes assertion of conditional independence is built. It includes the problem random variables, nodes conditional probability distribution and nodes dependencies.

The basic structure of the graphical model is built based on the prior knowledge of the problem (see Fig. 3.). According to (3), *ACE* signal is dependent to the frequency and tie-line power deviations, then they will be the parent nodes of the *ACE* signal in the BN graphical model, and since frequency deviation is dependent to the load disturbance and tie-line power deviation [1], then they will be parent nodes of Δf,

$$ACE_i = \beta_i \Delta f_i + \Delta P_{tie-i} \tag{3}$$

Since, the ΔP_c is considered to be dependent to *ACE* signal only (Fig. 3), *ACE* node will be the parent node for the control action signal. Another approach for construction the graphical model of the BN can be considered based on the following observations:
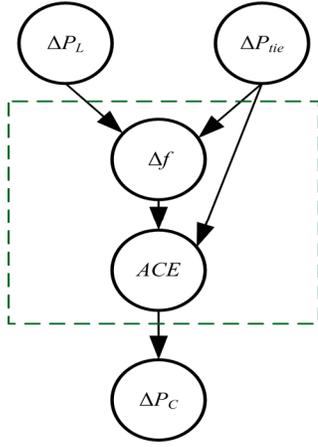
Fig. 3. The graphical model for an area

From the chain rule of probability, we have

$$p(x) = \prod_{i=1}^{n} p(x_i | x_1, \cdots, x_{i-1}) \qquad (4)$$

Now, for every x_i, there will be some subset $\pi_i \subseteq \{x_1, ..., x_{i-1}\}$ such that x_i and $\{x_1, ..., x_{i-1}\} \backslash \pi_i$ are conditionally independent given π_i. That is for any x,

$$p(x_i | x_1, \cdots, x_{i-1}) = p(x_i | \pi_i) \qquad (5)$$

Combining (4) and (5),

$$p(x) = \prod_{i=1}^{n} p(x_i | \pi_i) \qquad (6)$$

Comparing (4) and (6), shows that the variables sets $(\pi_1, ..., \pi_n)$ correspond to the BN parents $(pa_1, .., pa_n)$, which in turn fully specify the arcs in the network structure S. Consequently, to determine the structure of a BN, (*i*) The variables should be ordered somehow, and (*ii*) The variables sets which satisfy (5) should be determined for $i = 1, ..., n$.

Here, using the ordering $(\Delta P_{tie}, \Delta P_L, \Delta f, ACE, \Delta P_c)$ and according to (5), the conditional dependencies are as follows,

$$p(\Delta P_L | \Delta P_{tie}) = p(\Delta P_L)$$

$$p(\Delta P_{tie} | \Delta P_L) = p(\Delta P_{tie})$$

$$p(\Delta f | \Delta P_L, \Delta P_{tie}) = p(\Delta f | \Delta P_L, \Delta P_{tie}) \qquad (7)$$

$$p(ACE | \Delta P_{tie}, \Delta P_L, \Delta f) = p(ACE | \Delta P_{tie}, \Delta f)$$

$$p(\Delta P_c | ACE, \Delta P_{tie}, \Delta P_L, \Delta f) = p(\Delta P_c | ACE)$$

The graphical model of the problem (Fig. 3.) is based on the right side of the above relationships. In the next step of BN construction (parameter learning), the local conditional probability distribution(s) $p(x_i | pa_i)$ are computed from the training data. Probability distributions and conditional probability distribution related to this problem, according to Fig. 3, are $p(\Delta P_L)$, $p(\Delta P_{tie})$, $p(\Delta f | \Delta P_L, \Delta P_{tie})$, $p(ACE | \Delta P_{tie}, \Delta f)$,

and $p(\Delta P_c | ACE)$.

Here, Bayesian networks toolbox (BNT) [9] is used to probabilistic inference of the model. The BNT toolbox uses the training data matrix and finds the conditional probabilities related to the graphical model variables (This is the parameter learning phase).

D. Bayesian Network Inference

Once, a BN has been constructed (from prior knowledge, data or a combination), various probabilities of interest from the model are determined. For the problem at hand, it is desired to compute the posterior probability distribution on a set of query variables, given the observation of another set of variables called the evidence. The posterior probability that should be found is $p(\Delta P_c | \Delta P_{tie}, \Delta P_L)$. This probability is not stored directly in the model, and hence needs to be computed. In general, the computation of a probability of interest given a model is known as *probabilistic inference*.

V. SIMULATION RESULTS

To demonstrate the effectiveness of the proposed control design, some simulations are carried out. In these simulations, the proposed controllers are applied to the model described in Section III. Similar to a real-world power system, in the performed application the important inherent requirement and basic constrains such as governor dead band and generation rate constrain imposed by physical system dynamics are considered. Here, the performance of the closed-loop system using the MARL based controllers [3, 5] compared to the designed BNs based controllers is tested for the various possible load disturbances.

As a severe test scenario, the following load disturbances (step increase in demand) are applied to three areas: In Area 1, 3.8% of total area load at bus 8, 4.3% of total area load at bus 3 in Area 2, and 6.4% of total area load at bus 16 in Area 3 have been simultaneously increased in a step form. The frequency deviation (Δf), and area control error (ACE) signals of the closed-loop system are shown in Figs. 4, 5 and 6.

In the proposed simulations, to clearly show the wind turbine impacts on the overall system frequency behavior, the wind farms are directly connected to the 39-bus power system, without using washout/low-pass filters. Therefore, fast movements in wind power output are combined with movements in load and other resources. That is why, the power system response is affected by the wind power fluctuation, and the recorded signals which are shown in figures 4-6 begin with a transient. In fact, when wind power is a part of the power system, additional imbalance is created when the actual wind output deviates from its forecast.

It is shown that using the proposed method, the area control error and frequency deviations in all areas are properly driven close to zero in the presence of wind turbines and load disturbance. Furthermore, the intelligent controllers provide smoother control action signals, and the areas frequency deviation is less than the frequency deviation in the system with MARL based controllers.
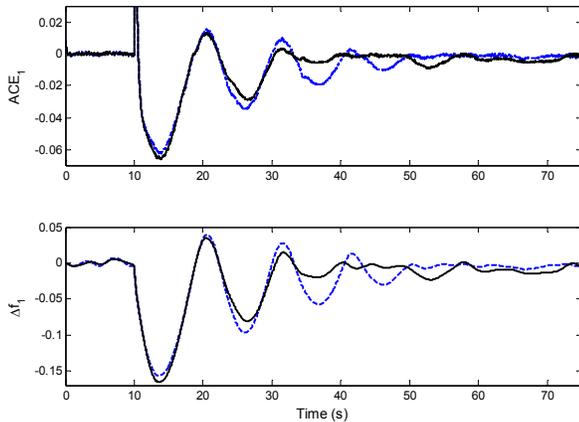
Fig. 4. Area-1 responses; proposed multi-agent BNs method (solid line), and MARL method (dashed line)
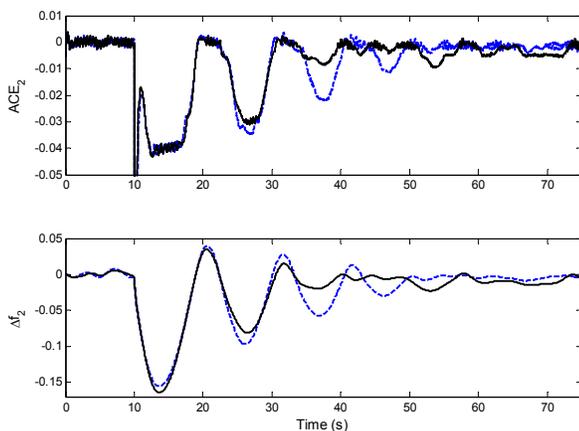

Fig. 5. Area-2 responses; proposed multi-agent BNs method (solid line), and MARL method (dashed line)
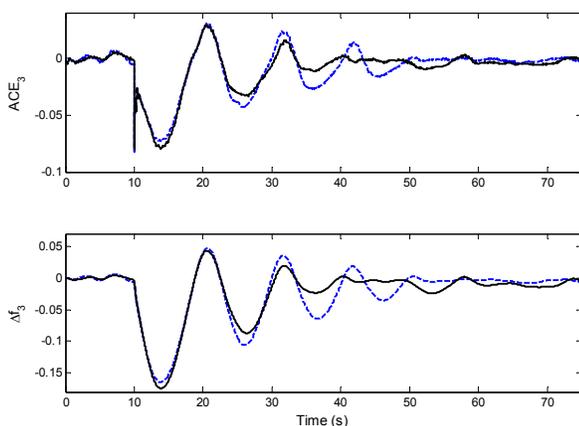

Fig. 6. Area-3 responses; proposed multi-agent BNs method (solid line), and MARL method (dashed line)

In summary, flexibility, higher degree of intelligence, model independency, and handling of incomplete measured data (uncertainty consideration) can be considered as some important advantages of the proposed methodology.

VI. CONCLUSION

A new method for AGC design using a Bayesian networks multi-agent is proposed for a large-scale power system. The proposed method is applied to a 3-control area power system. The results show that in comparison of RL based intelligent controllers, the new algorithm presents a desirable performance.

VII. REFERENCES

[1] H. Bevrani, "Robust power system frequency control," Springer Press, 1st Ed., NY, USA, 2009.
[2] H. Bevrani, T. Hiyama, Y. Mitani, K. Tsuji, and M. Teshnehlab, "Load-frequency regulation under a bilateral LFC scheme using flexible neural networks," Engineering intelligent systems for electrical engineering and communications, 2002, 14(2), pp. 109-117.
[3] H. Bevrani, F. Daneshfar, P. R. Daneshmand, "Intelligent power system frequency regulation concerning the integration of wind power units", Book chapter of *Wind Power Systems: Applications of Computational Intelligence*, L. F. Wang, C. Singh, and A. Kusiak (Eds), Springer Book Series on Green Energy and Technology, Springer-Verlag, Heidelberg, expected in 2010.
[4] F. Daneshfar, "Automatic Generation Control using Multi-agent Systems", MSc. dissertation, Department of Electrical and Computer engineering, University of Kurdistan, Sanandaj, Iran, 2009. Available online: http://www.bevrani.com/Disertations/Daneshfar.rar
[5] F. Daneshfar, H. Bevrani, "Load-frequency control: a GA-based multi-agent reinforcement learning," *IET Gener. Transm. Distrib.*, vol. 4, no. 1, pp. 13-26, 2010.
[6] J. Pearl, "Probabilistic reasoning in intelligent systems: networks of plausible inference," Morgan Kaufman Publishers Inc: San Francisco CA, 1988.
[7] K, Murphy, "An introduction to graphical models," Technical report, Intel Research, 2001.
[8] RE. Neapolitan, "Probabilistic reasoning in expert systems," Wiley Interscience: New York, 1990.
[9] K. Murphy, "The bayes net toolbox for matlab", Computing Science and Statistics 2001, 33
[10] H. Bevrani, and T. Hiyama, "On load-frequency regulation with time delays: design and real-time implementation", *IEEE Transaction on Energy Conversion*, 2009, vol. 24, pp. 292-300.