A CONTEXT-AWARE LEARNING, PREDICTION AND MEDIATION FRAMEWORK FOR RESOURCE MANAGEMENT IN SMART PERVASIVE ENVIRONMENTS

by

NIRMALYA ROY

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT ARLINGTON

August 2008

To my parents, for always supporting me in life and being always there for me. You have made all of this possible. Thank You.

ACKNOWLEDGEMENTS

I would like to thank my advisor Prof. Sajal K. Das who has been amazingly patient, helpful, and supportive during my research. He introduced me to the field of pervasive computing, guided me throughout my graduate study and constantly encouraged for high quality research. I would also like to thank Prof. Hao Che, Prof. Yonghe Liu, Prof. Mohan Kumar and Prof. Bob Weems for their comments and suggestions regarding my work in ubiquitous computing. My sincere regards to Prof. Kalyan Basu for his technical guidance and help to me.

I take this opportunity to thank all my friends and colleagues at CReWMaN laboratory for their valuable discussions and support.

I like to acknowledge the Computer Science and Engineering Department of UT Arlington for providing me Teaching Assistantship and STEM Doctoral Fellowship and Dean's Fellowship. I would also like to acknowledge my previous institution Jadavpur University (Calcutta), India, for offering me the first light of Computer Science.

I would like to thank my parents and my brother for their constant encouragement and support without which this work would have not been possible. Finally, I would like to thank my wife, who had to endure long working hours while I was working on this thesis and always supported me emotionally.

JULY 18, 2008

ABSTRACT

A CONTEXT-AWARE LEARNING, PREDICTION AND MEDIATION FRAMEWORK FOR RESOURCE MANAGEMENT IN SMART PERVASIVE ENVIRONMENTS

NIRMALYA ROY, Ph.D.

The University of Texas at Arlington, 2008

Supervising Professor: Sajal K. Das

Advances in smart devices, mobile wireless communications, sensor networks, pervasive computing, machine learning, middleware and agent technologies, and human computer interfaces have made the dream of smart environments a reality. An important characteristic of such an intelligent, ubiquitous computing and communication paradigm lies in the autonomous and pro-active interaction of smart devices used for determining inhabitants' important contexts such as current and near-future locations, activities or vital signs. 'Context Awareness' is perhaps the most salient feature of such an intelligent computing environment. An inhabitant's mobility and activities play a significant role in defining his contexts in and around the home. Although there exists optimal algorithm for location and activity tracking of a single inhabitant, the correlation and dependence between multiple inhabitants' contexts within the same environment make the location and activity tracking more challenging. In this thesis, first we propose a cooperative reinforcement learning policy for location-aware resource management in multi-inhabitant smart homes. This approach adapts

to the uncertainty of multiple inhabitants' locations and most likely routes, by varying the learning rate parameters. Using the proposed cooperative game-theory based framework, all the inhabitants currently present in the house attempt to minimize this overall uncertainty in the form of utility functions associated with them. Joint optimization of the utility function corresponds to the convergence to Nash equilibrium and helps in accurate prediction of inhabitants' future locations and activities. Hypothesizing that every inhabitant wants to satisfy his own preferences about activities, next we look into the problem from the perspective of non-cooperative game theory where the inhabitants are the players and their activities are the strategies of the game. We prove that the optimal location prediction across multiple inhabitants in smart homes is an NP-hard problem and to capture the correlation and interactions between different inhabitants' movements (and hence activities), we develop a novel framework based on a non-cooperative game theoretic, Nash H-learning approach that attempts to minimize the joint location uncertainty of inhabitants. Our framework achieves a Nash equilibrium such that no inhabitant is given preference over others. This results in more accurate prediction of contexts and more adaptive control of automated devices, thus leading to a mobility-aware resource (say, energy) management scheme in multi-inhabitant smart homes. Experimental results demonstrate that the proposed framework is capable of adaptively controlling a smart environment, significantly reduces energy consumption and enhances the comfort of the inhabitants.

To promote independent living and wellness management services in this smart home environment we envision sensor rich computing and networking environments that can capture various types of contexts of patients (or inhabitants of the environment), such as their location, activities and vital signs. However, in reality, both sensed and interpreted contexts may often be ambiguous, leading to fatal decisions if

not properly handled. Thus, a significant challenge facing the development of realistic and deployable context-aware services for healthcare applications is the ability to deal with ambiguous contexts to prevent hazardous situations. In this thesis, we propose a quality assured context mediation framework, based on efficient context-aware data fusion and information theoretic system parameter selection for optimal state estimation in resource constrained sensor network. The proposed framework provides a systematic approach based on dynamic Bayesian network to derive context fragments and deal with context ambiguity or error in a probabilistic manner. It has the ability to incorporate context representation according to the applications' quality requirement. Experimental results demonstrate that the proposed framework is capable of choosing a set of sensors corresponding to the most economically efficient disambiguation action and successfully sensing, mediating and predicting the patients' context state and situation.

Energy-efficient determination of an individual's context (both physiological and activity) is an important technical challenge for this assisted living environments. Given the expected availability of multiple sensors, context determination is viewed as an estimation problem over multiple sensor data streams. We develop a formal, and practically applicable, model to capture the tradeoff between the accuracy of context estimation and the communication overheads of sensing. In particular, we propose the use of *tolerance ranges* to reduce an individual sensor's reporting frequency, while ensuring acceptable accuracy of the derived context. We introduce an optimization technique allowing the context service to compute both the best set of sensors, *and* their associated tolerance values, that satisfy the QoINF target at minimum communication cost. Experimental results with SunSPOT sensors are presented to attest to the promise of this approach.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Advances in smart devices, mobile wireless communications, sensor networks, pervasive computing, machine learning, middleware and agent technologies, and human computer interfaces have made the dream of smart pervasive environments a reality. According to Cook and Das [22], a "smart environment" is one that is able to autonomously acquire and apply knowledge about its users and their surroundings, and adapt to the users' behavior or preferences with the ultimate goal to improve their experience in that environment. The type of experience that individuals expect from an environment varies with the individual and the type of environment considered. This may include the safety of users, reduction of cost of maintaining the environment, optimization of resources (e.g., energy bills or communication bandwidth), task automation or the promotion of an intelligent independent living environment for healthcare services and wellness management. An important characteristic of such an intelligent, pervasive computing and communication paradigm lies in the autonomous and pro-active interaction of smart devices used for determining users' important contexts such as current and near-future locations, activities, or vital signs.

In this sense, *'context awareness'* is a key issue for enhancing users living experience during their daily interaction with computer systems, as only a dynamic adaptation to the task at hand will make computing environments just user friendly and supportive. The combination of awareness with information appliances, or rather

the implementation of awareness in information appliances became known as context awareness [104], since a device should act within its current context of use, by being aware of the various aspects of its current environment. Context awareness is concerned with the situation a *device* or *user* is in, and with adapting applications to the current situation. But knowing the current context an application or system is used in and dynamically adapting to it only allows to construct reactive systems, i.e., systems which run after changes in their environment. To maximize their usefulness and user support, systems should rather adapt in advance to a new situation and be prepared before they are actually used. This demands the development of proactive systems, i.e., systems which predict changes in their environment and act in advance. To this end, we strive to develop methods to learn and predict future context, to mediate ambiguous context, enabling systems to become proactive with regard to their context of use. Our concept is to provide applications not only with information about the current user context, but also with predictions of future user context. When equipped with various sensors, a system should classify current situations and, based on those classes, learn the user's behaviors and habits by deriving knowledge from historical data. The focus of this thesis is to forecast future user contexts lucidly by extrapolating the past and derive techniques that enable context prediction in pervasive systems and leaves decisions about starting actions to applications built on top of it.

1.2 Challenges

An instance of such an intelligent indoor environment is a *smart home* [27] that perceives the surroundings through sensors and acts on it with the help of actuators. In this environment, user's mobility and activity create an uncertainty of their locations and hence subsequent activities. In order to be cognizant of his contexts, the

smart environment needs to minimize this uncertainty. An analysis of his daily routine and life style reveals that there exist some well defined patterns of these contexts. Although these patterns may change over time, they do not change too frequently and thus can be learned. An optimal algorithm for location (activity) tracking in an indoor smart environment, based on dictionary management and online learning of the inhabitant's mobility profile, followed by a predictive location-aware resource management (energy consumption) scheme for a single inhabitant smart home is discussed in [94]. However, the presence of multiple inhabitants with dynamically varying profiles as well as preferences make such tracking much more challenging. This is due mainly to the fact that the relevant contexts of multiple inhabitants in the same environment are often inherently *correlated* and *inter-dependent* on each other. Therefore, the learning and prediction (decision making) paradigm needs to consider the joint (simultaneous) location/activity tracking of multiple inhabitants which we address in this thesis. Furthermore, hypothesizing that each inhabitant in a smart home behaves in such a way as to fulfill his own objectives and maximizes his utility, the residence of multiple inhabitants with varying preferences might lead to conflicting goals. Thus, a smart home must be intelligent enough to strike a balance between multiple preferences, eventually attaining an equilibrium state. This motivates us to investigate the multi-inhabitant location tracking problem from the perspective of stochastic game theory, where the inhabitants are the players of the game. The goal here is to achieve an equilibrium so that the system (i.e., smart home) is able to probabilistically predict the inhabitants' locations and activities with sufficient accuracy in spite of possible correlations.

In this thesis we also look into how the various types of contexts of patients (or inhabitants of the environment), such as their location, activities and vital signs can provide health related and wellness management services in an intelligent, energy-

efficient way so as to promote independent living. However, in reality, both sensed and interpreted contexts may often be ambiguous, leading to fatal decisions if not properly handled. Thus, a significant challenge facing the development of realistic and deployable context-aware services for healthcare applications is the ability to deal with ambiguous contexts to prevent hazardous situations.

1.3 Problem Statement

Our main focus of research is on user centered learning and prediction of context and presenting a context-aware middleware framework for autonomous resource management and ambiguous context mediation subsystem. Context, in the field of pervasive computing, has been defined in different ways. One of the first definitions of context in [106] states that it comprises computing, user and physical properties. The definition adopted within this thesis is the one by Dey et.al. [32], according to which context is any information which can be used to characterize the situation of an entity, where an entity is a person, place or object that is considered relevant to the interaction between a user and an application, including the user and the application themselves. We define context awareness as incorporating learned, predicted, future context into the device behavior and being prepared to future situations. We can define the problem statement as follows: What are the necessary concepts, architectures and methods for context learning and prediction, context modeling and mediation in smart pervasive systems? Thus, the research goal is to evaluate and, if necessary, develop methods for learning, predicting, modeling and mediating context with the limited resources of pervasive systems.

1.4 Scope and Methodology

A smart home aims at building intelligent automation with a goal to provide its inhabitants with maximum possible comfort, minimum resource consumption and thus reduced cost of home maintenance. 'Context Awareness' is perhaps the most salient feature of such an intelligent environment. An inhabitant's mobility and activities play a significant role in defining his contexts in and around the home. Although there exists optimal algorithm for location and activity tracking of a single inhabitant, the correlation and dependence between multiple inhabitants' contexts within the same environment make the location and activity tracking more challenging. In this thesis, first we propose a cooperative entropy learning policy for location-aware resource management in multi-inhabitant smart homes. This approach adapts to the uncertainty of multiple inhabitants' locations and most likely routes, by varying the learning rate parameters and minimizing the Mahalanobish distance. However, the complexity of multi-inhabitant location tracking problem was not characterized in this work. But the optimal location prediction across multiple inhabitants in smart homes is an NP-hard problem. Next, to capture the correlation and interactions between different inhabitants' movements (and hence activities), we develop a novel framework based on a game theoretic, Nash H-learning approach that attempts to minimize the joint location uncertainty of inhabitants. Our framework achieves a Nash equilibrium such that no inhabitant is given preference over others. This results in more accurate prediction of contexts and more adaptive control of automated devices, thus leading to a mobility-aware resource (say, energy) management scheme in multi-inhabitant smart homes. Experimental results demonstrate that the proposed framework is capable of adaptively controlling a smart environment, significantly reduces energy consumption and enhances the comfort of the inhabitants.

To promote independent living and wellness management services in a smart home environment we envision sensor rich computing and networking environments that can capture various types of contexts of patients (or inhabitants of the environment), such as their location, activities and vital signs. Given the expected availability of multiple sensors, context determination may be viewed as an estimation problem over multiple sensor data streams. We develop a formal, and practically applicable, model to capture the tradeoff between the accuracy of context estimation and the communication overheads of sensing. In particular, we propose the use of *tolerance ranges* to reduce an individual sensor's reporting frequency, while ensuring acceptable accuracy of the derived context. We introduce an optimization technique allowing the Context Service to compute both the best set of sensors, *and* their associated tolerance values, that satisfy the QoINF target at minimum communication cost. We also propose a novel framework for context mediation, based on efficient context-aware data fusion and information theoretic reasoning. The proposed framework provides a systematic approach based on dynamic Bayesian network to derive context fragments and deal with context ambiguity in a probabilistic manner. It has the ability to incorporate context representation within the applications and also easily composable rules to mediate ambiguous contexts. We have implemented a demonstration of the use of our model. Experimental results demonstrate that the proposed framework is capable of choosing a set of sensors corresponding to the most economically efficient disambiguation action and successfully predicting the patients' situation.

1.5 Results

The present thesis analyzes prerequisite for user centered learning and prediction of context and present a framework for autonomous resource management in smart home environment and ambiguous context mediation subsystem with appli-

cation to smart healthacre. The developed system is being implemented in terms of a flexible software framework and evaluated with real-world data from everyday situations.

1.6 Organization

The remainder of this thesis is split into six Chapters. Chapter 2, which defines the specific goals and presents the general concept for context-aware resource management through learning and prediction in a cooperative multi-inhabitant smart home. In Chapter 3 we look into the same problem from non-cooperative perspective to strike a balance between multiple preferences of the inhabitants. Chapter 4 then shows how context information is useful in providing health related and wellness management services in an intelligent way so as to promote independent living. Chapter 5 then presents the determination of this health related context in an efficient way using the resource constrained sensor network. In Chapter 6, related work is summarized and this thesis is positioned among and against other publications with regard to novelties in our approach and differences to previous work. Finally, in Chapter 7 the thesis is summarized by pointing out the main arguments and the scientific contribution and giving an outlook on possible future research.

CHAPTER 2

COOPERATIVE MOBILITY AWARE RESOURCE MANAGEMENT

2.1 Introduction

The vision of *ubiquitous computing* was first conceived by M. Weiser at Xerox PARC as the future model for computing [121]. The most significant characteristic of this computing paradigm lies in smart, pro-active interaction of the hand-held computing devices with their peers and surrounding networks, often without explicit operator control. Hence, the computing devices need to be imbued with an inherent sentience [54] about their important contexts. This *context-awareness* is perhaps the key characteristic of the next generation of intelligent networks and associated applications. The advent of smart homes is germinated from the concept of ubiquitous computing in an indoor environment with a goal to provide the inhabitants with sufficient comfort at minimum possible operational. Obviously, the technology needs to be weaved into the inhabitants' everyday life such that it becomes "technology that disappears" [121]. A careful insight into the features of a smart home reveals that the ability to capture the current and near-future locations and activities (hence 'contexts') of different inhabitants often becomes the key to the environment's associated "smartness". Intelligent prediction of inhabitants' locations and routes aids in efficient triggering of active databases or guaranteeing a precise time frame of service, thereby supporting location-aware interactive, multimedia applications. This also helps in pro-active management of resources such as energy consumption.

Given the wide variety of smart, indoor location-tracking paradigms, let us summarize below some of the important ones. The Active Badge [45] and Active Bat [46]

use infra-red and ultrasonic time-of-flight techniques for indoor location tracking. On the other hand, the Cricket Location Support System [92] delegates the responsibility of location reporting to the mobile object itself. RADAR [4], another RF-based indoor location support system, uses signal strength and signal-to-noise ratio to compute 2-D positioning. The Easy-living and the Home projects [72] use real-time 3D cameras to provide stereo-vision positioning capability in an indoor environment. In the Aware Home [87], the embedded pressure sensors capture inhabitant's footfalls, and the system uses this data for position tracking and pedestrian recognition. The Neural Network House [82], Intelligent Home [77] and Intelligent House_n [57] projects focus on the development of adaptive control of home environments to anticipate the needs of the inhabitants.

In an earlier work [94], we proposed location-aware resource management considering a single-inhabitant smart home. However, the presence of multiple inhabitants with varying preferences and requirements makes the problem more challenging. A suitable balance of preferences arising from multiple inhabitants [108] needs to be considered. Thus, the environment (or system) needs to be more smart to extract the best performance while satisfying the requirements of the inhabitants as much as possible.

2.1.1 Our Contributions

In this chapter we have developed a framework for mobility-aware resource management in multi-inhabitant smart homes, based on a dynamic, cooperative learning technique. Here the resource management means the reduction of the consumption of energy. The movement pattern and various activities of the inhabitants always create an uncertainty of their locations and subsequent activities. In order to be cognizant of the inhabitants' contexts, the system needs to minimize this uncertainty

which can be measured by Shannon's entropy [26]. An analysis of inhabitants' daily routines reveals that every inhabitant has some patterns in daily-life that can be learnt. Although the life style (pattern) changes over time, such changes are not frequent and random. This observation helps us assume that the inhabitant's mobility and associated activities follow a *piece-wise stationary, ergodic, stochastic process* [7], with some value of entropy (uncertainty) associated with it. The novelty of our work lies in the development of a new framework based on *cooperative game theory and reinforcement learning* to minimize the overall uncertainty associated with multiple inhabitants currently present in the smart home. This is performed by developing a *joint utility function* of entropy. Optimization of this utility function asymptotically converges to *Nash Equilibrium* [8]. Minimizing the utility function of uncertainty helps in accurate learning and estimation of inhabitants' contexts (locations and associated activities). Thus, the system can control the operation of automated devices in an adaptive manner, thereby developing an amicable environment inside the home and providing sufficient comfort to the inhabitants. This also aids in minimizing the energy usage, leading to a reduction of overall maintenance cost of the house.

The rest of the chapter is organized as follows. The problem definition, basic concepts of cooperative framework and information theoretic estimation of location uncertainty are discussed in Section 2.2. The new game-theoretic learning framework that minimizes uncertainty associated with all inhabitants, is presented in Section 2.3. In Section 2.4 we present the analytical model for estimation and classification process of different values of uncertainty level. Section 2.5 demonstrates the use of the proposed framework in resource optimization in multi-inhabitant smart homes. Simulation results in Section 2.6 delineates the efficiency of our framework. Section 2.7 concludes the chapter with pointers to future researches.

2.2 Preliminaries

The smart home environment, basic concepts of cooperative framework, information theoretic estimation of location uncertainty and the learning in cooperative environments are discussed here.

2.2.1 Overview of Smart Homes

The MAVHome (**M**anaging **A**n intelligent **V**ersatile **Home**) [27] is a multi-disciplinary research project at the University of Texas at Arlington. It is focused on the creation of an intelligent home environment capable of perceiving its surroundings through the use of sensors, and thereby adopting suitable actions by using the actuators. In such a smart computing platform there exists movements of inhabitants interacting with their surrounding environments through the hand-held devices. The overall goal is to provide the inhabitant's comfort at an optimal cost. Efficient and intelligent estimate and prediction of inhabitants' contexts (location and activity) is the most necessary component of such a smart home.

2.2.2 Cooperative Framework for Inhabitants Mobility

Our proposed framework is based on symbolic interpretation of the inhabitant's movement (mobility) within the home, which is captured by sampling the in-building sensors (RF-ID readers or pressure switches). Thus, the movement history of an inhabitant is assumed as a string "$v_1 v_2 v_3 \ldots$" of symbols (sensor-ids) where $v_i \in \vartheta$ (the alphabet set). We argue that the inhabitant's mobility and current location is merely a reflection of his/her movement history (profile), which can be learned over time in an on-line fashion. Characterizing such mobility as a probabilistic sequence suggests that it can be defined as a stochastic process $\mathcal{V} = \{V_i\}$, while the *repetitive* nature of *identifiable patterns* adds *stationarity* as an essential property, leading to

$Pr[V_i = v_i] = Pr[V_{i+l} = v_i]$ for all $v_i \in \vartheta$ and for every shift l. The movement of the set of inhabitants inside the smart home always create an uncertainty in their locations and activities. The concept of entropy [26] in information theory is the most fair measure to estimate this uncertainty.

2.2.3 Information Theoretic Estimate for Location Uncertainty

The entropy $H_b(X)$ of a discrete random variable X with probability mass function $p(x)$, $x \in \mathcal{X}$, is defined by: $H_b(X) = -\sum_{x\in\mathcal{X}} p(x) \log_b p(x)$. The limiting value "$\lim_{p\to 0} p \log_b p = 0$" is used in the expression when $p(x) = 0$. The *relative entropy* between two probability mass functions $p(x)$ and $q(x)$, $x \in \mathcal{X}$, is given by $D(p||q) = \sum_{x\in\mathcal{X}} p(x)\ log\frac{p(x)}{q(x)}$. This relative entropy is a fair measure of the inefficiency of assuming that the distribution is q, when the actual distribution is p. Also, the *conditional entropy* is defined as $H(Y|X) = \sum_{x\in\mathcal{X}}\ p(x)\, H(Y|X = x)$. For any set $\{V_1, V_2, \ldots, V_k\}$ of k discrete random variables with distribution given by $p(v_1,\, v_2,\, \ldots,\, v_k) = \Pr[V_1 = v_1,\, V_2 = v_2,\, \ldots,\, V_k = v_k]$, where $v_i \in \vartheta$, the joint entropy is given by $H(V_1,\, V_2,\, \ldots,\, V_k) = \sum_{i=1}^{k} H(V_i \mid V_1,\, V_2,\, \ldots,\, V_{i-1})$. The additive terms on the right-hand side carry necessary information which makes the higher-order context models more information-rich as compared to the lower-order ones.

2.2.4 Learning in Cooperative Environments

Our investigation in this chapter is focussed on n-player cooperative repeated games. Let n denote the number of inhabitants, s the set of states, a_i the set of actions available to inhabitant i with $\mathcal{A} = (a_1 \times a_2 \times ... \times a_n)$ as the joint action space, $\pi : s \times \mathcal{A} \times s \to [0, 1]$ the probability of selecting a policy/route of moving from state s to $s^{'}$ on performing action $\mathcal{A}$, and H_i the utility function of the i-th inhabitant defined by $s \times \mathcal{A} \to H$. We assume the inhabitants are fully rational in the sense

that they can fully use their available histories or beliefs to construct future route strategy. Each inhabitant i keeps a count $\mathcal{C}^j_{a_j}$ which represents the number of times user j has followed an action for a specific route in the past for each j and $a_j \in \mathcal{A}_j$, where $1 \leq j \leq n$ and $i \neq j$. When the game is encountered, inhabitant i believes the relative frequencies of each of j's move as indicative of j's current route. So for each inhabitant j, inhabitant i assumes j plays action $a_j \in \mathcal{A}_j$ with probability [20]:

$$\pi(a_j)_i = \frac{\mathcal{C}^j_{a_j}}{\sum_{\flat_j \in \mathcal{A}_j} \mathcal{C}^j_{\flat_j}} \tag{2.1}$$

We consider these counts as reflecting the observations an inhabitant has regarding the route strategy of the other inhabitants. As a result, the decision making component should not directly repeat the actions of the inhabitants but rather learn to perform actions that optimize a given reward or utility function.

2.3 Inhabitant's Utility Function based on Cooperative Learning

In a smart home environment, an inhabitant's goal is to optimize the total utility it receives. To address these requirements of optimization, the decision making component of smart home uses reinforcement learning to acquire a policy that optimizes overall uncertainty of the inhabitants which in turn helps in accurate prediction of inhabitants' locations and activities. In this section we present an algorithm from an information-theoretic perspective for learning a value function that maps state-action pairs to future discounted reward using Shannon's entropy measure.

2.3.1 Entropy Learning based on Individual Policy

Most reinforcement-learning (RL) algorithms use evaluation or value functions to cache the results of experience for solving discrete optimal control problems. This is useful in our case because close approximations to optimal entropy value function

lead the inhabitant directly towards its goal by possessing some good control policies. Here we closely follow the Q-learning (associate values with state-action pairs, called Q values as in Watkins' Q-learning) [120] for our Entropy learning (H-learning) algorithm that combines new experience with old value functions to produce new and statistically improved value functions in different ways. First, we discuss how the algorithm uses its own system beliefs to change its estimate to optimal value functions called *update rule*. Then we discuss a *learning policy* that maps histories of states visited, probability of action chosen ($\pi(a_j)_i$), current hamming distance (d_h) and the utility received ($H_t(s_t, a_t)$); into a current choice of action. Finally, we claim that this learning policy results in convergence when combined with the H-learning update rule.

To achieve the desired performance of smart homes, a reward function, r, is defined that takes into account the success rate of achieving the goal using system beliefs. Here r is the instantaneous reward received which we have considered as success rate of the predicted state. One measure of this prediction accuracy can be estimated from per-symbol Hamming distance (d_h) which provides the normalized symbol-wise mismatch between the predicted and the actual routes followed by the inhabitants. Intuitively, this measure should have correspondence with the relative entropy between the two sequences. A direct consequence of information theory helps in estimating this relationship [94].

Using the state space and reward function, the H-learning is used similar to Q-learning algorithm to approximate an optimal action strategy by incrementally estimating the entropy value, $H_t(s_t, a_t)$, for state/action pairs. This value is the pre-

dicted future utility that will be achieved if the inhabitant executes action a_t in state s_t. After each action, the utility is updated as

$$H_{t+1}(s_t, a_t) = (1 - \alpha)H_t(s_t, a_t) + \alpha[r_t + \gamma \min_{a \in \mathcal{A}} H_t(s_{t+1}, a_{t+1})] \tag{2.2}$$

where H_t is the estimated entropy value at the beginning of the t-th time step, and s_t, a_t, r_t are the state, action and reward at time step t. Update of $H_{t+1}(s_t, a_t)$ depends on $\min_{a \in \mathcal{A}} H_t(s_{t+1}, a_{t+1})$ which relies on comparing various predicted actions [109]. The parameters α and γ are both in the range 0 to 1. When the learning rate parameter α is close to 1, the H-table changes rapidly in response to new experience. When the discount rate γ is close to 1, future interactions play a substantial role in defining the total utility values. After learning the optimal entropy value, a_t can be determined as

$$a_t = \min_{a \in \mathcal{A}} H_t(s_t, a_t) \tag{2.3}$$

Here we propose a learning policy that selects an action based on the function of the history of the states, actions and utility. This learning policy makes decision based on a summary of history consisting of the current state s, current estimate of the entropy value function as a utility, number of times inhabitant j has used its action a_j in the past and Hamming distance (d_h). Such a learning policy can be expressed as the probability $Pr(a|s, H_t(s_t, a_t), \pi(a_j)_i, d_h)$, that the action a is selected given the history. An example of such a learning policy is a form of Boltzmann exploration [109]:

$$Pr(a|s, H_t(s_t, a_t), \pi(a_j)_i, d_h) = \frac{e^{\pi(a_j)_i} H_t(s_t, a_t)/d_h}{\sum e^{\pi(a_j)_i} H_t(s_t, a_t)/d_h} \tag{2.4}$$

The differential distance parameter, d_h, will be decreased over time as the inhabitant reaches its goal. Consequently, the exploration probability is increased ensuring the convergence.

2.3.2 Entropy Learning based on Joint Policy

For cooperative action learners (CAL), the selection of the actions should be done carefully. To determine the relative values of their individual actions, each inhabitant in a CAL algorithm maintains beliefs about the strategy of other inhabitants. From this perspective, inhabitant i predicts the Expected Entropy Value (EEV) of its individual action a_i at t-th time step as follows

$$EEV_t(a_i) = \sum_{a_{-i} \in \mathcal{A}} H_t\{(s_t, a_{-i(t)}) \cup (s_t, a_{i(t)})\} \prod_{j \neq i} \pi(a_{-i})_j \tag{2.5}$$

2.3.3 A New Algorithm for Optimizing Joint Uncertainty

In this section we describe an algorithm (see Figure 2.1) for a rational and convergent cooperative action learner. The basic idea is to vary the learning rate used by the algorithm so as to accelerate the convergence, without sacrificing rationality. In this algorithm we have a simple intuition like "learn quickly while predicting the next state incorrectly", and "learn slowly while predicting the next state correctly". The method used here for determining the prediction accuracy is by comparing the current policy's entropy with that of the expected entropy value earned by the cooperative action over time. This principle aids in convergence by giving more time for the other inhabitants to adapt to changes in the inhabitant's strategy that at first appear beneficial, while allowing the inhabitant to adapt more quickly to the other inhabitants' strategy changes when they are harmful [8]. We use two learning rate parameters, namely "succeeding" (δ_s) and "failing" (δ_f), where $\delta_s < \delta_f$. The term $|\mathcal{A}_i|$ denotes the number of available joint actions of i-th inhabitant. The policy is improved by increasing the probability so that it selects the highest valued action according to the learning rate. The learning rate used to update the probability depends on whether the inhabitant is currently succeeding (δ_s) or failing (δ_f). This is

Procedure CAL
Input: Individual and joint expected entropy values
Output: Decision on the learning rate
1. Let α and $\delta_f > \delta_s$ be the learning rates. Initialize
$H_t(s_t, a_t) \leftarrow 0$, $Pr(a|s, H_t(s_t, a_t), \pi(a_j)_i, d_h) \leftarrow \frac{1}{|\mathcal{A}_i|}$.
2. Repeat
a) From state s select action a with probability
$Pr(a|s, H_t(s_t, a_t), \pi(a_j)_i, d_h)$
b) Observing reward H_t and next state s_t, update
$H_{t+1}(s_t, a_t) \leftarrow (1-\alpha)H_t(s_t, a_t) + \alpha[r_t + \gamma \min_{a \in \mathcal{A}} H_t(s_{t+1}, a_{t+1})]$
c) Calculate Joint Entropy value as
$EEV_t(a_i) = \sum_{a_{-i} \in \mathcal{A}} H_t\{(s_t, a_{-i(t)}) \cup (s_t, a_{i(t)})\} \prod_{j \neq i} \pi(a_{-i})_j$
d) Update $Pr(a|s, H_t(s_t, a_t), \pi(a_j)_i, d_h)$ as
$Pr(a|s, H_t(s_t, a_t), \pi(a_j)_i, d_h) \leftarrow Pr(a|s, H_t(s_t, a_t), \pi(a_j)_i, d_h) +$
$\begin{cases} \delta & \text{if } a = argmin_{a_t} H_t(s_t, a_t) \\ \frac{-\delta}{|\mathcal{A}_i|-1} & \text{otherwise} \end{cases}$
where,
$\delta = \begin{cases} \delta_s & \text{if } H_{t+1}(s_t, a_t) > EEV_t(a_i) \\ \delta_f & \text{otherwise} \end{cases}$

Figure 2.1. Procedure of a Cooperative Action Learner (CAL) .

determined by comparing the current estimation of the entropy value following the current policy, π, in the current state with that of following the joint policy. If the individual entropy value of the current policy is smaller than the joint expected entropy value, then the larger learning rate δ_f is used in the sense that the inhabitant is currently "failing".

Proposition 1 Our CAL algorithm converges to a Nash Equilibrium if the following two conditions hold:

i) Optimization towards Believing in Rationality:

$EEV_t(a_i) \in argmin_{a_t}(H_{t+1}(s_t, a_t)) \forall t$

The joint expected entropy value tends to be one of the candidates of the set of all optimal entropy values followed by our H-learning process defined previously.

ii) Convergence towards Playing in Believing:

$\lim_{t\to\infty} |H_{t+1}(s_t, a_t) - EEV_t(a_i)| = 0$

The difference between the current entropy value following the current policy π in the current state with that of the joint entropy value tends to 0.

These two properties guarantee that the inhabitant will converge to a stationary strategy that is optimal given the actions of the other inhabitants. As is standard in the game theory literature, it is thus reasonable to assume that the opponent is fully rational and chooses actions that are in its best interest. When all inhabitants are rational, if they converge, then they must have converged to a Nash equilibrium. Since all inhabitants converge to a stationary policy, each rational inhabitant must converge to the best response to the opponent choice of actions. After all, if all inhabitants are rational and convergent with respect to other inhabitant strategies, then convergence to a Nash equilibrium is guaranteed [8].

Proposition 2 The learning rate α $(0 \leq \alpha \leq 1)$ decrease over time such that it satisfies $\sum_{t=0}^{\infty} \alpha = \infty$ and $\sum_{t=0}^{\infty} \alpha^2 \leq \infty$

Proposition 3 Each inhabitant samples each of its actions infinitely often. Thus probability of inhabitant i choosing action a_t is nonzero. Hence $Pr_i(a_t) \neq 0$

Proposition 4 The probability of choosing some nonoptimal action in the long run tends to zero since each inhabitant's exploration strategy is exploitive.

Hence, $\lim_{t\to\infty} Pr(a|s, H_t(s_t, a_t), \pi(a_j)_i, d_h) = 0$

Proposition 2 and 3 are required conditions for our Entropy learning algorithm. They ensure that inhabitants could not adopt deterministic exploration strategies and

become strictly correlated. The last proposition states that the inhabitants always explore their knowledge. This is necessary to ensure that an equilibrium will be reached.

2.4 Classification and Estimation of the Uncertainty Level

Mahalanobis distance [102] is a very useful way of determining the "similarity" of a set of values from an unknown sample to a set of values measured from a collection of "known" samples. In our scenario, the entropy values calculated by the inhabitants once in an individual mode and on the other hand in a cooperative mode in the smart home environment are correlated to each other. From this perspective we have used Mahalanobis distance as the basis for our analysis which takes distribution of the entropy correlations into account compared to the traditional Euclidean distance. The advantage of using this approach lies in extending the inhabitants to choose the most efficient route with the minimum entropy value.

To provide the most efficient route to the inhabitants of smart home, we consider an N-dimensional space of individual Entropy Value Level (EVL) $\wp = [\wp_1, \wp_2, \wp_3, ..., \wp_N]$ evolved by N different actions at different time instant. In our model, due to cooperative learning among the inhabitants, another set of EVL such as $e = [e_1, e_2, e_3, ..., e_N]$ could be evolved due to the joint actions of the inhabitants. Thus we have two different estimation of the entropy values. One estimation has been done due to individual action and the other estimation is due to joint actions in a cooperative environment. Therefore we have two points, $\wp$ and e, in the N-dimensional space representing two different EVL "states".

Let us have two groups, $\mathcal{G}_1$ and $\mathcal{G}_2$, consisting of different inhabitants distinguished by their EVL measures. For example, group $\mathcal{G}_1$ may contain inhabitants who provide route in accordance with EVL, $\wp$, and group $\mathcal{G}_2$ in accordance with e. If we

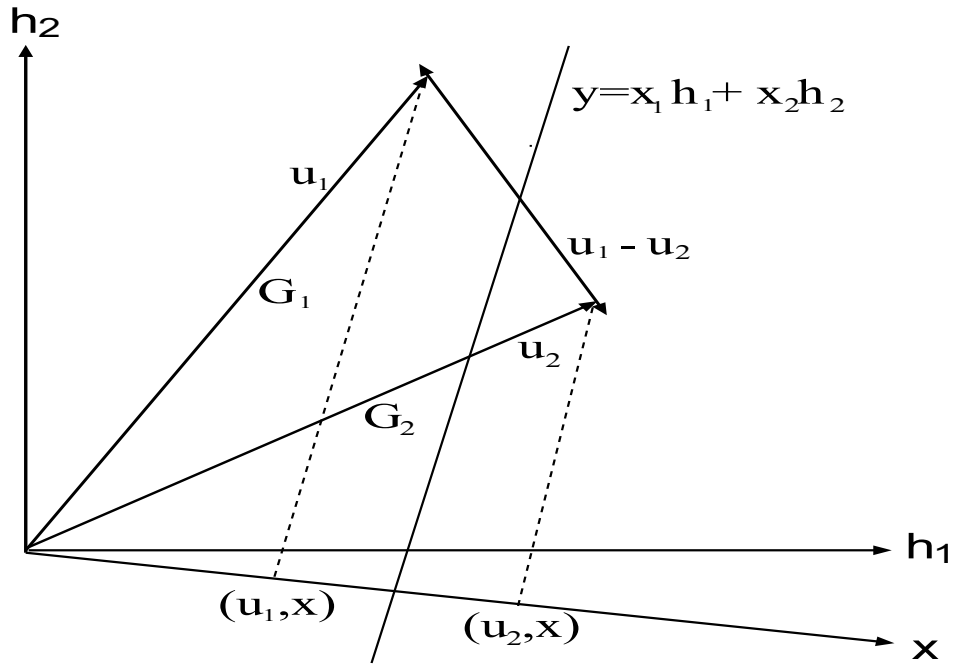

Figure 2.2. Geometric Interpretation of Entropy Value Level Classification.

now have one new entropy value h, the problem is to classify it as either belonging to $\mathcal{G}_1$ or $\mathcal{G}_2$. We reduce this problem to the classification of two Gaussian groups by means of multi-dimensional statistical analysis. For characterizing these two groups, we choose two N-dimensional Normal (Gaussian) distributions $\mathcal{N}_n(\mu_1, \mathbf{V})$ for group $\mathcal{G}_1$ and $\mathcal{N}_n(\mu_2, \mathbf{V})$ for group $\mathcal{G}_2$, respectively. Therefore for these two cases, we have the following characteristic functions:

1) $\mu_1 = (\mu_{11}\mu_{12}...\mu_{1N})^T$ for $\mathcal{G}_1$ assuming as "succeeding" cases, and $\mu_2 = (\mu_{21}\mu_{22}...\mu_{2N})^T$ for $\mathcal{G}_2$ assuming "failing" cases, where T denotes transposition. Here μ_1 and μ_2 represent the means for all the entropy in the multivariate space defined by the EVL in the model. These points can be called as group *Entropy Centroid*. For each new entropy values, we can then compute the Mahalanobis distances from each of the group Entropy Centroid. We would classify the EVL as belonging to the group to which it is the closest, that is, where the Mahalanobis distance is the smallest.

2) The Covariance matrix $\mathbf{V} = [\sigma_{\mathbf{ij}}]$ is the same for both the distributions.

Our N-dimensional EVL measures are given by $h = [h_1, h_2, ..., h_N]$. For the two-group case, we use a linear discriminant function that can also be thought of as multiple regression. In general, we fit a linear equation of the type: $z = x_1h_1 + x_2h_2 + ... + x_Nh_N$ which is a scalar product of vectors x and h, where the vector $x = [x_1, x_2, ..., x_N]$ represents unknown regression coefficients. We have defined the following decision rule depending upon some threshold value y, such that $h \in \mathcal{G}_1$ if $z \leq y$, otherwise $h \in \mathcal{G}_2$.

Thus we reduce the classification issue into two problems: a) to determine the N unknown coefficients $x_1, x_2, ...x_N$ so that the distance between the projections of mean vectors μ_1 and μ_2 on vector x is maximal, and b) to choose point y between these projections on vector x, minimizing the probability of wrong classification which in turn provides the optimal EVL to the inhabitants.

The overall classification process is shown in Figure 2.2 for two naturally occurring EVL groups $\mathcal{G}_1$ and $\mathcal{G}_2$, which can be divided by the line $x_1h_1 + x_2h_2 = y$.

Mahalanobis distance: Mahalanobis distance [102], $\mathcal{D}_m^2$, is a generalized measure of the distance between two correlated groups as it adequately accounts for the correlations. If our point h belongs to group $\mathcal{G}_1$, then variable z defined previously has one-dimensional normal distribution with mean and variance as follows [102].

$$z_1 = \sum_{i=1}^{N} x_i\mu_{1i} = x^T\mu_1 \quad \sigma_z^2 = \sum_{i=1}^{N}\sum_{j=1}^{N} x_ix_j\sigma_{ij} = x^T\mathbf{V}x \tag{2.6}$$

In a similar way if h belongs to group $\mathcal{G}_2$, then z has a normal distribution with mean z_2 and the same variance.

$$z_2 = \sum_{i=1}^{N} x_i \mu_{2i} = x^T \mu_2 \quad \sigma_z^2 = \sum_{i=1}^{N} \sum_{j=1}^{N} x_i x_j \sigma_{ij} = x^T \mathbf{V} x \tag{2.7}$$

The distance between groups $\mathcal{G}_1$ and $\mathcal{G}_2$ can be expressed as

$$\mathcal{D}_m^2 = (\mu_2 - \mu_1)^T \mathbf{V^{-1}} (\mu_\mathbf{2} - \mu_\mathbf{1}) \tag{2.8}$$

using Equations (2.6) and (2.7). Now we need to find out the constants $x_1, x_2, ...x_N$ maximizing the so called Mahalanobis distance $\mathcal{D}_m^2 = \frac{(z_2 - z_1)^2}{\sigma_z^2}$. The solution of x as obtained from [102] is $x = \mathbf{V^{-1}}(\mu_\mathbf{2} - \mu_\mathbf{1})$. Thus, the guaranteed best entropy value level can be determined as $z = x_1 h_1 + x_2 h_2 + ... + x_N h_N$.

Next we need to minimize the misclassification probability. Classification is the process by which a decision is made whether a particular inhabitant belongs to a particular group. Let N_1 denote the number of inhabitants that truly belong to group $\mathcal{G}_1$, and let N_2 denote the number of inhabitants that truly belong to group $\mathcal{G}_2$. Let N_{11} be the number of inhabitants that actually belong to group $\mathcal{G}_1$ and assigned to group $\mathcal{G}_1$ (i.e., correctly classified). Let N_{12} be the number of inhabitants that belong to group $\mathcal{G}_1$ but are assigned to group $\mathcal{G}_2$ (i.e., incorrectly classified). Similarly, N_{21} denote the number of inhabitants that belong to group $\mathcal{G}_2$ but are incorrectly classified into $\mathcal{G}_1$, and N_{22} denote the number of inhabitants that belong to group $\mathcal{G}_2$ and are correctly classified into $\mathcal{G}_2$. Then the total number of incorrectly classified inhabitants is $N_{12} + N_{21}$ and hence the probability of incorrectly classified inhabitants is $\psi = \frac{N_{12} + N_{21}}{N}$ where N is the total number of inhabitants. Thus ψ denotes the probability of choosing group $\mathcal{G}_1$ when the correct group is $\mathcal{G}_2$ or vice

versa. The probability (ψ_1) of choosing group $\mathcal{G}_2$ when the true one is $\mathcal{G}_1$ can be expressed as [95]

$$\psi_1 = Pr[\mathcal{G}_2|\mathcal{G}_1] = Pr[z > y|\mathcal{G}_1] = 1 - \Phi(\frac{y - z_1}{\sigma_z}) = \frac{N_{12}}{N} \tag{2.9}$$

where Φ denotes the normal distribution function. Similarly, the probability (ψ_2) of choosing group $\mathcal{G}_1$ when the true one is $\mathcal{G}_2$ can be expressed as

$$\psi_2 = Pr[\mathcal{G}_1|\mathcal{G}_2] = Pr[z \leq y|\mathcal{G}_2] = 1 - \Phi(\frac{z_2 - y}{\sigma_z}) = \frac{N_{21}}{N} \tag{2.10}$$

Assuming the threshold value of the entropy value level y as $\frac{z_1+z_2}{2}$, the total probability of misclassification can be expressed as

$$\begin{aligned} \psi = \psi_1 + \psi_2 &= Pr[\mathcal{G}_2|\mathcal{G}_1] + Pr[\mathcal{G}_1|\mathcal{G}_2] \\ &= \{1 - \Phi(\frac{y - z_1}{\sigma_z})\} + \{1 - \Phi(\frac{z_2 - y}{\sigma_z})\} \\ &= 2\{1 - \Phi(\frac{z_2 - z_1}{2\sigma_z})\} = 2\{1 - \Phi(\frac{\mathcal{D}_m}{2})\} \\ &= 2\Phi(-\frac{\mathcal{D}_m}{2}) = \frac{N_{12} + N_{21}}{N} \end{aligned} \tag{2.11}$$

2.5 Resource and Comfort Management in Smart Homes

One of the objectives behind the development of smart homes is to provide the inhabitants with maximum possible comfort at minimum possible energy consumption. However, the inhabitants' location uncertainty inside the house leads to uncertainty in their activities and operation of smart indoor appliances. Once this uncertainty is minimized for the entire set of inhabitants, the house becomes intelligent enough to make more accurate estimations of the inhabitants' activities and aids them with smart control of automated devices. The novelty of our approach lies in the development of mobility-aware resource management framework, which considers

multiple inhabitants inside the house. Efficient estimation of most likely locations and routes used by these set of inhabitants helps in pro-active, automated operations of smart devices, thus developing an amicable environment inside the house, while conserving the energy dissipation as much as possible.

2.5.1 Mobility-Aware Energy Conservation

The energy consumption over the entire smart home needs to be optimized for reducing the maintenance cost. At the same time we need to consider the inhabitant's comfort by reducing the explicit manual operation and control of smart devices and appliances. Today's houses mostly use static energy management scheme, where a fixed number of devices (electric lights, fans, etc) are kept on for a certain fixed amount of time. Intuitively, this results in sufficient loss of valuable energy inside the house. One obvious solution is to manually control these devices while leaving or entering particular locations inside the house. However, such manual operations are in the opposite pole of inhabitants' comfort and automation. Hence, a smart energy management system needs to be designed that will operate in a proactive fashion while considering unnecessary wastage of in-house resources. We argue that location awareness is the key behind such energy management framework. The automated devices (e.g., lights and fans) operate in a *pro-active mode* to conserve energy during the absence of any inhabitant in particular locations inside the house. These devices also attempt to bring the indoor environment in an amicable condition before the user actually enters into those specific locations. Also, whenever a particular location/region of the house becomes unoccupied by the inhabitants, the automated devices are switched off to conserve the energy.

Let $\mathcal{P}_{ij}$ denote the power of the i-th device in the j-th zone, η denote the maximum number of devices which remained turned on in the particular zone, $\tilde{R}$ denote

the number of zones, $t_1 \leq t \leq t_2$ denote the time that device remains turned on, and $p(t)$ denote the probability density function of uniform time distribution. Then the expected average energy ($\mathcal{E}$) consumed due to lights and devices will be given by [94]:

$$\mathcal{E} = \frac{t_2 - t_1 + \Delta_t}{2} \sum_{j=1}^{\tilde{R}} \sum_{i=1}^{\eta} \mathcal{P}_{ij}, \tag{2.12}$$

where Δ_t is the time-lag between the time of device-operation and the first inhabitant's entrance in the zone (e.g., room).

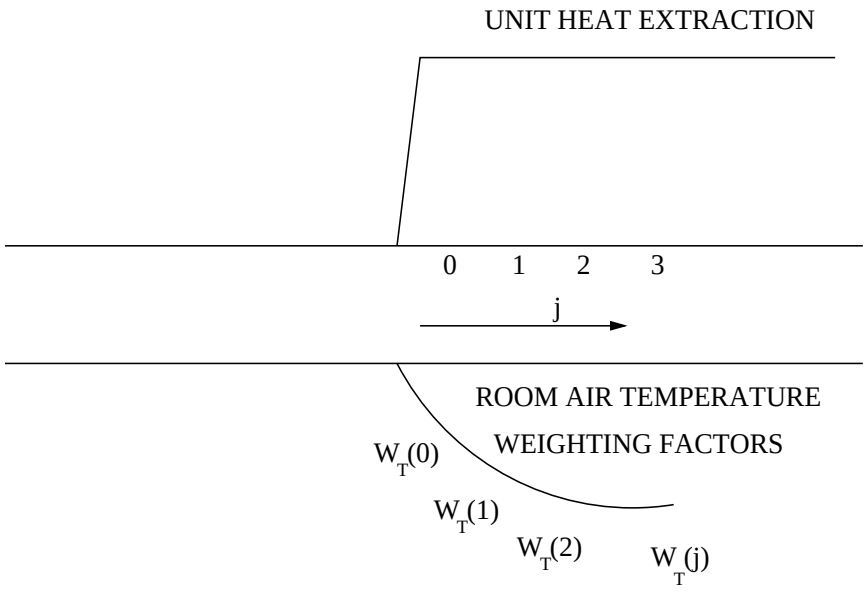

Figure 2.3. Room Air Temperature Weighting Factors.

2.5.2 Smart Temperature Control System

We have developed a distributed temperature control system in various locations of the house, for energy conservation. The temperature control system is intelligent enough to bring the temperature of specific locations (inside the home) into a comfortable one before the inhabitant enters those locations. The operation of temperature control is termed as *pre-conditioning*. The time needed for this pre-conditioning is

pre-conditioning period and the *rate of energy* required during this period is known as *pre-conditioning load*. When the inhabitant is about to leave a particular location, say l_1, the predictive location management system estimates its most probable set of routes and near future location (say l_2). The pre-conditioning period (W_T) is obtained by estimating the time taken by the inhabitant to move from l_1 to l_2, i.e., $W_T = t_{l_1} - t_{l_2}$. During this period, the constant rate of energy at full capacity is supplied to bring down the temperature to the comfort level. The shorter the duration of pre-conditioning period W_T, the larger is the pre-conditioning load. In order to estimate this load, it is required to know the characteristics of air temperature variation caused by constant unit rate of heat extraction from the specific locations. As depicted in Figure 2.3, W_T is often termed as *room air temperature weighting factors for unit heat extraction* [64]. Modern air-conditioning systems usually express this in *time series*, which might be defined as temperature weighting factors for unit heat extractions. If $\tilde{H}(t)$ and $\varphi(t)$ respectively denote heat extraction and air temperature deviation at time t, then the relation is:

$$\tilde{H}(t) = \sum_{j=0}^{\infty} W_z(j)\varphi(t-j), \tag{2.13}$$

where $W_z(j)$ is known as the weighting factor for heat extraction in the indoor environment. In our smart home, we have considered three major components of $W_z(j)$ responsible for heat exchange, namely walls, glass-windows and furniture. Thus, we have,

$$\begin{aligned} W_z(j) &= W_{zw}(j) + W_{zg}(j) + W_{zf}(j) \\ &= -\sum_{i=1}^{k_w} Z_w(i,j)\Lambda_w(i) \\ &\quad -\sum_{i=1}^{k_g} Z_g(i,j)\Lambda_g(i) - C_f\tilde{U}, \end{aligned} \tag{2.14}$$

where $Z_w(i,j)$, $Z_g(i,j)$ are respectively Z-response factors [64] for i-th wall and glass window, $\Lambda_w(i)$, $\Lambda_g(i)$ are the respective areas of i-th wall and glass window, C_f is the heat capacity of the furniture and $\tilde{U}$ is the volume of room space. Using the values of $\tilde{H}(t)$ and $\varphi(t)$, $\forall t = 0$ to ∞, we can derive a series of equations from Equation (2.13):

$$\begin{aligned} W_z(0)\varphi(0) &= 1 \\ W_z(0)\varphi(1) + W_z(1)\varphi(0) &= 1 \\ W_z(0)\varphi(2) + W_z(1)\varphi(1) + W_z(2)\varphi(0) &= 1 \\ \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots & \end{aligned} \tag{2.15}$$

The solutions for $\varphi(j)$ for all j can now be obtained successively from the above set of equations. The temperature deviation without heat extraction until the occupancy of the inhabitants in that particular location is calculated first. Let the total temperature deviation during start of occupancy be represented by $\Delta(\varphi)$. Let $\tilde{H}_{he}(t)$ denotes the rate of heat extraction during t hours of pre-conditioning. Then

$$\tilde{H}_{he}(t) = \frac{\Delta(\varphi)}{\varphi(t)} \tag{2.16}$$

In the cooling mode, once the air conditioning is stopped (inhabitant's departure from specific location of the house), the temperature of that region increases rapidly. The same mechanism is repeated whenever the inhabitant is about to move into the specific locations inside the house. The pre-conditioning period is followed by the conditioned period, when the room temperature is kept constant at a reference level.

2.5.3 Estimation of Inhabitants' Comfort

While the goal behind the deployment of smart homes lies in providing the inhabitants with sufficient comfort, this comfort is actually a subjective measure ex-

perienced by the inhabitants themselves. Thus, it is quite difficult to objectively estimate their comfort in smart homes. In-building climate, specifically temperature, plays an important role in defining this comfort. Moreover, the amount of manual operations and the time spent by the inhabitants in performing the house-hold activities also have significant influence on the inhabitants' comfort. We define the comfort as a joint function of temperature, manual operations and time spent by the inhabitants. Obviously, increase in the temperature-deviation, the number of manual operations and the amount of time spent reduces the overall comfort experienced by the inhabitant. If $\Delta(\varphi)$, $\mathcal{M}$ and τ represent the temperature deviation, number of manual operations and time an inhabitant spent in house-hold activities, then the associated comfort for that inhabitant is represented by the following equation:

$$Comfort = f\left(\frac{1}{\Delta(\varphi)}, \frac{1}{\mathcal{M}}, \frac{1}{\tau}\right) \tag{2.17}$$

It should be noted that the reduction of joint entropy by using the co-operative learning algorithm, described in Figure 2.1, endows the house with sufficient knowledge of the inhabitants' contexts. This helps in accurate estimate of current and future contexts (location, routes and activities) of the multiple inhabitants present in the house. Successful estimate of these contexts results in adaptive control of environmental conditions and automated operation of devices. This is necessary to reduce the empirical values of $\Delta(\varphi)$, $\mathcal{M}$ and τ, thereby increasing the overall comfort.

2.6 Simulation Experiments

In this section, we study the performance of our mobility-aware resource optimization framework for multi-inhabitant smart home. After describing our simulation environment and assumptions, we present the performance results.

We have developed an object-oriented discrete-event simulation for support-

ing inhabitants' movements, estimation of their locations, and comfort management scheme. The data used for simulation is obtained from the $X10$ controller Active-Home kit [117] deployed in the appliances in the MAVHome [27]. The time spent by the inhabitant in different locations is obtained from the motion-sensors placed along the walls. The different events are inhabitants' actions (behaviors), which result in the probabilistic movement of one or more inhabitants from one station to another depending on their lifestyle. An event queue is used for holding and scheduling these dynamic events. During the inhabitants' probabilistic movements across the house from one location to the another, the set of sensor-ids are collected. The inhabitants are also assumed to follow a different lifestyle in the weekends and holidays, with more household activities than during the weekdays.

Before presenting the details of the experimental results, let us enumerate a set of common assumptions used in our simulation: (i) The co-operative, game-theoretic framework for uncertainty minimization is performed in the smart home with an average number of 5 regular inhabitants and 3 visitors. (ii) The time spent at each destination is assumed to be uniformly distributed between the maximum and minimum stay at that particular destination. This maximum stay is different for regular inhabitants and visitors. (iii) The delay between sensory data-acquisition, processing and triggering the actuators is assumed to be negligible. (iv) The decision-making associated with the resource and comfort management is performed as the inhabitants leave every location for their next station. (v) The entire set of results is presented by sampling every sensor at a time and observing the simulation environment for a period of 10 weeks.

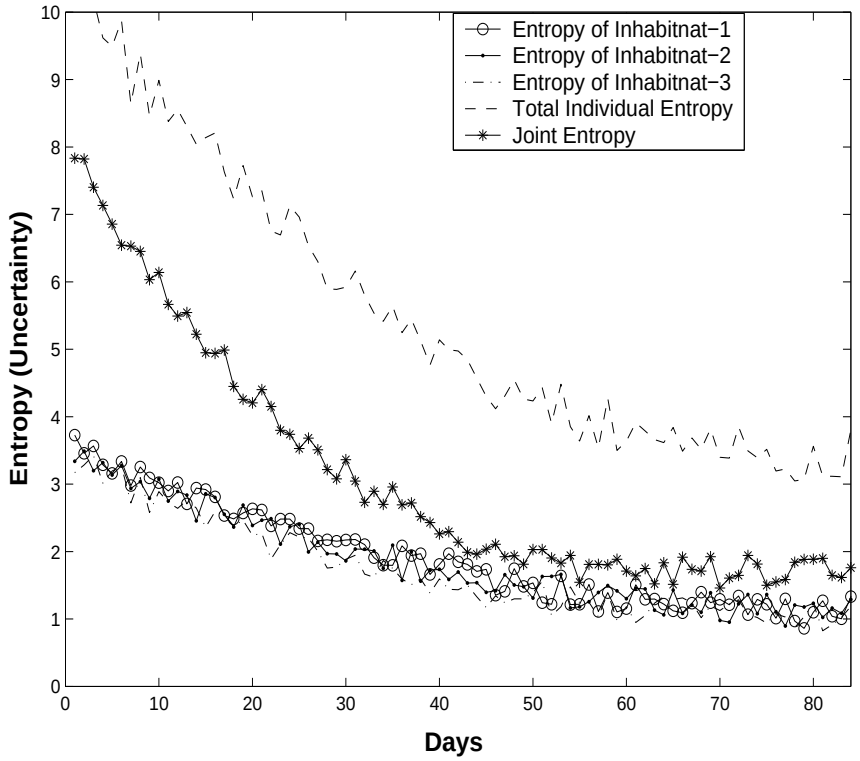

Figure 2.4. Variation of Entropy (Uncertainty).

2.6.1 Performance Results

The main objective of the co-operative, learning framework is to reduce the location uncertainty (entropy) associated with individual and the entire set of inhabitants. Figure 2.4 shows the variation of the individual and joint entropy over the entire time period of the simulation. It should be noted that the game-theoretic learning framework reduces the joint entropy quickly to a low value. While the entropy of every inhabitant lies in the range $\sim$ 1–3, the visitor's entropy is typically higher ~ 4. This is quite logical as the house finds the location contexts of the visitors more uncertain than the residents (inhabitants). The joint entropy of all inhabitants and visitors is even reduced to a further lower value (≤ 1). This entropy minimization procedure formulated by co-operative learning helps increase the efficiency of the lo-

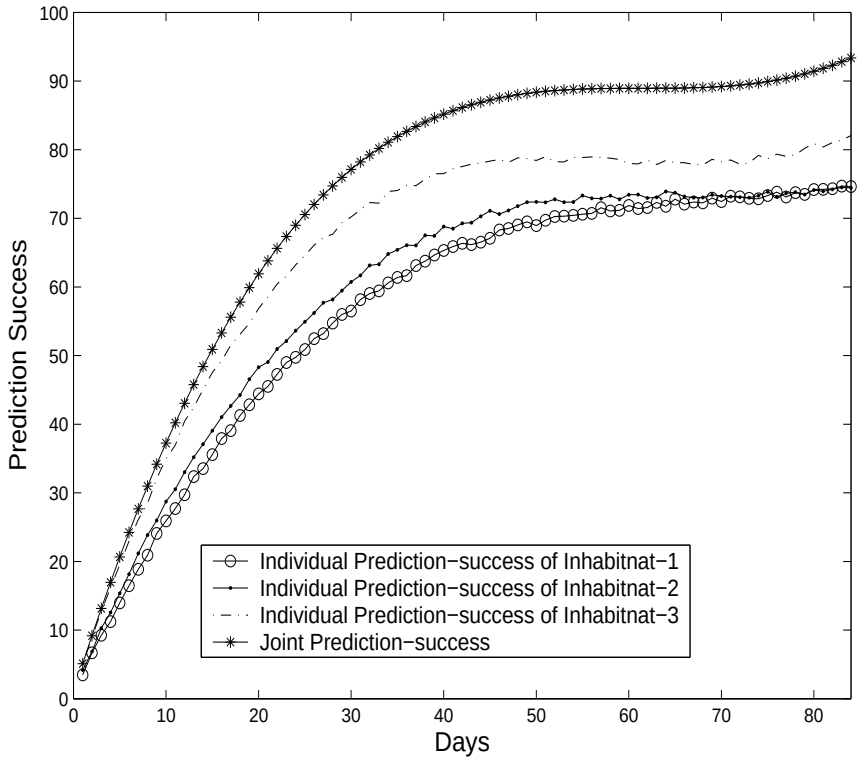

Figure 2.5. Accuracy in Location Estimation.

cation estimation technique. Figure 2.5 demonstrates that our proposed co-operative learning strategy is capable of estimating the location of all the inhabitants with almost $\simeq 90\%$ accuracy within 3 weeks span. The house takes this time to learn the joint movement patterns of all inhabitants. The success rate of location estimation for visitors is however 50%–60%, as the house finds it difficult to get the knowledge of the random visitors.

Efficient location estimation is a key factor to meet the minimum energy consumption in the house. While moving from a particular zone to another, correct estimation of location and routes helps in triggering the actuators *only* along those predicted (estimated) locations and routes, thereby attempting to minimize the energy consumption. In Figure 2.6 we have compared this amount of energy consump-

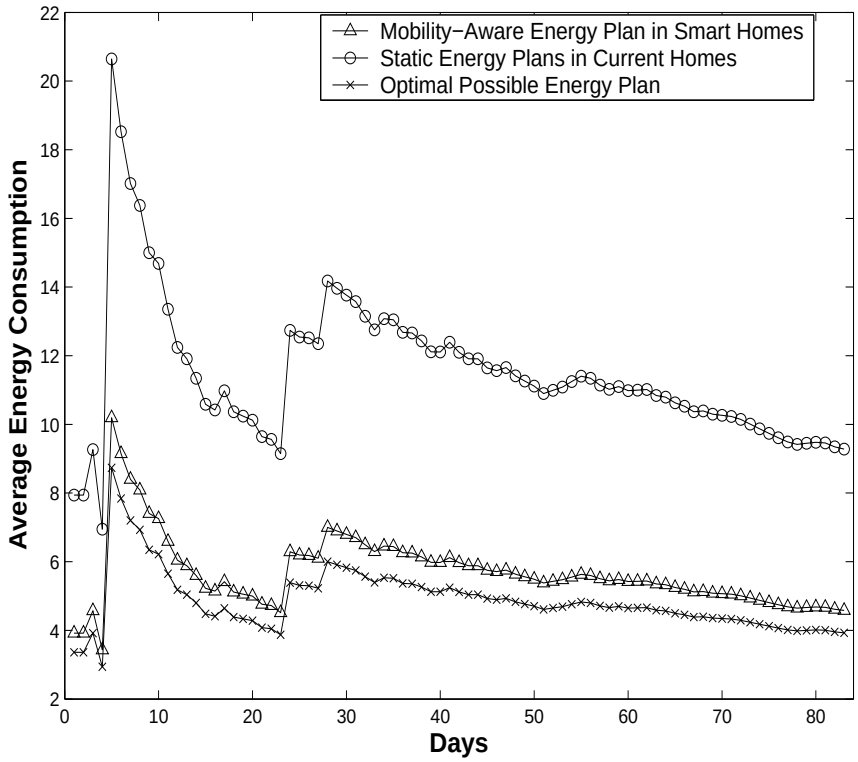

Figure 2.6. Average Energy Consumption.

tion scheme resulting from our mobility-aware resource management framework with the static energy plans and observed that our scheme is capable of saving almost 50% of the energy consumption in comparison with today's houses, using static energy model. We believe that this scheme has the power to reduce the maintenance cost by conserving sufficient amount of energy. As discussed earlier, the comfort of an individual inhabitant is a subjective quality and is rather difficult to quantify. While there exists no appropriate model for analyzing this comfort, we rely on measuring the individual parameters responsible for this comfort. Figure 2.7 points out that successful estimation of inhabitants' routes and locations reduces the manual operations performed by the inhabitants and the time required for performing those operations. The scheme results in almost $\sim$ 12%–17% manual operations and time spent by

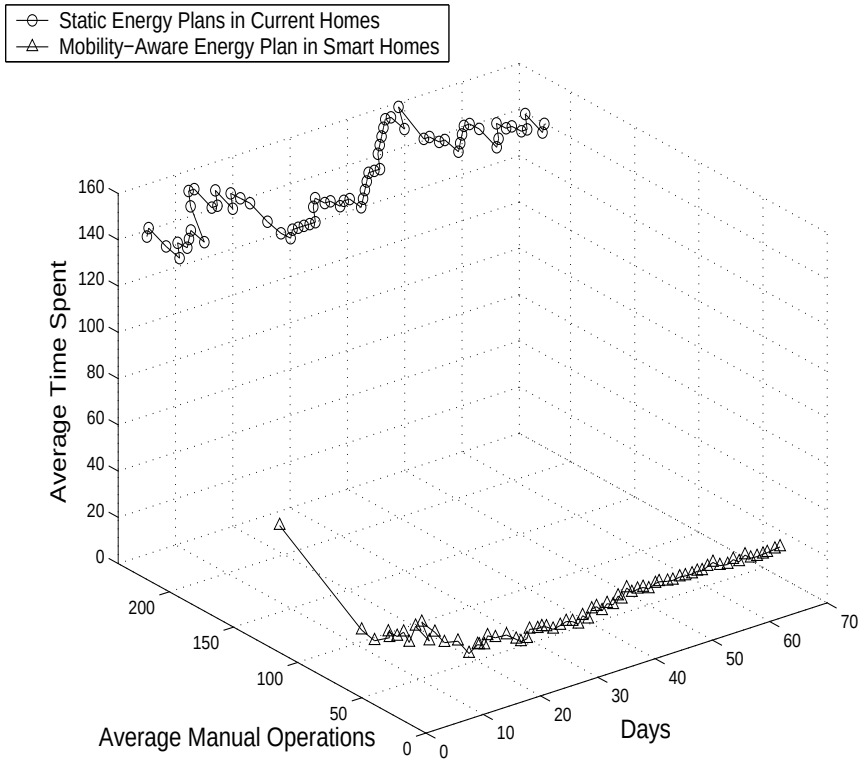

Figure 2.7. Dynamics of Manual Operations and Time Spent.

the inhabitants in comparison to the current houses using static energy management scheme.

2.7 Summary

In this chapter, we have developed a novel mobility-aware resource management framework for a multi-inhabitant smart home [96]. Characterizing the mobility of inhabitants as a stationary, ergodic, stochastic process, our framework uses the information theoretic measure to estimate the uncertainty associated with all the inhabitants present in the house. A co-operative learning paradigm based on dynamic game theory is formulated, which learns and estimates the inhabitants' location (route) profiles by minimizing the overall entropy (uncertainty) associated with

it. Automated activation of devices and conservation of energy along these estimated locations and routes provide the inhabitants with necessary comfort at a near optimal cost. We believe that this is an integral step toward realization of smart pervasive computing paradigm. In the next chapter we will focus on multi-inhabitant joint location uncertainty problem from non-cooperative point of view using stochastic game theory.

CHAPTER 3

NON-COOPERATIVE CONTEXT-AWARE RESOURCE MANAGEMENT FRAMEWORK

3.1 Introduction

Advances in smart devices, mobile wireless communications, sensor networks, pervasive computing, machine learning, middleware and agent technologies, and human computer interfaces have made the dream of smart environments a reality. According to Cook and Das [22], a "smart environment" is one that is able to autonomously acquire and apply knowledge about its inhabitants and their surroundings, and adapt to the inhabitants' behavior or preferences with the ultimate goal to improve their experience in that environment. The type of experience that individuals expect from an environment varies with the individual and the type of environment considered. This may include the safety of inhabitants, reduction of cost of maintaining the environment, optimization of resources (e.g., energy bills or communication bandwidth), or task automation. An instance of such an indoor environment is a *smart home* (e.g., MavHome[1]) that perceives the surroundings through sensors and acts on it with the help of actuators.

An important characteristic of such an intelligent, ubiquitous computing and communication paradigm lies in the autonomous and pro-active interaction of smart devices used for tracking inhabitants' important contexts such as current and near-future locations as well as activities. *"Context awareness"* is indeed a key to build a smart environment and associated applications. For example, the embedded pressure sensors in the Aware Home [87] capture inhabitants' footfalls, and the system (i.e.,

[1]**M**anaging an **A**daptive **V**ersatile **Home** [27]

smart home) uses these data for position tracking and pedestrian recognition. The Neural Network House [82], the Intelligent Home [77], the Intelligent House_n [57] and the MavHome [27, 118] projects focus on the development of adaptive control of home environments by also anticipating the location, routes and activities of the inhabitants. The Active Badge [45] and Active Bat [46] takes the help of infra-red and ultrasonic time-of-flight techniques to provide indoor location tracking framework. On the other hand, MIT's Cricket Location Support System [92] delegates the responsibility of location reporting to the mobile object itself. RADAR [4], another RF-based indoor location support system uses signal strength and signal-to-noise ratio to compute 2-D positioning. Microsoft's Easy-living and Microsoft Home [72] projects use real-time 3D cameras to provide stereo-vision positioning capability in an indoor environment. Intelligent prediction of these contexts helps in efficient triggering of mobility-aware services.

Now, it is not difficult to understand that an inhabitant's mobility and activity create an uncertainty of their locations and hence subsequent activities. In order to be cognizant of his contexts, the smart home needs to minimize this uncertainty as captured by Shannon's entropy measure [26]. An analysis of his daily routine and life style reveals that there exist some well defined patterns of these contexts. Although these patterns may change over time, they do not change too frequently and thus can be learned. This simple observation leads us to assume that the inhabitant's mobility or activity is a *piece-wise stationary, ergodic, stochastic process* with an associated uncertainty (entropy), as originally hypothesized by Bhattacharya and Das [7] for personal mobility tracking in wide area wireless cellular networks.

In an earlier work [94], we designed an optimal algorithm for location (activity) tracking in an indoor smart environment, based on dictionary management and online learning of the inhabitant's mobility profile, followed by a predictive location-aware

resource management (energy consumption) scheme for a single inhabitant smart home. However, the presence of multiple inhabitants with dynamically varying profiles as well as preferences make such tracking much more challenging. This is due mainly to the fact that the relevant contexts of multiple inhabitants in the same environment are often inherently *correlated* and *inter-dependent* on each other. Therefore, the learning and prediction (decision making) paradigm needs to consider the joint (simultaneous) location/activity tracking of multiple inhabitants. In another preliminary work [97], we proposed a cooperative entropy learning policy for location-aware resource management in multi-inhabitant smart homes. This approach adapts to the uncertainty of multiple inhabitants' locations and most likely routes, by varying the learning rate parameters and minimizing the Mahalanobish distance [95]. However, the complexity of multi-inhabitant location tracking problem was not characterized which we address in this Chapter [98].

Furthermore, hypothesizing that each inhabitant in a smart home behaves in such a way as to fulfill his own objectives and maximizes his utility, the residence of multiple inhabitants with varying preferences might lead to conflicting goals. Thus, a smart home must be intelligent enough to strike a balance between multiple preferences, eventually attaining an equilibrium state. If each inhabitant is aware of the situation facing all others, a *Nash equilibrium* is a combination of deterministic or randomized strategies, one for each inhabitant, from which no inhabitant has an incentive to unilaterally move away. This motivates us to investigate the multi-inhabitant location tracking problem from the perspective of stochastic game theory, where the inhabitants are the players of the game. The goal here is to achieve a Nash Equilibrium so that the system (i.e., smart home) is able to probabilistically predict the inhabitants' locations and activities with sufficient accuracy in spite of possible correlations.

The major contributions of this work are summarized below.

- We characterize the joint location uncertainty (entropy) of multiple inhabitants in a smart environment. In particular, we prove that optimal tracking and hence prediction of location across multiple inhabitants is an NP-hard problem where optimality is defined as attaining a lower bound on entropy.
- Based on the stochastic game theory and following the Nash Q-learning approach, we develop a novel Nash H-learning framework that exploits the correlation of mobility patterns across multiple inhabitants and attempts to minimize the joint uncertainty. This is achieved by developing a new *joint utility function* of entropy. We prove that our game theoretic framework attains Nash equilibrium. Minimizing the joint utility function helps in accurate learning and estimation of inhabitants' locations and activities. We also derive worst-case performance bounds of our proposed framework.
- Although there may exist an exponential number of possible routes (sequence of locations) that the inhabitants may follow in a smart indoor environment, we have developed an efficient scheme to predict the most likely routes jointly followed by multiple inhabitants. This scheme is based on the concepts of *joint-typical-set* of sequences and asymptotic equipartition property (AEP) in information theory, that provide only a small subset of sequences with a large probability mass.
- The knowledge of the inhabitants' contexts such as locations and associated activities, helps the smart home control automated devices in an intelligent manner, thus providing sufficient comfort to the inhabitants. The predictive Nash H-learning framework leads to an efficient mobility-aware resource management scheme that brings intelligence automation with reduced energy consumption and hence the overall maintenance cost of the smart home.

- We perform extensive experiments using a combination of simulation traces and real data collected from the X10 controller ActiveHome kit [117], deployed in the MavHome [119]. Experimental results demonstrate that the Nash H-learning framework performs better than predictive schemes optimized for only individual inhabitants' location/activity.

The rest of the chapter is organized as follows. Section 3.2 brings out the motivation and illustrates with the scenario of an indoor floor plan of a smart home. Section 3.3 reviews an existing information theoretic approach for optimal location tracking of individual inhabitants, and also discusses its limitation in optimally handling multiple inhabitants. In Section 3.4 we prove that the optimal (joint) location prediction problem across multiple inhabitants is NP-hard. The game theory based Nash H-learning framework that minimizes joint uncertainty associated with multiple inhabitants, is then presented in Section 3.5. Subsequently, we prove the convergence to Nash equilibrium and derive worst-case performance bounds. Section 3.6 describes how to capture the inhabitants' most likely routes and Section 3.7 develops a predictive, mobility-aware resource management scheme in multi-inhabitant smart homes. Experimental results in Section 3.8 delineates the efficiency of our proposed framework, and Section 3.9 concludes this chapter.

3.2 An Illustrative Example

Figure 3.1 gives the floor-plan of a typical smart home together with the placement of motion-sensors along the inhabitant's routes. A quick look into the floor-plan reveals that this smart home's coverage area can be partitioned into different *zones*. While moving from one zone to another, the inhabitant goes through an array of coverage areas of different sensors along the path. When the system needs to contact the inhabitant, it will initiate a prediction scheme to predict the inhabitant's current

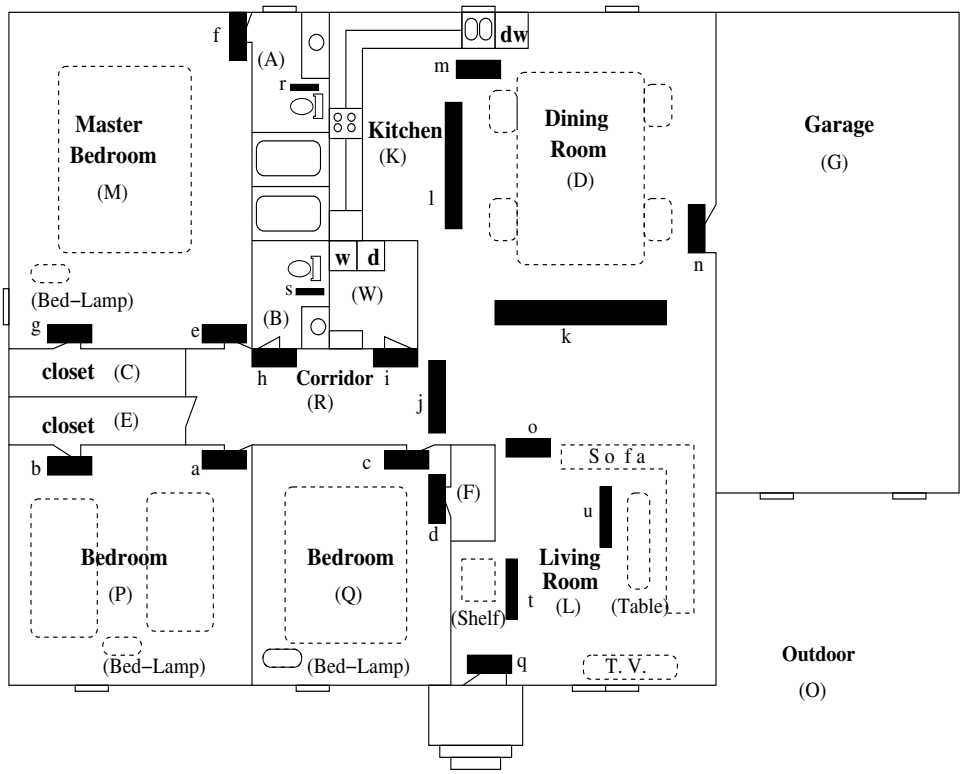

Figure 3.1. Example Floorplan of a Smart Home.

location together with his most likely paths. In order to control the location uncertainty of the inhabitant, the system also relies on location information provided by the in-building sensors from time to time. This helps in reducing the search space for the next prediction. As shown in Figure 3.2, the smart home network corresponding to Figure 3.1 can be represented by a connected graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$, where the node-set $\mathbf{V} = \{A, B, C, D, E, F, G, K, L, M, O, P, Q, R, W, \ldots\}$ represents the zones and the edge-set $\mathbf{E}$ represents the action/movement between a pair of zones.

As a motivating example for multi-inhabitant tracking, let us consider two inhabitants in our smart home indoor environment. We assume that inhabitant 1 starts from zone (node) C and attempts to reach to the destination zone (goal) G, while inhabitant 2 starts from G and wishes to reach to the destination A. Assume an

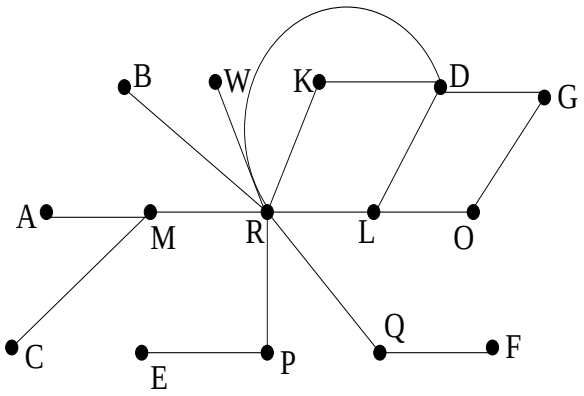

Figure 3.2. Graph Representing the Connectivity of Zones/Locations.

inhabitant can cross only one zone at a time, and all possible edges through which he can travel define the degree of the currently residing node. Reaching the goal earns a positive reward for the inhabitants. In case both inhabitants reach their goals at the same time, both are rewarded with positive payoffs. They do not know the locations of their goals at the beginning of learning period. Furthermore, the inhabitants choose their actions simultaneously. They can observe the previous actions of both inhabitants and the current state (joint location). They can also observe the immediate rewards after both inhabitants choose their actions.

The objective of an inhabitant in this case is therefore to reach its goal/destination zone with a minimum number of states yielding a minimum value of the cumulative entropy associated with the trace path. We will follow the above scenarios throughout this chapter to validate our proposed model.

3.3 Single Inhabitant Location Tracking

As mentioned earlier, an inhabitant's mobility creates an uncertainty of his location and thus activity. In order to minimize such uncertainty and adapt to fluctuations, one needs to build personal mobility profiles dynamically. From an in-

formation theoretic perspective, *entropy* [26] is an appropriate measure to quantify this uncertainty. In the context of personal mobility tracking in cellular wireless networks, Bhattacharya and Das [7] proved that it is impossible for any location tracking scheme to track down an inhabitant by exchanging any less information, on the average, than the uncertainty generated due to its mobility. A model-independent, predictive framework based on on-line compression and learning, was also proposed in [7] that minimizes location uncertainty and meets this information theoretic lower bound on entropy.

In smart indoor environment, the above framework was adopted in [94] to derive a location prediction scheme that is optimal only for single inhabitants. This framework is based on *symbolic* interpretation of the inhabitant's movement (mobility) history or profile, as captured by sampling the in-building smart devices such as sensors, RFID readers, or pressure switches. More precisely, the inhabitant's movement history is assumed to be a string $\nu_1\nu_2\nu_3\ldots$ of symbols (e.g., sensor-ids) where ν_i is an element of the alphabet set, ϑ. Given that our daily life has repetitive activity patterns, we argue that the inhabitant's current location is merely a reflection of his mobility/activity profile that can be learned over time in an on-line fashion. Characterizing the mobility as a probabilistic sequence of symbols suggests that it can be defined as a stochastic process $\mathcal{V} = \{V_i\}$. The repetitive nature of identifiable patterns (routes) adds *piece-wise stationarity* as an essential property, leading to $Pr[V_i = \nu_i] = Pr[V_{i+\ell} = \nu_i]$, for all $\nu_i \in \vartheta$ and for every shift ℓ. The family of optimal Lempel-Ziv text compression algorithms such as LZ-78 [123] is suitable for efficient encoding of these variable length routes or contexts (substrings of symbols from the mobility profile) such that the conditional entropy corresponding to the uncertainty due to the inhabitant's mobility is minimized. For details, refer to [7, 94].

Before proceeding further, let us formally define *entropy* and *conditional entropy* of random variables of a stochastic process from information theoretic stand point [26].

Definition 1 *For a discrete random variable X of a stochastic process, with probability mass function $p(x)$, its entropy is defined as $H(X) = -\sum_{x \in \mathcal{X}} p(x) \lg p(x)$. When $p(x) = 0$, the limiting value "$\lim_{p \to 0} p \lg p = 0$" is used.*

Definition 2 *For a set $\{V_1, V_2, \ldots, V_k\}$ of k discrete random variables with joint probability distribution $p(\nu_1, \ldots, \nu_k) = Pr[V_1 = \nu_1, \ldots, V_k = \nu_k]$, $\forall \nu_i \in \vartheta$, the joint entropy is given by $H(V_1, V_2, \ldots, V_k) = \sum_{i=1}^{k} H(V_i \mid V_1, V_2, \ldots, V_{i-1})$, where $H(V_i \mid V_1, V_2, \ldots, V_{i-1})$ is the conditional entropy of random variable V_i given the history of previous $(i-1)$ random variables $V_1, V_2, \ldots, V_{i-1}$.*

The additive terms on the right-hand side of the equation in the above definition carry necessary information which makes the higher order context models (explained in the next subsection) more information-rich as compared to the lower order ones.

The above location tracking strategy is optimal for individual inhabitants only. This is because it treats every inhabitant independently and fails to consider the correlation between the activity and hence mobility patterns of multiple inhabitants within the same home environment. Intuitively, independent application of the above scheme for each individual actually increases the joint location uncertainty. Mathematically, this can be observed from the fact that *conditioning reduces entropy* [26].

Result 1 *For a stochastic ergodic process $\mathcal{V}$ containing the set of random variables V_1, V_2, ..., V_k, with distribution $Pr(V_1 = \nu_1, V_2 = \nu_2, \ldots, V_k = \nu_k)$,*

$$H(\mathcal{V}) = H(V_1, V_2, \ldots, V_k) = \sum_{i=1}^{k} H(V_i | V_1, \ldots, V_{i-1}) \leq \sum_{i=1}^{k} H(V_i)$$

Therefore, the optimal location/activity tracking problem across multiple inhabitants needs to incorporate their correlation so as to minimize the joint uncertainty as measured by the entropy.

Before presenting the complexity of the location prediction problem across multiple inhabitants in the next section, let us illustrate the concept of different order contexts in the location profile represented in the symbolic domain and compute their entropy leading to models of different orders.

3.3.1 Contexts in Location Profile

Let us consider the movement history of a typical inhabitant within the smart home as shown in Fig 3.1. For simplicity, we only record the movement within the different zones of the MavHome network. This means that the inhabitant must be in one of the zones $\{A, B, C, D, E, F, G, K, L, M, O, P, Q, R, W \ldots\}$ at any point of time. Suppose the inhabitant wakes up at 7:00 am in the morning in a weekend day. We track down his movement profile until 7:00 pm in the smart home. Table 3.1 shows the time at which the zone has been changed and reported to the system. Consequently, all that the system captures in the location profile is a sequence of zone-id's.

Table 3.1. Inhabitant's Location Profile between 7:00 am and 7:00 pm

Time	9:05am	9:31am	9:45am	1:15pm	2:02pm
Changing zone	$M \rightarrow R$	$R \rightarrow M$	$M \rightarrow R$	$R \rightarrow M$	$M \rightarrow R$
Time	2:42pm	3:15pm	4:05pm	4:22pm	4:44pm
Changing zone	$R \rightarrow K$	$K \rightarrow D$	$D \rightarrow K$	$K \rightarrow R$	$R \rightarrow M$

From Table 3.1, let us now compare and contrast the sample sequences generated by the inhabitant's movement with respect to time threshold and his action

which is basically the transition from one zone to another. For the time based scheme, we have considered two values of T, i.e., 1 hr and 1/2 hr. Observe that a smaller value of T captures finer route granularities of the inhabitant as depicted in Table 3.2. Regarding the action based scheme, $\mathcal{A} = 1$ captures movement in the finest details due to its one state transition feature at a time compared to $\mathcal{A} = 2$ where we have considered the state after a two step transition. A combined approach of time and action based schemes makes the movement history more informative as it traces in detail the routes taken by the inhabitant. The last row of Table 3.2 shows the sequence which generates with an hourly basis starting from 7:00am in the morning, as well as when the zone has been changed.

Table 3.2. Zone Sequence Extracted from the Location Profile of the Inhabitant

Time dependent (T = 1hr)	$MMMRRRRMKDMMM\ldots$
Time dependent (T = 1/2 hr)	$MMMMMRRRRRRRRMMRKDDKMMMMM\ldots$
Action dependent ($\mathcal{A} = 1$)	$MRMRMRKDKRM\ldots$
Action dependent ($\mathcal{A} = 2$)	$MMMKKM\ldots$
Time and Action dependent (T = 1hr, $\mathcal{A} = 1$)	$MMMRMRRRRRMMRKKDDKRMMMM\ldots$

The iid (independent and identically distributed) model [7] takes the first step towards learning from movement history. Unfortunately, the iid model does not carry any information about the symbols order of appearance and falls short in such situations which we call order-0 Markov model in our context. The single step transition or order-1 Markov model carries a little more information about the ordering, at least to the extent of one symbol context.

In the *iid model*, where V_i's are independently and identically distributed, the relative frequencies of the symbols are listed in Table 3.3. Thus the inhabitant's

Table 3.3. Contexts of Orders 0, 1 and 2 with Occurrence Frequencies

Order-0	Order-1		Order-2		
$M(10)$	$M\|M(6)$	$R\|K(1)$	$M\|MM(3)$	$M\|RM(2)$	$M\|KR(1)$
$R(8)$	$R\|M(3)$	$K\|K(1)$	$R\|MM(2)$	$R\|RM(1)$	$D\|KK(1)$
$K(3)$	$M\|R(3)$	$D\|K(1)$	$M\|MR(1)$	$M\|RR(1)$	$D\|KD(1)$
$D(2)$	$R\|R(4)$	$K\|D(1)$	$R\|MR(1)$	$R\|RR(3)$	$R\|DK(1)$
	$K\|R(1)$	$D\|D(1)$	$K\|MR(1)$	$K\|RK(1)$	$K\|DD(1)$

residence probabilities are estimated as $\pi_M = 10/23, \pi_R = 8/23, \pi_K = 3/23, \pi_D = 2/23$, and $\pi_A = \pi_B = \pi_C = \pi_E = \pi_P = \pi_W = \pi_L = \pi_O = \pi_G = \pi_Q = \pi_F = 0$.

The corresponding entropy value is given by,

$$\begin{aligned} H(\mathcal{V}) &= -\sum_{\nu\in\mathcal{V}} p(\nu)\lg\, p(\nu) = -\sum_{\nu\in\mathcal{V}} \pi_\nu \lg\, \pi_\nu \\ &= \frac{10}{23}lg\frac{23}{10} + \frac{8}{23}lg\frac{23}{8} + \frac{3}{23}lg\frac{23}{3} + \frac{2}{23}lg\frac{23}{2} = 1.742bits \end{aligned} \tag{3.1}$$

Similarly, let us compute the entropy value for order-1 Markov model. From the Markov chain in Figure 3.3, the probability transition matrix is given by

$$P = \begin{bmatrix} 2/3 & 1/3 & 0 & 0 \\ 3/8 & 1/2 & 1/8 & 0 \\ 0 & 1/3 & 1/3 & 1/3 \\ 0 & 0 & 1/2 & 1/2 \end{bmatrix}$$

Let $\Pi = [\pi_M, \pi_R, \pi_K, \pi_D]^{\mathcal{T}}$ be the steady state probability vector. Solving for $\Pi = \Pi \times P$ with $\pi_M + \pi_R + \pi_K + \pi_D = 1$, we obtain $\pi_M = 9/22, \pi_R = 4/11, \pi_K = 3/22, \pi_D = 1/11$, and $\pi_A = \pi_B = \pi_C = \pi_E = \pi_P = \pi_W = \pi_L = \pi_O = \pi_G = \pi_Q = \pi_F = 0$. Therefore, the entropy is given by

$$\begin{aligned} H'(\mathcal{V}) &= -\sum_i \pi_i \left(\sum_j P_{i,j} \lg\, P_{i,j} \right) \\ &= \frac{9}{22}\left(\frac{2}{3}lg\frac{3}{2} + \frac{1}{3}lg3\right) + \frac{4}{11}\left(\frac{3}{8}lg\frac{8}{3} + \frac{1}{2}lg2 + \frac{1}{8}lg8\right) + \frac{3}{22}\left(3\times\frac{1}{3}lg3\right) + \frac{1}{11}\left(2\times\frac{1}{2}lg2\right) \\ &= 1.194bits \end{aligned} \tag{3.2}$$

Now, let us compute the conditional entropy

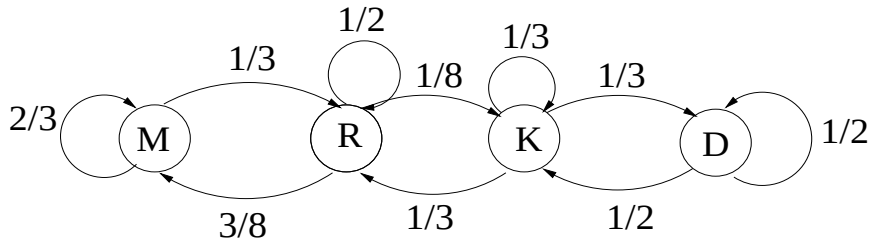

Figure 3.3. Order-1 Markov Model for Location Profile.

$$\begin{aligned} H(V_2|V_1) &= -\sum_i \pi_i \left(\sum_j P_{i,j} \lg P_{i,j} \right) \\ &= \frac{10}{23}\left(\frac{2}{3}lg\frac{3}{2} + \frac{1}{3}lg3\right) + \frac{8}{23}\left(\frac{3}{8}lg\frac{8}{3} + \frac{1}{2}lg2 + \frac{1}{8}lg8\right) + \frac{3}{23}\left(3 \times \frac{1}{3}lg3\right) + \frac{2}{23}\left(2 \times \frac{1}{2}lg2\right) \\ &= 1.182 bits \end{aligned} \tag{3.3}$$

Since the joint entropy is given by

$$H(\mathcal{V}) = H(V_1, V_2, \ldots, V_k) = \sum_{i=1}^{k} H(V_i|V_1, \ldots, V_{i-1})$$

we get $H(V_1, V_2) = H(V_1) + H(V_2|V_1) = 1.742 + 1.182 = 2.924$ and by taking the running average we arrive at an estimate of 1.462. Thus, we observe that the joint entropy value is less than the individual entropy value according to the Result 1.

3.4 Multi-Inhabitant Location Prediction

The multi-inhabitant location prediction problem is defined as follows: *For a group of η location predictions, one for each of η inhabitants residing in the smart home consisting of L different locations, the objective is to maximize the number of successful predictions.* The following theorem characterizes the complexity of this problem.

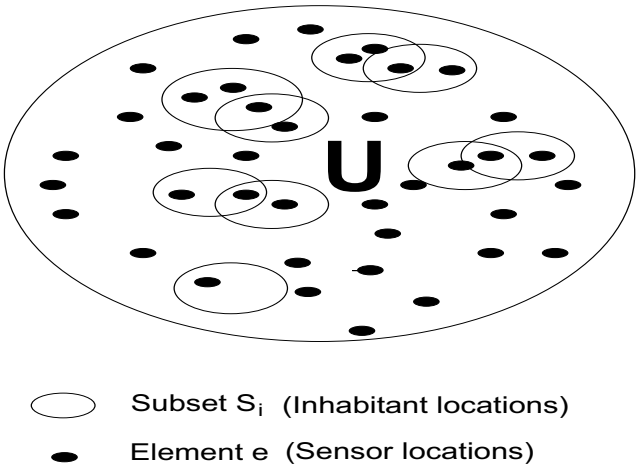

Figure 3.4. Analogy of Set-Packing Problem.

Theorem 1 *The problem of maximizing the number of successful predictions of multiple inhabitants' locations in a smart home is NP-hard.*

Proof: We reduce this problem to the *Set Packing* problem, which is known to be NP-hard [37]. The Set Packing problem arises in partitioning elements under strong constraints on what is allowable partitions. The key feature is that no element is permitted to be covered by more than one set. As shown in Figure 3.4, the input to the Set Packing problem is a set $\mathbf{S} = \{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_\xi\}$ of ξ subsets of the universal set $\mathbf{U} = \{1, 2, \ldots, \eta\}$, where η is the number of prediction requests as defined above. The goal is to maximize the number of mutually disjoint subsets from $\mathbf{S}$. In other words, given the condition that each element from the universal set $\mathbf{U}$ can be covered by *at most one subset* from $\mathbf{S}$, the objective is to maximize the number of mutually disjoint subsets from $\mathbf{S}$. In order to prove the theorem, we assume that each location as identified by the sensor is occupied by at most one inhabitant. The sensor deployment and coverage in a smart home is assumed to be dense enough to make this distinction.

The maximum successful prediction process in a smart home having L locations and η prediction requests, is equivalent to the Set Packing problem with η subsets

and a universal set $\mathbf{U}$ of L elements. At any instance of time, an inhabitant i can actually reside under the coverage of one or more sensors (locations), say l_i. Then the prediction process, $predict_i$, for inhabitant i is a collection of its possible locations, i.e., $predict_i = \{l_i\}$. Every such prediction is mapped to a particular subset $\mathcal{S}_i$. Each single location (sensor coverage-area) of the smart home is mapped to an element of the subset $\mathcal{S}_i$. The strategy that maximizes the number of successful predictions is basically the one that maximizes the number of disjoint subsets from $\mathbf{S}$. Thus, we conclude that the multi-inhabitant optimal location prediction is NP-hard. •

Therefore, it is computationally infeasible to find an optimal strategy for maximizing the number of successful location predictions across multiple inhabitants. In the following, we devise a suboptimal solution based on game theory. It attempts to reach an equilibrium and maximizes the number of successful predictions across all inhabitants.

3.5 Predictive Nash H-learning Framework

Hypothesizing that every inhabitant wants to satisfy his own preferences about activities, we assume he behaves like a selfish agent to fulfill his own goals. Under this circumstance, the objective of the system is to achieve a suitable balance among the preferences of all inhabitants residing in the smart home. This motivates us to look into the problem from the perspective of non-cooperative game theory where the inhabitants are the players and their activities are the strategies of the game. Moreover, there can be conflicts among the activity preferences. Our proposed game theoretic framework aims at resolving these conflicts among inhabitants, while predicting their activities (and hence locations) with as much accuracy as possible. Before going into

the details of our framework, let us briefly review the relevant concepts of game theory required for our purpose.

3.5.1 Stochastic Games and Equilibrium

Stochastic games model multi-agent systems where the agents are the house and the inhabitants, pursuing their individual (often conflicting) goals. We assume there exists no enforceable agreement on the joint actions of the inhabitants.

Definition 3 *[58] An n-player stochastic game,* Γ*, is defined as a tuple* $< S, \mathcal{A}^1, \ldots, \mathcal{A}^n, r^1, \ldots, r^n, p >$*, where* S *is the state space and* $\mathcal{A}^i$ *is the action space of player* i*;* $r_t^i : S \times \mathcal{A}^1 \times \mathcal{A}^2 \ldots \times \mathcal{A}^n \rightarrow \mathcal{R}$ *is the payoff or reward function for player* i *at instant* t*;* $p : S \times \mathcal{A}^1 \times \mathcal{A}^2 \ldots \times \mathcal{A}^n \rightarrow \Delta(S)$ *is the transition probability map, where* $\Delta(S)$ *is the set of probability distributions over the state space* S*.*

Given a state s, the inhabitant agents independently perform their actions $a^1, \ldots, a^n$, for $a^i \in \mathcal{A}^i$, and receive rewards $r_t^i(s, a^1, \ldots, a^n)$, for $i = 1, \ldots, n$. The state s changes to the next state s' based on transition probabilities, satisfying the constraint

$$\sum_{s,s' \in S} p(s'|a^1, \ldots, a^n) = 1$$

In a stochastic game, the objective of each player is to maximize the sum of rewards, with factor $\beta \in [0, 1)$. If π^i denotes the strategy of player i for choosing the optimal state action pair , then for a given initial state s, the objective of player i is to maximize the sum of rewards:

$$\Re^i(s, \pi^1, \pi^2, \ldots, \pi^n) = \sum_{t=0}^{\infty} \beta^t E(r_t^1 | \pi^1, \ldots, \pi^n, s_0 = s) \tag{3.4}$$

where $E(.)$ is the expected value.

Definition 4 *[58] A Nash equilibrium is a joint strategy where each agent is a best response to the others. For a stochastic game, each agent strategy is defined over the entire time horizon of the game. Hence, in a stochastic game* Γ*, a Nash equilibrium point is a tuple of* n *strategies* $(\pi_*^1, \pi_*^2, \ldots, \pi_*^n)$ *such that for all* $s \in S$*,* $i = 1, \ldots, n$ *and* $\forall \pi^i \in \Pi^i$*,*

$$\Re^i(s, \pi_*^1, \ldots, \pi_*^i, \ldots \pi_*^n) \geq \Re^i(s, \pi_*^1, \ldots, \pi_*^{i-1}, \pi^i, \pi_*^{i+1}, \ldots \pi_*^n) \tag{3.5}$$

where Π^i *is the set of all strategies available to agent* i*.*

A fundamental result related to equilibria in stochastic games states that every n-player stochastic game possesses at least one Nash equilibrium point in stationary strategies [88]. Let us now develop a suitable multi-agent learning framework that maximizes the number of successful location predictions in smart homes.

3.5.1.1 Representation of Stochastic Games

Considering the previous example, the individual action space of inhabitant i is given by $a^i = \{\text{all possible edges from the current residing node}\}$ for $i = 1, 2$. The individual state space is $s^i = \{A, B, C, D, E, F, G, K, L, M, O, P, Q, R, W\}$ for $i = 1, 2$. The joint state space is given by $S = \{(A, B), (A, C), \ldots, (W, R)\}$ where a state $s' = (s^1 \times s^2)$ represents the inhabitants' joint location.

Instead of calculating the entropy at each and every step, we have considered three different values of entropy to generate Nash H values from our proposed algorithm. If inhabitant 1 and inhabitant 2 respectively reach their goal/destination zones, then they achieve the minimum entropy value (assume 0.01 instead of 0 for our calculation purpose). Here we define the reward (r_t^i) function as inversely proportional to this entropy value. Thus in this case , $r_t^i = 100$.

If they would come into the same state, we consider the entropy as 1.0 and added a penalty factor with the reward function for accelerating the convergence towards the goal. Therefore, $r_t^i = -1$.

If they appear in any other distinct zones than the destination, then we assume entropy achieves a higher value and reward (r_t^i) becomes 0.

So, if an inhabitant reaches the goal state, it receives a reward of 100. If it reaches another state without colliding with the other inhabitant, its reward is zero. If it collides with the other inhabitant, it receives -1 and both inhabitants are bounced back to their previous states. Let $s' = \ell(s, a)$ be the potential new state resulting from choosing action a in state s. The reward function is, for $i = 1, 2$, is defined as

$$r_t^i = \begin{cases} 100, & \text{if } \ell(s_t^i, a_t^i) = Goal_i \\ -1, & \text{if } \ell(s_t^1, a_t^1) = \ell(s_t^2, a_t^2) \text{ and } \ell(s_t^2, a_t^2) \neq Goal_j, j = 1, 2 \\ 0, & \text{otherwise} \end{cases} \tag{3.6}$$

3.5.2 Entropy (or H) Learning

The concept for general-sum games builds from the Nash equilibrium [83], in which each player effectively holds a correct expectation (generally expressed in terms of payoff, reward or utility value) about the other players behaviors, and acts rationally with respect to this expectation. Acting rationally means the agent follows the strategy which corresponds to a best response to the others' strategies. Any deviation would make that agent worse off from achieving that equilibrium point. In extending the Q-learning [58] to our multi-inhabitant smart home context aware resource management problem we adopt the basic framework of general sum stochastic games. In single-agent systems, the concept of optimal Q-value can be defined in terms of an agent maximizing its own expected payoffs with respect to a stochastic environment. In multiagent systems, Q-values are contingent on other agents strate-

gies. In the framework of general-sum stochastic games, the optimal Q-values are the subset of the Q-values received in a Nash equilibrium, and referred as Nash Q-values. The goal of learning is to find Nash Q-values through repeated game. Based on learned Q-values, the agent can then derive the Nash equilibrium and choose its actions accordingly. In Nash Q-learning [58] algorithm, the agent attempts to learn its equilibrium Q-values, starting from an arbitrary guess. Thus here the Nash Q-learning agent maintains a model of other agents Q-values and uses that information to update its own Q-values based on the payoff value and takes their equilibrium actions in each state.

Our proposed Nash H-learning algorithm in this section enhanced the Nash Q-learning algorithm in that it captures the location uncertainty in terms of entropy at each and every step of the inhabitants' path. Thus, in our case, Nash H-value is determined which satisfies both Nash condition as well as our imposed entropy minimization constraint.

We assume that the inhabitants are fully rational in the sense that they can fully use their location histories to construct future routes. Each inhabitant i keeps a count $\mathcal{C}_a^j$ representing the number of times an inhabitant j has followed an action $a \in \mathcal{A}$. When the game is encountered, inhabitant i believes the relative frequencies of each of j's movements as indicative of j's current route. So for each inhabitant j, the inhabitant i believes j plays action $a \in \mathcal{A}$ with probability:

$$\mathcal{P}(a)^i = \frac{\mathcal{C}_a^j}{\sum_{b \in \mathcal{A}} \mathcal{C}_b^j} \tag{3.7}$$

This set of route strategies forms a reduced profile of strategies for which inhabitant i adopts a best response. After the game, inhabitant i updates its possible belief of its neighbor appropriately, given the actions used by other inhabitants. We consider these counts as reflecting the observations an inhabitant has regarding the

route strategy of the other inhabitants. As a result, the decision making component should not directly repeat the actions of the inhabitants but rather learn to perform actions that optimize a given reward (or utility) function.

Indeed, the decision making component of a smart home applies learning to acquire a policy that optimizes joint uncertainty of the inhabitants' activities which in turn helps in accurate prediction of their activities and thus locations. For this optimization, our proposed entropy learning algorithm, called *Nash H-learning* (NHL), learns a value function that maps the state-action pairs to future reward using the entropy measure, H. It combines new experience with old value functions to produce new and statistically improved value functions. The proposed multi-agent Nash H-learning algorithm updates with future Nash equilibrium payoffs.

```
Procedure NHL
Input:  Individual entropy values
Output:  Joint entropy values
1.  Let the learning agent be indexed by i;
2.  t := 0, H_t^j(s, a^1, ..., a^n) := 0, ∀s ∈ S and a^j ∈ A^j, j = 1, ..., n;
3.  Repeat
4.     Choose action a_t^i;
5.     Compute r_t^1, ..., r_t^n, a_t^1, ..., a_t^n and s_{t+1} = s';
6.     for (j = 1, ..., n),
7.          H_{t+1}^j(s, a^1, ..., a^n) = (1 − α_t) H_t^j(s, a^1, ..., a^n) + α_t [r_t^j + β Nash H_t^j(s')],
            where α_t ∈ (0, 1) is the learning rate
            and Nash H_t^j(s') = ∏_{k=1}^n π^k(s') H_t^j(s')
8.     t := t + 1;
9.  until (true)
```

Figure 3.5. Nash H Learning Algorithm (NHL).

Figure 4.4 describes the pseudo-code of the Nash H-learning algorithm which has been explained next with a reference to each line number of the algorithm. (1.&2.)

A learning agent, indexed by i, learns about its H-values by forming an arbitrary guess at time 0. We have assumed this initial value to be zero, i.e., $H_0^i(s, a^1, \ldots, a^n) = 0$. (4.) At each time t, the agent i observes the current state and takes its action. (5.) After that, it observes its own reward, actions taken by all other agents and their rewards, and the new state s'. (7.) It then calculates a *Nash Equilibrium* $\pi^1(s'), \pi^2(s'), \ldots, \pi^n(s')$ at that stage and updates its own H-values as follows.

$$\begin{aligned} H_{t+1}^i(s, a^1, \ldots, a^n) &= (1 - \alpha_t) H_t^i(s, a^1, \ldots, a^n) + \alpha_t \left[r_t^i + \beta Nash\, H_t^i(s') \right], \\ \text{where } \; Nash\, H_t^i(s') &= \prod_{j=1}^{n} \pi^j(s') H_t^i(s') \end{aligned} \tag{3.8}$$

where the learning rate parameters α_t and β are in the range 0 to 1. For every agent, information about other agents' H-values is not given, so agent i must learn about those values too. Agent i forms conjectures about those H-functions at the beginning of the game. We have assumed $H_0^j(s, a^1, \ldots, a^n) = 0$, for all j and all $s, a^1, \ldots, a^n$. As the game proceeds, agent i observes other agents' immediate rewards and previous actions. That information can then be used to update agent i's conjectures on other agents' H-functions. Agent i updates its beliefs about agent j's H-function, i.e., $H_{t+1}^j(s, a^1, \ldots, a^n)$ according to the same updating rule it applies to its own. Thus, we have

$$H_{t+1}^j(s, a^1, \ldots, a^n) = (1 - \alpha_t) H_t^j(s, a^1, \ldots, a^n) + \alpha_t \left[r_t^j + \beta Nash\, H_t^j(s') \right] \tag{3.9}$$

3.5.3 Convergence of NHL Algorithm

The convergence proof of the proposed Nash H-learning algorithm is based on two basic assumptions:

1. Every state $s \in S$ and every action $a^k \in \mathcal{A}^k$ for $k = 1, \ldots, n$ are visited infinitely often.

2. The learning rate α_t satisfies the following condition: $0 \leq \alpha_t(s, a^1, \ldots, a^n) < 1$, and $\alpha_t(s, a^1, \ldots, a^n) = 0$ if $(s, a^1, \ldots, a^n) \neq (s_t, a_t^1, \ldots, a_t^n)$. In other words, the updates occur only on H-function elements which correspond to current state s_t and actions $a_t^1, \ldots, a_t^n$.

Our proof relies on the following result, which establishes the convergence of a general functional-learning process updated by a pseudo-contraction operator. Let $\mathcal{U}$ be the space of all utility functions.

Result 2 *[58]: Let there exists a number γ such that $0 < \gamma < 1$ and a sequence $\lambda_t \geq 0$ converging to zero with probability 1 such that $|P_t U - P_t U_*| \leq \gamma |U - U_*| + \lambda_t$ for all $U \in \mathcal{U}$ and $U_* E[P_t U_*]$. Then the following condition holds:*

$$Pr\left[(U_{t+1} = (1 - \alpha_t)U_t + \alpha_t[P_t U_t]) \rightarrow U_*\right] = 1 \tag{3.10}$$

where P_t is a pseudo-contraction operator. In other words, we can say that the iterative utility function U_t converges to the Nash Equilibrium U_* with probability 1.

Replacing the general utility function U by the entropy or H function corresponding to H-learning, we get

$$Pr\left[(H_{t+1} = (1 - \alpha_t)H_t + \alpha_t[P_t H_t]) \rightarrow H_*\right] = 1.$$

For our n-player stochastic game we define the operator P_t as:

$$P_t H^k(s, a^1, \ldots, a^n) = r_t^k(s, a^1, \ldots, a^n) + \beta \pi^1(s') \ldots \pi^n(s') H^k(s'), \text{ for } k = 1, \ldots, n \tag{3.11}$$

where s' is the state at time $t+1$ and $\pi^k(s')$ is an equilibrium strategy at that stage of the game corresponding to the utility function $H^k(s')$. We now state the main result along with its proof:

Result 3 *For n-player stochastic game in smart homes, $E[P_t H_*] = H_* = (H_*^1, \ldots, H_*^n)$*

Proof: If $v^k(s', \pi_*^1, \ldots, \pi_*^n)$ is agent k's equilibrium payoff and $(\pi_*^1(s), \ldots, \pi_*^n(s))$ is its Nash Equilibrium point, then $v^k(s', \pi_*^1, \ldots, \pi_*^n) = \pi_*^1(s), \ldots, \pi_*^n(s) H_*^k(s')$ according to [58]. Based on this relation, we can state that

$$\begin{aligned} & H_*^k(s', a^1, \ldots, a^n) = r_t^k(s, a^1, \ldots, a^n) + \beta \sum_{s' \in S} p(s'|s, a^1, \ldots, a^n) \pi_*^1(s') \ldots, \pi_*^n(s') H_*^k(s') \\ = & \sum_{s' \in S} p(s'|s, a^1, \ldots, a^n) \times \left[r_t^k(s, a^1, \ldots, a^n) + \beta p(s'|s, a^1, \ldots, a^n) \pi_*^1(s') \ldots, \pi_*^n(s') H_*^k(s') \right] \\ = & E\left[P_t^k H_*^k(s, a^1, \ldots, a^n) \right] \end{aligned} \tag{3.12}$$

Combining Equations (3.10)–(3.12), we arrive at the following conclusion:

Result 4 *The predictive H-learning framework described by the iterative Equation (3.9) almost surely converges to the Nash Equilibrium. That is,*

$$Pr\left[H_{t+1} \rightarrow H_* \right] \rightarrow 1, \textit{where } \; H_{t+1} = (1 - \alpha_t) H_t + \alpha_t \left[r_t^k + \beta \prod_{j=1}^{n} \pi^j(s') H_t(s') \right] \tag{3.13}$$

We have proved the convergence of Nash H-learning under the assumption of some technical conditions expressed in equations (3.10) (3.11) (3.12) (3.13). If there is a unique equilibrium H-function then learning consistently converges, but sometimes it fails to converge if it has different equilibrium H-functions. Specifically the learning process converges to Nash H-values if every game that arises during learning has a global optimum point, and the agents update the H-values according to the rules. It will also converge if every game has a saddle point, and agents update in terms of these. In general, properties of convergence during learning are difficult to ensure. Nonetheless, establishing sufficient convergence conditions for this learning process may provide a useful insight.

3.5.4 Computing Nash H-values

A Nash equilibrium for two inhabitants consists of a pair of strategies (π_*^1, π_*^2) in which each strategy is a best response to the other. Two shortest paths that do not

interfere with each other constitute a Nash equilibrium, since each path (strategy) is a best response to the other. Different variants of the Nash equilibrium path followed by the two inhabitants are shown in Table 3.4.

Table 3.4. Zone Sequences Extracted from Location Profile

Inhabitant 1	Inhabitant 2
$CMRDG$	$GOLRMA$
$CMRDG$	$GDKRMA$
$CMRDG$	$GDRMA$
$CMRKDG$	$GOLRMA$
$CMRKDG$	$GDKRMA$
$CMRKDG$	$GDRMA$

Table 3.5. Stationary Strategy for Inhabitant 1

State	$\pi^1(s)$
(C x)	Action CM
(M x)	Action MR, MA
(R x)	Action RM, RB, RW, RK, RD, RL, RQ, RP
(D x)	Action DK, DR, DL, DG

An example strategy for inhabitant 1 is shown in Table 3.5. In the right column, all possible actions are represented for a given state of the sequence "$CMRDG$". The notation (s x) refers to any state where the first inhabitant is in zone s with an option to transit to zone x after the action (transition) being performed. States that cannot be reached given the path are omitted in the table. The strategy shown represents the path for the inhabitant 1 to reach its destination in Figure 3.6. This is the best response to inhabitant 2's path in that graph.

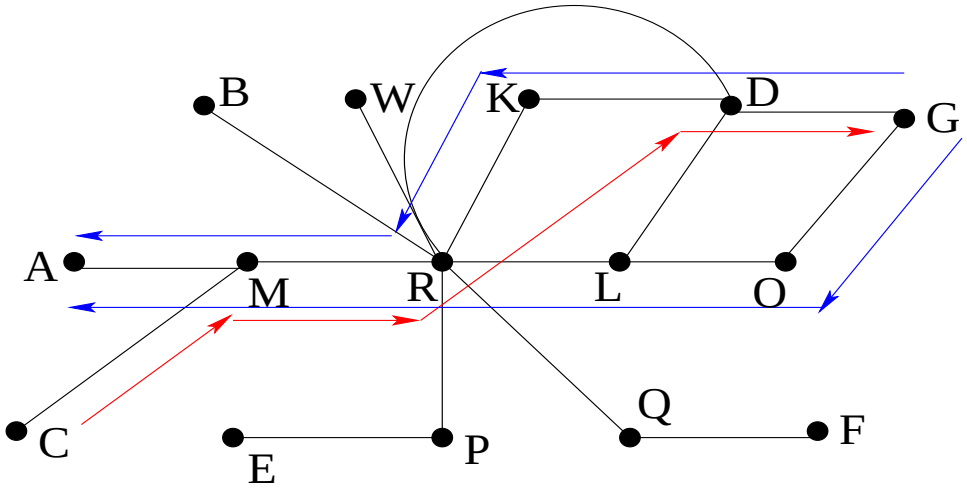

Figure 3.6. Nash Equilibrium Paths for Two Inhabitants.

The value of the game for inhabitant 1 is defined as its accumulated reward when both inhabitants follow their Nash equilibrium strategies,

$$\Re^1(s_0) = \sum_{t=0}^{\infty} \beta^t E(r_t^1 | \pi_*^1, \pi_*^2, s_0)$$

Considering initial state as $s_0 = (CG)$ and location profile as "$CMRDG$" for inhabitant 1, this reward becomes, given $\beta = 0.99$,

$$\Re^1(CG) = 0 + 0.99 \times 0 + (0.99)^2 \times 0 + (0.99)^3 \times 100 = 97.0$$

$$\Re^1(MO) = 0 + 0.99 \times 0 + (0.99)^2 \times 100 = 98.0$$

Considering the location profile as "$CMRKDG$", this reward becomes,

$$\Re^1(CG) = 0 + 0.99 \times 0 + (0.99)^2 \times 0 + (0.99)^3 \times 0 + (0.99)^4 \times 100 = 96.05$$

Based on the values for each state, we can then derive the Nash H-values for inhabitant 1 in state s_0,

$$H^1(s_0, a^1, a^2) = r_t^1(s_0, a^1, a^2) + \beta \sum_{s'} p(s' | s_0, a^1, a^2) \Re^1(s')$$

Therefore, when inhabitant 1 is on the path "$CMRDG$" and inhabitant 2 is on the path "$GDRMA$", we can derive the Nash H-values as follows by considering a collision at state "R",

$$H^1_*(s_0, CM, GD) \;=\; -1 + 0.99\Re^1(CG) = -1 + 0.99 \times 97.0 = 95.03,$$

Again when inhabitant 1 is on the path "$CMRDG$" but inhabitant 2 is on the path "$GOLRMA$" or "$GDKRMA$", the Nash H-value has been increased due to the absence of any conflict,

$$H^1_*(s_0, CM, GO) \;=\; 0 + 0.99\Re^1(CG) = 0 + 0.99 \times 97.0 = 96.03.$$

Now if we look to the other way round as inhabitant 1 is on the path "$CMRKDG$" and inhabitant 2 is on the path "$GDRMA$" we can derive the Nash H-values as follows by considering a collision at state "R",

$$H^1_*(s_0, CM, GD) \;=\; -1 + 0.99\Re^1(CG) = -1 + 0.99 \times 96.05 = 94.08,$$

Again when inhabitant 1 is on the path "$CMRKDG$" and inhabitant 2 is on the path "$GOLRMA$" or "$GDKRMA$", the Nash H-value has been increased due to the absence of any conflict,

$$H^1_*(s_0, CM, GO) \;=\; 0 + 0.99\Re^1(CG) = 0 + 0.99 \times 96.05 = 95.08.$$

The Nash H-values for both the inhabitants in state (CG) are shown in Table 3.6. There are two Nash equilibria for this game $(H^1(s_0), H^2(s_0))$, and each is a global optimal point with the value $(96.03, 96.03)$.

3.5.5 Worst-Case Analysis

In a smart home environment, multiple inhabitants act autonomously without an authority regulating their day-to-day activities in order to achieve some "social

Table 3.6. Nash H-Values

	GOLRMA	GDKRMA	GDRMA
CMRDG	96.03, 96.03	96.03, 96.03	95.03, 95.03
CMRKDG	95.08, 95.08	95.08, 95.08	94.08, 94.08

optimum" such as minimization of overall (joint) uncertainty across all inhabitants' locations and activities. In our system where multiple inhabitants share a common resource, we use the ratio between the worst possible Nash equilibrium and social optimum as a measure of the effectiveness of the system. Basically, we are investigating the cost of the lack of *coordination* as opposed to the lack of *information* (on-line algorithms) or lack of *unbounded computational resources* (approximation algorithms). The basic assumption here is that every inhabitant always attempts to benefit from the underlying utility function associated with him. Now the question is: how much performance is lost because of this? The answer to this question provides the basis for *worst-case analysis* or *coordination ratio*, given by the ratio of worst possible cost and optimal cost. Note that, although Nash Equilibrium attains a balance between the preferences of all inhabitants, it is not necessarily optimal. The deviation from optimality in this environment can be estimated using this worst-case analysis [70].

Result 5 *The worst-case coordination ratio for m inhabitants taking m actions is given by* $\Omega\left(\frac{\lg m}{\lg \lg m}\right)$.

Proof: The problem is identical to that of throwing m balls in m bins and attempting to find expected maximum number of balls in a bin. The bound follows from [70].

We believe that this lower bound is tight and if $\mathcal{N}$ denotes the expected maximum number of balls in a bin, we conjecture that the coordination ratio of any number of inhabitants taking m actions is also $\mathcal{N}$.

Theorem 2 *The coordination ratio of any number of inhabitants with m actions is at most* $\mathcal{N} = 3 + \sqrt{4m \lg m}$.

Proof: A quantity associated with an equilibrium in our context is the expected entropy over all actions for a specific route. From this perspective, inhabitant i maintains beliefs about the strategy of other inhabitants and predicts the Expected Entropy Value (EEV) of its individual action a^i at $(t+1)$-th time step as follows:

$$EEV^i_{t+1}(a^i) = \sum_{a^{-i} \in \mathcal{A}} H^i_{t+1}\{(s, a^1, \ldots, a^n) \cup (s, a^{-1}, \ldots, a^{-n})\} \prod_{j \neq i} \mathcal{P}(a^{-i})^j \qquad (3.14)$$

We call it the Nash equilibria cost which we wish to compare with the social optimum entropy, Ψ. More precisely, we want to estimate the coordination ratio as the worst case ratio, $C = \max\{\text{Nash equilibria cost / } \Psi\}$ where the maximum is taken over all equilibria. Computing the social optimum (Ψ) is an NP-hard problem (equivalent to the partition problem, see Theorem 1). However, for the purpose of upper bounding C, it suffices to use simple approximations: $\Psi \geq \max\{H^i_{t+1}(s, a^1, \ldots, a^n), EEV^i_{t+1}(a^i)/n\}$

Using a martingale concentration bound known as the Azuma-Hoeffding inequality [2] [41], we will show that the utility (entropy) of a given action a^j exceeds $(\mathcal{N}-1)\Psi$ with probability at most $\frac{1}{m^2}$. Then, the probability that the maximum utility on all actions does not exceed $(\mathcal{N}-1)\Psi$ is at least $\frac{1}{m}$. It follows that the expected maximum utility is bounded by $(1-\frac{1}{m})(\mathcal{N}-1)\Psi + \frac{1}{m}(m\Psi) \leq \mathcal{N}\Psi$. It remains to show the probability that the utility of a given action a^j exceeds $(\mathcal{N}-1)\Psi$ is indeed small, at most $\frac{1}{m^2}$.

Let X_i be a random variable denoting the contribution of inhabitant i towards the utility of action a^j. In particular, $Pr[X_i = H_1] = \mathcal{P}$ and $Pr[X_i = 0] = 1 - \mathcal{P}$.

[2] Azuma-Hoeffding inequality: Suppose that for each $i \geq 0$ there exist real numbers a_i and b_i such that $P(Y_i \in [a_i, b_i]) = 1$. Then for any $\varepsilon \geq 0$ we have $P[S_n - E(S_n) \geq n\varepsilon] \leq exp\left(-2n^2\varepsilon^2 \,/\, \sum_{i=0}^{n-1}(b_i - a_i)^2\right)$

Clearly, the random variables $X_1, \ldots, X_n$ are independent. We are interested in estimating the probability $Pr[\sum X_i \geq (\mathcal{N}-1)\Psi]$. Since the entropy H_{t+1} and probabilities $\mathcal{P}$ may vary a lot, we do not expect the sum $\sum X_i$ to exhibit the good concentration bounds of sum of binomial variables. However, we can get a weaker bound using Azuma-Hoeffding inequality which gives very good results for probabilities around 0.5. In our case, the probabilities are either 0 or 1.

Let $\mu_i = E[X_i]$ and consider the martingale $S_t = X_1 + \ldots + X_t + \mu_{t+1} + \ldots + \mu_n$. Now notice that $\mid S_{t+1} - S_t \mid = \mid X_{t+1} - \mu_{t+1} \mid \leq H_{t+1}$. We can then apply the Azuma-Hoeffding's inequality:

$$P[S_n - E(S_n) \geq x] \leq exp(-\frac{x^2}{2} / \sum_i {H_{t+1}^i}^2)$$

Let $x = (\mathcal{N}-3)\Psi$. Since $E(S_n) = \sum \mu_i = EEV_{t+1}^i(a_i) \leq 2\Psi$, we get that the entropy of action a^j exceeds $(\mathcal{N}-1)\Psi$ with probability at most $exp(-\frac{x^2}{2} / \sum_i {H_{t+1}^i}^2)$. It is easy to establish that

$$\sum_i {H_{t+1}^i}^2 \leq \max\{mH_1^2, m(\sum_i H_{t+1}^i/m)^2\} \leq m\Psi^2$$

Thus, the probability that the entropy of action a^j exceeds $(\mathcal{N}-1)\Psi$ is at most $exp(-\frac{1}{2}(\mathcal{N}-3)^2/m)$. For $\mathcal{N} = 3 + \sqrt{4m \lg m}$, this probability becomes $1/m^2$ and the proof is complete [70].

3.6 Inhabitants' Joint-Typical Routes

The collection of indoor locations inside the smart homes actually forms the routes (paths) of the inhabitants. Although there may be an exponential number of possible routes in general, in the long run the inhabitants typically follow only a small subset of them according to the mobility profiles. The concepts of *jointly-typical set* and *asymptotic equipartition property* (AEP) [26] in information theory help us derive this small subset of *highly probable* routes maintained by a particular inhabitant.

While the concept of jointly-typical set is valid for any number of sequence-sets, for the sake of simplicity, we discuss with the help of only two sets of sequences. Let $\mathcal{Z}$ and $\mathcal{Y}$ denote discrete and finite sets and let $Pr_{\mathcal{Z},\mathcal{Y}}$ be a probability mass function (pmf) on $\mathcal{Z} \times \mathcal{Y}$. Let $z^n = (z_1, \ldots, z_n) \in \mathcal{Z}^n$ denote an n-length sequence of symbols from $\mathcal{Z}$. Similarly, let y^n denote an element of $\mathcal{Y}^n$. Also, let $(\mathcal{Z}^n, \mathcal{Y}^n) = [(\mathcal{Z}_1, \mathcal{Y}_1), \ldots, (\mathcal{Z}_n, \mathcal{Y}_n)]$ denote an n-length sequence of random variables drawn according to the product measure on $\mathcal{Z}^n \times \mathcal{Y}^n$ obtained from the pmf $Pr_{\mathcal{Z},\mathcal{Y}}$. Then

$$Prob\left[\mathcal{Z}^n = z^n, \mathcal{Y}^n = y^n\right] = Pr_{\mathcal{Z}^n,\mathcal{Y}^n}(z^n, y^n) = \prod_{i=1}^{n} Pr_{\mathcal{Z},\mathcal{Y}}(z_i, y_i)$$

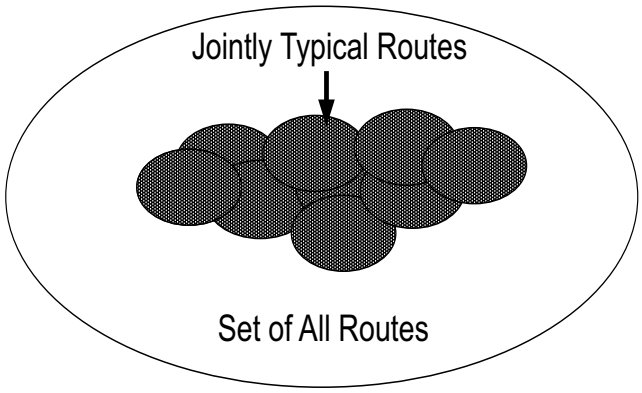

Figure 3.7. Jointly-Typical Routes.

Result 6 *[26] The set of jointly-typical sequences* $\mathcal{T}_\epsilon^{(n)} = \{(z^n, y^n) \in \mathcal{Z}^n \times \mathcal{Y}^n\}$ *for the joint probability mass function* $Pr_{\mathcal{Z},\mathcal{Y}}$ *is a set of sequences which hold the following relations*

$$\begin{aligned} \left|\frac{-1}{n} \lg Pr_{\mathcal{Z}^n,\mathcal{Y}^n}(z^n, y^n) - H(\mathcal{Z}, \mathcal{Y})\right| &\leq \epsilon \\ \left|\frac{-1}{n} \lg Pr_{\mathcal{Z}^n}(z^n) - H(\mathcal{Z})\right| &\leq \epsilon \\ \left|\frac{-1}{n} \lg Pr_{\mathcal{Y}^n}(y^n) - H(\mathcal{Y})\right| &\leq \epsilon \end{aligned} \tag{3.15}$$

As shown in Figure 3.7, the most important feature of the jointly-typical set is that it is sufficiently small and contains most of the probability mass of the set of sequences, i.e., $Pr[(\mathcal{Z}^n, \mathcal{Y}^n) \in \mathcal{T}_\epsilon^{(n)}] \rightarrow 1$. This is basically the AEP for stationary ergodic process [26]. This encompasses the inhabitant's most likely routes and determines the *average nature* of the large route-sequences.

Result 7 *[26] AEP assures that that asymptotically almost all the probability mass is concentrated in the* jointly-typical set*. This encompasses the inhabitants' most likely activities and paths and determines the* average nature *of the large route-sequences. Formally, for fixed* $\epsilon > 0$*, as* $n \rightarrow \infty$*,*

$$Pr\left[(\mathcal{Z}^n, \mathcal{Y}^n) \in \mathcal{T}_\epsilon^{(n)}\right] \rightarrow 1 \tag{3.16}$$

If $Pr[\phi_1, \phi_2]$ denotes the joint probability of the two inhabitants' contexts (routes) $\mathcal{Y}$ and $\mathcal{Z}$, each of length $\mathcal{L}(\phi)$, their *probabilistic difference* is computed as: $\delta = |Pr[\phi_1, \phi_2] - 2^{-\mathcal{L}(\phi)H(\mathcal{Z},\mathcal{Y})}|$. Clearly, δ provides the gap between the ideal probability of typical routes and the probability of a particular route stored in the dictionary. Choosing a higher value of δ leads to the inclusion of a large number of *typical* mobility profiles and the framework starts deviating from the typical-set of routes. In our experiments, we have used $\delta \leq 0.01$. Thus, the system captures a typical set of inhabitant's movement profiles from the H-learning scheme and uses them to predict the inhabitants' most likely routes.

To clarify the concept of *jointly-typical set*, we consider the following two sequences from Figure 3.6: $CMRDG$ for Inhabitant 1 and $GOLRMA$ or $GDKRMA$ Inhabitant 2, which correspond to a Nash equilibrium path. Now the joint sequence generated by both the inhabitants is given by $C, G, M, O, R, L, D, R, G, M, A$ or $C, G, M, D, R, K, D, R, G, M, A$. All other joint sequences are as follows: "$C, G, M,$

D, R, R, D, M, G, A"; "$C, G, M, O, R, L, R, R, D, M, G, A$"; "$C, G, M, D, R, K, K, R, D, M, G, A$" and "$C, G, M, D, R, R, K, M, D, A, G$".

Table 3.7. Context of Orders 0 with Occurrence Frequencies

$C(1)$	$G(2)$	$M(2)$	$O(1)$	$R(2)$	$L(1)$	$D(1)$	$A(1)$	-
$C(1)$	$G(2)$	$M(2)$	$D(2)$	$R(2)$	$K(1)$	$A(1)$	-	-
$C(1)$	$G(2)$	$M(2)$	$D(2)$	$R(2)$	$A(1)$	-	-	-
$C(1)$	$G(2)$	$M(2)$	$O(1)$	$R(2)$	$L(1)$	$K(1)$	$D(1)$	$A(1)$
$C(1)$	$G(2)$	$M(2)$	$D(2)$	$R(2)$	$K(2)$	$A(1)$	-	-
$C(1)$	$G(2)$	$M(2)$	$D(2)$	$R(2)$	$K(1)$	$A(1)$	-	-

Then we calculate $Pr_{\mathcal{Z},\mathcal{Y}}$, considering the first row of Table 3.7,

$$Prob\left[\mathcal{Z}^n = z^n, \mathcal{Y}^n = y^n\right] = Pr_{\mathcal{Z}^n,\mathcal{Y}^n}(z^n, y^n) = \prod_{i=1}^{n} Pr_{\mathcal{Z},\mathcal{Y}}(z_i, y_i)$$

$$= \left(\frac{1}{11} \times \frac{2}{11}\right) \times \left(\frac{2}{11} \times \frac{1}{11}\right) \times \left(\frac{2}{11} \times \frac{1}{11}\right) \times \left(\frac{1}{11} \times \frac{1}{11}\right) \approx 4 \times 10^{-8} \quad (3.17)$$

and verify that

$$\left|\frac{-1}{n} \lg Pr_{\mathcal{Z}^n,\mathcal{Y}^n}(z^n, y^n) - H(\mathcal{Z}, \mathcal{Y})\right| \leq \epsilon \Rightarrow \left|0.68 - \frac{1}{96.03}\right| \leq \epsilon \Rightarrow |0.67| \leq \epsilon \quad (3.18)$$

Now here the set of the sequences which contains most of the probability mass is "GMR". So, in this case, the joint typical routes of both the inhabitants is "GMR". Similarly, for all other paths of both the inhabitants from Figure 3.6, we can obtain the typical route segment as "$GMDR$", "$GMDR$", "GMR", "$GMDRK$", "$GMDR$". Considering all of the instances visited by both the inhabitants the joint typical route is given by

"$GMRGMDRGMDRGMRGMDRKGMDR$".

3.7 Resource and Comfort Management

The objectives of a smart home include how to efficiently automate device control, provide the inhabitants with maximum possible comfort, minimize operational cost and consumption of resources, say energy. By managing the uncertainty related to the inhabitant's location, the house can facilitate accurate predictions of inhabitants' activities that help smart control of automated devices and appliances, leading to better resource utilization. Minimizing energy consumption reduces the maintenance cost, furthermore, reduction in explicit manual operations and control, in turn, increases the inhabitants' comfort. In the following, we develop a mobility-aware resource management scheme for multiple inhabitant smart homes.

3.7.1 Mobility-Aware Energy Conservation

Let us first consider two simple but extremely useful energy management schemes. In the worst-case scenario, a house may use a static scheme where a certain number of devices (electric lights, fans, etc.) are switched on for a fixed amount of time during a day. Intuitively, this results in unnecessary energy consumption. On the other hand, in the best-case scenario, devices are manually controlled every time while leaving or entering particular locations inside the house. However, such manual operations are against the smart home's goals of intelligent building automation and support of calm computing. We believe a smart energy management scheme ought to use the predicted routes and activities from the NHL algorithm for smart control of devices, thus minimizing unnecessary consumption of valuable resources. This will allow devices like lights, fans or air-conditioner operate in a *pro-active* manner to conserve energy during the inhabitant's absence in specific locations (zones) in the home. These devices also attempts to bring the indoor environment, such as temper-

ature and light control, to amicable conditions before the inhabitant actually enters into those locations.

3.7.2 Estimation of Inhabitants' Comfort

The comfort is a subjective measure experienced by the inhabitants, and hence quite difficult to derive analytically. In-building climate, specifically temperature, plays the most important role in defining this comfort. Moreover, the amount of manual operations and the time spent by the inhabitants in performing the house hold activities also have significant influence on the inhabitants' comfort. We define the comfort as a joint function of temperature deviation, $\Delta(\theta)$, number of manual device operations ($\mathcal{M}$) and time spent (τ) for those activities by the inhabitants in the last Chapter. Our mobility-aware resource management framework attempts to reduce empirical values of these controlling parameters, thereby increasing the inhabitants' comfort. Note that the reduction of joint entropy by using our proposed NHL algorithm described in Figure 4.4, endows the house with sufficient knowledge for accurate estimate of current and future contexts (locations, routes and activities) of multiple inhabitants in the house. Successful estimate of these contexts results in adaptive control of environmental conditions and automated operation of devices.

3.8 Experimental Study

In this section, the proposed Nash H-learning framework is implemented and we conduct a series of experiments in MavHome [118] smart home environment to study its performance on a group of three inhabitants in a smart home equipped with smart devices and wireless sensors. The inhabitants wear radio frequency identification (RFID) tags and are tracked by RFID-readers. The house is equipped with explicit monitoring of inhabitants' activities and locations for performing a trace-

driven simulation of the inhabitant's mobility followed by the resource management scheme.

3.8.1 Simulation Environment

We have developed an object-oriented discrete-event simulation platform for generating and learning inhabitants' mobility profiles, and predict the likely routes that aid in the resource and comfort management scheme. In order to collect the test data associated with the inhabitants' life-style, the appliances in the MavHome are equipped with $X10$ ActiveHome kit and HomeSeer [117], thus allowing the inhabitants to automatically control the appliances. The identity of the inhabitants, their locations and activities are captured by wireless sensors placed inside the home. The inhabitants wear the RF-tags, which are sensed by the RF-readers to gather their identities. The raw data [118][119] as shown in Table 4.1 is first parsed using parsing tools like Perl and Tcl to remove unnecessary information. The different column headings in Table 4.1 have the following meanings: Mark as the data and time stamp, Zone and Number as unique sensor zone identifier and sensor number within it, State as binary 'on' or 'off' of the sensor, Level as specific value if on, Source as the network mode. Subsequently, we use these data to validate the mobility-aware resource management scheme. The energy and comfort management framework is compared with two reference platforms: (i) energy management without any predictive scheme, and (ii) energy management associated with per-inhabitant location prediction. The results are presented by sampling every sensor at a time and performing simulation experiments for a period of 12 weeks over 3 inhabitants and 2 visitors.

Table 3.8. A Snapshot of the Collected RAW Data

Mark	Zone	Number	State	Level	Source
$2005-01-03\ \ 09:47:30$	i	5	1	100	$X10$
$2005-01-03\ \ 09:56:17$	i	5	0	0	$X10$
$2005-01-03\ \ 13:04:45$	a	1	1	100	$X10$
$2005-01-03\ \ 13:05:37$	i	3	1	100	$X10$
$2005-01-03\ \ 13:06:11$	c	4	1	100	$X10$
$2005-01-03\ \ 13:06:22$	c	4	0	0	$X10$
$2005-01-03\ \ 13:16:32$	S	1	1	10	$ArgusMS$
$2005-01-03\ \ 13:16:33$	S	2	1	152	$ArgusMS$
$2005-01-03\ \ 13:16:33$	S	3	1	13	$ArgusMS$
$2005-01-05\ \ 23:59:00$	V	23	1	100	$ArgusD$
$2005-01-05\ \ 23:59:01$	V	23	0	0	$ArgusD$
$2005-01-05\ \ 23:59:04$	V	21	0	0	$ArgusD$
$2005-01-05\ \ 23:59:12$	V	21	1	100	$ArgusD$
$2005-01-05\ \ 23:59:12$	V	21	0	0	$ArgusD$

3.8.2 Performance Results

We have divided the entire set of simulation results into three categories. First, we demonstrate the accuracy of our proposed predictive scheme in multi-inhabitant smart homes and compare the results with our previous H-learning algorithm [97] with current modified Nash H-learning approach. Then we show the storage and computational overhead associated with it. Finally, we discuss the effect of this predictive framework in terms of energy conservation and inhabitants' comfort.

3.8.2.1 Predictive Location Estimation

Recall that the Nash H-learning framework aims at reducing the location uncertainty (entropy) associated with individual and multiple inhabitants. Figure 3.8 shows the variation of the individual and joint entropy over the entire time period of the simulation using H-learning approach. Note that the our existing H-learning

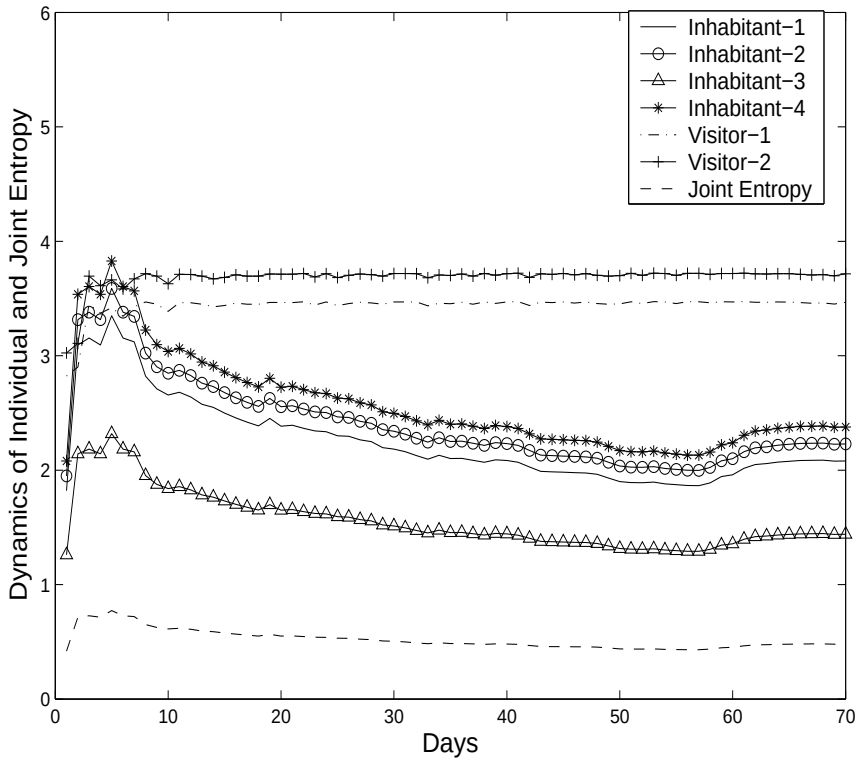

Figure 3.8. Variation of Entropy (Uncertainty) using H-Learning.

framework [97] reduces the joint entropy quickly to a low value. While the entropy of every inhabitant lies in the range $\sim$ 1–3, the visitor's entropy is typically higher ~ 4. This is quite logical as the house finds the location contexts of the visitors more uncertain than the residents (inhabitants). In comparison, Figure 3.9 shows that initially the entropy associated with three individual inhabitants is around 4.0 using Nash H-learning approach. As the predictive framework becomes knowledgable of the inhabitants' life-style, the individual entropy values reduce to 1.0. Therefore, the joint entropy is quite less than the total entropy of all the inhabitants. Initially the joint entropy is close to 8.0, but gradually it reduces to almost 1.0. The total entropy, on the other hand, lies in the range 4.0–10.0. In this way, the entropy minimization

procedure formulated by Nash H-learning helps increase the efficiency of the location estimation technique.

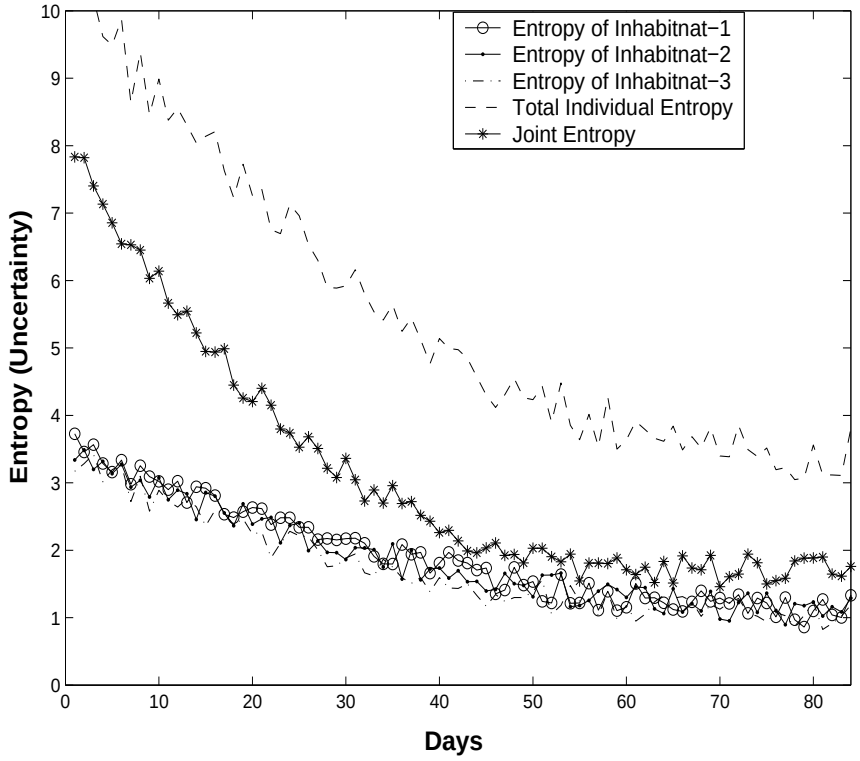

Figure 3.9. Variation of Entropy (Uncertainty) using Nash H-Learning.

The goal of our first experiment is to investigate into the dynamics of this entropy. The Nash H-learning framework also leads to higher success rate than simple H-learning. Figure 3.10 demonstrates that our co-operative H- learning strategy is capable of estimating the location of all the resident inhabitants with almost 90% accuracy within 3 weeks span. The house takes this time to learn the joint movement patterns of all inhabitants. The success rate of location estimation for visitors is however 50%–60%, as the house finds it difficult to get the knowledge of the random visitors. In comparison, Figure 3.11 shows the variation of prediction success for in-

dividual inhabitants and joint prediction success using Nash H-learning framework. Initially, the success-rate is pretty low as the system proceeds through the learning stage. Once the system becomes cognizant of inhabitants' profiles, the success rate increases and saturates at a particular value. The individual prediction process does not consider the correlation among different inhabitants. Thus, it fails to capture some important contexts and results in comparatively lower prediction success upto 80%. The joint prediction, however, takes the correlation among different inhabitants into account and results in higher success rate (close to 95%) than the simple H-learning framework.

The collection of the inhabitants' joint typical-set is the key behind the devel-

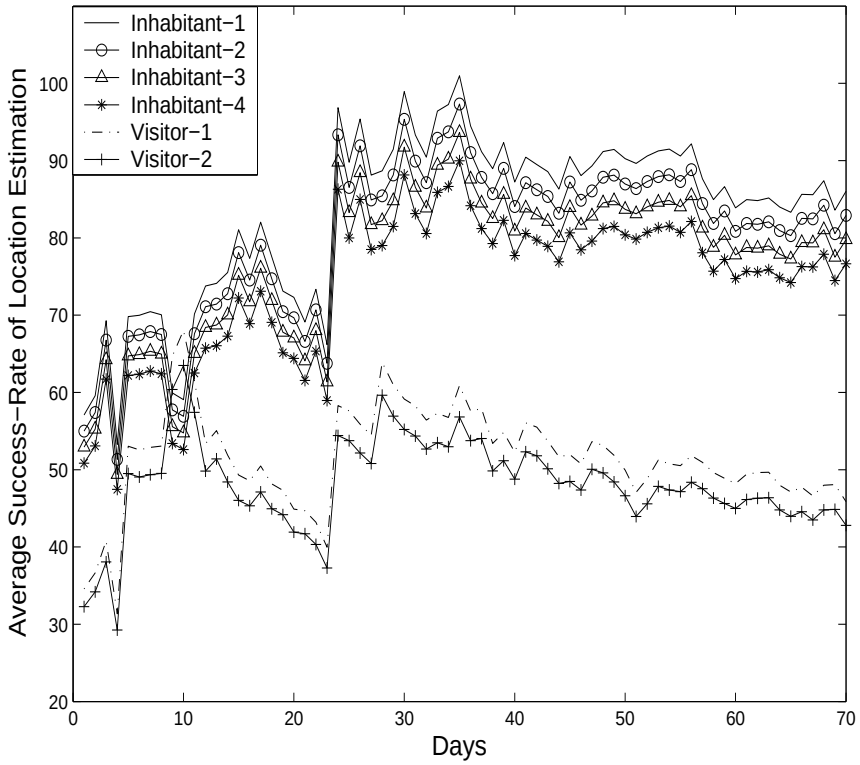

Figure 3.10. Dynamics of Prediction Success using H-Learning.

opment of efficient energy and temperature control system in the smart home. As discussed earlier, the joint-typical set is relatively a small subset of all routes (of all inhabitants) containing most of the probability mass (i.e., set of most probable routes). Figure 3.12 provides the percentage of total routes categorized as individual and joint typical routes. It is clear that the size of the individual and joint typical set is initially less than 50% of total routes. This size then gradually shrinks to as low as about 10% as the system captures the relevant contexts of inhabitants' movement-profiles.

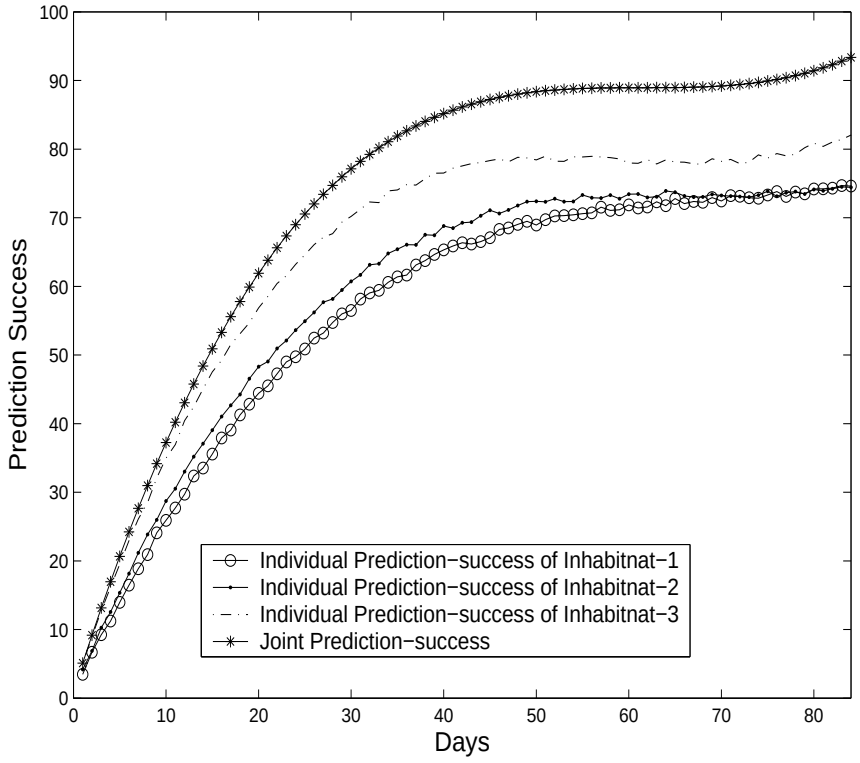

Figure 3.11. Dynamics of Prediction Success using Nash H-Learning.

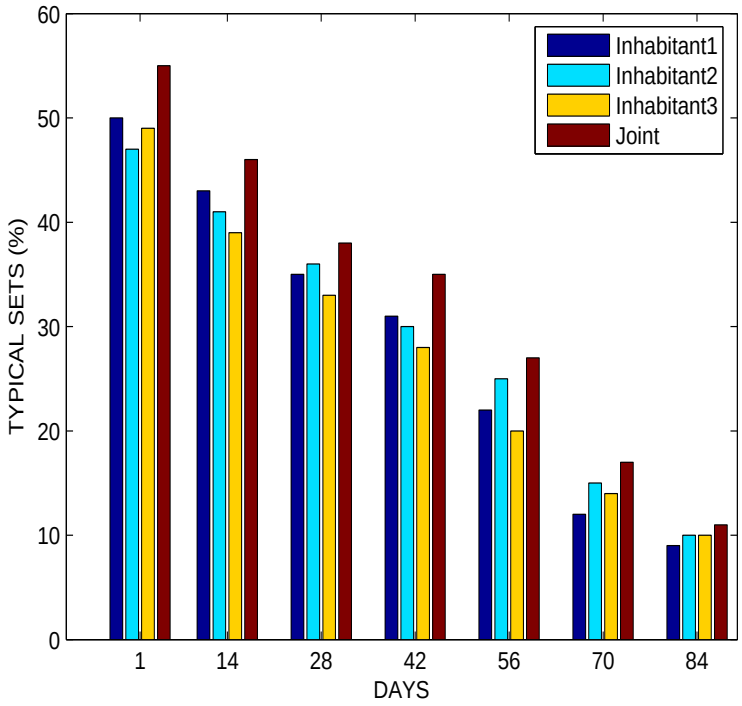

Figure 3.12. Dynamics of Typical Routes.

3.8.2.2 Storage and Computational Overhead

Another important criteria of our predictive framework is its low storage (memory) requirements. Figure 3.13 shows that the storage requirement of the joint prediction scheme is sufficiently less than the total storage requirement of the individual prediction schemes. The storage requirement of joint prediction initially starts increasing and then saturates at a reasonable value of 10 Kbytes, whereas the storage overhead for individual prediction is around 40 Kbytes.

For practical use, it is important to ensure that the savings in storage is not negated entirely by the additional computational cost of the proposed algorithm. For this purpose, we computed the average time complexity per day in the smart home for our multi-inhabitant predictive framework, as well as for the existing per-inhabitant

location-prediction algorithm [94], applied over all inhabitants. We observe that the average number of operations for the proposed multi-inhabitant prediction is around 13414 where as the same for per-inhabitant prediction is 22357. Thus, the multi-inhabitant predictive framework reduces the time complexity by 40% in comparison to the per-inhabitant location tracking framework.

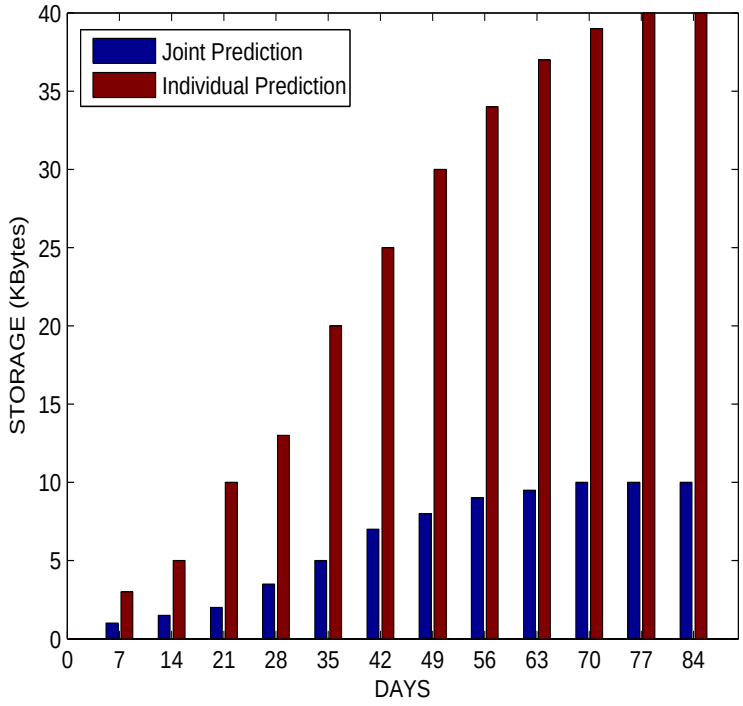

Figure 3.13. Storage Overhead.

3.8.2.3 Energy Savings and Inhabitants' Comfort

With a goal to maximize the inhabitants' comfort with minimum energy consumption, the predictive framework makes the system knowledgeable of inhabitants' profiles. The smart temperature control system and energy management framework

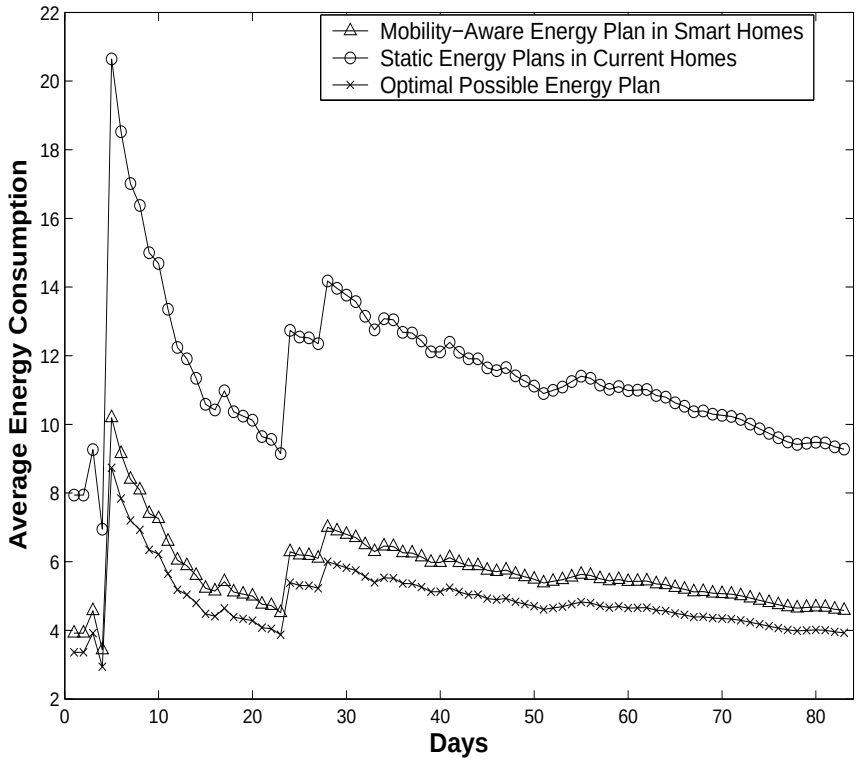

Figure 3.14. Energy Consumption.

makes intelligent use of these profiles to conserve energy. Figure 3.14 shows that using the predictive framework, the daily average energy consumption can be kept about 5 KiloWatt-hour (KW-Hr), in comparison to 9 KW-Hr for energy management scheme without the predictive framework. Figure 3.15 shows the reduction of manual operations and time spent for all the inhabitants. The predictive Nash H-learning scheme aids the system with sufficient automation, by reducing the overall manual operations performed by the inhabitants and the time spent behind all such operations which in turn increases the overall comfort.

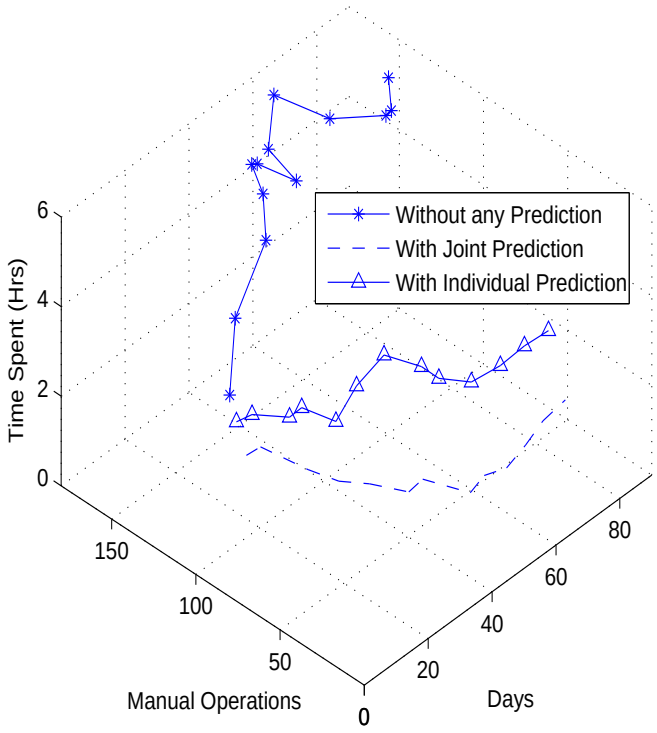

Figure 3.15. Manual Operations and Time Spent.

3.9 Summary

In this chapter, we have developed a novel mobility-aware resource management framework in a multi-inhabitant smart home. Characterizing the mobility of inhabitants as a stationary, ergodic, stochastic process, the framework uses the information theoretic measure to estimate the uncertainty associated with all the inhabitants in the house. It has also been shown that the direct use of per-inhabitant location tracking fails to capture the correlation among multiple inhabitants' locations or activities. We have proved that the multi-inhabitant location tracking is an NP-hard problem. We also formulated a non-cooperative learning paradigm based on stochastic game theory, which learns and estimates the inhabitants' most likely location (route) profiles by minimizing the overall entropy associated with them. The convergence and

worst-case performance bounds of this framework are also derived. Automated activation of devices along the predicted locations/routes provide the inhabitants with necessary comfort while minimizing energy consumption and cost. In the next chapter we will focus how such context information is useful in providing health related and wellness management services in an intelligent way to promote independent living in a smart home environment.

CHAPTER 4

AMBIGUOUS CONTEXT MEDIATION FRAMEWORK

4.1 Introduction

Current research and development in smart environments [23, 28, 98] technology offer a promising solution to the increasing needs of the elderly in home based healthcare applications. Essential to such applications is what is called *human-centric* computing and communication, where computers and devices are designed to adapt to the user needs and preferences. The objective here is to create a total solution for the perennial connection of the human with the environment, rather than focussing merely on the devices for the sole purpose of obtaining input from the human. This form of computing platforms are becoming ubiquitous in healthcare and nursing industry, thus transforming the patients from passive to active consumers of healthcare benefits [29]. To this end, current research efforts have largely focused on the development of communication technologies and intelligent user interfaces [34].

In this chapter we focus on the computational aspect of user-centric data to provide context-aware services [28, 98] that promotes intelligent independent living. Context-aware applications typically derive their desired context information (implicit input) from physical sensors and other information sources. Though sensing is becoming more and more cost-effective and ubiquitous, the interpretation of sensed data as contexts is still imperfect or ambiguous. Therefore, a critical challenge facing the development of realistic and deployable context-aware services, particularly in health related applications, is the ability to handle ambiguous contexts. The conversion of raw data into high-level context information requires middleware to pre-process such

as filter, transform, and even aggregate the data collected from homogeneous or heterogeneous distributed sensors, with a goal to minimize the ambiguity of the derived contexts. Only with reasonably accurate context(s), can applications be confident to make adaptative and better decisions. The context processing could involve simple filtering based on a value match, or sophisticated data correlation or data fusion techniques [16, 80]. Contexts may also include various aspects of relevant information; they may be instantaneous or durative, ambiguous or unambiguous. Furthermore, heterogeneous information source sensors usually have different measurement objects, different resolutions and accuracies, and different data rates and formats. Thus, the mapping from sensory output to the context information is a non-trivial task. We believe context-aware data fusion plays a critical role in improving the accuracy of the derived contexts by reducing their ambiguity, although the exact fusion technique to use is application and domain specific. This motivates our work.

4.1.1 Related Work

The *ubiquitous computing* paradigm [121] implies smart (i.e., pro-active) interaction of computing and communication devices with their peers and surrounding networks, often without explicit operator control. Hence, such devices need to be imbued with an inherent sentience [54] about their important contexts that can automatically or implicitly sense information about their state and the presence of users (inhabitants) in order to take action on those contexts. This concept has led to various projects smart homes or environments in general [22, 23]. Existing work such as the Reactive Room [24], Neural Network House [82], Intelligent Room [21] and House_n [60] do not provide explicit reusable support for users to manage or correct uncertainty in the sensed data and their interpretations, and thereby assume that the sensed contexts are unambiguous. The work reported in [34] provided a toolkit to enable the

integration of context data into applications, however, no mechanism is provided for sensor fusion or reasoning about contexts to deal with ambiguity. Although other works such as [61] proposed mechanisms for reasoning about contexts, yet they do not provide well defined context-aware data fusion model nor address the challenges associated with context ambiguity and users' situation prediction. Distributed mediation of ambiguous contexts in aware environments was discussed in [31] that allow the user to correct ambiguity in the sensed input. Multimodal Maps [15] for travel planning addresses ambiguity by using multimodal fusion to combine different inputs and then prompting the user for more information to remove the remaining ambiguity as much as possible. Remembrance Agent [93] uses context to retrieve information relevant to the user and explicitly addresses ambiguity in its manual interface.

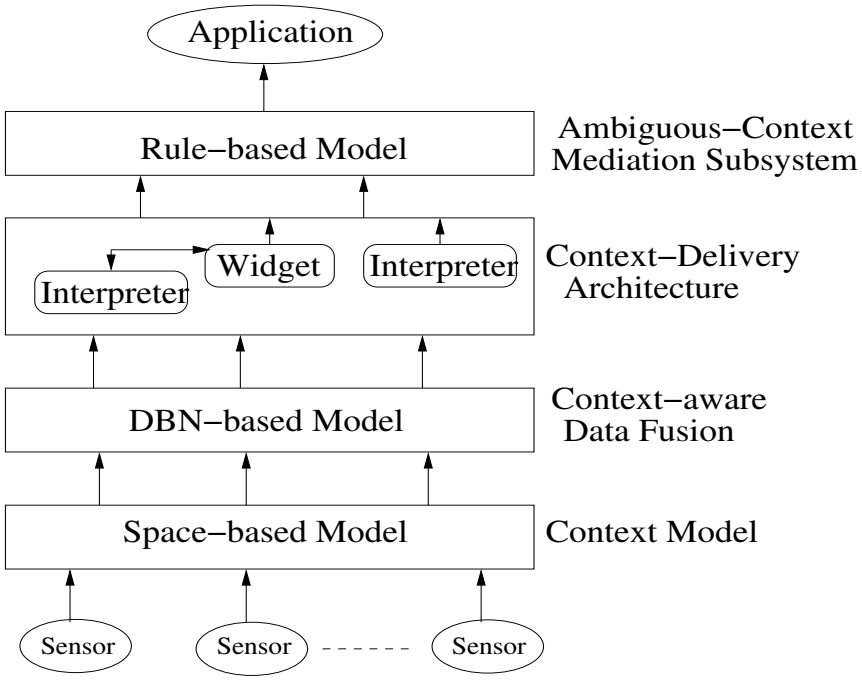

Figure 4.1. A Middleware Framework for Ambiguous Context Mediation.

Alongside, significant efforts have been made to develop middleware systems that can effectively support context-aware applications in the presence of resource constraints (e.g., sensor networks), also considering requirements for sensory data or information fusion from middleware perspective [1]. For example, DFuse [73] is a data fusion framework that facilitates dynamic transfer of different application level information fusion into the network in order to save power. In adaptive middleware [55] for context-aware applications in smart home setups, the application's quality of context (QoC) requirements is matched with the QoC attributes of the sensors with the help of a utility function. Similarly, in MiLAN [49], application's quality of service (QoS) requirements are matched with the QoS provided by the sensor networks. However, in this scheme, the QoS requirements of the applications are assumed to be predetermined, which the applications should know in advance in addition to the quality associated with the type of sensors it can make use of. Given that in ubiquitous computing environments, the nature (number, types and cost off usage, and benefits) of such sensors available to the applications usually vary, it is impractical to include a priori knowledge about them. The selection of right sensor with right information at the right moment was originally introduced in [114], while the structure of an optimal sensor configuration constrained by the wireless channel capacity was investigated in [13]. By eliminating the simplifying assumption that all contexts are certain, in an earlier work [99], we designed a context-aware data fusion algorithm based on dynamic Bayesian network to mediate ambiguous context. But an intelligent sensor management that provides energy-efficiency as well as a way to manage quality of context requirements, which may change over time with changes in patient's state, has not been considered before. In this chapter an information theoretic approach is taken to decide an optimal sensor configuration to determine the best current state of the patient while satisfying the application QoC requirements. For end user an

ontological rule based approach using semantic web technology is proposed for further reduction of context ambiguity with applications to context-aware healthcare services. By eliminating the simplifying assumption that all contexts are certain, in this chapter we propose a middleware architecture as shown in Fig. 4.1 (explained in Section 4.5.2) that supports a variety of services, ranging from context-aware data fusion to ambiguous context mediation subsystem with applications to context-aware healthcare services [99]. The major contributions of this work are summarized later.

4.1.2 Example Scenario

As an example, let us take the scenario of a home care patient after hospitalization for cardiac infarction. Although such a patient should be guaranteed a good quality of life and wellness management services in an independent way, he/she still needs to be in constant contact with an expert physician so that his/her cardiac activity (e.g., the heart rate and peripheral blood pressure), body temperature and breathing frequency can be continuously monitored. However, the health condition of a patient can only be partially evaluated through his vital signals and must be mediated and integrated by other signals and information coming both from personal characteristics (risk factors, degree of disease, age, sex, family history, psychological features, etc.) and from the environmental context (e.g., whether in bed or mobile, by him/herself or in company, at work or at home, the season and the temperature, etc.). The monitoring system should be able to deduce the context from the available data to provide a feedback to the patient as well as notifying his status to somebody else, such as a relative, the family doctor, or the hospital, depending on the degree of alert detected, and possibly adapting the level of service (i.e., the intensity of the monitoring activity).

The above scenario requires the integration of patients' vital signs monitored by different sensory medical devices, of environmental data acquired by sensors located near the patient, of patient data available from the electronic medical records stored by the hospital. Although the current technologies offer the necessary means to support this kind of health care, in our opinion without a contextual realization that tailors the available data into usable information, the healthcare applications will become practically unusable. Contextual information deals with information about the user environment (e.g., location, activity) that enables this tailoring and reduces efforts required to develop healthcare applications.

Application scenarios of the type presented above give rise to several issues. The sensory devices constantly attached to the patient produce huge streams of physiological data which must be collected and related to environmental conditions. To achieve this we need a technique that can fuse acquired data with different modalities to infer the current context state (activity) and situation space (behavior or sickness) associated with the monitored person. Again these sensors should be light and portable to reduce their impact on the patient's well-being and thus must be constrained in terms of energy capacity. Consequently, the amount of information transmitted to the sensor fusion mediator (the data aggregator) should be minimized in order to prolong its lifetime by selecting the structure of an optimal set of sensors based on the QoC guarantees and cost of information acquisition.

4.1.3 Our Contributions

In this chapter, we propose a framework that fuses data from disparate sensors, represents context state (activity) and reasons efficiently about this state, to support context-aware services that deal with ambiguity and allow users to mediate ambiguity, if any. For environments with ambiguous contexts, our goal is to build a framework

that resolves information overlap conflicts, and also ensures the conformance to the application's quality of context (QoC) bound based on an optimal sensor configuration. For this purpose, we propose layered and modularized system design using Dynamic Bayesian Networks (DBNs) [63] in which the sensed data is used to interpret context state and the sensor fusion process is analogous to the human perception and reasoning processes. The use of DBNs as our baseline sensor fusion mechanism reflects this analogy whereas an information theoretic reasoning selects an optimal context attribute (sensor data) value to satisfy the application QoC bound. However, our proposed technique can not remove all the ambiguity in the sensed data, leaving it up to the programmer and inhabitants to deal with. To alleviate this problem, we propose to leverage off a rule based model [101] and involve end users in removing any remaining ambiguity through a process, called *ambiguous-context mediation subsystem*. We use Semantic Web [110] technology to implement this rule based model to visualize wellness management services to the elderly person. Experimental results demonstrate that the proposed framework is capable of adaptively enhancing the effectiveness of the probabilistic sensor fusion scheme and patient's situation prediction by selectively choosing the sensor corresponds to the most economically efficient disambiguation action.

The rest of the chapter is organized as follows. Section 4.2 describes the basic concepts of context model and quality of context in resource limited sensor network. Section 4.3 describes the context-aware (active) data fusion model based on the DBNs for resolving ambiguity. In Section 4.4 we study the structure of an optimal sensor configuration from an information theoretic point of view. A rule based model with its prototype for the realization of unambiguous context has been discussed in Section 4.5. The performance of our proposed system is evaluated for health monitoring

application in a smart home, and the results are presented in Section 4.6. Finally, Section 4.7 concludes this Chapter.

4.2 Context Model

Context-aware data fusion in the face of ambiguities is a challenging research problem as the data sent to the sensor fusion mediator collected from network of multiple sensors are often characterized with a high degree of complexity due to the following challenges: (i) data are often acquired from sensors of different modalities and with different degrees of uncertainty and ambiguity, (ii) decision must be made quickly, and (iii) the situation as well as sensory observations always evolve over time. We make use of the space-based context model [89] and extend it with quality of context (QoC) attributes. This model captures the underlying description of context related knowledge and attempts to incorporate various intuitions that should impact context inference, to produce a better fusion results. This approach is specifically intended for use in context-aware applications, and exhibits many characteristics desirable in reasoning about context.

4.2.1 Space-based Context Model

This model defines the following concepts:

Definition 5 Context Attribute: *A context attribute, denoted by a_i, is defined as any type of data that is used in the process of inferring situations. A context attribute is often associated with sensors, virtual or physical, where the values of the sensor readings denote the context attribute value at a given time t, denoted by $\tilde{a}_i^t$.* The body temperature "100^0 F" of a patient measured by i-th sensor at a given time t is an example of a context attribute.

Definition 6 Context State: *A context state describes the application's current state in relation to chosen context, and is denoted by a vector S_i. It is a collection of N context attribute values that are used to represent a specific state of the system at time t. Thus, a context state is denoted as $S_i^t = (\tilde{a}_1^t, \tilde{a}_2^t, \ldots, \tilde{a}_N^t)$.* Suppose the body temperature is "100^0 F" and the location is in "gym", then the context state of the patient is "doing physical exercise".

Definition 7 Situation Space: *A situation space represents a real-life situation. It is a collection of ranges of attribute values corresponding to some predefined situation (sickness, normal behavior) and denoted by a vector space $\mathcal{R}_i = (\tilde{a}_1^R, \tilde{a}_2^R, \ldots \tilde{a}_{\mathcal{M}}^R)$ (consisting of $\bar{M}$ acceptable ranges R for these attributes). An acceptable range $\tilde{a}_i^R$ is defined as a set of elements $\bar{V}$ that satisfies a predicate $\bar{\mathcal{P}}$, i.e., $\tilde{a}_i^R = \bar{V}|\bar{\mathcal{P}}(\bar{V})$.* For example the context attribute body temperature can take values within "98^0 F" to "100^0 F" when the patient context state is "doing physical exercise" with predefined situation space "normal". But if the context attribute body temperature takes values within this range "98^0 F" to "100^0 F" when the patient context state is "lying on the bed" then the situation space is "not normal".

4.2.2 Quality of Context

Despite recent development in sensing and network technology, continuous monitoring of individuals vital signs (e.g., the heart rate and peripheral blood pressure, body temperature and respiratory rate) and environmental context (e.g., whether in bed or mobile, by him/herself or in company, at work or at home, the season and the temperature, etc.) in normal setting is still challenging due to the resource constrains of sensor networks. The sensors should be light and portable to reduce their impact on the patient's well-being and thus must be constrained in terms of energy capacity. Consequently, the amount of information transmitted to the sensor fusion mediator

should be minimized in order to prolong its lifetime. We define the Quality of Context (QoC) [56] as a metric for minimizing resource usage (e.g., battery life, communication bandwidth) while maintaining a minimum quality of the data received. QoC is essential to our model in choosing the best data values among the monitored ones for reporting a specific type of context. For example, if the blood pressure of an inhabitant in a smart home monitoring environment lies in between the predefined normal range (120/80 mmHg), or frequency of getting up from the bed at night is $(2-3$ times) then the sensor need not to report that value to the sensor fusion mediator again. But if the aggregated value computed at the mediator is beyond the tolerance level of QoC (± 10 mmHG for BP or $> 5-6$ times for Frequency), then the sensor needs to report its samples back to the mediator.

Thus sensor fusion mediator always ensures that the aggregated value computed by it dose not diverge from the true reading by more than a specified "tolerance". The key is to have the mediator communicate a precision range or interval to an individual sensor, with an idea that a sensor need not report its samples back to the mediator as long as they fall into this specified range. Such tolerance is expressed in terms of "Quality of Context" (QoC) metric, and is especially useful for applications issuing aggregation queries. But in case of an emergency medical situation when all the sample values lie outside this range, the mediator gets the update from all available sensors in order to compute the best possible estimate of the patient's state. We assume that the information fusion issues an aggregation query with its QoC specified by a precision range Q, which implies that the aggregate value computed at the mediator at any instant should be accurate within $\pm Q$. Our primary objective is to evaluate this update cost of a sensory action A for a given task while ensuring the conformance to the application's QoC bound. Let us denote the update cost

(communication overhead) as $\bar{\mathcal{U}}_i^j$ if indeed sensor B_i has to report its sample value at time j to the mediator. Then, the objective is to

$$\text{minimize} \sum_{i \in B_m} \bar{\mathcal{U}}_i(q_i) \tag{4.1}$$

where $\bar{\mathcal{U}}_i$ denotes the expected average Update (reporting) cost and explicitly indicates its dependence on the specified precision interval q_i. Intuitively, $\bar{\mathcal{U}}_i$ is inversely proportional to q_i, since the value of the reporting cost would be high as the precision interval keeps on shrinking.

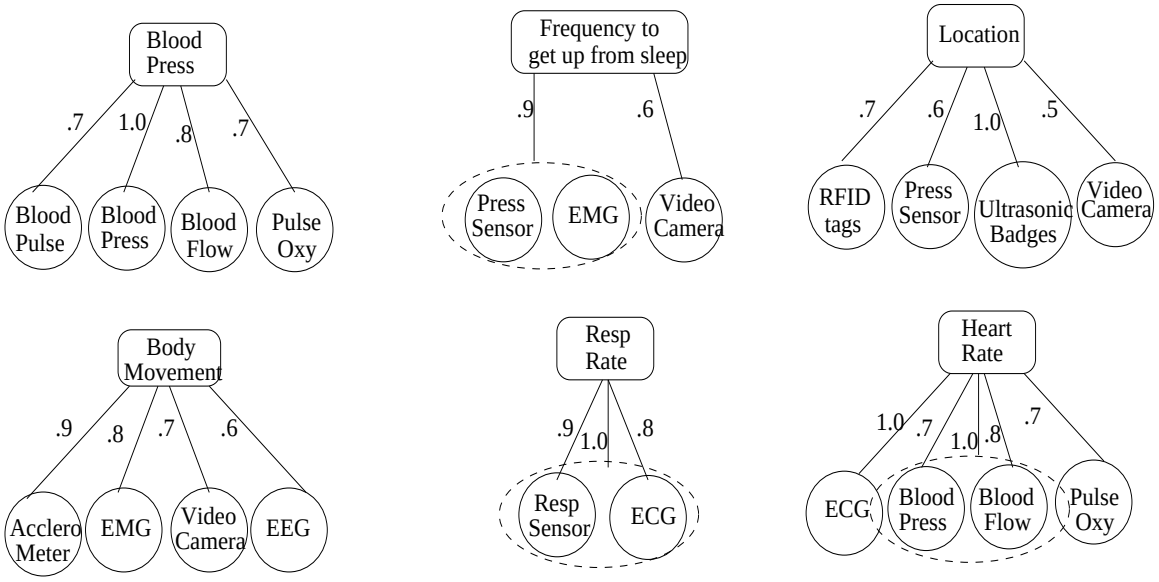

Figure 4.2. Subset of Sensors with Required QoC.

But quality for each context attribute can be satisfied using data from one or more sensors. Context-aware data fusion plays an important role when multiple sensors are fused to provide a certain quality level to a context attribute. Fig. 4.2[1]

[1]EMG: Electromyography; EEG: Electroencephalography; ECG: Electrocardiogram with 1, 3, 5 or 12 leads

illustrates the important context attributes to monitor when determining a patient's state and indicates the group of sensors that can meet the QoC bound to the measurements of these variables. The line between the sensor and context attribute represents the quality that the sensor can provide to the measurement of that variables. For example, using data from a respiratory sensor, the respiratory rate can be determined with a 0.9 quality level, but combining this with data from a ECG sensor increases the quality level to 1.0.

4.3 Context-Aware Data Fusion

A characteristic of a sensor-rich smart healthcare environment is that it senses and reacts to *context*, information sensed about the environment's occupants and their daily activities, by providing context-aware services that facilitates the occupants in their everyday actions. Here we develop an approach for sensor data fusion in context-aware healthcare environment considering the underlying space-based context model and a set of intuitions it covers. In the case of context-aware services, it is really difficult to get an accurate and well defined context which we can classify as 'unambiguous' since the interpretation of sensed data as context is mostly imperfect and ambiguous. To alleviate this problem, we view context information as ambiguous or unambiguous and propose a DBN model for ambiguous contexts and a rule based mediation technique for unambiguous contexts. For example, a user location can be sensed using ultrasonic badges, RFID-tags, video cameras or even pressure sensors in the floor. All of these sensors have some degree of precision in the data they sense. For instance, ultrasonic badges can determine location with a precision of up to 3 cm, while RF lateration is limited to 1-3 m. Similarly, a video camera system which identifies the user posture based on the current position (sitting, standing, lying on the floor in distress) has a different probability of correctness than using pressure

sensors in furniture. The ambiguity problem becomes worse when the application derives implicit higher-level context state (activity of the person) based upon those inputs. For example, an application may infer that a person is lying in distress. However, there may be other explanation of this phenomenon such as the person might be lying to perform normal exercises. Thus, we design a context-aware (active) data fusion framework based on DBNs to reduce this ambiguity as much as possible during the situation inference (patient's behavior or sickness) process.

4.3.1 Dynamic Bayesian Network Based Model (DBN)

Our motivation is to use the data fusion algorithm to develop a context-aware model to gather knowledge from sensor data. We look for a technique which is appropriate for performing context-aware data fusion with the flexibility of both top-down and bottom-up inference mechanisms. The top-down inference can be used to predict the utility of a particular sensory action with respect to a goal at the top. For example, in the case of a given context state (going to restroom), it will fuse the most relevant context attributes (time, frequency of getting up from the bed, blood sugar level etc.). The bottom-up inference allows the integration of the context attributes from a sensory action and update each node about the context state in the network. Dynamic Bayesian Networks can be used for similar problems since it provides a coherent and unified hierarchical probabilistic framework for sensory data representation, integration and inference. Figure 4.3 (adopted from [122]) illustrates a DBN based framework for context-aware data fusion system consisting of a situation space, context states, context attributes, a sensor fusion mediator and network of information sensors.

The selection of an information source (sensor) or the activation of a process to compute new information is simply regarded as a set of actions available to the

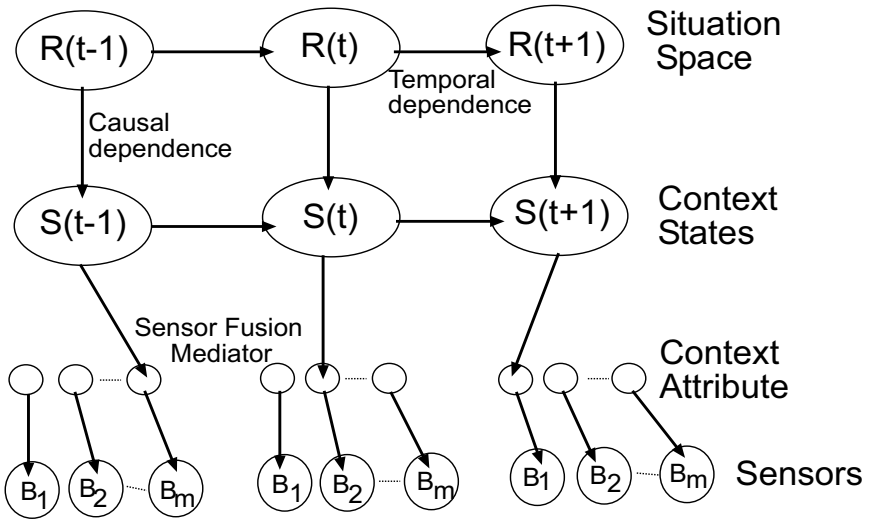

Figure 4.3. Context-Aware Data Fusion Framework based on Dynamic Bayesian Networks.

```
Procedure ACMA
1.  t = 0, Compute ambiguity-reducing utility {V̄_1^t, ..., V̄_m^t} by Eqn.4.2
2.  Calculate utility value {Ū_1^t, ..., Ū_m^t} using Eqn.4.4
3.  Select the most economically efficient disambiguation
    sensor action based on Eqn.4.5
4.  Instantiate the subset of corresponding sensors B
5.  Run ACMA inference algorithm to update probability
    distribution P(R, A) of situation space
6.  If P(R, A*) ≥ confidence threshold, then terminate; otherwise
7.  Add a new time stamp t = t + 1, and go to step 1
```

Figure 4.4. Ambiguous Context Mediation Algorithm (ACMA).

decision maker in decision theory [63]. For example, the value of context attribute location can be measured by ultrasonic badges, RFID-tags, video cameras and even pressure sensors in the floor. In our case, the information module needs to determine the next optimal context attributes and the corresponding sensory action such as triggering ultrasonic badges or RFID-tags or video cameras or even pressure sensors. But selecting an action always has a consequence which can be measured by the cost

of information acquisition, QoC bound, varying the confidence of the situation space. If we can devise a cost measure to each possible consequence, this can be used by the system to decide what action to perform, and which sensor to activate on. We have to choose the action that maximizes its ambiguity reducing utility which depends on the current available sensory data, the internal context state, and the current situation space.

Let us assume that we have a situation space $\mathcal{R}_i$ to confirm using the sensory information sources $B = \{B_1, \ldots, B_m\}$ which is a set of measurements taken from sensors labeled from 1 to m as shown in Fig. 4.3. The context attribute which is most relevant in our case should decrease the ambiguity of the situation space $\tilde{a}_j^R$ the most; and we will select the one that can lead the probabilities of situation space close to near one (for maximum) and zero (for minimum). Let $\bar{\mathcal{V}}_i$ be the ambiguity reducing utility to the situation space $\mathcal{R}_i$ with N states. Then the expected value of $\bar{\mathcal{V}}_i$, given a context attribute $\tilde{a}_i^t$ from a sensor B_i, which has K possible values, can be represented as

$$\bar{\mathcal{V}}_i = \max_{i=0}^{K} \sum_{j=0}^{N} [P(\tilde{a}_j^R | \tilde{a}_i^t)]^2 - \min_{i=0}^{K} \sum_{j=0}^{N} [P(\tilde{a}_j^R | \tilde{a}_i^t)]^2 \tag{4.2}$$

where $i \in \{1, 2, \ldots m\}$ is a sensor tag which identifies the sensor that provides the context attribute. This context attribute can be measured by propagating the possible outcome of an information source, i.e.,

$$P(\tilde{a}_j^R | \tilde{a}_i^t) = \frac{P(\tilde{a}_j^R, \tilde{a}_i^t)}{P(\tilde{a}_i^t)} \tag{4.3}$$

However, quantification of this conditional probability needs a detailed model depending upon the usage of different types of sensors and their applications. Consider, for example, an audio sensor. Evaluating the benefit of using audio in disambiguating whether a person is moaning in pain or singing, is really hard. It depends on how far

the person is from the microphone, which way the person is facing, the time of day (at night it is more quiet so sounds can be heard more clearly), the state of other potentially interfering audio sources (such as air conditioning, TV, radio, refrigerator), etc. Computing the disambiguating utility therefore, needs very detailed models of how the above factors affect the efficacy of the audio sensor.

Considering the information update cost from Eqn. 4.1 and ambiguity reducing utility from Eqn. 4.2, the overall utility can be expressed as

$$\bar{U}_i = \zeta \bar{\mathcal{V}}_i + (1-\zeta)(1-\bar{\mathcal{U}}_i) \tag{4.4}$$

where $\bar{\mathcal{U}}_i$ is the update cost to acquire the information by sensor with tag i with a knowledge of QoC bound, ζ denotes the balance coefficient between the ambiguity reduction and the cost of information acquisition. Eqn. 4.4 represents the contribution to ambiguity reduction and the cost associated with information retrieval to achieve the desired level of confidence to the situation space. We can observe from Eqn. 4.4 that the utility value of a context attribute $\tilde{a}_i$ increases with the ambiguity reducing utility and decreases as the cost to acquire that attribute increases. So the most economically efficient disambiguation sensor action $\mathcal{A}^*$ can be chosen with the help of the following decision rule

$$\mathcal{A}^* = \arg\max_{\mathcal{A}} \sum_j \bar{U}(B, \tilde{a}_j^R) P(\tilde{a}_j^R | B) \tag{4.5}$$

where $B = \{B_1, \ldots, B_m\}$ is a set of measurements taken from sensors labeled from 1 to m at a particular point of time. By incorporating the temporal dependence between the nodes as shown in Fig. 4.3, the probability distribution of the situation space we want to achieve can be generally described as

$$P(\mathcal{R}, \mathcal{A}) = \prod_{t=1}^{T-1} P(S_t | S_{t-1}) \prod_{t=1}^{T-1} P(\mathcal{R}_t | B_t) P(\mathcal{R}_0) \tag{4.6}$$

where T is time boundary; the situation $\mathcal{R} = \{\mathcal{R}_0, \ldots, \mathcal{R}_t, \ldots, \mathcal{R}_T\}$ and the subset of sensed information $B = \{B_0, \ldots, B_t, \ldots, B_T\}$, on time sequence of T. Here $S = \{S_0, \ldots, S_t, \ldots, S_T\}$ represents a context state relevant on time sequence of T that has temporal links between corresponding nodes in two neighboring time frames. The sensor action strategy must be recalculated at each time slice since the best action varies with time. The ambiguous context mediation algorithm is presented in Fig. 4.4.

4.4 Information Theoretic Reasoning

Consider the personal health monitoring scenario discussed in Section 4.1.2 running on a PDA that receives and analyzes data from a number of sensors (e.g., ECG, EMG, blood pressure, blood flow, pulse oxymeter). The monitor reacts to potential health risks and records health information in a local database. Considering that most sensors used by the personal health monitor will be battery operated and use wireless communication, it is clear that this application can benefit from intelligent sensor management that provides energy-efficiency as well as a way to manage QoC requirements, which may change over time with changes in the patient's state. For example, higher quality might be required for certain health-related context attributes during high stress situations such as a medical emergency, and lower quality during low stress situations such as sleep. Thus application performance can be described by QoC of different context attributes of interest, where the QoC of different context attributes depends on which sensors in which context state (exercising, lying in distress) provide data to the application. In personal health monitoring scenario, context attributes such as blood pressure, respiratory rate, heart rate, location etc may be determined based on measurement obtained from any of the several sensors as shown in Fig. 4.2. Each sensor has a certain quality on characterizing each of the application's context attributes. As an example in Fig. 4.2, a blood pressure sensor can directly measure

blood pressure with a quality of 1.0 [2] in determining this context attribute, where as heart rate measured by this sensor can have quality less than 1.0. But the quality of the heart rate measurement could be improved through high-level fusion of the blood pressure measurements with data from additional sensors such as blood flow sensor depending upon the context state. Fig. 4.5 shows the context attributes requirement graph for personal health monitor which includes multiple states for each vital signs that can be monitored depending upon the context state of the patient. For example, the Fig. 4.5 shows that when a patient is in lying in distress state and the blood pressure is low, the blood oxygen level must be monitored with a quality of .7 and the blood pressure must be monitored with a quality of .8. So the problem here is to decide what type of information each sensor should send to the fusion center to estimate the best current state of the patient while satisfying the application QoC requirements for each context attribute.

We introduce a formalism for optimal sensor parameter selection for state estimation. The optimality is defined in terms of reduction in ambiguity or error in state estimation process. The main assumption is that state estimation becomes more reliable and accurate if the ambiguity/error in the underlying state estimation process can be minimized. We investigate this from an information theoretic perspective [26] where information about the context attribute is made available to the fusion center by a set of smart sensors. The fusion center produces an estimate of the state of the situation based on the intelligent analysis on the received data. We assume that the noisy observation across sensors are independent and identically distributed (i.i.d) random variable conditioned on the binary situation $\mathcal{R}$ (assume situation $\mathcal{R}$ here as binary for ease of modeling). Now each sensor attribute has a source entropy rate $H(\tilde{a}_i)$. Any sensor wishing to report this attribute must send $H(\tilde{a}_i)$ bits per unit

[2] Quality with 1.0 corresponds to 100% reliability

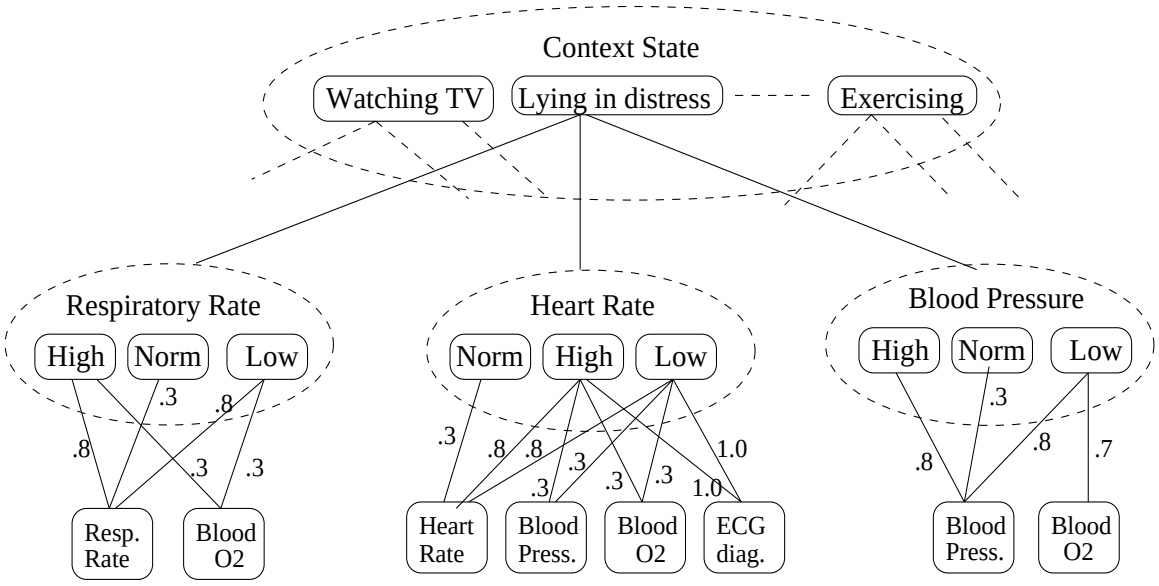

Figure 4.5. State-based Context Attribute Requirement Graph with the Required QoC.

time which is the entropy of the source being measured assuming that the sensor is sending the 'exact' physical state. Of course, different sensors (due to perhaps their sensing limitations) contribute in different measures to the 'error' in state estimation. So, the problem is to minimize the error (or keep it within a specified bound), while not exceeding the shared link rate $\mathcal{Q}$. Thus by maximizing the posteriori detector probability we can minimize the estimation error of the random variables based on noisy observation from a set of sensors at the fusion center to accurately reconstruct the state of the situation [13].

Problem 1 *Let* $\mathbf{B}$ *be the vector of sensors and* $\mathbf{A}$ *be the set of attributes, then imagine a* $(\mathbf{B} \times \mathbf{A})$ *matrix where* $\mathbf{B}_{mi} = 1$ *when sensor* m *sends attribute* $\tilde{a}_i$*. Then, the goal is to find a matrix* $(\mathbf{B} \times \mathbf{A})$ *within the capacity constraint* $\mathcal{Q}$ *which minimizes the estimation error of the situation space.*

$$\sum_{m} \sum_{i} H(a_i) * \mathbf{B}_{mi} < \mathcal{Q} \quad \textit{and} \quad \texttt{minimize}\,[P_e = P\{\tilde{\mathcal{R}} \neq \mathcal{R}\}] \tag{4.7}$$

where $\mathbf{B}_{mi}$ *is an indicator function taking on the value* 1 *only if the sensor* m *actually sends attribute* $\tilde{a}_i$*,* $H(\tilde{a}_i)$ *is the source entropy rate of the attribute* $\tilde{a}_i$ *and* $\tilde{\mathcal{R}}$ *is an estimate of the original state* $\mathcal{R}$*,*

4.4.1 Problem Explanation

We assume $\mathcal{R}$ be the random variable drawn from the binary alphabet $\{\mathcal{R}_0, \mathcal{R}_1\}$ with prior probabilities p_0 and p_1, respectively. In our case each sensor needs to determine a sequence of context attributes for a sequence of context state $\{S_{m,t} : \forall t = 1, 2, \ldots, T\}$ about the value of situation $\mathcal{R}$. We assume that random variable $S_{m,t}$ are i.i.d., given $\mathcal{R}$, with conditional distribution $p_{S|\mathcal{R}}(.|\mathcal{R}_i)$. The sensors could construct a summary $Z_{m,t} = \pi_m(S_{m,t})$ of their own observation to a fusion center at discrete time t. The fusion center produces an estimate $\tilde{\mathcal{R}}$ of the original situation $\mathcal{R}$, upon reception of the data. Thus we need to find out an admissible strategy for an optimal sensor-attribute mapping matrix $(\mathbf{B} \times \mathbf{A})$ that minimizes the probability of estimation error $P_e = P\{\tilde{\mathcal{R}} \neq \mathcal{R}\}$

Definition 8 *A set of decision rules* π_m *for an observation* $X \rightarrow \{1, 2, \ldots \bar{a}_m\}$ *where* $\bar{a}_m$ *is the number of attribute admissible to sensor* B_m *with the admissible strategy denoted by* π*, consists of an integer* M *in* $(\mathbf{B} \times \mathbf{A})$ *matrix, such that*

$$\sum_{m=1}^{M} \sum_{i} H(\bar{a}_m.\tilde{a}_i) * \mathbf{B}_{mi} < \mathcal{Q}$$

Evaluation of message $z_{m,t} = \pi_m(s_{m,t})$ by sensor B_m is forwarded to the fusion center at time t. Since we are interested in a continuous monitoring scheme here we consider the observation intervals T tends to ∞. But the associated probability of error at the fusion center goes to zero exponentially fast as T grows unbounded. Thus we can

compare the transmission scheme through the error exponent measure or Chernoff information

$$E(\pi) = -\lim_{T\to\infty} \frac{1}{T} \log P_e^{(T)}(\pi) \tag{4.8}$$

where $P_e^{(T)}(\pi)$ denotes the probability of error at the fusion center for strategy π considering maximum posteriori detector probability. We use $\Pi(\mathcal{Q})$ to capture all admissible strategies corresponding to a multiple access channel with capacity $\mathcal{Q}$ and redefine our problem as follows:

Problem 2 *Find an admissible strategy $\pi \in \Pi(\mathcal{Q})$ that maximizes the Chernoff information*

$$E(\pi) = -\lim_{T\to\infty} \frac{1}{T} \log P_e^{(T)}(\pi) \tag{4.9}$$

4.4.2 Results

Let us consider an arbitrary admissible strategy $\pi = (\pi_1, \pi_2, \ldots, \pi_M)$ and denote the space of received information corresponding to this strategy by

$$\gamma = \{1, 2, \ldots, \bar{a}_1\} \times \{1, 2, \ldots, \bar{a}_2\} \times \ldots \times \{1, 2, \ldots, \bar{a}_M\} \tag{4.10}$$

where

$$(\pi_1(x_1), \pi_2(x_2), \ldots, \pi_M(x_M)) \in \gamma \tag{4.11}$$

for all observation vectors $(x_1, x_2, \ldots, x_M) \in X^M$. Since the maximization of posteriori detector is basically the minimization of the probability of estimation error at the fusion center, we could just approximate this probability of error for a finite observation interval T and can measure the error exponent corresponding to strategy π by using the Chernoff's theorem [26].

Next we consider $p_{\tilde{Z}|\mathcal{R}}(.|\mathcal{R}_0)$ and $p_{\tilde{Z}|\mathcal{R}}(.|\mathcal{R}_1)$ as the conditional probability mass functions on γ, given situation $\mathcal{R}_0$ and $\mathcal{R}_1$. Now for $\tilde{z} = (z_1, z_2, \ldots z_M)$ and $i \in 0, 1$

$$\begin{aligned} p_{\tilde{Z}|\mathcal{R}}(\tilde{z}|\mathcal{R}_i) &= P_i\left\{\tilde{x} : (\pi_1(x_1), \pi_2(x_2), \ldots, \pi_M(x_M)) = \tilde{z}\right\} \\ &= \prod_{m=1}^{M} P_i\{\pi_m(u_m)\} \end{aligned} \tag{4.12}$$

where the probability of event $\bar{W}$ is $P_i\{\bar{W}\}$ under situation $\mathcal{R}_i$, and $\pi_m(u_m) = \{x : \pi_m(x) = z_m\}$

Theorem 3 *[26] Using the Chernoff's theorem, the best achievable exponent in the probability of error at the fusion center is given by*

$$E(\pi) = -\min_{0\leq k\leq 1} \log\left[\sum_{\tilde{z}\in\gamma} (p_{\tilde{Z}|\mathcal{R}}(\tilde{z}|\mathcal{R}_0))^k (p_{\tilde{Z}|\mathcal{R}}(\tilde{z}|\mathcal{R}_1))^{1-k}\right]$$

where $\pi \in \Pi(\mathcal{Q})$ is given. Using Theorem 3 we can restate our original problem as follows

Problem 3 *Maximized the Chernoff information*

$$E(\pi) = -\min_{0\leq k\leq 1} \log\left[\sum_{\tilde{z}\in\gamma} (p_{\tilde{Z}|\mathcal{R}}(\tilde{z}|\mathcal{R}_0))^k (p_{\tilde{Z}|\mathcal{R}}(\tilde{z}|\mathcal{R}_1))^{1-k}\right]$$

corresponding to an admissible strategy $\pi \in \Pi(\mathcal{Q})$.

The problem of finding the optimal decision rules $\pi = (\pi_1, \pi_2, \ldots, \pi_M)$ is hard even when the assignment vector $(\bar{a}_1, \bar{a}_2, \ldots, \bar{a}_M)$ is fixed a priori. Hence we try to derive a set of simplified conditions for Problem 3. Thus we state the following Lemma, where we upper bound the contribution of a single sensor to the Chernoff information and find sufficient conditions for which having $\mathcal{Q}$ sensors in $(\mathbf{B} \times \mathbf{A})$ matrix, each sending one bit of information is optimal.

Lemma 1 *For strategy π, the contribution $E_{B_m}(\pi)$ from a single sensor B_m to the Chernoff information $E(\pi)$ is bounded above by the Chernoff information E^* contained in one context state S,*

$$E_{B_m}(\pi) \leq E^* \equiv -\min_{0\leq k\leq 1} \log \left[\int_X (p_{S|\mathcal{R}}(x|\mathcal{R}_0))^k . (p_{S|\mathcal{R}}(x|\mathcal{R}_1))^{1-k} dx \right] \tag{4.13}$$

Proof: Considering the contribution of sensor B_m, the Chernoff information for strategy $\pi = (\pi_1, \pi_2, \ldots, \pi_M)$ is given by

$$\begin{aligned}
E(\pi) &= -\min_{0\leq k\leq 1} \log \left[\sum_{\tilde{z}\in\gamma} (p_{\tilde{Z}|\mathcal{R}}(\tilde{z}|\mathcal{R}_0))^k (p_{\tilde{Z}|\mathcal{R}}(\tilde{z}|\mathcal{R}_1))^{1-k} \right] \\
&= -\log \left[\prod_{m=1}^{M} \left(\sum_{z_m=1}^{\bar{a}_m} (P_0\{\pi_m(u_m)\})^{k^*} (P_1\{\pi_m(u_m)\})^{1-k^*} \right) \right] \\
&= -\sum_{m=1}^{M} \log \left[\sum_{z_m=1}^{\bar{a}_m} (P_0\{\pi_m(u_m)\})^{k^*} (P_1\{\pi_m(u_m)\})^{1-k^*} \right] \\
&= -\log \left[\sum_{z_1=1}^{\bar{a}_1} (P_0\{\pi_m(u_m)\})^{k^*} (P_1\{\pi_m(u_m)\})^{1-k^*} \right] \\
&- \sum_{m=2}^{M} \log \left[\sum_{z_m=1}^{\bar{a}_m} (P_0\{\pi_m(u_m)\})^{k^*} (P_1\{\pi_m(u_m)\})^{1-k^*} \right]
\end{aligned} \tag{4.14}$$

where the Chernoff information $E(\pi)$ is maximized at k^*. So we can conclude that contribution of sensor B_m to the Chernoff information $E(\pi)$ can not exceed

$$-\min_{0\leq k\leq 1} \log \left[\sum_{z_m=1}^{\bar{a}_m} (P_0\{\pi_m(u_m)\})^k (P_1\{\pi_m(u_m)\})^{1-k} \right] \tag{4.15}$$

which in turn is upper bounded by the Chernoff information contained in one context state S. So, the Lemma 1 confirms that the contribution of a single sensor to the total Chernoff information is no way greater than the information contained in each observation. Hence we derive the sufficient condition based on the Lemma 1 where having $\mathcal{Q}$ binary sensors is optimal.

Let us represent $E_1(\pi_m)$ as the Chernoff information corresponding to a single sensor with decision rule π_m, i.e.,

$$E_1(\pi_m) = -\min_{0\leq k\leq 1} \log \left[\sum_{z_m=1}^{\bar{a}_m} (P_0\{\pi_m(u_m)\})^k (P_1\{\pi_m(u_m)\})^{1-k} \right] \tag{4.16}$$

and let Π_b be the set of binary functions on the observation space X.

Lemma 2 *Consider a binary function $\tilde{\pi}_b \in \Pi_b$ such that $E_1(\tilde{\pi}_b) \geq \frac{E^*}{2}$ then having $\mathcal{Q}$ identical sensors, each sending one bit of information is optimal.*

Proof: Let strategy $\pi = (\pi_1, \pi_2, \ldots, \pi_M) \in \Pi(\mathcal{Q})$ and rate $\mathcal{Q}$ be given. We construct an admissible strategy $\pi' \in \Pi(\mathcal{Q})$ such that $E(\pi') \geq E(\pi)$. We divide the collection of decision rules $\{\pi_1, \pi_2, \ldots, \pi_M\}$ into two sets, a first set contains all the binary functions, whereas the other is composed of the remaining decision rules. We also consider I_b to be the set of integers for which the function π_m is a binary decision rule

$$I_b = \{m : 1 \geq m \geq M, \pi_m \in \Pi_b\} \tag{4.17}$$

Similarly, we define $I_{nb} = \{1, 2, \ldots, M\} - I_b$. Considering binary decision rule $\hat{\pi}_b \in \Pi_b$, we express

$$E_1(\hat{\pi}_b) \geq \max\{\max_{m\in I_b}\{E_1(\hat{\pi}_b)\}, \frac{E^*}{2}\} \tag{4.18}$$

Since by assumption $\tilde{\pi}_b \in \Pi_b$ and $E_1(\tilde{\pi}_b) \geq \frac{E^*}{2}$, we could infer that such a function $\hat{\pi}_b$ always exits. Observing that $m \in I_{nb}$ implies that $\bar{a}_m \geq 2$, which in turn yields $H(\bar{a}_m.\tilde{a}_i) \geq 2$. Hence without exceeding the capacity ($\mathcal{Q}$ bits per unit time) of the multiple access channel, we can replace each sensor with index in I_{nb} by two binary sensors. Considering the alternative scheme π', where π' is an admissible strategy, we replace every sensor with index in I_{nb} by two binary sensors with decision rule $\hat{\pi}_b$. This new scheme outperforms the original strategy π as shown in Equation 4.19.

$$
\begin{aligned}
E(\pi') = (\mid I_b \mid +2|I_{nb}|)\, E_1(\hat{\pi}_b) \geq |I_b| E_1(\hat{\pi}_b) + |I_{nb}| E^* \\
\geq & \sum_{m=1}^{M} \left[-\min_{0 \leq k \leq 1} \log \left[\sum_{z_m=1}^{\bar{a}_m} (P_0\{\pi_m(u_m)\})^k (P_1\{\pi_m(u_m)\})^{1-k} \right] \right] \\
\geq & -\min_{0 \leq k \leq 1} \log \left[\sum_{\tilde{z} \in \gamma} \left(\prod_{m=1}^{M} (P_0\{\pi_m(u_m)\})^k (P_1\{\pi_m(u_m)\})^{1-k} \right) \right] \\
= & \quad E(\pi)
\end{aligned} \tag{4.19}
$$

This proof is based upon the noisy observations across sensors which are independent and identically distributed random variable conditioned on situation $\mathcal{R}$. We also observe that the Chernoff information at the fusion center is monotonically increasing in the number of sensors for a fixed decision rule $\tilde{\pi}_b$. State estimation error can be minimized by augmenting the number of sensors in π' until the capacity constraint $\mathcal{Q}$ is met with equality. The strategy π being arbitrary, we conclude that having $\mathcal{Q}$ identical sensors in $(\mathbf{B} \times \mathbf{A})$ matrix, each sending one bit of information is optimal in terms of reducing the state estimation error. This configuration also conveys that the gain offered through multiple sensor fusion exceeds the benefits of getting detailed information from each individual sensor.

4.5 Rule Based Model

Unambiguous context can be realized to some extent if we can directly involve the user action as one of the input to the model to mediate the current associated ambiguity with the context. We have used Bayesian network and information theoretic reasoning to deal with imperfect contexts. However, it is next to impossible to make the sensed data as completely free from ambiguity, leaving it up to the context-aware environment programmer and the inhabitants to deal with. An application can choose to ignore the ambiguity and just take some action (e.g., act on the most likely

choice) or can use rule-based techniques to ask the end user about his/her actual intent. So we need to model a framework that is able to capture context and deliver it to the caregiver/user who can make use of mediation technique for reduction of this ambiguity.

4.5.1 Example Rule Sets

Through the application of active database [91] technology, it is possible to detect and mediate, urgent and ambiguous contexts and react as soon as possible when potentially unusual or hazardous situations develop. We have used the following rule format derived from the active database for identifying the ambiguous contexts.
on *measuring* <context attribute> **if** *true* <context state> **do** *evaluation of* <situation space>
The underlying idea is the following. The situation evaluation should be done if a particular context state is true on capturing some context attribute value. To illustrate, consider the following rules that can be used to actively monitor an elderly person in a smart home [3].

***R*ule 1:**

on Context Attribute (CA) : <body temperature>, <respiratory rate>, <blood pressure>

if Context State (CS) : <doing physical exercise>, <lying on the bed>

do Situation Space (SS): <general sickness of a person>

Suppose using some specific sensors in an indoor smart home environment, we monitor the value of a few context attributes such as body temperature, respiratory rate and blood pressure of the person in the home. Now if the body temperature, respiratory rate and blood pressure are measured as higher than a specified range, we can infer about the situation of the person by observing the context state. If the

context state is doing physical exercise, then the situation is *normal*. Otherwise, if the context state is lying on the bed, then the situation is *abnormal*. Let us consider another rule where we consider the context attribute as time instant, time span and location of the inhabitant in the home.

***R*ule 2:**

on CA: <time instant>, <time span>, <location>

if CS: <doing morning/evening walk>, <talking with neighbors>, <sleeping>

do SS: <normal behavior>

If it is 2am at morning and the location of the person is outside the home where the context state is walking, then we can conclude that the behavior of the person is not *normal*. But if it is 6am in the morning and the location of the person is outside the home where the context state is walking, then we can conclude that the behavior of the person is *normal*. We can make different variants of this rule by considering different context attributes with a different context state. Thus detecting that a person has been engaged in a specific activity for an unusual time may be an indicator of a health problem or a potentially hazardous domiciliary situation.

***R*ule 3:**

on CA: <time>, <frequency for getting up from the bed>

if CS: <watching the television>, <going to restroom>

do SS: <sickness of a person>

We can frame another rule where the context attribute are time and frequency for getting up from the bed. If the time is late night and frequency is too high we have to take a look at the context state. If the context state is watching the television then we don't need to comment regarding the person situation but if it's going to restroom then might be there is a need for taking care of the person (measuring the blood sugar level, blood pressure etc.).

Again, the inhabitant's current activity (e.g., cooking vs. watching television) or location in the environment (e.g., bedroom vs. navigating the stairs) can affect the choice of sensors to use, and thus represent an ambiguous context. In the next section we will discuss different components of this rule based model.

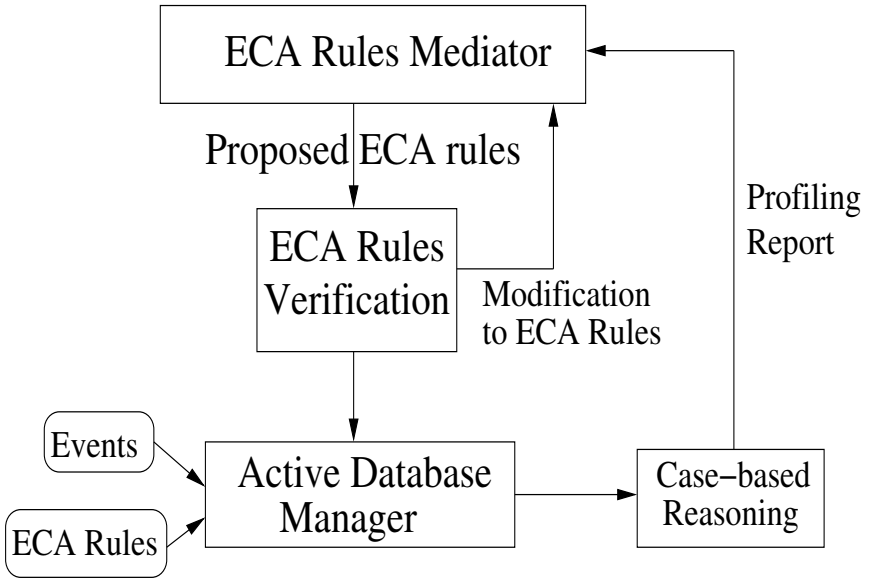

Figure 4.6. Ambiguous Context Mediation Subsystem.

4.5.2 Architecture

We have built support for the mediation of this ambiguous context by incorporating the rule based approach in the Context Toolkit Model [34] on top of our underlying context-aware data fusion and context delivery model. There are two basic building blocks – context widgets and context interpreters – that are relevant to our discussion as shown in Fig. 4.1. Context widgets are basically GUIs which are responsible for collecting contextual information about the patient's normal daily activities in the smart home environment. They are responsible for providing a uniform

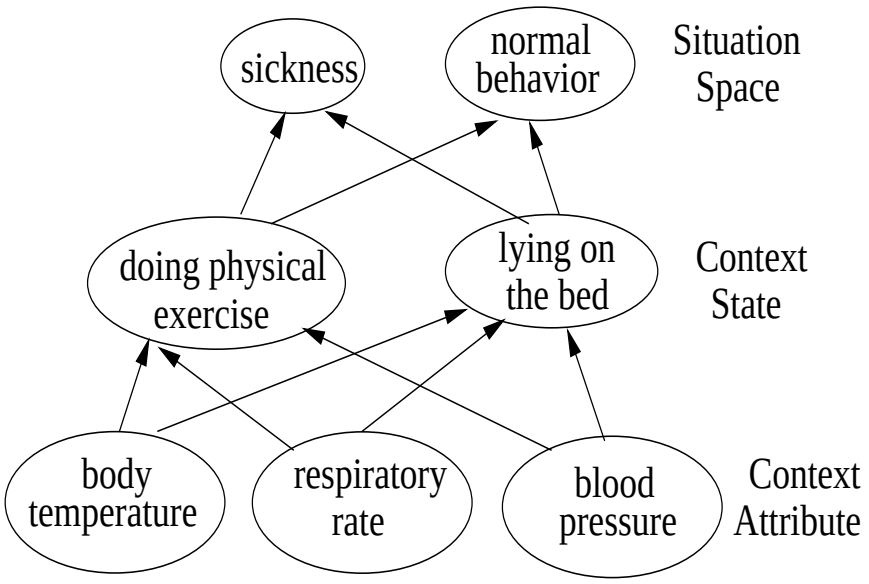

Figure 4.7. An Event Graph Representing the ECA Rules.

interface to its fellow components or applications that use the context, hiding the details of the underlying derivation of higher level context. They maintain a persistent record of all the context they sense and allow applications and other widgets to query those context information. A context interpreter is used to abstract or interpret context. A context widget may forward the context attribute location as inside the bedroom, but an application may require the exact geometric location. A context interpreter can be used to provide this kind of abstraction.

The architecture of ambiguous context mediation subsystem is shown in Fig. 4.6 which consists of an Event-Condition-Action (ECA) rules mediator, ECA rules verification and active database manager. This subsystem provides an internal rule-based model which encapsulates information about the context and the relationship between input and interpretations of that input, produced by ECA rules in a graph as shown in Fig. 4.7. This graph keeps track of source events and their interpretations. This ECA rules mediator displays a portion of the graph to the user. Based on the user's

response, the mediator accepts or rejects events in the graph. Once the ambiguity has been resolved, it passes through the ECA rules verification to ensure its correctness as a whole. The active database manager can detect complex event generated by a person through the event-condition-action rules. According to the context in which these events occur, the system will warn when potentially unusual or hazardous situations develop. There are also some feedback loops such as suggested modification to ECA rules, profiling report detailing improved suggestions which can enhance the performance of this mediation technique. An implementation of this rule based model is discussed next.

4.5.3 Rule based Engine Implementation

We represent the internal rule based model using Semantic Web Technology and OWL (Web Ontology Language) [110]. The OWL is an ontology markup language that enables context sharing and context reasoning. The ontology is described in OWL as a collection of RDF (Resource Description Framework) triples, each statement being in the form of (*subject, predicate, object*), where *subject* and the *object* are the ontology's objects or individual and *predicate* is a property relation defined by the ontology.

4.5.3.1 RDF Vocabulary

We create a RDF vocabulary for ambiguous context mediation subsystem based on the rule based approach using an ontology compliance level (OWL Lite) and create the model and generate the RDF/XML schema using the SemanticWorks interface from Altova Inc [107]. We have defined three classes *ElderlyPerson*, *GrandParents* and *ContextAttribute* for the rules defined in Section 4.5.1. We define *GrandParents* as a subclass of *ElderlyPerson*, which essentially states that any instance of the

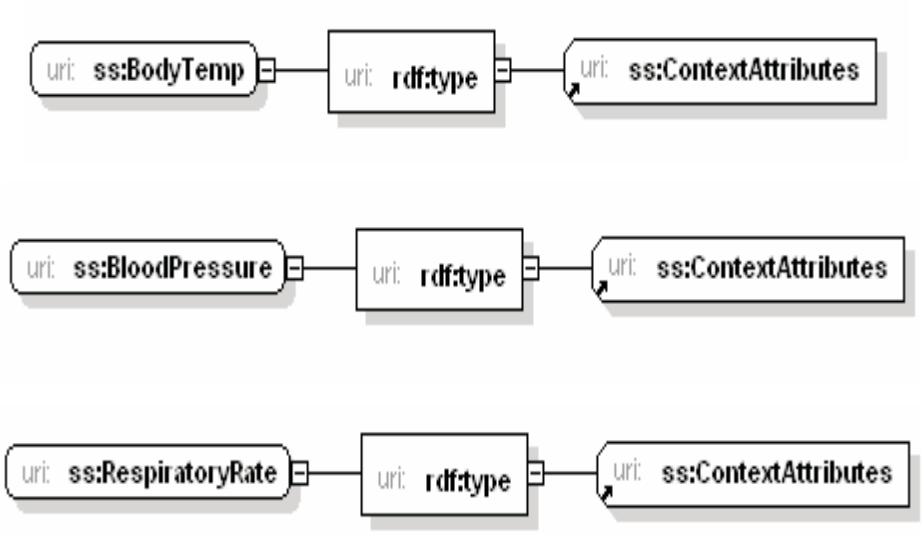

Figure 4.8. Different Instances of ContextAttribute Class.

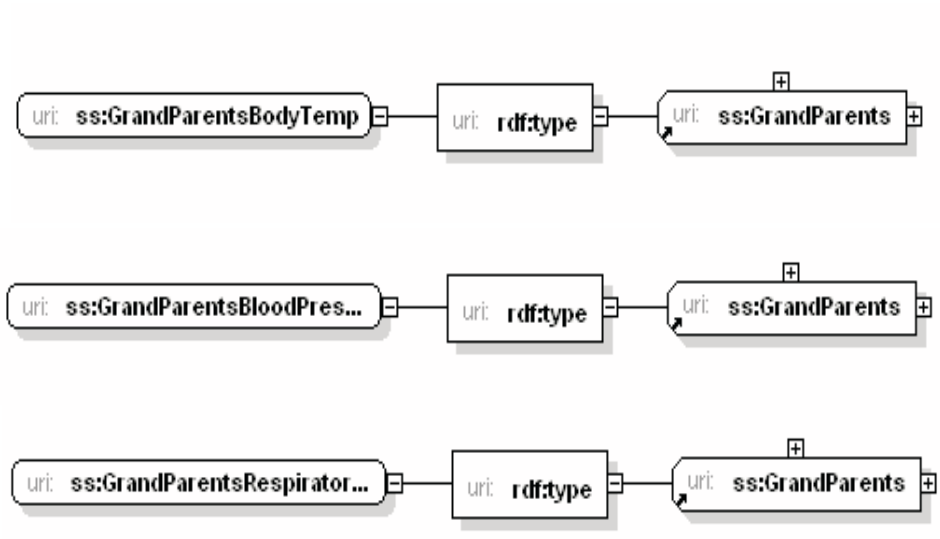

Figure 4.9. Different Instances of GrandParents Class.

GrandParents class must also be an instance of the *ElderlyPerson* class. The class (or classes) that the property applies to is called the property's domain, while the set of values the property can take is called the property's range. Properties are created at a global level and then related to different classes. In our ontology, we deal with two properties:

- Object property: *hasExercising*, *hasLying* to carry information about the type of the *ContextAttribute*. The *ContextAttribute* can be *BodyTemp*, *RespiratoryRate* and *BloodPressure*. We will create this property as an object property. Doing this enables us to relate one resource to another. In this case we wish to relate instances of the *GrandParents* class to instances of the *ContextAttribute* class via the *hasExercising*, *hasLying* property.
- Datatype property: *name*, which is a literal value indicating the name of the GrandParents. We will create this property as a datatype property.

We use the *ContextAttribute* class to (i) define it as the range of a property called *hasExercising* or *hasLying* and (ii) create instances of *ContextAttribute*. We define the class *GrandParents* to be the domain of the property *hasExercising*, *hasLying* and the class *ContextAttribute* to be the range of the property *hasExercising*, *hasLying*. This would mean that the property *hasExercising*, *hasLying* applies to the class *GrandParents* and takes values that are instance of the class *ContextAttribute*.

We create three instances of the *ContextAttribute* class as shown in Fig. 4.8, which will be simple instances like *BodyTemp*, *RespirtoryRate* and *BloodPressure*. Then we define three more instances of the *GrandParents* class and add predicates with them. The instance *GrandParentsBodyTemp*, *GrandParentsRespiratoryRate*, *GrandParentsBloodPressure* has therefore been defined to:

- Be an instance of the class *GrandParents*,

- Have object property *hasExercising*, *hasLying* that takes the instance *ContextAttribute* as its object, and
- Have a datatype property *name* that takes the string as its literal value.

4.5.3.2 Demonstration

To validate our work and to demonstrate the expressiveness of the model we implemented a reasoning component, which implements this rule model. The java based reasoning component gets RDF-information about these classes, available context state, context attribute, and can then trigger actions and provide prioritization information about situation space. Jena 2.5.3, an open source Semantic Web Toolkit [62] is utilized within the reasoning component for parsing the RDF descriptions. We implemented demonstration software, which can be used for experimenting with the reasoning component. The demonstration software includes the RDF vocabulary. This vocabulary has been converted to java class file using the schemagen utility from Jena API which helps to get access to different properties of RDF graph based model. A RDF metadatabase is developed which contains several datavalue used for querying context. We use SPARQL [111], a query language which can select RDF triples from this database. Applications based on our ambiguous context mediation subsystem can query contexts by specifying a SPARQL query. The queries are executed directly with Jena's SPARQL support for querying RDF models. Variables in the rules present the resources (users, situations), which are to be found with the SPARQL query. The RDF descriptions and rules in the demonstration were written by using an XML/RDF definition for presenting RDF models. Here is an example of a statement from a rule, which is supported by our implementation:

string rules = "[Rule1: *(?GrandParentsBodyTemp ss:hasLying ?BodyTemp)* $\wedge$ *(?BodyTemp ss:hasGreaterThan ?TempValue)* $\wedge$ *(?GrandParentsBloodPressure*

ss:hasLying ?BloodPressure) $\wedge$ *(?Bloodpressure ss:hasGreaterThan ?BPValue)* $\Rightarrow$ *(?GrandParents ss:hasSituation sick)*]"

Currently conditions in our model correspond to Boolean "and". "Or" can be achieved by creating several rules. Relational operations $(>, <)$ are being implemented, as well as support for spatial and temporal reasoning, e.g., by introducing event sequences. The following are the examples of a partial rule set based on the forward-chaining rule engine.

string rules = "[Rule2: *(?ElderlyPerson ss:hasLocation ?location)* $\wedge$ *(?ElderlyPerson ss:hasTime ?time)* $\wedge$ *(?ElderlyPerson ss:hasWalking ?walking)* $\Rightarrow$ *(?ElderlyPerson ss:hasSituation parasomnias)*]"

string rules = "[Rule3: *(?ElderlyPerson ss:hasWatching ?television)* $\wedge$ *(?ElderlyPerson ss:hasTime ?time)* $\wedge$ *(?ElderlyPerson ss:hasFrequency ?toRestroom)* $\Rightarrow$ *(?ElderlyPerson ss:hasSituation normal)*]"

The context widget is used for managing the user's RDF, including the rules and context information presented within. The end-user UI demonstrates how the situation are presented to the user based on the prioritization obtained via reasoning. The UI also tells when context-based actions are triggered. The reasoning component uses SPARQL queries for checking if a rule presents a situation that can be found in the available RDF information. SPARQL queries are automatically generated from the rules of the user's or situation's description in order to find the relevant matches. If a match exists, the rule is true and the relevant actions can be triggered, and/or the matching situation can be categorized. In the demonstration implementation a simple relevance metrics is used for the situation prioritization; categories have different fixed priorities, but all the situations within a category are prioritized equally.

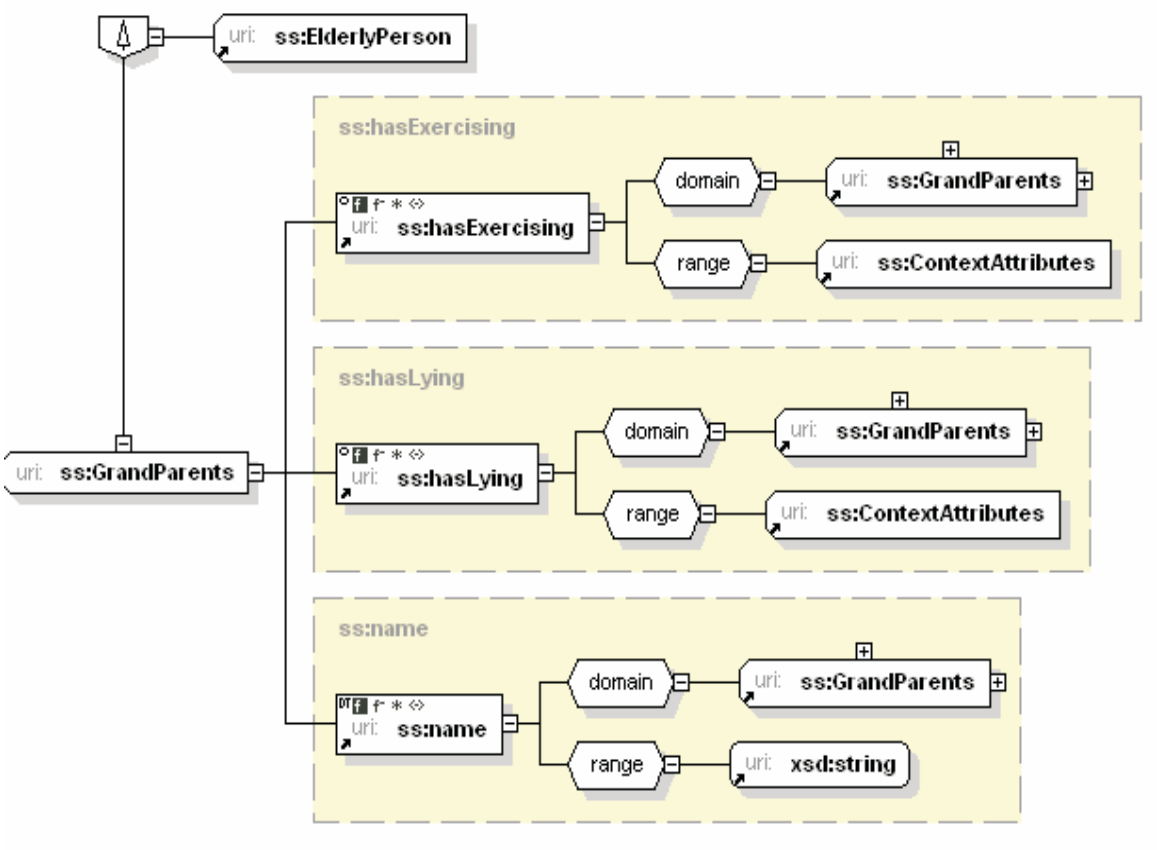

Figure 4.10. RDF Model View of GrandParents Class.

4.6 Simulation Study

We conducted simulation experiments to evaluate the performance of the proposed ambiguous context mediation framework in a smart home health monitoring environment and report the results in this section. The ambiguous context mediation algorithm (ACMA) given in Fig. 4.4 was applied during our evaluation. In our application, the goal is to determine a set of sensors and the situation level (emergency or non-emergency) of a patient based on most economically efficient disambiguation sensor action. Let's assume situation level has three states, high, medium and low. Fig 4.11 represents a snapshot of the Bayesian Network model for this application

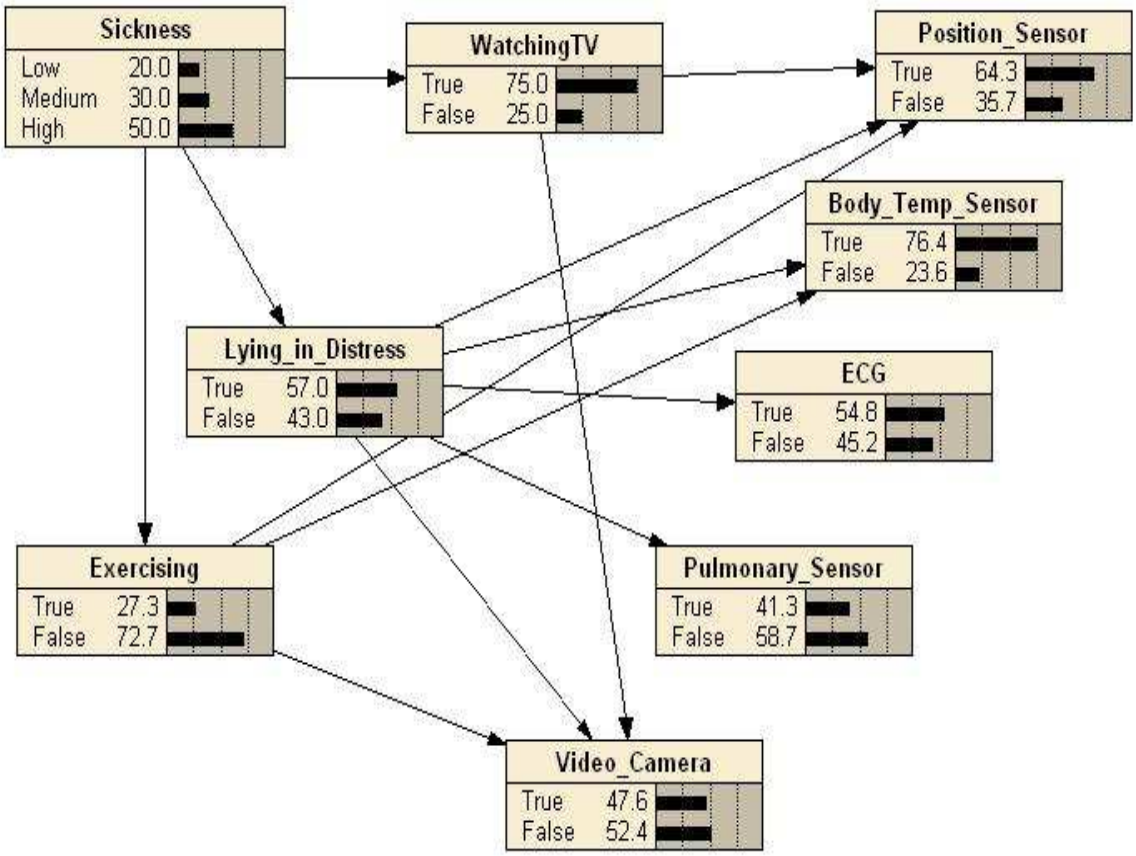

Figure 4.11. Bayesian Network.

using Netica BN software [84]. In this figure there is one situation space sickness to confirm with three context states – WatchingTV, Lying_in_Distress and Exercising. The sensors selected by ACMA for this application are Position_Sensor, ECG, Body_Temp_Sensor, Video_Camera and Pulmonary_Sensor. The conditional probability tables at a particular state of the application are also shown in Fig. 4.11. The numbers in this Figure are completely fictional, chosen for the purpose of explaining our scheme.

We conduct a series of experiments in the MavHome [118] on a group of three inhabitants in a smart home equipped with smart devices and wireless sensors. The

inhabitants wear radio frequency identification (RFID) tags and are tracked by RFID-readers. The house is equipped with explicit monitoring of inhabitants' activities and locations to get the context attribute values for performing a trace-driven simulation. We have developed an object-oriented discrete-event simulation platform for generating context attribute values, deriving context state and inferring the situation space using the DBN model. In order to collect the test data associated with the inhabitants' life-style as shown in Table 4.1, the appliances in the MavHome are equipped with $X10$ ActiveHome kit and HomeSeer [117], thus allowing the inhabitants to automatically control the appliances. The identity of the inhabitants, their locations and activities are captured by wireless sensors placed inside the home. The inhabitants wear the RF-tags, which are sensed by the RF-readers to gather their identities. The raw data [119] as shown in Table 4.1 is first parsed using parsing tools like Perl and Tcl to remove unnecessary information. The different column headings in Table 4.1 have the following meanings: Mark as the data and time stamp; Zone and Number as unique sensor zone identifier and sensor number within it; State as binary 'on' or 'off' of the sensor; Level as specific value if the sensor is on. Subsequently, we use these data to effectively select the set of sensors for an application performing context-aware data fusion and decision making.

4.6.1 Performance Results

The sensors used for our application are classified according to their numbers such as Sensor-1 (Position Sensor), Sensor-2 (Body_Temp_Sensor), Sensor-3 (ECG), Sensor-4 (Pulmonary Sensor), Sensor-5 (Video Camera), Sensor-21 (RFID 1) and Sensor-23 (RFID 2) as shown in Table 4.1. We have calculated the utility value for different combination of sensors with varying set size through the successive iteration of the ambiguous context mediation algorithm. The balance coefficient α is set to 1

Table 4.1. A Snapshot of the Collected RAW Data

Mark	Zone	Number	State	Level
$2005-01-03\ \ 09:47:30$	i	5	1	100
$2005-01-03\ \ 09:56:17$	i	5	0	0
$2005-01-03\ \ 13:04:45$	a	1	1	100
$2005-01-03\ \ 13:05:37$	i	3	1	100
$2005-01-03\ \ 13:06:11$	c	4	1	100
$2005-01-03\ \ 13:06:22$	c	4	0	0
$2005-01-03\ \ 13:16:32$	S	1	1	10
$2005-01-03\ \ 13:16:33$	S	2	1	152
$2005-01-03\ \ 13:16:33$	S	3	1	13
$2005-01-05\ \ 23:59:00$	V	23	1	100
$2005-01-05\ \ 23:59:01$	V	23	0	0
$2005-01-05\ \ 23:59:04$	V	21	0	0
$2005-01-05\ \ 23:59:12$	V	21	1	100
$2005-01-05\ \ 23:59:12$	V	21	0	0

Table 4.2. Utility Value without Considering Information Acquisition Cost

Time Stamp	Selected Sensor	Utility Position	Utility Temp	Utility ECG
1	$Temp$	0.1523	0.1671	0.0621
2	$Temp$	0.1712	0.1790	0.1472
3	$Temp$	0.1643	0.2438	0.2178
4	$Position$	0.2135	0.2078	0.1970
5	$Temp$	0.2073	0.2532	0.2239
6	$Temp$	0.2348	0.2791	0.1524
7	$Temp$	0.1921	0.3054	0.1832
8	$Position$	0.2476	0.2320	0.1645
9	$Temp$	0.2521	0.3226	0.1587
10	$Position$	0.2891	0.2721	0.1283

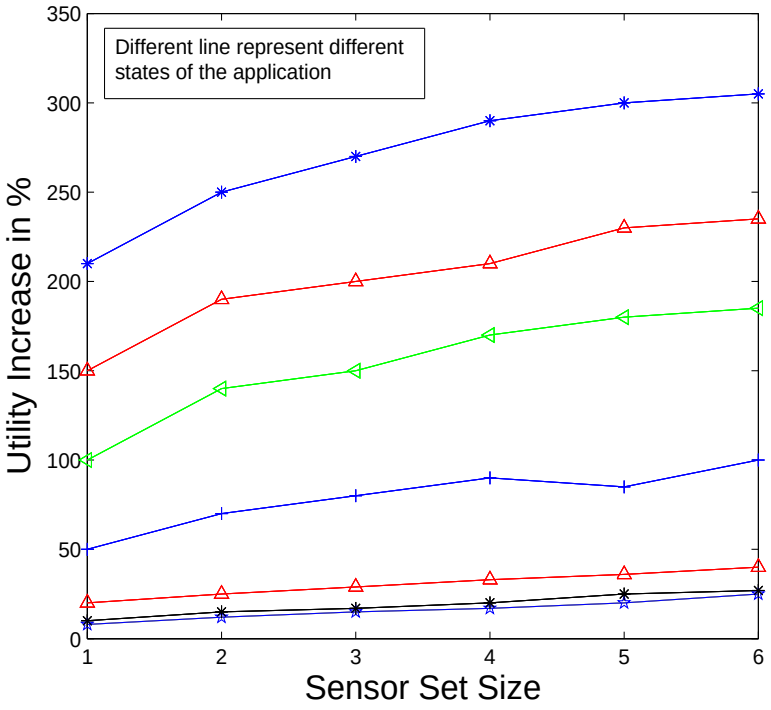

Figure 4.12. Reduction in Ambiguity for Different States of the Application.

to ignore the update cost of sensory information which in turn maps the utility value to the ambiguity reducing only. The different sets of sensors are as follows:

Sets of 1: $\{1, 2, \ldots, 21, 23\}$

Sets of 2: $\{1, 2\}, \{1, 3\}, \ldots \{21, 23\}$

Sets of 3: $\{1, 2, 3\}, \{1, 2, 4\}, \ldots \{5, 21, 23\}$

Sets of 4: $\{1, 2, 3, 4\}, \{1, 2, 3, 5\}, \ldots \{4, 5, 21, 23\}$

Sets of 5: $\{1, 2, 3, 4, 5\}, \{1, 2, 3, 4, 21\}, \ldots \{3, 4, 5, 21, 23\}$

Sets of 6: $\{1, 2, 3, 4, 5, 21\}, \{1, 2, 3, 4, 21, 23\}, \ldots \{2, 3, 4, 5, 21, 23\}$

From Fig. 4.12, we observe that the utility increases (reduces ambiguity) as the number of selected sensors increases for different states of the application. But the increase in utility (reduction in ambiguity) achieves a steady state after a cer-

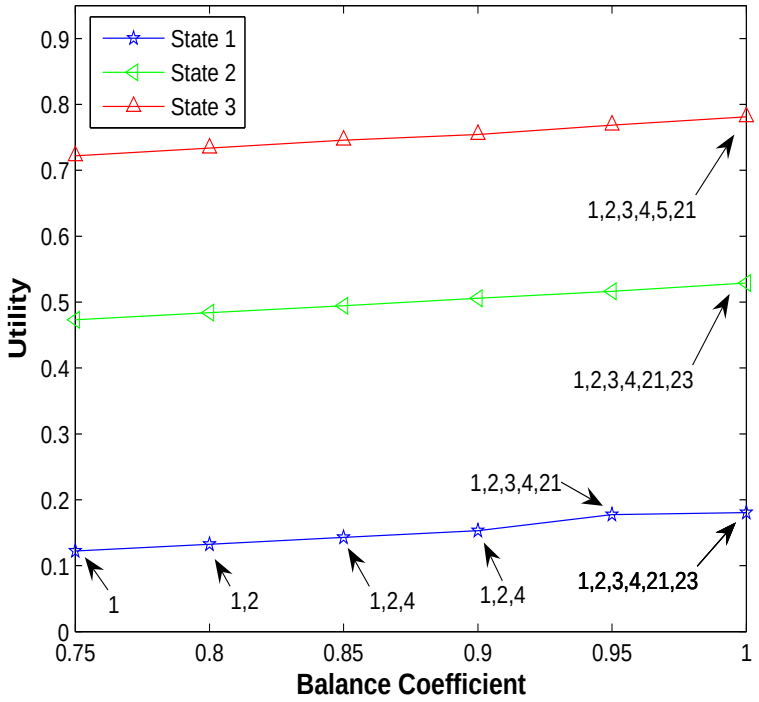

Figure 4.13. Best Sensor Set for Different Values of Balance Coefficient α.

tain sensor set size. The initial utility is calculated using Eqn. 4.2 considering a single sensor. The maximum utility values obtained by increasing sensor set size for three different states (different probability values) is shown in Table 4.3. This table demonstrates that ACMA dynamically selects the best set of sensors that gives the maximum utility. For example if the selected sensor set size is 2, the best set is $\{2, 21\}$ for application state 3 in Table 4.3, whereas if the sensor set size is 3, the best set is $\{3, 4, 5\}$. The best set of sensor varies from one application state to another. In states 1 and 2, the set $\{1, 2, 3, 4, 21, 23\}$ is the best set for the application. With different balance coefficient, the best set of sensors for an application having multiple states is also different as shown in Fig. 4.13. The above results also confirm the gain obtained by having more sensors exceeds the benefits of getting detailed information from each

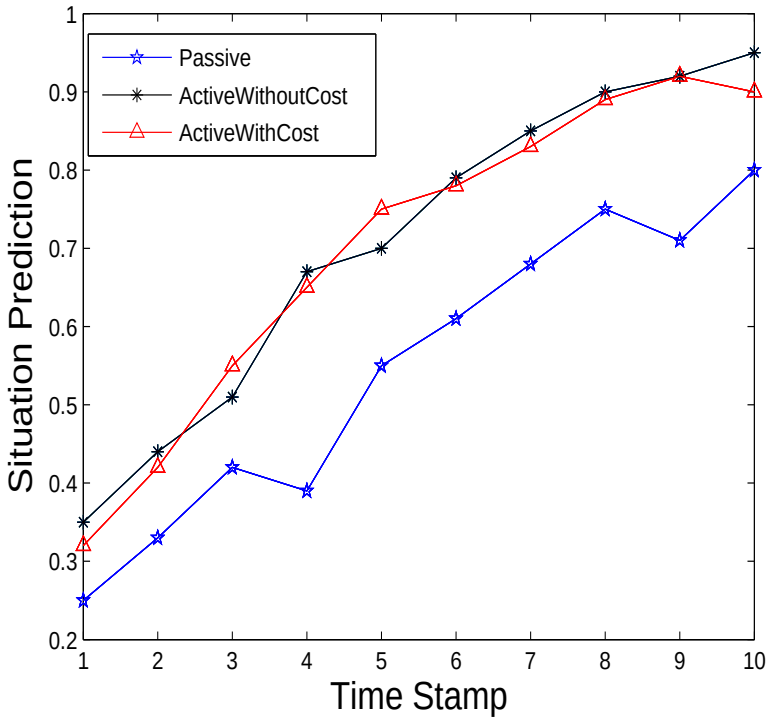

Figure 4.14. Situation Prediction Probability using Passive and Active Fusion.

individual sensor in accordance to our information theoretic analysis.

Next, we experimentally analyze the performance of active (context-aware) and passive (non context-aware) fusion in order to illustrate how the proposed active fusion system basically works. The time constraints do not allow the fusion system to activate all possible sensors in a passive manner during the continuous monitoring of the three elderly person in smart home. We therefore need to determine the best sensory action scheme to accelerate situation prediction. For situation determination where it is a general sickness type, we need the attributes that best characterize the sickness. The attributes may include location, time, body temperature, respiratory rate, blood pressure etc. We classified the person's body area sensor network as Thermoregulatory System (temperature Sensor), Cardiovascular System (ECG, pres-

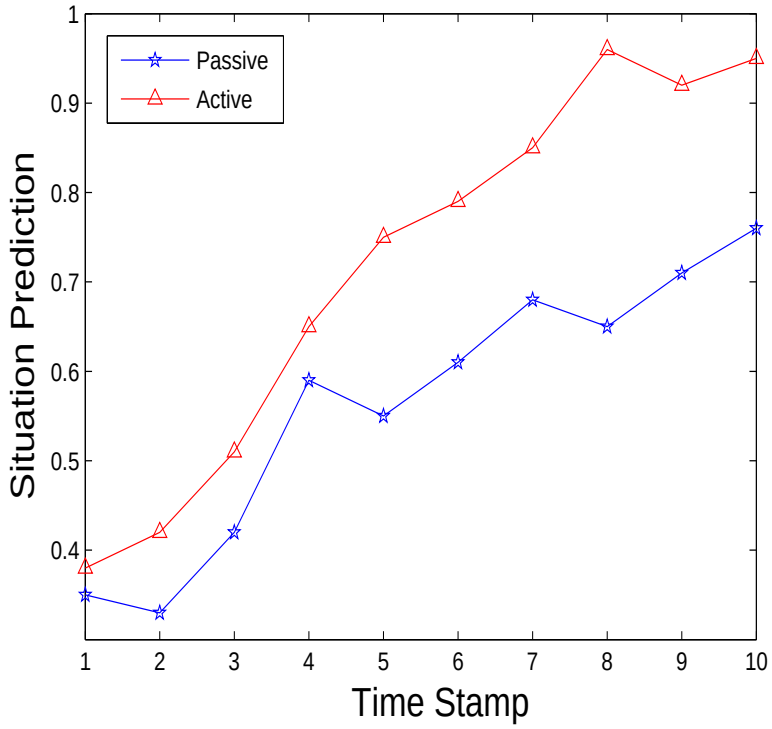

Figure 4.15. Situation Prediction Probability using Multi Sensor Fusion.

sure sensor), Pulmonary System (respiration sensor). Based on the expected utility computation, location and body temperature are the most important attributes. To obtain these information, we may need to activate the position sensors to determine the location and context state of the person and the temperature sensor from thermoregulatory system to determine the body temperature of the person. The choice of which sensor to activate depends on the expected utility of each sensor. After activating the sensor, the information we obtain on the location and body temperature of the person can help determine the situation type. If we still are not confident at the situation space type, we need to decide what will be the next sensor action at the next time stamp. This repeats until we identify the situation type with sufficient confidence.

Table 4.3. Best Sensor Set with Different Set Size for Different States of the Application ($\alpha = 1$)

Set Size	State 1 (Low) Sensor Set	Utility	State 2 (Medium) Sensor Set	Utility	State 3 (High) Sensor Set	Utility
1	$\{1\}$	0.17230	$\{2\}$	0.51310	$\{21\}$	0.76211
2	$\{1, 3\}$	0.17243	$\{2, 3\}$	0.51401	$\{2, 21\}$	0.76372
3	$\{1, 2, 4\}$	0.17283	$\{2, 3, 5\}$	0.51438	$\{3, 4, 5\}$	0.76578
4	$\{1, 2, 3, 5\}$	0.17302	$\{1, 2, 3, 21\}$	0.51578	$\{1, 2, 3, 5\}$	0.77270
5	$\{1, 2, 3, 4, 21\}$	0.17732	$\{2, 3, 4, 21, 23\}$	0.51632	$\{1, 2, 3, 4, 5\}$	0.77839
6	$\{1, 2, 3, 4, 21, 23\}$	0.18048	$\{1, 2, 3, 4, 21, 23\}$	0.51910	$\{1, 2, 3, 4, 5, 21\}$	0.78124

Table 4.4. Utility Value Considering Information Acquisition Cost

Time Stamp	Selected Sensor	Utility Position	Utility Temp	Utility ECG
1	*Position*	0.2143	0.2127	0.1371
2	*ECG*	0.2205	0.2245	0.2542
3	*Temp*	0.2304	0.2493	0.1789
4	*Temp*	0.2293	0.2356	0.1921
5	*Temp*	0.2367	0.2390	0.1940
6	*Position*	0.2598	0.2454	0.2122
7	*Temp*	0.2134	0.2610	0.2099
8	*Position*	0.2610	0.2439	0.1845
9	*Temp*	0.2391	0.2394	0.2257
10	*ECG*	0.2217	0.2280	0.2418

Table 4.2 and Table 4.4 convey how the required sensors got selected during the context-aware data fusion based on the context mediation algorithm. Table 4.2 gives the result without considering the information acquisition cost and Table 4.4 provides the result considering the information acquisition cost. Selection of multiple (two) sensors at each time stamp are shown in Table 4.5. The activation sequence for passive fusion has been generated randomly. We can observe from Table 4.2 that few sensors dominate compared to the others. This repetition of sensors accelerate the decision to be taken on situation space compared to the passive fusion as shown

Table 4.5. Multi Sensor Utility Value Considering Information Acquisition Cost

Time Stamp	Selected Sensor	Utility Position	Utility Temp	Utility ECG
1	$Temp - Position$	0.1723	0.2393	0.1213
2	$Position - ECG$	0.2421	0.1987	0.2193
3	$Temp - Position$	0.2187	0.2267	0.2105
4	$Position - ECG$	0.2378	0.2198	0.2237
5	$Temp - ECG$	0.2303	0.2498	0.2378
6	$Temp - Position$	0.2421	0.2590	0.2376
7	$Temp - ECG$	0.2204	0.2653	0.2393
8	$Temp - Position$	0.2535	0.2843	0.2487
9	$ECG - Position$	0.2810	0.2703	0.2906
10	$Position - Temp$	0.3034	0.2979	0.2852

in Fig. 4.14. But sometime it leads to information redundancy if it repeats the same value of the attribute consecutively. However, it may be beneficial for reducing the imprecision and increasing the reliability.

Active sensors alternate more frequently in Table 4.4 when the acquisition cost has been considered. But Fig. 4.14 shows no significant performance difference by considering the acquisition cost. So the information redundancy can be overcome due to the frequent alternate between the active sensors with almost same performance gain. Fig. 4.15 represents the similar performance graph as two sensors are activated simultaneously. So we can conclude that context-aware data fusion (active) outperforms the context-non-aware data fusion (passive) to predict the situation space in terms of time spent.

Fig. 4.16 represents the confidence of situation prediction for different specified QoC constraints. The confidence level achieves the higher value for a rigid QoC constraint compared to a loosely coupled system. Though we got a better confidence level for a tight QoC constraint, more uniformity is achieved at the loosely bound

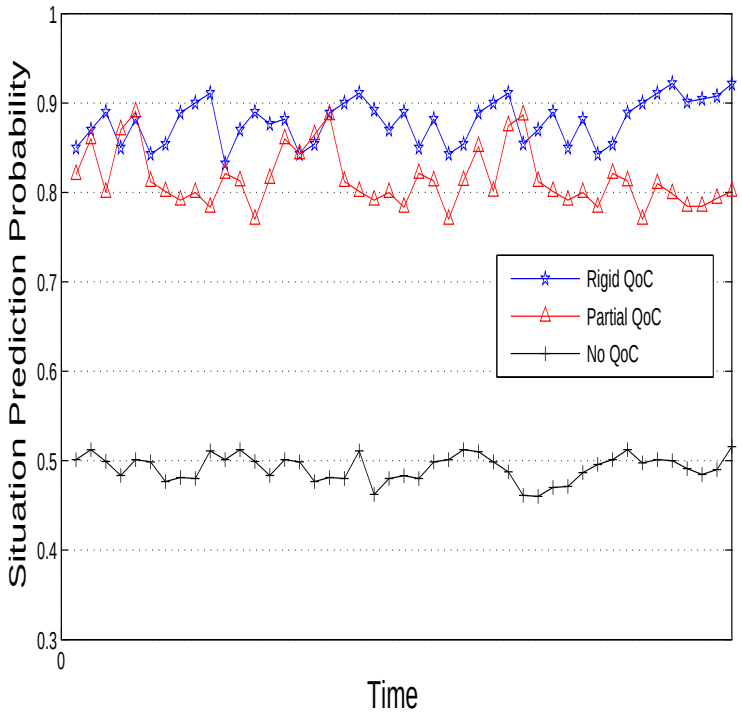

Figure 4.16. Variation of Situation Prediction Probability using QoC Constraint.

system. This observation confirms the participation of more sensors during the non-rigid QoC bound fusion process yields a more stable value though fails to achieve the higher confidence gain. Next we examine the situation prediction when we selectively choose the different sensors using the context mediation algorithm. Fig. 4.17 depicts the variation of situation prediction with different set of context attributes from different sensors. In the first scenario, all context attributes are fused following the specified algorithm according to their QoC specification. In the second scenario, values are only partially satisfied due to their inherent inaccuracy and experimental settings. The fusion of selective context attribute yields a better results compared to the non-selective one.

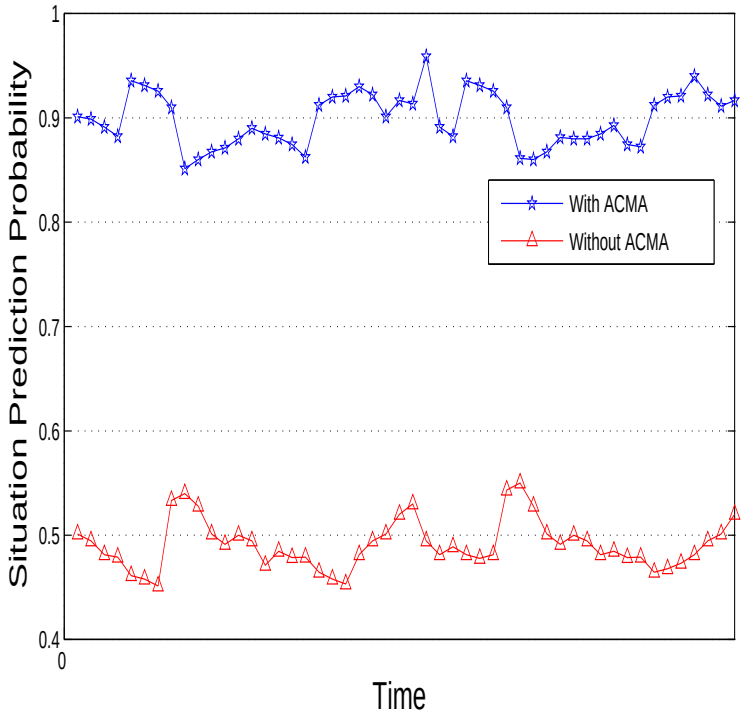

Figure 4.17. Variation of Situation Prediction Probability using Sensor Values.

4.7 Summary

This chapter presents a framework which supports ambiguous context mediation and user-centric situation prediction based on dynamic Bayesian networks and information theoretic reasoning, leading to context-aware healthcare applications in smart environments. Our framework provides a Bayesian approach to fuse context fragments and deal with context ambiguity in a probabilistic manner, an information theoretic approach to satisfy context quality within the application, and a semantic web technology to easily compose rules to reason efficiently to mediate ambiguous contexts. An algorithm is proposed and subsequent experimental evaluation is done to perform cost/benefit analysis to engage the most economically efficient actions for context disambiguation in resource constrained sensor environments. In next chapter

we discuss an efficient Quality-of-Inference (QoINF)-Aware context determination in pervasive care environments.

CHAPTER 5

QUALITY-OF-INFERENCE (QoINF)-AWARE CONTEXT DETERMINATION FRAMEWORK

5.1 Introduction

Remote medical monitoring of elderly individuals and chronically ill patients is widely perceived as an emerging transformative technology for healthcare delivery. In particular, we are already beginning to witness commercial activity (e.g., [2]), centered on remote monitoring within 'smart assisted-living homes', using a combination of body-worn medical and non-medical sensors (e.g., SpO2 monitors and accelerometers) and *in-situ* sensors (e.g., thermal and motion detectors). A key use of the retrieved sensor data involves the automated determination of a person's activity or medical *context* from the raw sensor data values where such context is exploited by many monitoring-based applications (e.g., alerting a first responder if the individual is judged to be 'sleeping for an abnormal period' or 'lying immobile after a sudden fall').

The determination of a specific context attribute may be viewed as an inference or estimate obtained by fusing the values from multiple sensor data streams. Much of the context-aware computing literature has focused on the questions of how *a)* the mapping from sensor readings to appropriate context states may be automatically computed, or *b)* empirically establishing whether the accuracy of the inferred context is high enough to enable automated context-based adaptation. The implicit assumption in such work is that the *quantity of the instrumented sensor data is invariant*;

accordingly, the primary goal is to determine the best-possible context estimate, given an underlying set of sensor data streams.

This chapter introduces a somewhat **opposing** perspective [100]: given the constraints on device battery capacities and communication capacity (especially for low-power wireless standards such as IEEE 802.15.4(Zigbee)) typical of many assisted-living environments, the goal should be to *reduce the volume of sensor data that needs to be actually collected with the minimal overhead to assure sufficiently high accuracy of the estimated context.* This approach explicitly highlights the 'energy/bandwidth cost associated with context-based computing, and is based on the observation that different applications require their context estimates to varying levels of accuracy (statistical confidence).

We suggest a formal approach to minimum-cost (cost defined in terms of metrics such as energy or bandwidth), *continuous* determination of an individual's *context* in smart environments. Our framework presupposes the use of an *event-driven data framework*, where each individual sensor is associated with a *tolerance range*, indicating the amount of imprecision that can be tolerated by the monitoring application. Individual sensors employ a 'dead-reckoning' based communication strategy, communicating their data samples only when their deviation from an estimated value exceeds the specified tolerance range. Thus a larger tolerance range results in a reduction in the sensor reporting rate.

Central to our model is the notion of a *'Quality of Inference' (QoInF)* specification, defined as the *error probability in estimating a context state, given the imprecision in the values of the contributing sensors.* There are two main observations driving our current work:

- Smart environments typically contain several sensors, with a particular activity context capable of being estimated to varying degrees of accuracy via data

from different sensors. More importantly, the *accuracy of the inferred context increases with the use of a progressively larger sensor set* (often with different modalities). As a simple example, a combination of data from a body-worn accelerometer and ceiling mounted motion sensors provides a more accurate estimation of whether 'a person is immobile after a fall', compared to deductions based solely on each individual sensor.

- The quality of the inferred context is not just a function of the chosen sensors, but also of the permitted inaccuracy in the sensor values; in general, the larger the uncertainty in the precise value of a data sample, the lower the inferencing accuracy. There is, effectively, a *tradeoff between the energy overheads of monitoring and the achievable QoINF value*.

Broadly speaking, this chapter advocates the development of a formal methodology for answering the following question:

> Given an application-defined specification of a minimal acceptable QoINF value, how do we compute *both* the *optimal set of sensor data streams* that are needed for inferencing, and the *optimal tolerance ranges permissible for each selected sensor?*

5.1.1 Related Work

The tradeoff between communication overhead and the quality of the reconstructed data was first studied in [86], which envisioned the effect of tolerance ranges on the relative frequency of sink-initiated fetching vs. source-initiated proactive refreshes. The focus, however, is on snapshot queries and not on *continually satisfying* the QoINF bound of a long-standing subscription. The idea of exploiting temporal correlation across the successive samples of individual sensors for reducing the communication overhead, for snapshot queries, is addressed in [30], which used training

data to parameterize a jointly-normal density function. While a precursor to our work, the focus there was on meeting the QoINF requirements for a class of 'aggregation queries', whereas our focus is on arbitrary relationships between a context variable and its underlying sensor data. The CAPS algorithm [56] is designed for long-running aggregation queries (such as $\{min,\ max\}$) and computes the optimal set of tolerance ranges for a given set of sensors, that minimizes communication overhead while guaranteeing the accuracy of the computed response. However, our aim is to compute both the best subset of available sensors, and their tolerance ranges, that achieves the desired accuracy for arbitrary context variables.

5.1.2 Contributions

This chapter makes the following key initial contributions towards developing a flexible formalization for efficient context inferencing in assisted-living environments.

- It proposes that the problem of minimum cost context estimation be quantitatively defined using a generic inference 'quality' function that captures the relationship between QoINF and the set of sensors (and their tolerance range) used. Such an inference function generalizes earlier work (e.g., [30, 56]) that focused on specific aggregation queries over sensor data.
- Besides presenting the mathematical optimization approach, we also suggest how a practical framework for computing the QoINF function may be realized using *machine learning over sensor data* and provide initial experimental evidence that a QoINF-aware approach to context extraction may indeed provide significant cost savings.

The rest of this chapter is organized as follows. Section 5.2 defines the concept of context state estimation and defines the notion of a QoINF functional specifies using a motivating example. Section 5.3 then defines the selection of sensors and

associated tolerance ranges as a non-linear optimization problems and presents a Lagrangian-based optimization algorithm. Subsequently, Section 5.4 describes how a QoINF function may be computed in practice, by employing past observations. Section 5.5 presents early results using SunSPOT sensors to investigate the sort of accuracy vs. overhead tradeoffs that may be realized. While Section 5.6 summarizes this chapter.

5.2 Context Inference and the QoINF Model

An important component of automated and proactive health monitoring technologies is the computation of specific *context variables*, based on an underlying set of sensor samples. To precisely quantify the concepts, we assume an underlying set of S sensors; let $v_i(t)$ represent the 'value' of the i^{th} sensor at time t. Also, let Λ_i represent the range of feasible values of sensor s_i. The act of determining the value of a context variable C may be viewed as a multi-dimensional mapping function $f_C(.)$, that takes as input the values from a set of sensors $\theta : \ \theta \subset S$ and maps them onto the state space (denoted by Λ_C) of the output context. Mathematically:

$$f_C(\Upsilon) : \prod_{i \in \theta} \Lambda_i \Rightarrow x : \ \ x \in \Lambda_C. \tag{5.1}$$

Different values of the same context may be inferred to varying degrees of 'accuracy' by employing different sets of sensors. In general, one may thus associate an accuracy function $QoINF_C(\theta(S))$, representing the *average accuracy* in estimating the context C based on the values for the sensors in the $\theta(S)$. To make our definitions more specific, we define $QoINF(.)$ to be the 'one minus the average estimation error resulting from the inferencing function $f_C(\theta(S))$, i.e.,

$$QoINF_C(\theta(S)) = 1 - \sum_{x \in \Lambda_C} p(i) * err_C(x, \{s_i \in \theta(S)\}), \tag{5.2}$$

where $err_C(x, \{s_i \in \theta(S)\}))$ is the probability of error, given accurate sensor readings from the sensor set $\theta(S)$, when the individual's value for context variable C is actually x. In general, alternative definitions of accuracy (e.g., estimation error subject to maximal 'false alarm' rates) are possible; such definitions may, for our purposes, be viewed merely as alternative definitons of $QoINF(.)$. If the state space of context C is discrete (e.g., a choice between the values {walking, sitting, sleeping}, then the estimation error is computed by the normalized number of *wrong* inferences made by the 'best possible estimator'.[1]

Figure 5.1 illustrates this notion of multiple sensors and the accuracy (or QoINF) value associated with different context variables. For example, as shown in Fig. 5.1, the activity state of an individual may be computed with an inferencing accuracy of 0.9 (i.e., with 10% error rate) using data from a respiratory sensor, but only with 0.8 accuracy using data from a low-quality ECG sensor. However, by fusing the data available from respiratory, ECG and accelerometer sensors, we can achieve an inferencing accuracy of 0.95 (i.e., only a 5% error rate).

5.2.1 Role of Tolerance Ranges in Context Estimation Errors

The model above, illustrated in Figure 5.1, can capture the relationship between context inferencing quality and the *choice of sensors*. Such a model does not, however, completely capture the continuous event-driven remote monitoring scenario, where the value of a sensor sample may be known to different *degrees of precision*

[1]In contrast, if the context state space is continuous, the error function also depends on the tolerance in the computed output state (e.g., it may be acceptable for an inferred location context to be inaccurate by ±3 feet). Due to reasons of space, we do not delve into these technicalities, which all relate to the precise specification of the $err(.)$ function. Our focus here is on how to exploit a *given function* $err(.)$; for the actual determination of $err(.)$; it suffices to note that there exists an extensive body of literature on optimal estimators and the resulting lowest possible error bounds.

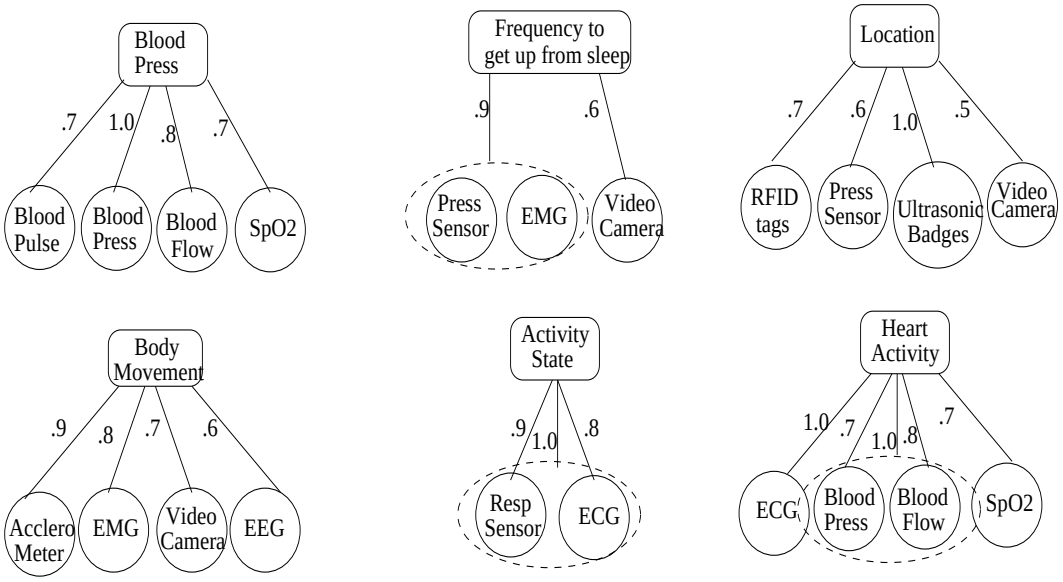

Figure 5.1. Impact of Choice of Different Sensor Subsets on QoINF (No Consideration of Tolerance Range).

for different sensors. In particular, as demonstrated by earlier research in [30, 56], one may achieve significant reduction in the communication and energy overheads of sensing, by providing even small, non-zero tolerance ranges to individual sensors.

In general, the accuracy of context inferencing should decrease with increasing uncertainty in specific sensor values. For example, the estimation of the "blood pressure" context in Figure 5.1 will certainly be less accurate if the SpO2 sensor tolerance range is $\pm 20\%$ (indicating that the true reading may be up to 20% higher or lower than the reported value), as opposed to a tolerance range of, say, 5%. To capture this additional 'degree of freedom', we now denote the selection of a particular sensor s_i via the symbol $s_i(q_i)$, where q_i denotes the tolerance range assigned to s_i. In other words, the infrastructure is now free to not just select a sensor s_i, but also assign a tolerance range q_i to it.

We can then represent the error associated with a specific choice of sensors $s(C)$ through the modified dependency relationship:

$$QoINF_C(\theta, q_\theta) = 1 - \sum_{x \in \Lambda_C} p(i) * err_C(x, \{(s_i, q_i) : \ s_i \in \theta(S)\})$$

The above equation expresses the fact the accuracy of context sensing is a function of *both* the sensors chosen and the tolerated amount of inaccuracy in each sensor's value.

5.2.2 Context Sensing Architecture

Figure 5.2 shows our vision of the high-level functional components of a context-determination service. External applications (e.g., activity monitoring applications deployed by stakeholders such as wellness professionals) subscribe to a specific context C, indicating a *minimally acceptable* QoINF value, $QoINF_{min}$. (Of course, monitoring applications may change their $QoINF_{min}$ specification in response to external 'situational awareness', thereby triggering another optimization cycle). The *Context Optimizer* component must then determine the best (least-cost) combination of sensors and tolerance ranges that can meet this QoINF threshold. To perform this optimization, it requires the $QoINF_C(.)$ function that is computed by the *Context Modeler* component. Finally, the transmitted sensor data samples are received by the *Context Estimator* component, which provides continuous updates on the subscribed context variable.

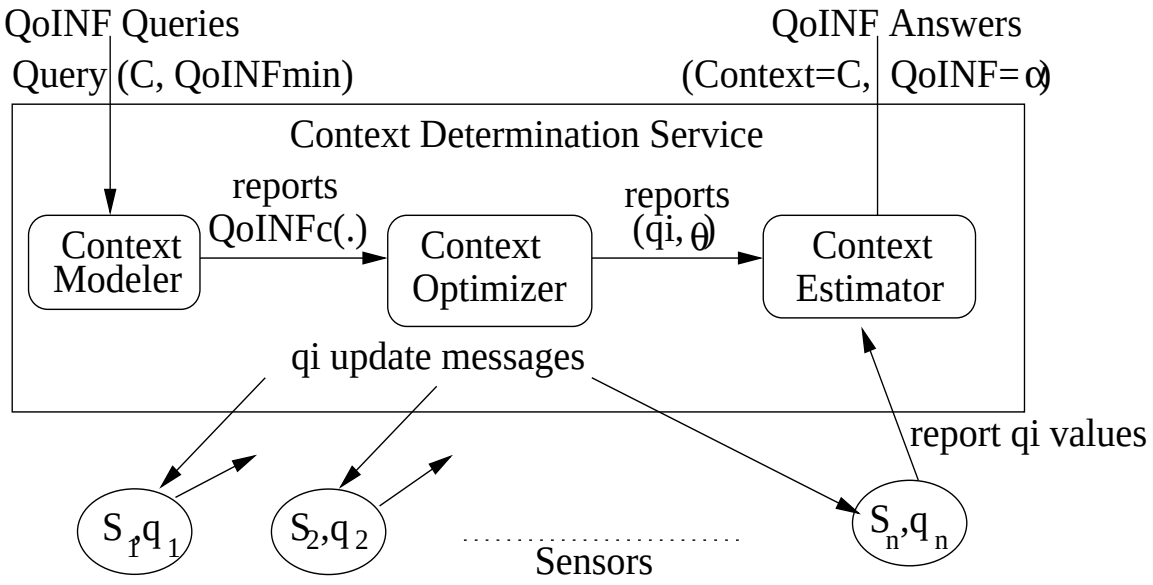

Figure 5.2. Overall Architecture of QoINF-Aware Context Determination.

5.2.3 Minimum-Cost QoINF-Aware Problem

Note that the least-cost objective can be expressed in terms of a variety of cost metrics–the most common metrics, of course, are either transmission bandwidth (volume or frequency of transmissions) or communication energy. In general, this cost will be a function of the sensors involved in the sensing and their associated tolerance ranges. Mathematically, given a sensor set θ, the cost can be expressed as:

$$COST(\theta, q_\theta) = \sum_{i \in \theta} c_i(q_i), \tag{5.3}$$

where the cost $c_i(.)$ associated with sensor s_i is a function of its assigned tolerance range q_i.

Given this formulation, the *best* sensor selection and parameter scheme (given a minimum context inferencing accuracy $QoINF_{min}$), denoted by $(\hat{\theta}, \hat{q_\theta})_{QoINF_{min}}$, is mathematically given by the optimization problem:

MIN-COST-INFERENCE:

$$(\hat{\theta}, \hat{q_\theta})_{QoINF_{min}} = \underset{\theta \subset S, q_\theta}{\arg\min} \; COST(\theta, q_\theta)$$
$$\text{such that} \quad QoINF_C(\theta, q_\theta) \geq QoINF_{min} \tag{5.4}$$

5.3 QoINF Cost Optimization

Our computing infrastructure consists of a declarative query processing engine that takes application bounds on $QoINF_{min}$ as input and optimally 'tasks' the individual sensors to provide the necessary inputs to a context estimator engine. The actual optimal parameters will depend on the specific cost (e.g., energy or reporting frequency) that we seek to minimize and the structure of the $QoINF(.)$ function. In this section, for specificity, we focus on minimizing a measure of the average *communication overhead* and provide insights into the resulting algorithms.

5.3.1 Average Reporting Cost Optimization

One natural cost to optimize is the communication overhead incurred by the sensor in reporting its values to the Context Estimator component. Let us denote the average update cost (communication overhead) of sensor s_i, given a tolerance range q_i as $c_i(q_i)$. Intuitively, c_i is a decreasing function of q_i, since the communication cost would be higher (more frequent reports) for smaller tolerance ranges. In a setting where the sensor data traverses multiple hops to get to the Context Estimator Engine, the update cost is also proportional to $\hbar_i$, the length of the uplink path from sensor s_i to the Aggregation Engine.[2] As shown in [56], *if the underlying data samples evolve*

[2]It is easy to generalize the model to other cost formulations–e.g., where the cost for s_i is weighted inversely proportional to its residual battery capacity.

as a random-walk model, we have $c_i(q_i) \propto \frac{\hbar_i}{q_i^2}$. In this case, the resulting cumulative cost function is given by:

$$COST(\theta, q_\theta) = \kappa * \sum_{i \in \theta} \frac{\hbar_i}{q_i^2} \tag{5.5}$$

where κ is a scaling constant and $\hbar_i$ is the hop count.

If the set of sensors to be used, i.e., the set θ is given, then the problem of optimally computing the q_is can be represented by the Lagrangian:

$$\text{minimize} \sum_{i \in \theta} \frac{\hbar_i}{q_i^2} + \lambda \times \left[QoINF_C(q_1, q_2, \ldots, q_\theta) - QoINF_{min} \right]. \tag{5.6}$$

Finding an exact solution to Equation 5.4, for any arbitrary $QoINF(.)$ is an NP-complete problem [30]. For the general case of a function $QoINF(.)$, the only solution to determine the most optimal set of sensors (i.e., $\hat{\theta}$) is to iterate over all the $2^S - 1$ elements of the power-set of S.

5.3.2 Suggested Optimization Heuristic

While a completely arbitrary $QoINF(.)$ function requires a brute-force search, there are certain forms of $QoINF(.)$ that prove to be more tractable and lend themselves to more efficient optimization heuristics. In particular, a particularly attractive case occurs when the i^{th} sensor's individual $qoinf(.)$ is represented by an 'Inverse-Exponential' distribution of the form

$$qoinf(i) = 1 - \frac{1}{\nu_i} exp(\frac{-1}{\eta_i q_i}), \tag{5.7}$$

where η_i and ν_i are sensitivity constants for sensor s_i. A larger value of ν_i indicates a lower contribution from sensor s_i to the inference of context C. Moreover, for a selection of θ sensors, the resulting $QoINF(.)$ function is modeled as:

$$QoINF_C(\theta) = 1 - \prod_{i \in \theta} (1 - qoinf_C(i)) \tag{5.8}$$

The above equation satisfies three of the key properties a $QoINF(.)$ function must have: **a)** $0 < QoINF_C(.) < 1\ \forall\theta$, **b)** $QoINF_C()$ is non-decreasing in θ (in other words, incorporating data from an additional sensor should not lead to a reduced QoINF value, the other sensors' q_i values remaining the same, and **c)** $\text{qoinf}_C(i) \rightarrow 0$ as $q_i \rightarrow \infty$.

Now, the first-level optimization problem is to choose the values of $q_1, q_2, \ldots, q_\theta$, *given a set* θ, such that we minimize the total update cost while ensuring that the minimum accuracy level is achieved. Mathematically,

$$\textbf{min}\ COST(\theta, q_\theta)\ \textbf{subject to}\text{: } QoINF_C(\theta) \geq QoINF_{min}. \tag{5.9}$$

We solve this by taking the Lagrangian form of the constraints, i.e, solve for

$$\text{minimize} \sum_{i\in\theta} \frac{\hbar_i}{q_i^2} + \lambda \left[1 - \prod_{i\in\theta} [\frac{1}{\nu_i} exp(\frac{-1}{\eta_i q_i})] - QoINF_{min}\right]. \tag{5.10}$$

Lemma 3 *If the* $QoINF(.)$ *function for any set of sensors* θ *follows the form specified by Equations 5.7 and 5.8, then the optimal choices of the* q_i*s that minimize the cost function follow the relationship:*

$$\frac{2 * \eta_1 * \hbar_1}{q_1} = \frac{2 * \eta_2 * \hbar_2}{q_2} = \ldots = \frac{2 * \eta_\theta * \hbar_\theta}{q_\theta}, \tag{5.11}$$

and the optimal value of q_i*, denoted by* $\hat{q}_i$ *is given by:*

$$\hat{q}_i = \frac{\hbar_i * \eta_i * \left[\sum_{i\in\theta} \frac{1}{\hbar_i * \eta_i^2}\right]}{log(1 - QoINF_{min}) + \sum_{i\in\theta} \nu_i} \tag{5.12}$$

Moreover, for a given θ*, the minimal cost to achieve the specified inference accuracy is given by:*

$$\widehat{COST}(\theta) = \frac{\left[log(1 - QoINF_{min}) + \sum_{i\in\theta} \nu_i\right]^2}{\sum_{i\in\theta} \frac{1}{\hbar_i * \eta_i^2}} \tag{5.13}$$

Proof: The above expression follows immediately by taking partial derivatives of the Lagrangian in Equation 5.10 and setting them to 0. The details are omitted here due to space constraints.

5.3.2.1 Search Heuristic for the Best θ

In addition to finding the minimum cost for a given θ, we also need to determine the 'best' θ (i.e., the subset that minimizes the overall update cost). Clearly, one solution is to iterate through all possible combinations, computing $\widehat{COST}(.)$ for each combination of sensors. However, for efficient operation, we now propose a selection heuristic that is only linear in the number of sensors.

The heuristic is based on the observation that the 'additional cost (increase or decrease)' (based on Equation 5.13) in adding a sensor s_x to an existing set θ is, roughly speaking, dependent on the term $log(\nu_x) * \hbar_x * \eta_x^2$; this can be more rigorously derived by considering the limiting case when $QoINF_{min} \rightarrow 1$. A lower value of this term indicates a greater preference for selecting a sensor (ideally, a sensor has a small $\hbar_i$ (small update cost)) and small η, ν terms (indicating a smaller degradation in $QoINF$ with increasing q_i). Accordingly, the selection heuristic sorts the available sensor set S in ascending order of this term, and keeps adding additional sensors until the overall cost either increases or the percentage decrease in cost falls below a specified threshold. Figure 5.3 shows the pseudocode for our proposed search heuristic.

5.4 Techniques for Deriving $QoINF_C(.)$

One of the main challenges in the application of our suggested formalism is the establishment of appropriate $QoINF_C(.)$ functions for specific context variables. Indeed, much of the work on utility-based context models has failed to achieve the desired impact due to the difficulty of computing useful utility functions. To overcome this challenge, we propose to employ *statistical learning or regression techniques* to construct the $QoINF(.)$ functions from the empirically observed data. Such statis-

```
Procedure Selective_Fusion(input set S, QoINF_min)
1.  Initialize an empty set of sensors; Q = φ;  MinCost = ∞;
2.  Sort the sensor set θ into a list L in
increasing order of sensitivity term log(ν_x) * ħ_x * η_x^2;
3.  For (i = 1; i < |S|; i++)
4.       θ = θ + L(i); //set-theoretic addition
5.       Compute the optimal update cost COST^(θ) for QoINF_min
5.       if (COST^(θ) − MinCost) > 0 OR below threshold
6.          break;
7.       else MinCost = COST^(θ).
7.  End-For
8.  return {θ, COST^(θ)}.
```

Figure 5.3. Proposed Sensor Set Selection Heuristic.

Table 5.1. Calibrated Accelerometer Sample Values for Different Context State

Range($5-95$th percentile) of Tilt Values (in degree)	Context State
85.21 to 83.33	*Sitting*
68.40 to 33.09	*Walking*
28.00 to -15.60	*Running*

tical techniques are needed to provide the necessary robustness in the face of sensor errors (due to noise and miscalibration) and incomplete data (due to network losses).

If a particular form of the function $QoINF(.)$ *is assumed*, we can use least-squares regression or any other appropriate statistical technique to estimate the best values for the functional coefficients. To obtain the necessary 'samples' during the 'training phase', we need to obtain both the true 'context state' as well as the corresponding sensor readings for various values of the tolerance range q_i. The true context state may either be obtained from explicit user feedback or implicit user actions (e.g., [71]).

5.5 Experimental Components and Evaluation

To develop some initial knowledge of the interplay between tolerance ranges $\{q_i\}$ and the resulting inferencing error, we have experimented with several Sun SPOT [112] (Sun Small Programmable Object Technology (SPOT) devices. The Sun SPOT sensor board contains a 3-axis accelerometer (with two range settings: 2G or 6G), a light sensor and a temperature sensor. For our initial studies, we have utilized readings from the accelerometer (to estimate the motion and orientation of the SPOT) and the light sensor.

5.5.1 Empirical Determination of Context Estimates

We used the accelerometer to measure tilt value of the Sun SPOT (in degrees) when the monitored individual was in three different context states: *sitting, walking and running*. From the collected samples, we computed the 5^{th} and 95^{th} percentile of the tilt readings, corresponding to each state. Table 5.1 shows the resulting ranges in the accelerometer tilt readings observed for each of the three states. The results indicate that there is an observable separation in the ranges of the tilt values for the three different states. This suggests that the states can be distinguished reasonably accurately even under moderate uncertainty in the sensor's readings.

Similarly, we also used the Sun SPOT light sensor to measure the light level for different user contexts. Intuitively, low values of ambient light intensity may be indicative of a *'sleeping'* state, while higher values of light intensity are likely to result when the individual is *'active'*. Table 5.2 shows the observed ranges for the light values for each of these two states. The accuracy of context from the light sensor is, however, much lower, as users may often be inactive (e.g., sitting), even under high illumination.

Table 5.2. Light Sensor Values (lumen) for Different Context State

Avg. Range of Light level (lumen)	Context State
LightSensor.getValue() = 10 to 50	Turned on → active
LightSensor.getValue() = 0 to 1	Turned off → sleeping

5.5.2 Measurement of QoINF Accuracy & Sensor Overheads

To study the potential impact of varying the tolerance range on each sensor and the resulting tradeoff between the sensor reporting overhead, we collected traces for the SunSPOT motion and light sensors for a single user who engaged in a mix of three different activities (*sitting, walking and running*) for a total of ≈ 6 minutes (2000 samples at $5.5Hz$). We then used an emulator to mimic the samples that a sensor would have reported, given the trace, for a given q, and compared the context inferred from the values reported by the emulation against the ground truth. Figure 5.4 shows the resulting plots for the 'total number of samples reported' (an indicator of the reporting overhead) and the corresponding QoINF (defined as *1 - error rate*) achieved, for different values of the tolerance range (q_m) for the motion sensor. Figure 5.5 plots the corresponding values vs. the tolerance range (q_l) for the light sensor.

As the figures demonstrate, there is, in general, a continuous drop in the reporting overhead and the QoINF as q increases. However, as seen in Figure 5.4, a QoINF accuracy of ≈ 80% is achieved for a modestly large q value of 40; moreover, using this tolerance range reduces the reporting overhead dramatically by ≈ 85% (from 1953 → 248). This suggests that it is indeed possible to *achieve significant savings in bandwidth, if one is willing to tolerate marginal degradation in the accuracy of the sensed context.* A similar behavior is observed for the light sensor ($q = 4$ incurs a 5% loss in accuracy vs. a ≈ 65% reduction in reporting overhead). However, as the

difference between the lumen ranges for *Active* vs. *Sleeping* is only ≈ 10 (Table 5.2), increasing q beyond ≈ 10 leads to a sharp fall in the QoINF.

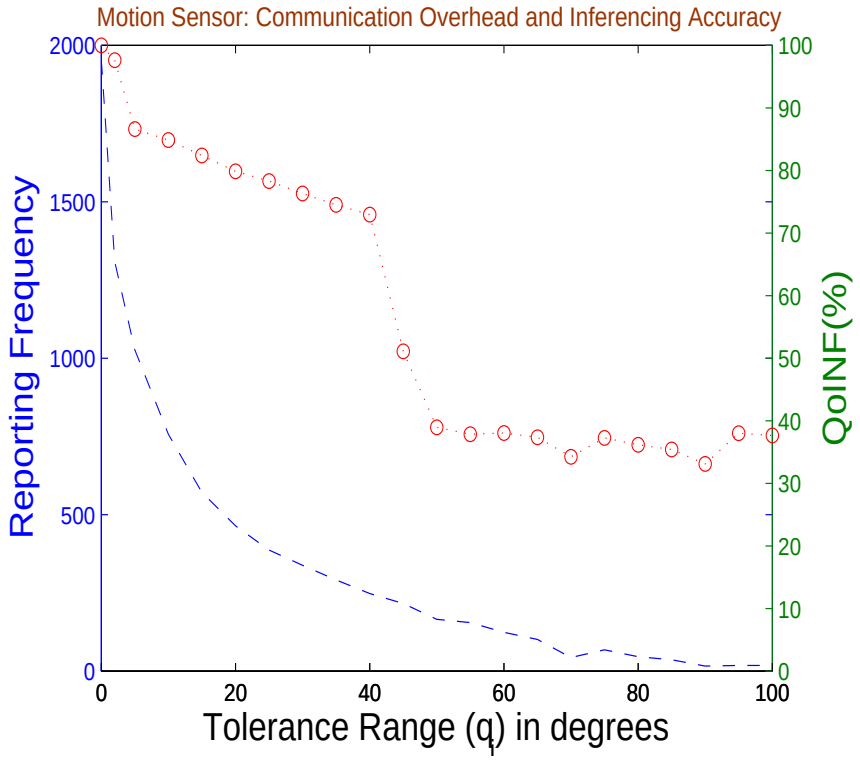

Figure 5.4. Communication Overhead & Inferencing Accuracy vs. Tolerance Range using Motion Sensor.

5.5.3 The Benefit of Joint Sensing

We also investigated how the use of readings jointly from both sensors affects the inferencing accuracy vs. tolerance ranges. We consider the individual to be in a *sitting, walking or running* state whenever the motion sensor tilt values lie within the corresponding range **AND** the light sensor values indicate an *active* state. Figure 5.6 uses a three-dimensional plot to illustrate the observed inferencing accuracy when the tuple (q_m, q_l) is jointly varied; we see how the QoINF is now less susceptible to

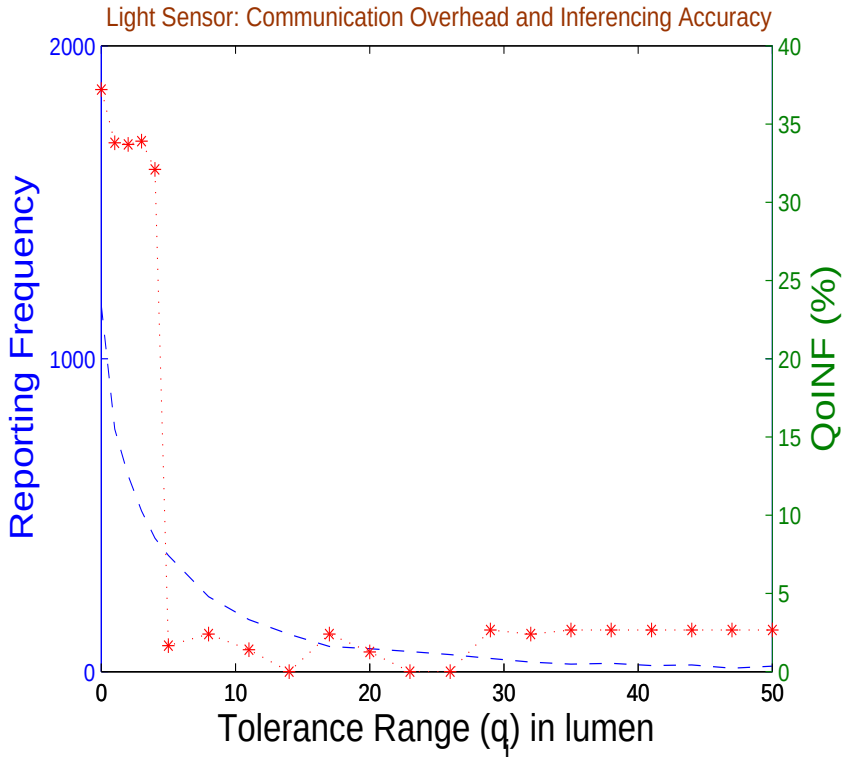

Figure 5.5. Communication Overhead & Inferencing Accuracy vs. Tolerance Range using Light Sensor.

individual q variations. Figure 5.7 confirms this benefit by plotting the QoINF vs. q obtained using just the light sensor against that obtained by using both sensors (the q ranges of both being identical). Clearly, the QoINF obtainable from the combination of the two sensors is much higher than that of a single sensor.

5.5.4 Next Steps and Ongoing Work

The initial results above provide preliminary evidence that significant savings that may be achieved, by relaxing the tolerance range for each sensor, without compromising the accuracy of context estimation. However, significant work remains to validate and quantify the benefits of our proposed approach. To apply these results to our proposed, formal $QoINF$ model, we are now working to fit the $QoINF(.)$ vs

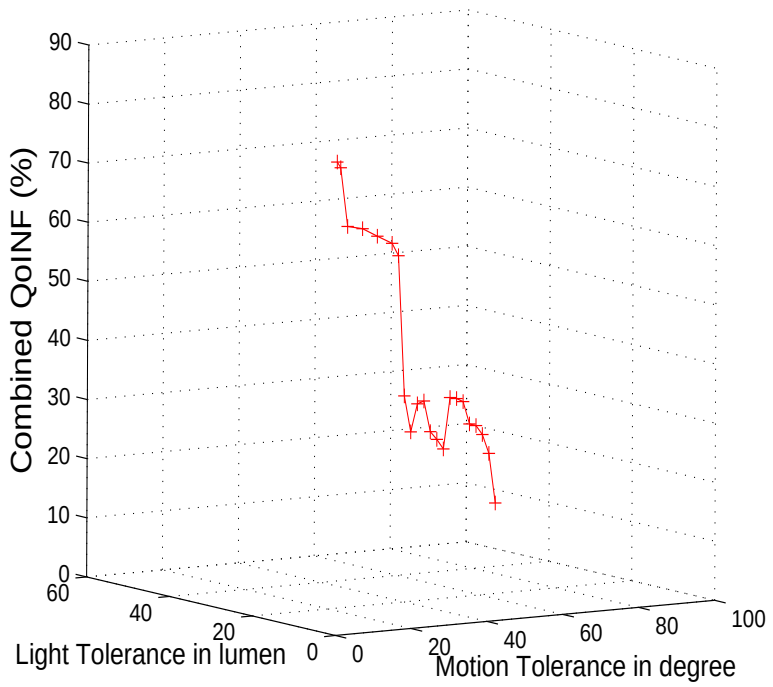

Figure 5.6. Inferencing Accuracy vs. Tolerance Range using both Motion and Light Sensor Together.

q curves from Figures 5.4 and 5.5 to the inverse exponential model of Equation 5.7. After obtaining the best parametric fit, we shall use the heuristic of Section 5.3.2.1 to compute the best predicted combination of q values for any target $QoINF_{min}$ value, and then use additional traces to verify if this approach can provide the required accuracy of context estimation. Of course, the experimental setup itself needs to be expanded to include additional sensors, both on-body and in-situ. Subsequently, we propose to extend this approach to the simultaneous estimation of *multiple* context attributes.

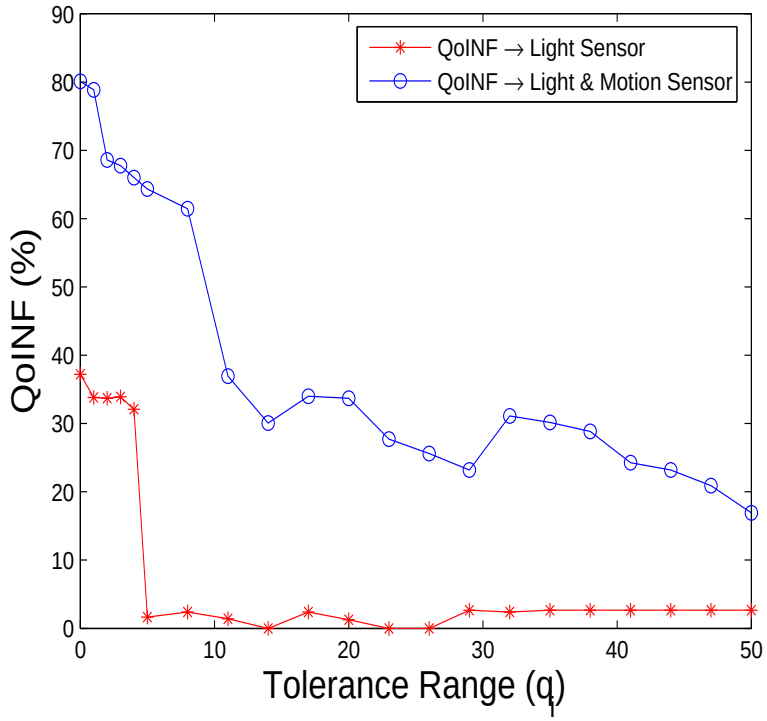

Figure 5.7. Comparison of Inferencing Accuracy Improvement using Multiple Sensor.

5.6 Summary

This chapter presents the initial design for a formal framework for energy-efficient determination of activity or physiological context in assisted living environments. The key idea is to express the accuracy of context estimation through a $QoINF(.)$ function, that captures the dependence of estimation accuracy on both the set of selected sensors and their specified tolerance ranges.

Besides implementing and empirically quantifying the validity of the proposed framework, our ongoing work must address several open challenges and issues. First, we shall continue to investigate alternative forms of $QoINF(.)$ functions that might lend themselves to provably optimal linear-time strategies for computing the optimal

$(\theta, \{q_\theta\})$ combination. Similarly, for our initial choice of the 'inverse exponential function', the quality of the proposed sensor selection heuristic requires evaluation.

CHAPTER 6

RELATED WORK

6.1 Introduction

This chapter presents recent research projects and groups which are related to context awareness in general or more specifically to context learning and prediction. The most important pioneering projects which helped to define the notion of context awareness and/or had great influence on the development of this field are briefly listed in section 6.3. Other related works regarding specific steps in our architecture or used methods and algorithms is mentioned in section 6.4. The term context prediction itself has been used by different research groups, but, as explained in more detail for the related projects, most predictions are performed for lower-level location information. We aim to predict high-level context identifiers, i.e., classes of situations a user or device usually is in [81].

6.2 Projects and Groups

Context awareness is still being defined by different research groups and in different projects. The following list of projects is non-exhaustive and by no means complete, as there have been too many publications to list which were dealing with context awareness in some way during the last few years. We present those publications that are more closely related to the main parts of this thesis: general frameworks and middleware for context awareness, work on context learning and prediction and characterizing uncertainty/ambiguity/error on sensor-driven decision process with limited resources.

Georgia Institute of Technology

The "Aware Home" is built as a living laboratory for empirical research on ubiquitous computing, with the goal to sense contextual information about itself and its inhabitants [66]. One of the motivations for this project is to enable support for elderly to be built into their homes. Kidd et.al. reported about the status of this project, which should also learn user's habits. The Context Toolkit [34] has been developed to support work on the Aware Home project. Consequently, it is a very flexible framework for abstracting context sensing from applications in a distributed, heterogeneous network environment. Aimed towards context sensing embedded into the infrastructure, it uses HTTP and XML for communication between sensors (represented by their "context widgets") and higher-level components (named "aggregators" or "servers" and "interpreters"). It is not directly addressing context prediction targeted at embedded systems; we aim to implement context recognition and prediction locally at each device, without the need for infrastructure components, while the Context Toolkit intentionally is an infrastructure approach for context sensing. There are some limitations of the context toolkit, described in [34]. It does not support continuous context and does not deal with unreliable/unavailable sensor data.

MIT Media Lab

In [19], Brian Clarkson et.al. describe a wearable system with a video camera and a microphone, capable of distinguishing coarse locations (although it is stated that the approach is not restricted to location). Similar to their previous work [18], they used Hidden Markov Models (HMMs), which are a well-known technique for recognizing time series, as a basic model for recognizing context. After feature extraction, they used unsupervised clustering to distinguish between user contexts.

MavHome

The MavHome project [25] by Diana J. Cook et.al. aims to create an agent-based intelligent home with a vision in the spirit of the Aware Home. Their multi-layer architecture consists of a "physical" (i.e. sensors), a "communication" (i.e. network), an "information" (i.e. database) and a "decision" layer and is based on CORBA for remote method invocation. This architecture is similar to the one used in this thesis, as it is also a bottom-up structure for recognizing context. The distinguishing factor of MavHome is its use of different prediction methods to automate actions within the house. Although the Neural Network Home also applies machine learning to predict inhabitant actions, there is a more detailed examination of prediction in the MavHome project. They have developed different algorithms for categorical time series prediction, a simple sequence-match algorithm, Active LeZi [42], a prediction by partial match algorithm based on Markov models, a Markov model based on higher-level action sequences and Episode Discovery [48] [27], which uses data-mining techniques and the minimum description length principle to predict periodic patterns. Those different prediction algorithms are used to forecast user actions, but parts of the prediction seem to rely on database support and batch training, i.e., it does not seem to be possible to use them in an online manner.

University of Washington

In [90], high-level user behavior is inferred from low-level sensor data by adding knowledge of real-world constraints to user location data. A variant of Dynamic Bayesian Networks (DBN) is used in an unsupervised way to predict transport routes based on GPS data. By adding constraints on the routes that could be learned by the training algorithm, the prediction accuracy was significantly improved. However, as we aim to perform prediction at the middleware, independent of the application area, we do not see a general way of incorporating real-world knowledge (that is not learned automatically from the sensor data) a priori into our framework. In the "Activity

Compass" application, this was possible because the work was restricted to location prediction with map material being available. Although it might improve context prediction accuracy, this approach does currently not seem applicable to our work on general, application-independent context.

Owl

The Owl context service supports heterogeneous context sources, privacy and meta information like age, i.e., the time since the sensor was last sampled, and confidence of context data [35]. It is one of the earlier works that already mentions the possibility of inferring future user behavior from learned habits. To this end, their context service was designed to manage context history in addition to the current context. It is also one of the few projects that explicitly deal with the issue of privacy by implementing access control measures. However, it was still developed as a centralized service with a database system as back end and thus requires a persistent connection between clients and infrastructure components and faces issues of fault-tolerance.

SOCAM

At the National University of Singapore, a "Service-Oriented Context-Aware Middleware" (SOCAM) has very recently been developed [43]. It is based on an ontology-oriented approach, i.e., on a vocabulary for representing context, and its model of context is defined using the Web Ontology Language (OWL), allowing to share context between different entities and giving rise to reasoning about context. Their middleware provides the standard services of acquiring, interpreting and disseminating context, but also takes steps towards deriving high-level context from low-level context, which they call "context reasoning" and which can be performed in description logic and first-order logic. To cope with limited resources in mobile devices, the authors divide possible situations into sub-domains (e.g. home domain,

office domain) and switch between the ontologies defined for these sub-domains. However, this middleware is also a service- and infrastructure-oriented approach with a service registry, which might make its use in infrastructure-independent embedded systems difficult. Adapting applications to changed context is performed via the standard way of predefined rules and triggers. The described implementation of the context interpreter, which performs the reasoning process within the OWL model, is suitable for infrastructure-based context services, but not for resource-limited devices. In recent work, the authors proposed a probabilistic extension to OWL and added a Bayesian Network approach to deal with uncertainty in sensor data [44]. Although the ontology-based approach and its automatic transformation to a Bayesian Network to deal with uncertainty could offer distinct advantages, currently there does not seem to be a method for automatically deriving this structure of the ontology from sensor data.

6.3 Pioneering Projects

In this section, the most important pioneering or initiating projects that had an influence on the development of context awareness are listed. They describe an important part of the history of context awareness and help to understand how research on context awareness evolved.

Active Badge

The Active Badge system [115] is a wearable, personal badge equipped with a microprocessor, an infrared sender and a button. Every 15 seconds, it sends a beacon containing its unique id to receivers distributed in the environment, allowing to gather location information at a central server. It used infrared as primary communication medium, which can be embedded in small devices with limited battery life. These badges have been used at ORL for pioneering applications in pervasive computing,

like a building-wide notification system, and have since then been deployed at the University of Kent, Imperial College, London, Lancaster University, the University of Twente, Xerox PARC, DEC research laboratories, Bellcore and MIT Media Lab. The largest system is still in use at Cambridge, with over 200 badges and 300 sensors in the environment. As one of the first building-wide location system that can be said to be unobtrusive (the badges are small enough to be worn comfortably), the Active Badge system had a significant influence on the development of subsequent location systems and already defined upper boundaries regarding the size and power consumption of worn devices. With the early Active Badge installations, even issues like privacy in practical applications or acceptance by test subjects could be tackled due to the large number of badges in use.

Smart Badge system

Inspired by and building upon the experiences with the Active Badge System, the Smart Badge system has been developed [5]. Similar to the Active Badge, a Smart Badge has an infrared transceiver and can be worn, but it additionally integrates tilt and heading sensors. Because infrared receivers are also available in addition to infrared senders, the Smart Badge can sense its environment (the spatial proximity of other Smart Badges) and transmit this sensor information bundled with its unique id to the infrastructure, where it is gathered at a network server. This usage of spatial proximity was one of the first in the pervasive computing research field.

Xerox ParcTab

At the Xerox Palo Alto Research Center (PARC), the ParcTab system has been developed, which is a palm-sized PDA with touch screen, complemented by an infrared communication infrastructure with room-sized cells. Most of its applications are executed on remote hosts and thus depend on the communication infrastructure, which also handles location tracking. One of the applications that have been imple-

mented on the ParcTab is a remote control system to control lights and temperature for the current location, others include the better known Forget-me-not system [76]. The ParcTab can be seen as a more complicated version of the Active Badge that includes a limited user interface for arbitrary applications. As for the Active Badge, infrastructure support is necessary, but this project still inspired many subsequent publications focusing on context aware handheld devices.

Cooltown

The Cooltown project in Palo Alto [67] is still one of the most prominent pervasive computing installations. Its main goal is to provide a "web presence" for people, places and things. To this end, things (i.e. devices) are equipped with web servers and URLs are used as the primary way of addressing information throughout the systems. By periodically sending URLs in infrared beacons and sensing those URLs, location awareness is provided to users in the form of location-aware web services. Their use of HTTP as communication protocol and WLAN for physical communication links allows arbitrary clients to access the infrastructure and thus make use of the web presence of other people, places and things. Example applications that have been implemented include the Cooltown museum and bookstore, which allow to retrieve information about real-world items that broadcast URLs, or the Cooltown conference room, which gives access to projectors, printers or whiteboards via URLs. This project has affected research on context awareness in two areas: Firstly, its distinction of physical entities into people, places and things has become one of the most often cited classifications of context aspects, although newer definitions are less location-specific. Secondly, the usage of HTTP and URLs as principles for referencing and sensing shifts the focus in research from applications to protocols. Using standard, well-known and established Internet protocols and only enhancing them by dynamic sensing technology might be an important direction for future context-aware systems.

Neural Network House

Learning user's habits has previously been explored in The Neural Network House [82], which is able to predict occupancy of rooms, hot water usage and likelihood that a zone is entered the next few seconds using trained feed-forward neural networks. The context information in the project was again mainly comprised of location, but additional state information from rooms like the status of lights or the temperature set by inhabitants were used. While this project is one of many "smart house" projects, it was one of the first to include prediction of user actions. It showed that prediction of user locations can help to save resources and support users by learning their behavior and automating simple tasks which is one of the main contribution of this thesis.

6.4 Summary

In this thesis we try to build an intelligent, pervasive computing and communication platform which can determine the inhabitants' important contexts through the autonomous and pro-active interaction of smart devices. *"Context awareness"* is indeed a key to build such a smart environment and associated applications. For example, the embedded pressure sensors in the Aware Home [87] capture inhabitants' footfalls, and the system (i.e., smart home) uses these data for position tracking and pedestrian recognition. The Neural Network House [82], the Intelligent Home [77], the Intelligent House_n [57] and the MavHome [27, 118] projects focus on the development of adaptive control of home environments by also anticipating the location, routes and activities of the inhabitants. The Active Badge [45] and Active Bat [46] takes the help of infra-red and ultrasonic time-of-flight techniques to provide indoor location tracking framework. On the other hand, MIT's Cricket Location Support System [92] delegates the responsibility of location reporting to the mobile object itself. RADAR

[4], another RF-based indoor location support system uses signal strength and signal-to-noise ratio to compute 2-D positioning. Microsoft's Easy-living and Microsoft Home [72] projects use real-time 3D cameras to provide stereo-vision positioning capability in an indoor environment. Intelligent prediction of these contexts helps in efficient triggering of mobility-aware services.

This *ubiquitous computing* paradigm [121] implies smart (i.e., pro-active) interaction of computing and communication devices with their peers and surrounding networks, often without explicit operator control. Hence, such devices need to be imbued with an inherent sentience [54] about their important contexts that can automatically or implicitly sense information about their state and the presence of users (inhabitants) in order to take action on those contexts. This concept has led to various projects smart homes or environments in general [22, 23]. Existing work such as the Reactive Room [24], Neural Network House [82], Intelligent Room [21] and House_n [60] do not provide explicit reusable support for users to manage or correct uncertainty in the sensed data and their interpretations, and thereby assume that the sensed contexts are unambiguous. The work reported in [34] provided a toolkit to enable the integration of context data into applications, however, no mechanism is provided for sensor fusion or reasoning about contexts to deal with ambiguity. Although other works such as [61] proposed mechanisms for reasoning about contexts, yet they do not provide well defined context-aware data fusion model nor address the challenges associated with context ambiguity and users' situation prediction. Distributed mediation of ambiguous contexts in aware environments was discussed in [31] that allow the user to correct ambiguity in the sensed input. Multimodal Maps [15] for travel planning addresses ambiguity by using multimodal fusion to combine different inputs and then prompting the user for more information to remove the remaining ambiguity

as much as possible. Remembrance Agent [93] uses context to retrieve information relevant to the user and explicitly addresses ambiguity in its manual interface.

Alongside, significant efforts have been made to develop middleware systems that can effectively support context-aware applications in the presence of resource constraints (e.g., sensor networks), also considering requirements for sensory data or information fusion from middleware perspective [1]. For example, DFuse [73] is a data fusion framework that facilitates dynamic transfer of different application level information fusion into the network in order to save power. In adaptive middleware [55] for context-aware applications in smart home setups, the application's quality of context (QoC) requirements is matched with the QoC attributes of the sensors with the help of a utility function. Similarly, in MiLAN [49], application's quality of service (QoS) requirements are matched with the QoS provided by the sensor networks. However, in this scheme, the QoS requirements of the applications are assumed to be predetermined, which the applications should know in advance in addition to the quality associated with the type of sensors it can make use of. Given that in ubiquitous computing environments, the nature (number, types and cost off usage, and benefits) of such sensors available to the applications usually vary, it is impractical to include a priori knowledge about them. The selection of right sensor with right information at the right moment was originally introduced in [114], while the structure of an optimal sensor configuration constrained by the wireless channel capacity was investigated in [13]. By eliminating the simplifying assumption that all contexts are certain we designed a context-aware data fusion algorithm based on dynamic Bayesian network to mediate ambiguous context. But an intelligent sensor management that provides energy-efficiency as well as a way to manage quality of context requirements, which may change over time with changes in patient's state, has not been considered before. An information theoretic approach is taken to decide an optimal sensor configuration

to determine the best current state of the patient while satisfying the application QoC requirements. For end user an ontological rule based approach using semantic web technology is proposed for further reduction of context ambiguity with applications to context-aware healthcare services.

Energy-efficient determination of an individual's context (both physiological and activity) is also an important technical challenge for assisted living environments. Given the expected availability of multiple sensors, context determination may be viewed as an estimation problem over multiple sensor data streams. This thesis develops a formal, and practically applicable, model to capture the tradeoff between the accuracy of context estimation and the communication overheads of sensing. But the tradeoff between communication overhead and the quality of the reconstructed data was first studied in [86], which envisioned the effect of tolerance ranges on the relative frequency of sink-initiated fetching vs. source-initiated proactive refreshes. The focus, however, is on snapshot queries and not on *continually satisfying* the QoINF bound of a long-standing subscription. The idea of exploiting temporal correlation across the successive samples of individual sensors for reducing the communication overhead, for snapshot queries, is addressed in [30], which used training data to parameterize a jointly-normal density function. While a precursor to our work, the focus there was on meeting the QoINF requirements for a class of 'aggregation queries', whereas our focus is on arbitrary relationships between a context variable and its underlying sensor data. The CAPS algorithm [56] is designed for long-running aggregation queries (such as $\{min,\ max\}$) and computes the optimal set of tolerance ranges for a given set of sensors, that minimizes communication overhead while guaranteeing the accuracy of the computed response. However, our aim is to compute both the best subset of available sensors, and their tolerance ranges, that achieves the desired accuracy for arbitrary context variables. In summary the distinguishing aspects of the work pre-

sented in this thesis are context learning, prediction, mediation and the combination of an open, cross-platform middleware framework with the concentration on limited resources, unobtrusiveness and quality of information.

CHAPTER 7

CONCLUSION AND FUTURE WORK

In this thesis, we have developed a novel mobility-aware resource management framework in a multi-inhabitant smart home. Characterizing the mobility of inhabitants as a stationary, ergodic, stochastic process, the framework uses the information theoretic measure to estimate the uncertainty associated with all the inhabitants in the house. It has also been shown that the direct use of per-inhabitant location tracking fails to capture the correlation among multiple inhabitants' locations or activities. We have proved that the multi-inhabitant location tracking is an NP-hard problem. We also formulated a non-cooperative learning paradigm based on stochastic game theory, which learns and estimates the inhabitants' most likely location (route) profiles by minimizing the overall entropy associated with them. The convergence and worst-case performance bounds of this framework are also derived. Automated activation of devices along the predicted locations/routes provide the inhabitants with necessary comfort while minimizing energy consumption and cost.

We also presented a framework which supports ambiguous context mediation and user-centric situation prediction based on dynamic Bayesian networks and information theoretic reasoning, leading to context-aware healthcare applications in smart environments. Our framework provides a Bayesian approach to fuse context fragments and deal with context ambiguity in a probabilistic manner, an information theoretic approach to satisfy context quality within the application, and a semantic web technology to easily compose rules to reason efficiently to mediate ambiguous contexts. An algorithm is proposed and subsequent experimental evaluation is done

to perform cost/benefit analysis to engage the most economically efficient actions for context disambiguation in resource constrained sensor environments.

At the end we developed a formal, and practically applicable, model to capture the tradeoff between the accuracy of context estimation and the communication overheads of sensing. In particular, we proposed the use of *tolerance ranges* to reduce an individual sensor's reporting frequency, while ensuring acceptable accuracy of the derived context. In our vision, applications specify their minimally acceptable value for a Quality-of Inference (QoINF) metric. We introduced an optimization technique allowing the Context Service to compute both the best set of sensors, *and* their associated tolerance values, that satisfy the QoINF target at minimum communication cost. Early experimental results with SunSPOT sensors are presented to attest to the promise of this approach.

REFERENCES

[1] H. Alex, M. Kumar and B. Shirazi, "MidFusion: An adaptive middleware for information fusion in sensor network applications", *Elsevier Journal of Information Fusion, 2005*

[2] Aperion Inspiration Communities, http://aperion.typepad.com/aperion_ companies_weblog/aperion_health_news/index.html

[3] J. C. Augusto, C. D. Nugent and N. D. Black, "Management and Analysis of Time-Related Data in Smart Home Environments", *Proceedings of Mediterranean Conference on Medical and Biological Engineering (Medicon'04).*

[4] P. Bahl and V. Padmanabhan, "RADAR: An In-Building RF-based User Location and Tracking System," *Proc. IEEE Infocom*, IEEE CS Press, Los Alamitos, CA, 2000, pp. 775-784.

[5] H. W. P. Beadle, B. Harper, G. Q. Maguire, and J. Judge, "Location aware mobile computing", *Proceedings of the IEEE/IEE International Conference on Telecommunications* (ICT97), April 1997, pp. 13191324.

[6] M. Beigl, H.-W. Gellersen, and A. Schmidt, "Mediacups: experience with design and use of computer-augmented everyday artifacts", *Computer Networks (Amsterdam, Netherlands)* 35 (2001), no. 4, 401409.

[7] A. Bhattacharya and S. K. Das, "LeZi-update: An information-theoretic approach for personal mobility tracking in PCS networks," *ACM/Kluwer Wireless Networks,* vol. 8, no. 2, pp.121-137, Mar-May 2002.

[8] M. Bowling and M. Velso, "Rational and Convergent Learning in Stochastic Games", *In Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence, Seattle, WA, Aug. 2001.*

[9] P. J. Brown and G. J. F. Jones, "Exploiting contextual change in context-aware retrieval", *Proceedings of the 2002 ACM Symposium on Applied Computing*, ACM Press, 2002, pp. 650656.

[10] S. B. H. Bruder, M. Farooq, and M. M. Bayoumi, "A feature-level approach to multisensor integration; emphasizing robotic applications",*Proceedings of IEEE Conference on Multisensor Fusion and Integration for Intelligent Systems*, October 1994, pp. 477484.

[11] S. B. H. Bruder, "An information centric approach to heterogeneous multisensor integration for robotic applications", *Robotics and Autonomous Systems 26* (1999), no. 4, 255280.

[12] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition", *Data Mining and Knowledge Discovery* 2 (1998), 121167.

[13] J. Chamberland and V. Verravalli, "Decentralized detection in sensor networks", *IEEE Transactions on Signal Processing*, 51(2), Feb 2003. 10

[14] G. Chen and D. Kotz, "Solar: A pervasive-computing infrastructure for context-aware mobile applications", *Tech. Report TR2002-421*, Dept. of Computer Science, Dartmouth College, February 2002.

[15] A. Cheyer and L. Julia, "Multimodal Maps: An Agent-based Approach", *In Proc. of the International Conference on Cooperative Multimodal Communication*, pp.103-113, Netherlands, May. 1995.

[16] G. Chen, M. Li and D. Kotz,"Design and implementation of a large-scale context fusion network", *Proceedings of the First Annual International Conference on*

Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous), Aug. 2004.

[17] B. Clarkson and A. Pentland, "Unsupervised clustering of ambulatory audio and video", *Tech. Report 471, MIT Media Lab*, Perceptual Computing Group, 1998.

[18] B. Clarkson, N. Sawhney, and A. Pentland, "Auditory context awareness via wearable computing", *Proceedings of the 1998 Workshop on Perceptual User Interfaces (PUI98)* (San Francisco, CA, USA), November 1998.

[19] B. Clarkson, K. Mase, and A. Pentland, "Recognizing user context via wearable sensors", *Proceedings of the 4th International Symposium onWearable Computers* (ISWC 2000), 2000, pp. 6976.

[20] C. Claus and C. Boutilier, "The Dyanamics of Reinforcement Learning in Co-Operative Multiagent System", *Proceedings of the Tenth Innovative Applications of Arti cial Intelligence Conference, Madison, WI, July 1998, pp. 746-752.*

[21] M. Coen,"The future of human-computer interaction or how I learned to stop worrying and love my intelligent room", *IEEE Intelligent Systems* 14, 2 (1999), 8-10.

[22] D. J. Cook and S. K. Das, "Smart Environments: Technology, Protocols and Applications," *John Wiley & Sons,* Nov 2004.

[23] D. J. Cook and S. K. Das, "How Smart Are Our Environments? An Updated Look at the State of the Art," *Pervasive and Mobile Computing* (Special Issue on Smart Environments), Vol. 3, No. 2, pp. 53-73, Mar 2007.

[24] J. Cooperstock et al., "Reactive environments: Throwing away your keyboard and mouse", *CACM* 40, 9 (1997), 65-73.

[25] D. J. Cook, M. Youngblood, III E. O. Heierman, K. Gopalratnam, S. Rao, A. Litvin, and F. Khawaja, "MavHome: An agent-based smart home", *Proceedings*

of the 1st IEEE International Conference on Pervasive Computing and Communications (PerCom03), IEEE Computer Society Press, March 2003, pp. 521524.

[26] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley, 1991.

[27] S. K. Das, D. J. Cook, A. Bhattacharya, E. Hierman, T. Y. Lin, "The Role of Prediction Algorithms in the MaVHome Smart Home Architecture", *IEEE Wireless Communications, Special Issue Smart Homes*, Dec. 2002.

[28] S. K Das, N. Roy and A. Roy "Context-Aware Resource Management in Multi-Inhabitant Smart Homes: A Framework based on Nash H-Learning", *Pervasive and Mobile Computing (PMC) Journal*, pp. 372-404, Vol. 2, Issue 4, Nov. 2006

[29] Datamonitor: The Changing Role of Consumers in Healthcare, *Health Data Management Report*, Aug. 2006.

[30] A. Deshpande, C. Guestrin, S. Madden, J. M. Hellerstein, W. Hong, "Model-based Approximate Querying in Sensor Networks", *Int'l Journal on Very Large Data Bases (VLDB Journal)*, 2005

[31] A. K. Dey, J. Mankoff, G. D. Abowd and S. Carter, "Distributed Mediation of Ambiguous Context in Aware Environments", *Proceedings of the 15th Annual Symposium on User Interface Software and Technology (UIST 2002)*, Oct. 2002. pp. 121-130.

[32] A. K. Dey, G. D. Abowd, and D. Salber, "A context-based infrastructure for smart environments", *Proceedings of the 1st International Workshop on Managing Interactions in Smart Environments* (MANSE 99), 1999, pp. 114128.

[33] A. K. Dey and G. D. Abowd, "Towards a better understanding of context and contextawareness", *Proceedings of the CHI 2000 Workshop on The What, Who, Where, When, and How of Context-Awareness*, April 2000.

[34] A. K. Dey, D. Salber and G. D. Abowd, "A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications", *Human-Computer Interaction (HCI) Journal*, Vol. 16 (2-4), pp. 97-166, 2001.

[35] M. R. Ebling, G. D. H. Hunt, and H. Lei, "Issues for context services for pervasive computing", *Proceedings of the Middleware 2001* Workshop on Middleware for Mobile Computing, 2001.

[36] T. D. Estop, *Applied Thermodynamics for Engineering Technologists: S.I. Units*, Longman Publishers, 1978.

[37] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman Publishers, 1979.

[38] D. Garlan, D. Siewiorek, A. Smailagic, and P. Steenkiste, "Project Aura: Towards distraction-free pervasive computing", *IEEE Pervasive Computing* 21 (2002), no. 2, 22 31.

[39] H.W. Gellersen, A. Schmidt, and M. Beigl, "Multi-sensor context-awareness in mobile devices and smart artefacts", *Mobile Networks and Applications* 7 (2002), 341351.

[40] K. Gopalratnam and D. J. Cook, "Active LeZi: An Incremental Parsing Algorithm for Sequential Prediction", *International Journal of Artificial Intelligence Tools*, vol. 14, no.(1-2), pp. 917-930, 2004.

[41] G. R. Grimmet and D. R. Stirzaker, *Probability and Random Processes, 2nd Ed.*, Oxford University Press, 1992.

[42] K. Gopalratnam and D. J. Cook, "Active Lezi: An incremental parsing algorithm for sequential prediction", *Proceedings of the Florida Artificial Intelligence Research Symposium*, 2003.

[43] T. Gu, H. K. Pung, and D. Q. Zhang, "A middleware for building context-aware mobile",*Proceedings of the IEEE Vehicular Technology Conference* (VTC-Spring 2004), May 2004.

[44] T. Gu, H. K. Pung, and D. Q. Zhang, "A bayesian approach for dealing with uncertain contexts", Advances in Pervasive Computing (A. Ferscha, H. Hrtner, and G. Kotsis, eds.), vol. 176, Austrian Computer Society (OCG), April 2004, part of the Second International Conference on Pervasive Computing (PERVASIVE 2004), pp. 205210.

[45] A. Harter and A. Hopper, "A distributed location system for the active office," *IEEE Network*, vol. 8, no. 1 January/February 1994, pp. 62-70.

[46] A. Harter, A. Hopper, P. Steggles, A. Ward and P. Webster, "The anatomy of a context-aware application," *Proc. 5th Annual Int'l Conference on Mobile Computing and Networking*, August 1999, pp. 59-68.

[47] R. Headon, "Movement awareness for a sentient environment", *Proceedings of the 1st IEEE International Conference on Pervasive Computing and Communications (PerCom03)*, IEEE Computer Society Press, March 2003, pp. 99106.

[48] E. Heierman and D. J. Cook, "Improving home automation by discovering regularly occurring device usage patterns", *Proceedings of the International Conference on Data Mining*, IEEE Computer Society Press, November 2003, pp. 537540.

[49] W. Heinzelman, A.L. Murphy, H.S. Carvalho and M.A. Perillo, "Middleware to Support Sensor Network Applications", IEEE Network 18, pp. 6-14, Feb. 2004.

[50] K. Hisazumi, T. Nakanishi, T. Kitasuka, and A. Fukuda, "Campus: A context-aware middleware", *Proceedings of the 2nd CREST Workshop on Advanced Computing and Communicating Techniques for Wearable Information Playing*, May 2003, pp. 4249.

[51] X10 web resources available at `http://www.x10.com`.

[52] A. Hopper, "Sentient computing," *The Royal Society Clifford Patterson Lecture, 1999.*

[53] E. Horvitz, P. Koch, C. M. Kadie, and A. Jacobs, "Coordinate: Probabilistic forecasting of presence and availability", *Proceedings of the 18th Conference on Uncertainty and Artificial Intelligence*, Morgan Kaufmann Publishers, July 2002, pp. 224233.

[54] A. Hopper, "Sentient computing," *The Royal Society Clifford Patterson Lecture, 1999.*

[55] M. C. Huebscher and J.A. McCann, "Adaptive Middleware for Context-aware Applications in Smart Homes", Proc. of the 2nd Workshop on Middleware for Pervasive and Ad-hoc Computing, Oct. 2004, pp. 111-116.

[56] W. Hu, A. Misra and R. Shorey, "CAPS: Energy-Efficient Processing of Continuous Aggregate Queries in Sensor Networks", *Fourth IEEE International Conference on Pervasive Computing and Communications (PerCom)*, 2006, pp. 190-199

[57] "House_n Living Laboratory Introduction", web resource available at `http://architecture.mit.edu/house_n/web/publications`.

[58] J. Hu and M. P. Wellman, "Nash Q-Learning for General-Sum Stochastic Games", *Journal of Machine Learning*, vol. 4, pp.1039-1069, 2003.

[59] 802-11: IEEE standards for information technology telecommunications and information exchange between systems local and metropolitan area networks specific requirements part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications, 1999.

[60] S.S. Intille, "The goal: smart people, not smart homes," *Proceedings of the International Conference on Smart Homes and Health Telematics*, IOS Press, 2006

[61] D. L. de Ipina, "An ECA Rule-Matching Service for Simpler Development of Reactive Applications", *Middleware*, Vol. 2, No. 7, Nov. 2001.

[62] Jena Semantic Web Toolkit: *http://www.hpl.hp.com/semweb/jena2.htm*

[63] F.V. Jensen, "Bayesian Networks and Decision Graphs", Springer-Verlag, New York, 2001

[64] W.P. Jones, "Air Conditioning Engineering", *Arnold Publishers*, 1995.

[65] G. Judd and P. Steenkiste, "Providing contextual information to pervasive computing applications", *Proceedings of the 1st IEEE International Conference on Pervasive Computing and Communications (PerCom03)*, IEEE Computer Society Press, March 2003, pp. 133142

[66] C. D. Kidd, R. Orr, G. D. Abowd, C. G. Atkeson, I. A. Essa, B. MacIntyre, E. D.Mynatt, T. Starner, and W. Newstetter, "The Aware Home: A living laboratory for ubiquitous computing research", *Proceedings of the Cooperative Buildings, Integrating Information, Organization, and Architecture*, Second International Workshop, CoBuild99, Lecture Notes in Computer Science, vol. 1670, Springer, 1999, pp. 191198.

[67] T. Kindberg, H. Morris, J. Schettino, B. Serra, M. Spasojevic, J. Barton, J. Morgan, G. Becker, D. Caswell, P. Debaty, G. Gopal, M. Frid, and V. Krishnan, "People, places, things: Web presence for the real world", *Proceedings of the 3rd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA00)*, IEEE Computer Society Press, December 2000, pp. 1930.

[68] Y. Korilis and A. Lazer, "On the existance of equilibria in noncooperative optimal flow control", *Journal of the ACM* vol. 42, no. 3, pp. 584–613, 1995.

[69] P. Korpip, M. Koskinen, J. Peltola, S.-M. Mkel, and T. Seppnen, "Bayesian approach to sensor-based context awareness", *Personal and Ubiquitous Computing* 7 (2003), no. 2, 113124.

[70] E. Koutsoupias and C. Papadimitriou, "Worst-case Equilibria", *16th Annual Symposium on Theoretical Aspects of Computer Science*, pp. 404–413, 1999.

[71] A. Krause, A. Smailagic, D. Siewiorek, "Context-Aware Mobile Computing: Learning Context-Dependent Personal Preferences from a Wearable Sensor Array," IEEE Transactions on Mobile Computing, pp. 113-127, Feb 2006.

[72] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale, S. Shafer "Multi-Camera Multi-Person Tracking for Easy Living," *Proceedings of 3rd IEEE International Workshop on Visual Surveillance*, IEEE Press, Piscataway, NJ, 2000, pp. 3-10.

[73] R. Kumar, M. Wolenetz, B. Agarwalla, J. Shin, P. Hutto, A. Paul and U. Ramachandran, "DFuse: a Framework for Distributed Data Fusion", Proc. of the 1st International Conference on Embedded Networked Sensor Systems, Nov. 2003, pp. 114-125.

[74] K. Van Laerhoven, "On-line adaptive context awareness starting from low-level sensors", *Masters thesis*, Free University of Brussels, 1999.

[75] K. Van Laerhoven, "Combining the Self-Organizing Map and K-Means clustering for on-line classification of sensor data", *Proceedings of the International Conference on Artificial Neural Networks (ICANN)*, Springer, 2001, pp. 464469.

[76] M. Lamming and M. Flynn, "Forget-me-not: Intimate computing in support of human memory", *Proceedings of FRIEND2194, The International Symposium on Next Generation Human Interface*, Feb. 1994.

[77] Lesser et al., "The Intelligent Home Testbed", *Proc. of Autonomy Control Software Workshop*, Jan 1999.

[78] P. Lukowicz, H. Junker, M. Stger, T. von Bren, and G. Trster, "WearNET: A distributed multi-sensor system for context aware wearables", *Proceedings of the 4th International Conference on Ubiquitous Computing (UbiComp02)* Lecture Notes in Computer Science, vol. 2498, September 2002, pp. 361370.

[79] P. Lukowicz, J. A. Ward, H. Junker, M. Stger, G. Trester, A. Atrash, and T. Starner, "Recognizing workshop activity using body worn microphones and

accelerometers", *Proceedings of the 2nd International Conference on Pervasive Computing (PERVASIVE 2004)* Lecture Notes in Computer Science, vol. 3001, Springer, April 2004, pp. 1832.

[80] H. Luo, Y. Liu and S. K. Das, "Routing Correlated Data with Fusion Cost in Wireless Sensor Networks," *IEEE Transactions on Mobile Computing*, Vol. 5, No. 11, pp. 1620-1632, Nov. 2006.

[81] Rene Mayrhofer, "An Architecture for Context Prediction," PhD Thesis, 2004

[82] M. C. Mozer, "The Neural Network House: An Environment that Adapts to its Inhavitants", *Proc. of the American Association for Artificial Intelligence Spring Symposium on Intelligent Environments*, pp. 110-114, 1998.

[83] John F. Nash. "Non-cooperative games", *Annals of Mathematics,* 54: 286295, 1951.

[84] Netica Bayesian Network Software http://www.norsys.com

[85] D. A. Norman, "The invisible computer", *MIT Press,* Cambridge, 1998.

[86] C. Olston and J. Widom, "Offering a precision performance tradeoff for aggregation queries over replicated data", *In Proceedings of VLDB*. Margan Kaufmann Publishers Inc., 2000, pp.144 -155.

[87] R. J. Orr and G. D. Abowd, "The Smart Floor: A Mechanism for Natural User Identification and Tracking," *Proceedings of Conference on Human Factors in Computing Systems* (CHI 2000), ACM Press, NY, 2000.

[88] G. Owen,"Game Theory", *Second Edition,Academic Press*,1982

[89] S. Padovitz, W. Loke, A. Zaslavsky, C. Bartolini and B. Burg, "An approach to Data Fusion for Context Awareness", *5th International and Interdisciplinary Conference, CONTEXT 2005*, pp. 353-367

[90] D. J. Patterson, L. Liao, D. Fox, and H. Kautz, "Inferring high-level behavior from lowlevel sensors", *Proceedings of the 5th International Conference on Ubiq-*

uitous Computing (UbiComp03) Lecture Notes in Computer Science, vol. 2864, Springer, October 2003, pp. 7389.

[91] N. Paton and O. Diaz, "Active Databse Systems" *ACM Computing Surveys*, 31(1): 63-103 (1999)

[92] N. Priyantha, A. Chakraborty and H. Balakrishnan, "The Cricket location support system," *Proc. 6th Ann. Int'l Conference on Mobile Computing and Networking*, August 2000, pp. 32-43.

[93] B. Rhodes, "The Wearable Remembrance Agent: a System for Augmented Memory", *Personal Technologies Journal Special Issue on Wearable Computing*, 1:218-224, 1997.

[94] A. Roy, S. K. Das Bhaumik, A. Bhattacharya, K. Basu, D. J. Cook and S. K. Das, "Location Aware Resource Management in Smart Homes", *Proc. of IEEE Int'l Conf. on Pervasive Computing and Communications (PerCom)*, pp. 481-488, Mar 2003.

[95] N. Roy, S. K. Das , K. Basu and M. Kumar, "Enhancing Availability of Grid Computational Services to Ubiquitous Computing Application", *IEEE International Conference on Parallel and Distributed Processing Symposium (IPDPS'2005)*, Denver, Colorado, USA, Apr. 2005.

[96] N. Roy, A. Roy, S. K. Das and K. Basu, "A Reinforcement Learning Framework for Location-Aware Resource Management in Multi-Inhabitant Smart Homes", *Proc. of 3rd International Conference on Smart Homes and Health Telematic (ICOST)*, pp. 180-187, July 2005.

[97] N. Roy, A. Roy, K. Basu and S. K. Das "A Cooperative Learning Framework for Mobility-Aware Resource Management in Multi-Inhabitant Smart Homes ", *Proc. of IEEE International Conference on Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous)*, pp. 393-403, July 2005.

[98] N. Roy, A. Roy and S. K. Das, "Context-Aware Resource Management in Multi-Inhabitant Smart Homes: A Nash H-learning based Approach", *Proc. of IEEE Int'l Conf. on Pervasive Computing and Communications (PerCom)*, pp. 148-158, Mar 2006. (Mark Weiser Best Paper Award).

[99] N. Roy, G. Pallapa and S. K. Das,"A Middleware Framework for Ambiguous Context Mediation in Smart Healthcare Application", *Proc. of IEEE Int'l Conf. on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Oct 2007.

[100] N. Roy, A. Misra and S. K. Das, "Efficient Long-Term Quality-of-Inference (QoINF)-Aware Context Determination in Pervasive Care Environments", *Proc. of ACM SIGMOBILE Workshop on Systems and Networking Support for Healthcare and Assisted Living Environments* (HealthNet 2008) in conjunction with MobiSys 2008.

[101] N. Roy, G. Pallapa and S. K. Das, "An Ontology-Driven Ambiguous Contexts Mediation Framework for Smart Healthcare Applications", Proc. of Int'l Conf. on PErvasive Technologies Related to Assistive Environments (PETRA 2008).

[102] M. Schneps-Schneppe and V. B. Iverson "Service Level Agreement as an issue of Teletraffic", *Proceedings of the ITC*, Berlin, Sept 2003.

[103] C. R. Shelton, "Balancing Multiple Sources of Reward in Reinforcement Learning", *Advances in Neural Information Processing Systems*, pp. 1082-1088, 2000.

[104] A. Schmidt, M. Beigl, and H.-W. Gellersen, "There is more to context than location", *Computers and Graphics* 23 (1999), no. 6, 893901.

[105] A. Schmidt, "Ubiquitous computing computing in context", *Ph.D. thesis, Lancaster University*, November 2002.

[106] B. N. Schilit, "A system architecture for context-aware mobile computing", *Ph.D. thesis*, Columbia University, May 1995.

[107] SemanticWorks: *http://www.altova.com/products/semanticworks/semantic_web_rdf_owl_editor.html*

[108] C. R. Shelton, "Balancing Multiple Sources of Reward in Reinforcement Learning", *Advances in Neural Information Processing Systems*, pp. 1082-1088, 2000.

[109] S. Singh, T. Jaakkola, M. L. Littman and C. Szepesvari, "Convergence Results for Single-Step On-Policy Reinforcement-Laerning Algorithms", *Machine Learning 38*, no. 3, 2000, pp. 287-290

[110] M. K. Smith, C. Welty, and D.L. McGuinness, "OWL Web Ontology Language Reference," World Wide Web Consortium (W3C) recommendation, Feb. 2004; www. w3.org/TR/owl-ref.

[111] SPARQL tutorial: *http://www.w3.org/TR/rdf-sparql-query/*

[112] SunSpotWorld-Home of Project Sun SPOT: *www.sunspotworld.com*

[113] E. M. Tapia, S. S. Intille, and K. Larson, "Activity recognition in the home using simple and ubiquitous sensors", *Proceedings of the 2nd International Conference on Pervasive Computing (PERVASIVE 2004)* Lecture Notes in Computer Science, vol. 3001, Springer, April 2004, pp. 158175.

[114] R. R. Tenney and N. R. Sandell, Jr., "Detection with distributed sensors," IEEE Trans. Aerosp. Electron. Syst., vol. AES-17, pp. 501-510, 1981.

[115] R. Want, A. Hopper, V. Falco, and J. Gibbons, "The active badge location system", *ACM Transactions on Information Systems* (TOIS) 10 (1992), 91102.

[116] M. Weiser, "The Computer for the 21st Century," *Scientific American*, vol. 265, no. 3, pp. 94-104, Sept. 1991.

[117] X10 web resources available at `http://www.x10.com`.

[118] M. Youngblood, D. J. Cook, and L. B. Holder. "Managing Adaptive Versatile Environments." *Proc. of IEEE Int'l Conf. on Pervasive Computing and Communications (PerCom)*, pp. 351-360; Mar 2005.

[119] M. Youngblood, "MavPad Inhabitant 2 Trial 2 Data Set." *The MavHome Project.* Computer Science and Engineering Department, The University of Texas at Arlington, 2005. *http://mavhome.uta.edu*

[120] Watkins, C. J. C. H., "Learning from delayed rewards", *PhD thesis*, King's College, Cambridge, UK, 1989

[121] M. Weiser, "The computer for the 21st century," *Scientific American*, vol. 265, no. 3, pp. 94-104, Sept. 1991.

[122] Y. Zhang, Q. Ji and C. G. Looney, "Active Information Fusion For Decision Making Under Uncertainty", *Proceedings of the Fifth International Conference on Information Fusion*, 2002, vol. 1, pp. 643- 650.

[123] J. Ziv and A. Lempel, "Compression of individual sequences via variable-rate coding," *IEEE Transactions on Information Theory*, vol. 24, no. 5, pp. 530-536, Sept 1978.

BIOGRAPHICAL STATEMENT

Nirmalya Roy received a Bachelor of Engineering degree in Computer Science and Engineering from Jadavpur University, Calcutta, India, in 2001. He began his graduate studies in the Department of Computer Science and Engineering at The University of Texas at Arlington in Fall 2002. He received his Master of Science degree in Computer Science and Engineering from The University of Texas at Arlington in May 2004.