2005 IEEE/RSJ International Conference on Intelligent Robots and Systems

Predictive Controller for Heterogeneous Sensor Network Operation in Dynamic Environments

Ashit Talukder[1], Syed Muhammad Ali, A. Panangadan, L Chandramouli,
CHLA/University of Southern California
[1]Joint affiliation with Jet Propulsion Laboratory/NASA
Email: talukder@usc.edu, Ph: 818 354 1000

ABSTRACT

***Abstract* - We discuss a novel control methodology for power management in heterogeneous distributed sensor networks. Many algorithms for resource management in sensor networks require complete knowledge of the external environment and the sensor network system, are rule-based and cannot handle rapidly changing environments; [*]this restricts their use in real-world environments. We present an event based control optimization formulation of the resource management problem and discuss a method to adaptively change desired system performance of the sensor network in response to events. This functionality is critical in field-deployable sensor networks where the available energy is extremely limited. This limitation disallows continuous operation as a very expensive option and necessitates system adaptation as a means to extend operational lifetime in the face of dynamic external events. We show results on synthetic sensor networks where only partially accurate information about the external world and the sensing system is available and illustrate the efficacy of the control algorithm in handling dynamic events with guaranteed minimum system lifespan via efficient usage of energy resources. We show that the control algorithm makes effective control decisions about the use of energy resources with varying sensor reliabilities.**

***Index Terms*—Evolutionary algorithm, control, distributed sensing, machine learning, sensor networks, pattern recognition, resource management, mathematical optimization, wireless communication.**

I. INTRODUCTION

Energy constrained mobile and static sensor networks lend themselves to a number of challenges that are normally not encountered in offline data processing or during the processing of signals that have access to lots of computing and transmission resources. The power usage and the communication load in a energy constrained distributed sensor network is highly dependent on the signal processing strategies used to process the observed data and on the resource management strategies used to control critical operations that consume power and use up available bandwidth. An often-ignored issue in sensor networks is the fact that the real-world being observed is dynamic in nature and varies over time. Therefore, ideal detection of dynamic external events in the real-world requires **truly adaptive** operation of multiple sensors in the sensor network.

Different sensors in a heterogeneous sensor network have varying sensitivities and specificities to various external events. Therefore, effective management of sensors and network resources is critical in detection of a variety of events with energy constrained mobile/static heterogeneous sensor networks. Past work on bio-medical sensor networks [1] and environment and health monitoring [2] has focused primarily on communication issues and data logging. The inter-disciplinary nature of this problem poses several hurdles to be overcome towards the ultimate goal of autonomous extended monitoring. One of the primary challenges which need to be addressed in sensor networks to achieve true autonomy is management of limited resources in the mobile system.

In our current work in MUSIC (Multi-modality Sensor network for Integrated event detection, Control optimization and resource management), we concentrate on the resource management algorithms to control operation of multiple distributed sensors in an optimal and coordinated fashion. We discuss a novel resource management technique for mobile distributed sensing platforms that allows monitoring and detection of environmental events for extended periods of time via dynamic adaptation of sensor network operational parameters in response to detected events. Power is the most critical resource in a sensor network. Power is needed to operate each sensor, run the computing unit to process measured sensory data, and power is absolutely critical for wireless transmission of any raw and processed data. Lowering the operating frequency lowers power consumption in three ways: (1) it lowers energy consumption in operating each sensor, (2) it reduces the data size to be processed in-situ, and thereby saves on power used by the CPU on the sensor, and (3) it reduces the data size during transmission and therefore uses less power during communication. Therefore, our focus in this paper is on power management in the sensor network by optimal coordinated scheduling of sensor operations.

The applications that we target range from unmediated data monitoring and record keeping of the environment to the health of a patient and handicapped individuals in a hospital environment. Figure 1 shows our mobile health monitoring using body wearable medical sensors. In mobile health monitoring, the simplest solution that has been employed up until now has been to run the system at a maximum possible

[*] This work has been sponsored by the NIAAA under Contract No. N01AA33004

frequency, capture all sensory data, and transmit all raw data to a central server. However, limited power, storage and bandwidth alongwith minimum desired operational lifetime makes such a simplistic solution infeasible. For example, our experiments have indicated that continuous transmission of data reduces the lifetime of a handheld by a factor of six, compared to continuous on-board data processing. Additionally, when a connection with the central server is unavailable for long periods of time, blind data logging becomes impractical. This illustrates the usefulness of managing resources, namely power and bandwidth, in a mobile sensing system.

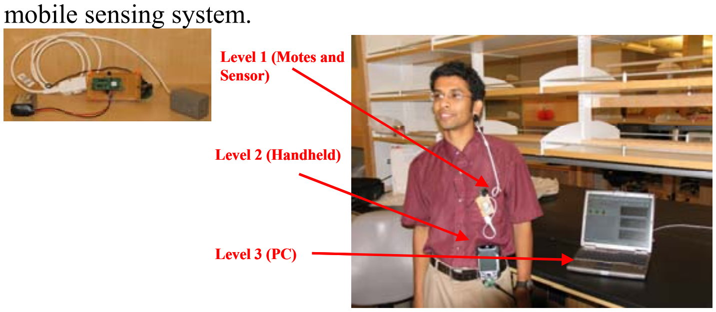


Figure 1 The MUSIC autonomous health monitoring hardware

We consider sensor networks where the sensors and computational devices can be toggled between an energy conserving "idle" and energy consuming "active" state. Therefore, the operational frequency of each device can be actively controlled in real-time. From an application and end-user perspective, running the system at its maximum possible frequency is not desirable when the external environment (greenhouse, battlefield) or human (health monitoring) is in normal condition since the information collected during such times is not critical. The limited resources should be conserved and used more at time periods when events are observed through the sensors (e.g. when the external environment or the health of patient deteriorates).

While a plethora of control optimization and resource management algorithms have been developed in the past, in diverse fields ranging from model-based control [3], to fuzzy control [4], reinforcement learning-based control [5], to agent-based control with roots in Artificial Intelligence [6], no single solution is suited to handle multi-parameter, multi-variable distributed systems with multiple conflicting objectives, and run with limited computational resources of a real-time system. We discuss a new multi-objective genetic algorithm (MOGA) formulation to handle the operational parameters of the distributed sensor network in response to dynamic events in the observed environment. The primary advantage of the MOGA is that it is capable of handling multiple conflicting objectives and constraints in the distributed sensing system, and a variety of variables types (integer, binary, and real-valued) that are present in our MUSIC system model. Our MOGA-based control ensures that optimal information about the external environment being observed is stored and conveyed back to the base station while maintaining an extended lifespan for the entire distributed sensor network. Novel algorithmic additions to the MOGA operations ensures an order of magnitude speedup in convergence speed without compromising accuracy.

The paper is organized as follows. In section II, we give a brief overview of our GA algorithm used in control optimization. Our pattern recognition algorithms for event detection are left out to maintain the technical focus of this paper. The reader is encouraged to read our previous publications on pattern recognition for sensor networks [7, 8]. We give an introduction to our adaptive control model formulation for MUSIC resource management in Section III, which is extended in Section IV to a predictive control model framework for better handling of dynamic environments. We then discuss our results for specific case-studies in Section V.

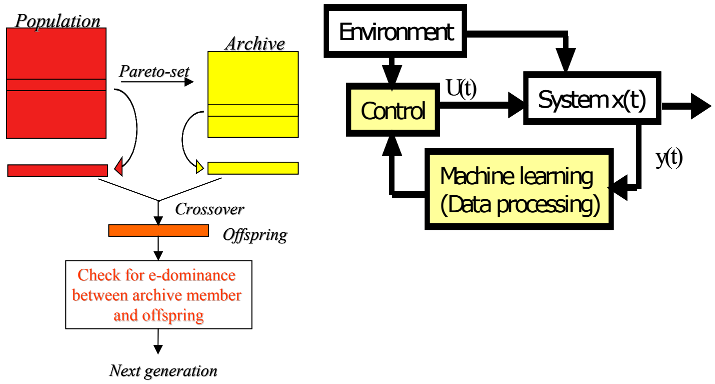
II. GENETIC ALGORITHMS

Genetic algorithms are a part of Evolutionary Computing, a rapidly growing area of artificial intelligence inspired by Darwin's Theory of Evolution. It is a stochastic search technique where the solution is evolved from random starting points through processes that resemble natural evolution. More about Genetic Algorithms (GA) can be found in [9]

Genetic Algorithms work by 'mixing' solutions during reproduction, and randomly altering them during mutation. The GA formulation allows specification of unusual, nonlinear constraint functions that are not smooth in nature, thereby allowing its use in problems that often confounds many traditional optimization techniques. A key advantage of GAs is that they are 'anytime' algorithms. The quality of the solution depends on the number of iterations, and increases monotonically over time. This allows the flexibility of trading off solution quality for solution time. During a control optimization step we can evolve our solutions for as long as feasible and use the best solution encountered till that point.

A number of refinements have been introduced into GA from the vanilla model of selection, reproduction, mutation over time. Since GAs are traditionally hindered by the slow run-times, *one enhancement that we use in our GA controller is elitism* (Figure 2a). Since genetic algorithms are a stochastic technique, it is possible for the algorithm to lose better individuals during evolution. Elitism is the idea of ensuring that better individuals persist throughout the period of evolution. In our case we maintain a set of elite solutions in an archive (Figure 2a), and the final solution is then selected from the archived set. This has the effect of guiding the search in the direction of better solutions and results in faster convergence, thereby enabling run-time performance.

An individual in our genetic algorithm controller is an assignment of a set of operational frequencies to the sensors in our system. The initial population of frequencies may be created randomly, but more intelligent methods can speed up convergence by an order of magnitude. Since the observed dynamic process in MUSIC changes slowly (relative to the control step resolution), successive control decisions are based on a world that is not too radically different over time. We can therefore expect to gain computational benefit from generating



(a) GA elitist optimization method (b) Control system representation

Figure 2 GA-based adaptive system control based on interaction with observed environment

the population for the current control step from the solution of the previous step. To guarantee successful evolution through diversity we generate the individuals of the GA population close to the previous control optimization solution with additive random Gaussian noise.

III. Control-Based Model for Resource Management and Real-Time Operation

A critical requirement in our distributed sensing system is the ability to detect and track important dynamic events while ensuring that certain system constraints are not compromised. This calls for an efficient dynamic control mechanism that takes into account all system resources, constraints, and measured/detected external events (derived from sensor data) in order to control the system operation in real-time.

Typically, real-world control optimization problems are characterized by constraints, multiple objectives, and dynamic changes in external and internal conditions. In particular real-world control problems are notoriously dynamic due to continuous changes in the external processes (environment) and, in many cases, the feedback between the controller and the controlled system. Additionally, changes in sensor sensitivities over time and in the internal system parameters makes the control problem more complex. A typical problem in hardware control is the drift of material parameters due to machinery wear-out and sensitivity to environmental changes, such as temperature and humidity. Therefore, the applied control optimization technique should be able to continuously search for the best solution at frequent time intervals.

We represent the dynamic (time-dependent) interaction between the controller, the observed process, and the controlled system as shown in Figure 2b. For this paper, we consider the problem of determining the operational frequency of each sensor in the network and deciding on whether data should be stored/discarded, given predefined system parameters, current conditions of dynamic state variables, and detected events by the pattern recognition system. Together, they can be combined with the state/control/environment variables that are specific to the problem being considered to form a utility profit-loss function that should be optimized dynamically at each time step. In this paper, we investigate control optimization strategies in profit maximization. Profit maximization involves measures that maximize a desired objective while minimizing the loss incurred in using system and environmental resources. The profit is a function of variables in the controller and in the controlled system.

The objective function can be dependent on several internal system and external environmental observed variables. In MUSIC, the objective is to maximize the information extracted (measured) and conveyed back to the end-user, while minimizing the usage of resources in the system, such as power, computational resources, and storage. Therefore, we devise a profit measure for the information measured from the sensors and a cost for deriving/extracting this information about the external environment. The profit of information measured should actually be adaptive and expressed as a function of the changing dynamic environment. We use rigorous information theoretic measures to formulate the information profit measure. It is expressed as a function of the event occurring at a given time (measurement of highly critical events have greater information utility than ones that are benign/normal) and the frequency of observations (a larger frequency of measurements also increases the information utility/profit). The information utility/profit is also a function of sensor/classifier reliability, and other internal variables. Further details of the information-theoretic measures are provided in Section III.A. The information loss or cost function expresses the cost incurred in trying to measure more information about the environment (larger sensor operational frequency incurs larger cost). This yields a set of profit and loss functions for various system requirements. The system operates under a predefined range of parameters and combinations thereof, and also has a minimum operational requirement, such as guaranteed minimum system lifespan. These measures are expressed as constraints. We therefore pose the control problem as one that maximizes (minimizes) all these profit (loss) functions, while satisfying all system constraints. We discuss our solution to this problem next.

III.A Information Utility Function

We have derived our information utility functions from the basics of information theory and probabilistic theory. From prior research in machine learning on ensemble of classifiers, it is well-known that a decision made from a combination of weak classifiers is far superior to using decisions from each classifier alone. We employ this fact in the design of our information utility from an ensemble of sensors. We view each sensor as a weak classifier/expert, where its raw sensory measurements can be used to make a classification decision about the presence/absence and the criticality of an event. **Therefore, in our discussions, we use the term classifier, expert, and sensor interchangeably**.

In ensemble learning [10] and mixture of experts models [11], the goal is to combine an ensemble of experts/classifiers in an optimal manner such that the overall error in classification is minimal. This is achieved by ensuring that the error diversity in the ensemble/mixture is maximal. When a quadratic error loss function is used, the generalization error is a combination (sum) of two terms [10]: the bias error and variance error. The bias-variance decomposition for quadratic loss states that the generalisation error of an estimator can be broken down into two components: bias and variance. These two usually work in opposition to each other: attempts to reduce the bias component will cause an increase in variance, and vice versa. The bias-variance error decomposition has been further extended in [12] to yield the overall bias-variance-covariance error in a simple weighted average ensemble of classifiers, and is given by

$$E\{ [1/S\ \Sigma\ (C_i - \langle D\rangle^2)] \} = \langle Bias\rangle^2 + 1/S \langle Var\rangle + (1 - 1/S) \langle Covar\rangle \qquad (1)$$

Where S is the number of experts in the ensemble, $\langle D\rangle$ is the desired output, C_i is the actual ensemble classification output, $\langle Bias\rangle$ is the averaged bias of the ensemble members, $\langle Var\rangle$ is the average variance in classification error of the ensemble members, and $\langle Covar\rangle$ is the average covariance classification error of the ensemble. Our main interest is in the cross variance error metric that measures the degree of correlation between the different sensor (or expert) errors and in the Bias error metric that measures the individual sensor/expert reliability or degree to which the sensor or expert can be trusted in detecting an event. Our information utility measures from these two error terms expressed as a function of sensory network operational parameters are detailed below.

III.B Information Theoretic Modeling of Sensor Network Joint Ambiguity

The statistical dependency between different sensors (experts) and the resultant error due to the sensor cross correlations are expressed as (from Eq. 1 above)

$$\langle Covar\rangle = 1/(S\ (S-1)) \sum_i \sum_j \sigma_{ij}$$

where σ_{ij} measures the cross covariance between sensor *i* and sensor *j*. This covariance model can be used to model dependency relationships between sensors. If two sensors observing an event measure similarly *underlying* phenomenon caused by occurrence of the event, it is probable that there is high correlation between them. Conversely, if two sensors have low correlation, the underlying phenomena they measure are likely to be weakly related. This reduces the chance of incorrect classification between weakly correlated (but reliable) sensors. We can therefore base our utility function on this term.

We formulate a correlation term reflecting the **sampled** overall cross correlation error between sensors and their operational frequency. If sensor i samples the event at a frequency of Fi (per minute), and the operational frequency of sensor j is Fj, the sampled cross covariance is

$$\text{CovErr} = \sum_i \sum_j \sigma_{ij} / (Fi + Fj) \qquad (2)$$

We can maximize our useful information by minimizing the overall cross covariance error amongst all sensors. With the sampled cross covariance error as defined above, the overall error will be lower (and information gathered will be higher) for higher operational frequency. However, given an upper bound on total sensor operational frequency due to power consumption constraints, the utility metric will be lower for a higher operational frequency of uncorrelated sensors than for correlated sensors.

While the above covariance error metric is good for static environments, it is unsuitable for use in dynamic environments where the tolerable error in estimating the class of an event is a function of the criticality of the event. More critical events should be observed with lower classification error, whereas less critical events can tolerate a higher degree of classification error. Therefore, our desired information utility metric is defined as a dynamic quantity that is a function of both, the sensor cross covariance models, and also the class label/criticality of the detected event. Therefore, our desired information utility to be maximized is the following

$$SIU = | (1-EC) - (\text{CovErr})_{norm}) | \qquad (3)$$

Here EC is the event criticality that ranges between 0 and 1, and $(CovErr)_{\text{norm}}$ is scaled version of CovErr (which is trivial for a given cross correlation model). In essence, this corresponds to minimizing the difference between the event criticality and the classification error of the sensor network ensemble. As events become more critical it becomes more important to be confident about the information that the sensor network extracts.

III.C Information Theoretic Modeling of Sensor Reliability (Bias)

From the basic ensemble and mixture of experts models formulations as discussed earlier, the overall classification error is a sum of three terms, of which the cross covariance error is used in the information utility metric described in the earlier section. Modeling the reliability or bias of each individual sensor is an important factor to be considered during the control decision process. While covariance error models cross correlation between sensors and our information utility metric attempts to minimize the cross covariance error, it fails to capture the reliability of each individual sensor in its measure. The bias error in an ensemble is given as

$$\langle Bias\rangle = 1/S \sum E\{ (C_i - \langle D\rangle) \} \qquad (4)$$

where $\langle D\rangle$ is the true class label or event criticality and C_i is the class label as estimated by class *i*. More reliable sensors will contribute in decreasing the Bias error, whereas less reliable sensors will increase the ensemble classification bias.

For a sensor i with statistical bias $B_i = \text{E}\{ (C_i - \langle D\rangle) \}$, we define the reliability of a sensor *i* as an inverse function of its statistical bias, $R_i = 1 - \text{E}\{ (C_i - \langle D\rangle) \}$. The sampled bias error in the sensor network is defined as $B_s = \Sigma\ B_i . F_i / (\Sigma F_i) \}$. Each sensor contributes to the bias error term proportionally to its operational frequency; more reliable sensors are therefore required to operate at a higher proportional frequency for lower overall bias error. We introduce the sampled bias as a constraint in the controller optimization function in the following manner:

$$B_s = \sum B_i . F_i / (\textstyle\sum_i F_i) \}. \leq MinBias \qquad (5)$$

Decreasing the constraint violation threshold *MinBias* forces the system to increasingly prefer use of more reliable sensors over less reliable sensors. To handle changing system requirements in dynamic environments, the constraint violation threshold *MinBias* is determined as a function of the detected class label and associated event criticality *D*. We weigh the constraint violation by the criticality of the event. This implies that the system would pay a heavier price for violating the classification bias constraint for more critical events than less critical events.

III.D Modeling Guaranteed Minimum Lifespan

One primary consideration in designing the sensor network is to ensure that it operates for a minimum desired amount of time. To ensure guaranteed lifespan QoS, we place constraints on the rate of energy consumption at each node in the sensor. Given an initial amount of available energy E_{MAX} and a minimum desired lifetime T_{MIN}, we assume a linear constraint for the energy consumed which implies that the total energy E_T expended upto time T cannot exceed $E_{MAX} \cdot T / T_{MIN}$,. In other words, the lifespan QoS guarantee is expressed as

$$E_T \leq E_{MAX} \cdot T / T_{MIN} \qquad (6)$$

III.E Control Optimization without Look-ahead

Based on the earlier models of information utility with the statistical bias and covariance error of an ensemble of sensors and the constraint design of lifespan QoS guarantee, the real-time control optimization for managing multiple energy resources given dynamic external observations is given as

$$\textbf{Minimize} \quad SIU = |\,(1-EC) - (\mathrm{CovErr})_{norm} \qquad (7)$$

$$\textit{Where} \quad \mathrm{CovErr} = \sum_i \sum_{j \neq i} \sigma_{ij} / (Fi + Fj)$$

$$\text{S.T.} \quad B_s = \sum B_i \cdot F_i \,/\, (\textstyle\sum_i F_i)\,\}. \leq MinBias$$

$$E_T \leq E_{MAX} \cdot T / T_{MIN}$$

The GA optimization algorithm discussed in Section II earlier is used to solve the optimization problem posed in Eq. (7) above. The elitist nature of the algorithm using stored archives during the crossover and mutation processes ensures swift convergence compared to traditional GA solutions that could be an order of magnitude or more slower.

IV. Predictive Control with Look-Ahead for Resource Management

Predictive controllers with look-ahead capability have been proposed in the past [3] The primary gain from controllers with look-ahead capability is that the ability to peek ahead into the future allows it to make an optimal decision not only from past and current observations and state changes, but also based on what will occur in the future. While this does not offer a clear advantage on problems where the system and the external environment change very slowly, it poses a number of benefits when the environment changes rapidly. Primarily, it allows the system to compensate for future changes by taking them into account while making its current control decision. In our application, if we can predict future (possibly more critical) events up until a finite time horizon, the controller then compensates by consuming less power in the current step and conserves the power for use in the future when it is more important to extract more information. This improved control performance however comes at the cost of increased model complexity, and potentially a larger optimization time. We discuss the details of our predictive controller next.

IV.A Event Prediction Model

A predictive model first requires a robust predictor that is able to estimate external event criticality and class labels in the future for a given history of events in the past. We assume an auto-regressive moving average (ARMA) model for the external event in the following manner

$$EC(t)_{pred} = \sum_{n=1}^{N} a_N \cdot EC(t-n) \qquad (8)$$

where N is the order of the model. Standard techniques such as least squares, fast orthogonal search, Yule Walker Estimation, Levinson Durbin Algorithm, or Maximum Entropy (Burg's algorithm) can be used to estimate the ARMA model parameters.

IV.B Predictive Ensemble Covariance Error and Bias Error Model

We include in our GA controller considerations about the patient's future condition and the future power requirements by modeling it as a set of constraints. We must ensure, based on our estimation of future states, that we have enough power to monitor future events appropriately.

While there are numerous ways to characterize the model and the requirements for future performance of the controller, one primary limitation is the computation time for the control optimizer in real-time systems. We limit the model complexity by estimating explicit constraints and explicit information predictions on the most critical event in the future within the time horizon of the controller, rather than a model that depends explicitly on all future events within the time horizon. The additional performance constraints are imposed on the most critical event in the future to ensure that the system meets all information error requirements at this point in the future.

We introduce additional notation for the text to follow. Let H be the prediction horizon and h is the time in the future corresponding to the most critical future event. We denote future time by t_p to distinguish it from current/past time. We call CovErr_h the covariance term calculated at time $h < H$ in the future. EC_h ε [0,1] denotes the event predicted at time h.

What prevents us from sensing 'appropriately' in the future is how much power we expend in the present, and what our upper bound on power consumption is at that time. Note that the covariance error can only decrease with increasing sensor frequency Fi. (from Eq. 2) We base our predictive information constraint on this idea. We estimate the highest frequency at which each sensor can operate in the future while meeting power and bias constraints. Based on this allowable range of frequencies we compute the resulting Information Utility in the future that we call Predictive Information Utility.

<u>Predictive Energy Resource Utilization Model</u>

In order to avoid a more complex model where we have to optimize over all unknown future frequencies, we assume an event-based frequency model for all future time steps from $t_p = T+1, T+2, \ldots. h-1$ up until the time instant before the most critical event. We assume that the future frequencies are proportional to the current (unknown) frequency to be optimized. We assume that the frequency of sensor i at time in the future is related to the current frequency via the relative estimated event class criticality in the future as:

$$F_{tp}[i] = F[i] * EC_{tp} / EC_T \qquad (9)$$

Where EC_{tp} is the predicted event class criticality at time t_p and EC_T is the event criticality at time T. Based on this predicted frequency behavior for future times $t_p=T+1,\ T+2,\ \ldots.\ h-1$, we can then estimate energy used up until time $h-1$ and determine the information covariance and bias errors at time h in the future that will meet desired error criteria, as discussed next.

$CovErr_h$ is the minimum allowed covariance error at time h in the future. If EC_h is the predicted future event, then based on our earlier information utilities measure, the ensemble Covariance error $CovErr_h$ should be as close to (1- EC_h) as possible. Therefore, we set our predictive ensemble covariance constraint to be

$$(1- EC_h) >= CovErr_h \qquad (10)$$

If this constraint is not satisfied, i.e. the minimum future covariance error at the most critical event at time h in the future would still be greater than the desired error, which means that even if were to operate at the highest frequency and expend all power that we can at that point without violating any other constraints, we would still be unable to sense the event appropriately. We use a similar constraint for the future ensemble Bias error at the most critical event at time h in the future, within the prediction horizon H.

V. RESULTS

We implemented a simplistic simulator for evaluating our GA controller. We assume a time-varying observed event that is classified with associate classification errors based on the sensor reliabilities and covariance models. The simulator feeds the controller with observed data, i.e. patient status and state data such as sensor power, handheld power, transmission power and bandwidth, and the controller determines the optimal control actions (frequency of operation of each sensor) for the next time step. The simulator recalculates the changes to the system (available energy etc.) based on the current control actions and which is input into the controller along with the next time step's inputs.

Sensors can be specified to have different energy consumption rates, which determine how long they can survive. The power consumption rate is the same for all sensors in our current simulations. The upper and lower bounds for sensor frequency varies between 3 and 25. We do not presently allow the controller to completely shut off the sensors to ensure continuous information capture at all times.

Since the controller may operate in a number of different conditions (sensor correlations, reliabilities, power consumption, etc.) and each setting would place different demands on the controller, we show results for our controller in two case studies, outlined below:

- Case 1- Three equally reliable sensors, the third sensor is less correlated with the first two highly correlated sensors
- Case 2- Four equally correlated sensors, with different reliabilities

In the second case study, the correlation between all sensors is 0.9. We tested our predictive controller with the controller without look-ahead capability. For the predictive controller, the event prediction horizon was five control steps with a minimum lifetime duration of 50 control steps.

V.A Controller without Look-Ahead

Figure 3 shows the results of the controller without lookahead. We see that the assignment of frequencies to sensors follows the trend of the class label and its criticality (patient state in health monitoring applications) in each case. We see that in both cases the controller gives preference to the correct sensor. In case 1 sensor 3 is the least correlated, and thereby is assigned more resources. In case II sensors 1 and 2 are successively the most reliable sensors. Sensor 1 is initially given the most precedence, and when it hits the power constraint sensor 2 takes over.

However in both scenarios we notice that the sensor frequencies peak too soon, and at the most critical event we do not extract the most information. This is because we utilize too much energy during relatively unimportant events and therefore when the class label becomes most critical (*ECs > 0.9*), the controller does not have the resources to extract sufficient information (increase the frequency of the least correlated sensor) without violating the constraints.

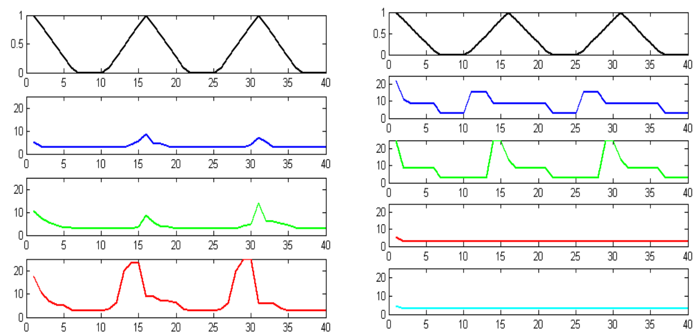


Figure 3 Results for cases I and II with simple controller (no look-ahead capability)

V.B Controller with Look-Ahead

In Section V.A earlier, we discussed results of the controller without any look-ahead capability. As noted in Section III.E in our theoretical discussions and observed in Case II above, the controller does not perform optimally since it makes decisions based on current and past events, without any expectations about possible changes in the future. This may result in undue consumption of power at certain occasions when it could be more warranted to act conservatively and preserve resources for the future.

Figure 4 shows the results from the controller with lookahead for the same cases I and II as in Section V.A. The predictive controller with look-ahead capability exhibit a number of attractive properties. Firstly, we note that this controller prefers the same sensors that the simpler controller does. However, because of its ability to predict the future, it can adjust its information gathering actions so as to accommodate the information needs of the future. Note that not only are the peaks of the sensor frequencies better aligned with event peaks, the controller also utilizes sensor 3 in case II to gather information, which it did not do earlier. This is because sensor 3 is the least important, and since it expects a very critical event to occur in the near future, the controller conserves the energy of sensors 1 and 2 by utilizing sensor 3.

The performance of our adaptive control algorithm must be able to match the real time needs of the MUSIC system. Figure 5 shows the average time (seconds) required to execute a control step with varying prediction horizons on a Pentium 4 processor. The control optimization algorithm scales well with an increasing prediction horizon, although we note that as we go from 3 sensors to 4 the impact of the horizon increases significantly. However, even while planning with a predictive control horizon of ten steps in the future, since the controller is executed every fifteen minutes, our duty cycle is only 0.15% and therefore applicable in a run-time environment.

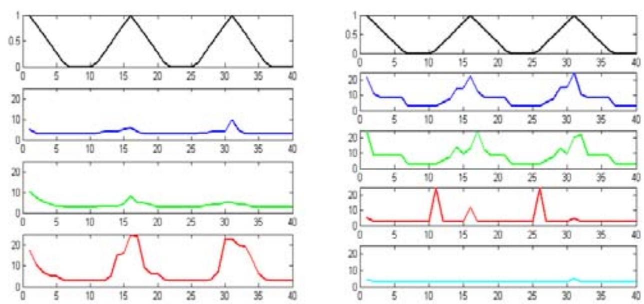


Figure 4 Results for cases I and II with predictive controller

VI. Conclusions

We have discussed a new GA-based adaptive controller to manage the resources in a heterogeneous sensor network to minimize error in information extracted, and extended the concept to predictive control for better power management in dynamic environments. Further tests on real data will be carried out in the future.

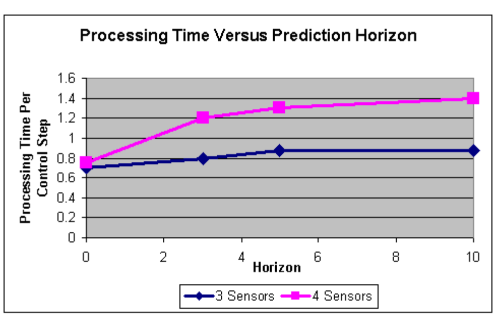


Figure 5 Execution time vs. Prediction horizon in GA controller for 3 and 4 sensors in sensor network

References

[1] L. Schwiebert, S. K. S. Gupta and J. Weinmann. *Research Challenges in Wireless Networks of Biomedical Sensors*. in **International Conference on Mobile Computing and Networking**. 2001. Rome, Italy: p. 151-165.

[2] D. Malan, T. Fulford-Jones, M. Welsh, and S. Moulton. *CodeBlue: An Ad Hoc Sensor Network Infrastructure for Emergency Medical Care*. in **International Workshop on Wearable and Implantable Body Sensor Networks**. 2004.

[3] J. Martin-Sanchez, and J. Rodellar, *Adaptive Predictive Control : From the concepts to plant optimization*. Prentice Hall International series in Systems and Control Engineering, ed. M.J. Grimble. 1996: Prentice Hall PTR.

[4] L.R. Ribler, S.J. Vetter, and H.S.D. Reed. *Autopilot: Adaptive Control of Distributed Applications*. in **Proc. 7-th IEEE Symposium on High Performance Distributed Computing**. 1998.

[5] R. Kretchmar, P.M. Young, and C.W. Anderson, *Robust Reinforcement Learning Control with Static and Dynamic Stability.* **Intl. Journal of Robust and Nonlinear Control**, 2001. **11**: p. 1469-1500.

[6] J. Modi, H. Jung, and M. Tambe. *Dynamic Distributed Resource Allocation: A distributed constraint satisfaction approach*. in **Intelligent Agents VIII: Proc. ATAL'01**. 2001.

[7] A. Talukder, T. Sheikh, and L. Chandramouli. *Real-time Intelligent Pattern Recognition, Resource Management and Control under Constrained Resources for Distributed Sensor Networks*. in **Intl. Joint Conference on Neural Networks**. 2004.

[8] A. Talukder, S. Monacos, T. Sheikh, and L. Chandramouli. *Data classification and resource management in sensor networks for remote health and environmental monitoring*. in **IEEE Intl. Conferences on Intelligent Signal and Image Processing. (ICISIP 2004) (Recipient of the Special Mention Paper Award)**. 2004.

[9] D. Corne, J. Knowles, and M. Oates. *The Pareto envelope-based selection algorithm for multiobjective optimization*. in **Sixth International Conference on Parallel Solving from Nature VI**. 2000: p. 839-848.

[10] S. Geman, E. Bienenstock, and R. Doursat., *Neural networks and the bias/variance dilemma.* **Neural Computation**, 1992. **4**(1): p. 1-18.

[11] R.A. Jacobs, M.I. Jordan, S.J. Nowlan, and G.E. Hinton, *Adaptive mixtures of local experts.* **Neural Computation**, 1991. **3**(1): p. 79-87.

[12] G. Brown and J.L. Wyatt. *The Use of the Ambiguity Decomposition in Neural Network Ensemble Learning Methods*. in **20th International Conference on Machine Learning (ICML'03)**. 2003.