
Actor coordination using info-gap decision theory in wireless sensor and actor networks

S. Chinnappen-Rimer*

Department of Electrical, Electronic and Computer Engineering,
University of Pretoria,
Pretoria, South Africa
E-mail: suvendic@uj.ac.za
*Corresponding author

G. P. Hancke

1) ISG Smart Card Centre,
Royal Holloway, University of London,
Egham, United Kingdom
2) Department of Electrical, Electronic and Computer Engineering,
University of Pretoria,
Pretoria, South Africa
E-mail: ghancke@ieee.org

Abstract: Mobile, unmanned, power and resource-rich devices, called actors, deployed within a Wireless Sensor Network (WSN) application area, enable faster response times to events. Due to cost constraints, only a few actors can be placed within a WSN application area. Determining which actor or set of actors should respond to an event is important, because the correct decision will increase the event response time, and reduce energy expenditure. Since the mobile actors are widely dispersed over the application area, the actors' accurate location and energy details will not always be available. In this paper, we show that using info-gap decision theory to choose the correct actors to respond to an event when uncertainty about an actor's location and/or energy exists, ensures that the actors chosen can adequately respond to the event. The robustness of the decision choice of the set of actor(s) assigned to respond to an event means that all chosen actor(s) have sufficient energy to respond to the event in real-time.

Keywords: info-gap decision theory; wireless sensor network; actor coordination.

Biographical notes: S. Chinnappen-Rimer received a BSc. Electrical Engineering degree from the University of the Witwatersrand and a Masters of Engineering degree from the University of Pretoria. She is currently working towards her PhD degree in Computer Engineering at the University of Pretoria. Her current research interests include wireless sensor networks, ad-hoc networks and energy-efficient message routing.

Gerhard Hancke received a Bachelor of Engineering degree in Computer Engineering from the University of Pretoria in 2002 and a Masters of Engineering degree from the same institution in 2003, both with distinction. He read for a PhD in Computer Science with the Security Group at the University of Cambridge's Computer Laboratory, which he completed in 2008. He joined the Information Security Group at Royal Holloway, University of London in 2007 and is responsible for the ISG Smart Card Centre's RFID/Contactless research track and RF/Hardware laboratory. In 2009 he was invited to act as part-time visiting lecturer at the University of Pretoria. His main interests are smart hardware tokens and their applications, mobile systems and pervasive computing.

1 Introduction

A Wireless Sensor Actor Network (WSAN) is a wireless network comprising a large number of low-cost, resource and energy constrained nodes within communication range of each other, together with a few resource and energy rich mobile devices, placed within the application area. In a typical wireless sensor network (WSN), after

an event occurs (e.g. temperature change), the sensing nodes activated by the event need to coordinate the messages between the nodes to send a single aggregated data message to a sink node, which then transmits the data to a remote end user for evaluation and response (Akyildiz et al., 2002). The real time response of the WSN thus depends on the reaction of the human entity in question. To enable better response

to events, mobile non-human devices called actors are placed in the application area to form a Wireless Sensor Actor Network (WSAN). Due to the limited and generally non-renewable energy source of sensor nodes, most applications of WSNs and WSANs try to limit the number of messages transmitted within the network. Actors must limit their communication with other actors as the wireless medium can result in other nodes inadvertently expending some energy on irrelevant message receptions. Thus, to conserve energy and ensure a faster response time, it may not be possible for all actors to communicate with each other via a round of negotiation type messages before reaching a decision on which actor(s) should respond to an event.

When an event is detected by a group of sensors in a WSAN, the sensors reach consensus to send a single message to one or more actors. When an actor receives a message, it informs all other actors of the event. Depending on where an event occurs and which actor is first informed of the event, the other actors in a WSAN will not have accurate data about the location and energy resources of other actor(s) before making a decision on which actor(s) should respond to the event. The actors need to coordinate a response and determine the actor(s) that will respond to the event, even when uncertainty about the other actor(s) location and available energy exists. Actor-actor coordination needs to be able to optimally choose an actor(s) to respond to an event as quickly as possible. Akyildiz et al. (Akyildiz et al., 2004) describe four types of scenarios for actor responses, namely, (1) Single actor centralised decision (SACD), (2) Single actor distributed decision (SADD), (3) Multi-actor centralised decision (MACD), and Multi-actor distributed decision (MADD). In each scenario, the real-time needs of the system implies that the exact state of each actor in the application area is uncertain. Therefore, one of the main problems in a WSAN is to determine which actor or set of actors should respond to an event, when the data about the current state of each actor's location and energy levels is not sufficiently certain.

To obtain the optimum actor decision, we can use one of the methods prescribed by the two main classes of decision theory to reach a conclusion with insufficient knowledge about everything, namely the probabilistic or the classical approach. The probabilistic approach assumes there is some previous record of decisions which can be used to learn (train) from in order to obtain the correct choice. The classical approach does not require knowledge of previous decisions. We propose the use of a classical approach to determine the optimum actor(s), based on modelling the inherent uncertainty in the environment as an information gap.

The primary motivation of using info-gap decision theory instead of probabilistic decision theory is because the probabilistic model requires "*history*", i.e. some previous results that can be used to predict future results. Probability theory represents uncertainty in terms of measurable values such as frequency of

recurrence of events. To ensure fast response to an event, actors cannot wait until all actors are aware of the correct energy levels and exact location of each actor in the application area. Actor-actor coordination needs to be able to optimally choose an actor(s) to respond to an event as quickly as possible, even when uncertainty about an actor(s)'s resources exist. Info-gap decision theory (IGDT) can be used where robust solutions are required in an uncertain environment. The robustness of the decision ensures that even if the optimal set of actors is not chosen to respond to an event, then those actor(s) chosen have sufficient resources to respond to the event.

The objective of this article is to show that information-gap decision theory can be used as an effective method to select a few optimum actors, which will adequately respond to an event, even when uncertainty about other actors' energy availability and location exists. We model uncertainty of an actor's energy and location as an information-gap, and then calculate the optimum set of actor(s) that should respond to the event. This article assumes that the WSAN has an automated architecture (Akyildiz et al., 2004). Sink interaction is removed from the decision making process, which should result in a more rapid response to events by actors.

2 Related Work

Duncan et al. (Duncan et al., 2008) use IGDT to determine correct decisions when assessing product life-cycle where there are many unknowns concerning product usage and performance. Cheong et al. (Cheong et al., 2008) used IGDT as a tool to assist companies bidding to supply electricity, where the information about competing bids is uncertain. Regan et al. (Regan et al., 2005) showed that IGDT can be used to rank different conservation management actions to protect the Sumatran rhinoceros. They illustrate the importance of understanding the impact uncertainty has in successful conservation management. More examples of the successful application of IGDT can be found at (www.info-gap.com, 2010).

There have been a number of different theories presented to enable effective coordination amongst sensors and actors in a WSAN. Melodia et al. (Melodia et al., 2007, 2005) propose dividing the application area into clusters with each cluster being assigned an actor. When an event occurs in a particular cluster, the sensors in that cluster will inform the actor(s) assigned to the cluster which will then coordinate the response to the event. The solution is based on a Distributed Event-driven Partitioning and Routing (DEPR) protocol that relies on local information and on greedy routing to reduce the number of messages and thus optimise sensor power. Actors bid in an auction based on their available energy to complete the task. The bidding process increases the actor response time and may not be suitable for real-time type applications. The authors

use clusters to reduce the energy dissipation of sensor nodes when informing an actor of an event. It is assumed that an actor is centrally located within the cluster to effectively receive an event message. The use of an auction mechanism is time-consuming and reduces actor response time.

This cluster does not change and the actor concerned is used for sending all the sensed data around that region. In the event driven clustering paradigm (Melodia et al., 2007, 2005), the formation of the cluster is triggered by an event. In this case the cluster is created in the most optimum way to react to the event. According to Melodia et al., coordination can be accomplished by defining a set of definitions. The first definition points to the existence of a maximum allowed latency between event sensing at sensor and event arrival at the actor. The second definition defines the existence of reliability of the event which is the ratio of the reliable data packets received to all the packets received during the decision interval. Using these two definitions an integral linear program can be used to minimize the latency and maximize the actor response to an event. One problem with clustering is that the actor assigned to the cluster may not be the best actor to respond to the event, depending on where within the cluster the actor is located. Melodia et. al. do not consider an actor's current energy levels or whether there are obstacles in the actor's path that may prevent an actor from reaching a specific location. If there are obstacles, even though an event message reaches an actor quickly, the actor cannot reach the event location.

The idea of sub-dividing the application area into smaller, more manageable pieces is also proposed by (Ngai et al., 2006). Event data is layered to separate event notification to the actor from the complete data message. After an event occurs, the main event data from the nearest sensors is transmitted to the closest actor. This reduces the response time as actors start to coordinate a response without receiving all event information. Ngai et. al do not consider if the actor(s) chosen is the best actor(s), but focuses on informing an actor of an event as quickly as possible.

A transactional framework (TRANSACT) to prevent collisions and ensure concurrent processing of event messages is proposed by Demirbas (Demirbas, 2007). The intent is to develop coordination programs that can be used to find the lead actor along the neighbourhood of a region. When the actor is not found then the program assigns a lead. TRANSACT provides guarantees on consistency and safety, but on the other hand does not guarantee tight time-lines due to the contending nature of channels access.

A solution to sensor-actor coordination via finding optimal routing paths based on the the tubular network formation behaviour of slime mold (Rossi et al., 2007) is proposed by Rossi et. al. By modelling routes between sensors and actors as slime mold, optimal routes (tubes) are computed by following the shortest field lines in a specific direction. This is mainly used to convey an event

message to an actor and not to coordinate which actor(s) should respond to the event.

Another proposed solution organises groups of sensors into a Voronoi region (cluster) with each region being assigned an actor (Bouhafs et al., 2006). During the initialisation phase all nodes inform the rest of the group members of their location and function. When an event occurs, the sensor sends the event message together with its data about the nodes nearest actor to a region/cluster head. The head determines the nearest actor and forwards the message to this actor. The cluster head limits distributed decision making in choosing an optimum actor to respond to an event and the network is dependent on the cluster head.

As far as we know, IGDT has not been proposed as a solution in actor-actor coordination. IGDT is a viable method for obtaining robust decisions in real-time when uncertainty about an actor's current energy resources and location exist. Using IGDT to model uncertainty about an actor's location, together with neighbouring topological information can increase the likelihood of choosing the correct set of actor(s) to respond to an event in real-time, without having to query each actor about their current location and energy levels.

3 Background: Info-gap Decision Theory

IGDT models uncertainty as an unbounded gap in available information. In info-gap, uncertainty represents that area between what is known and what needs to be known in order to make competent decisions. Info-gap models of uncertainty provide possible solutions when information is scarce or the system is susceptible to fundamental changes which is independent of current conditions.

In info-gap theory, robustness is the largest amount of uncertainty or deviation from an estimate that can be allowed in a decision without the possibility of the decision resulting in failure (BenHaim, 2006). IGDT does not ask how wrong the estimate *is*, which is a question that cannot be answered. Rather IGDT asks how wrong the estimate *can be* without allowing an unacceptable outcome. Large robustness values are required in general to confidently resist failure

Uncertainty is modelled by an unbounded family of nested sets of u -vectors. In IGDT, a decision algorithm $D(q, u)$ depends on among other things, the decision vector q , which specifies the structure and properties of the algorithm, as well as an uncertainty vector u , which includes measurements. The three sections of info-gap analysis is:

1. System model : specifies the decision algorithm q in terms of the input/output structure of the system to which a decision is to be applied. A reward function assesses the response of the system in terms of the quality of performance to the decision vector q and the uncertainty u .

2. Performance Requirements : specifies the requirements expected from the system.
3. Uncertainty model : models uncertainty as an info-gap model $U(\alpha, \tilde{u})$ in terms of any prior information about the uncertainty vector u . Variations on uncertainty variables u is represented by an info-gap model of uncertainty, $U(\alpha, \tilde{u}), \alpha \geq 0$. $U(\alpha, \tilde{u})$ is the set of all functions whose deviation from \tilde{u} is nowhere greater than α (BenHaim, 2006). For a fixed value of α , this set represents uncertainty by specifying a range of variation of u around the nominal function \tilde{u} . The larger the value of α , the greater the range of unknown variation, so α is called the *uncertainty parameter* or *horizon of uncertainty*.

4 Proposed Solution

The objective is to decide on an optimum set of actors to respond to an event in real-time, so that the actors chosen, can reach the event location and respond to the event adequately. In a real-time system, the time delay in reacting to an event should be as small as possible. As commonly found in real-life situations, there may be insufficient time or an inability to determine the exact state of all actors' resources. Thus actors will have to choose the optimal set of actors to react to an event when uncertainty about the other actors energy resources and location exists. We use IGDT to assess how uncertainty about actor data can exist between actors and still result in effective actor-actor coordination. IGDT considers by how much and in what ways the assessment of uncertain possibilities can differ between actors and still allow for agreement (BenHaim, 2006).

4.1 Sensor-Sensor and Sensor-Actor Coordination

To handle sensor-sensor coordination, after a sensor detects a specific event occurring (e.g. temperature change), the sensor will need to gain control of the communication medium to send a message to the nearest actor. This may require the suppression of data messages that are not immediately relevant, for e.g. many sensors in a specific region may want to transmit messages about the same event. If the target is static, one or two messages may suffice. If the target is moving, the actor(s) will need to know the direction (also probable direction based on estimates). In addition, mechanisms need to be in place to ensure actors are not reacting to an old event (e.g. timestamp, location etc).

If the triggered sensor does not receive a broadcast message informing it of the event and requesting access to the communication medium in order to transport urgent data, then the triggered sensor should initiate a special broadcast message to its immediate neighbours when the triggered sensor can transmit a message. The broadcast message should contain the following data:

- activated node ID
- event type
- timestamp
- location of activated sensor
- destination actor node ID.
- destination sensor node ID closest to actor.

If the triggered sensor does receive a broadcast message informing it of the event, then it should either stay silent or re-broadcast the message to the actor if it is the destination node.

How will the sensor know which actor to send the message to? On initialisation of a WSN, or when an actor changes its location, each actor node transmits a message containing the following data:

- actor ID
- actor location
- number of hops from actor, i.e proximity to actor.

When a sensor node receives this message it adds the data to an actor table. The actor table contains the actor ID, number of hops to actor, node ID of the transmitting message (for single hop, the transmitting node will be the actor). If sensor nodes receive more than one initial message, it orders the actors in the actor table from smallest hop count to largest. The receiving node increments the hop count and re-broadcasts the message to its neighbours. At the end of the initialisation phase of the WSN, all sensors should be aware of at least one actor in the system, and the number of hops the sensor is from the actor.

4.2 Actor-Actor Coordination using IGDT

In actor-actor coordination, actors need to cooperate with one another to determine the actor(s) to respond to an event. IGDT considers by how much and in what ways can assessment of uncertain possibilities differ between actors and still allow for agreement.

Assumptions:

1. On initialisation each actor is aware of all other actors in the application area and their location.
2. All actors know their location within the application area in terms of Cartesian coordinates (x, y) .
3. All actors are aware of their energy resources
4. All actors are aware of the geographic terrain and any obstacles, e.g. rivers, ditches etc. in the application area that may hinder movement to a particular area.
5. Each actor will use the equation $s = ut + \frac{1}{2}at^2$ to estimate time to reach event area.

6. The application area is small enough so that the distance from actor to event can be calculated as a straight line from its position to the position of the sensed event, by using Cartesian coordinates (ie. $d = \sqrt{(y_2 - y_1)^2 + (x_2 - x_1)^2}$)

The following describes the four different scenarios for actor-actor coordination (Akyildiz et al., 2004):

Centralised Decision : SACD or MACD

The controlling actor is designated at initialisation of the WSN. In single actor centralised decision (SACD) and multi-actor centralised decision (MACD), the controlling actor is informed about the event. The controlling actor calculates the other actors energy and time-of-arrival based on the specified uncertainty. The actor with the smallest arrival time but that has sufficient energy to reach the destination and respond to the event is chosen. The controlling actor transmits a message to alert the selected actor, or if it is the chosen actor, alerts other actors and sensors about the change in its position and the reason for the change (message will include event ID, location of event, and timestamp of event). If the controlling actor can change during the lifetime of the WSN, then the actors determine which actor should be the next controlling actor through consensus, or by using a rotation timetable.

Distributed Decision : SADD or MADD

The actor(s) that receive a sensor event message, sends a message to other actors in the application area informing the actors of the event. In a distributed decision scenario, each actor calculates its own as well as the other actors energy and time-of-arrival based on the specified uncertainty range. Due to time constraints, the actors may not be able to wait until all actors receive its calculations.

For single actor distributed decision (SADD), each actor sends a message to the actor that has the fastest arrival time and has sufficient energy to reach the destination and respond to the event. For multi-actor distributed decision (MADD), where the number of actor(s) may vary depending on the application, the specified number of actors with the smallest arrival time but that have sufficient energy to reach the destination and respond to the event are sent a message to respond to the event by each actor.

An actor must receive a pre-determined number of respond type messages (r), before it responds to the event. In the interim an actor compares the received message values against its calculated values. If an actor's calculated values indicate that it is a chosen actor than it will expect $r - 1$ respond type messages from other actors. If an actor does not receive the specified number of respond messages required within a pre-determined time-frame, from the other actors, it sends a message to all other actors alerting them that insufficient messages have been received. The message will include the actor's calculated values. The receiving actors can use this information to update its records and redo the calculation to determine if the actor should respond. For

the simulation, the actor waits to receive one message before it responds to an event.

4.2.1 System Model

In IGDT, each actor will have a decision vector q , which specifies the structure and properties of the decision algorithm. A decision algorithm $D(q, u)$ depends on among other things, the decision vector q as well as an uncertainty vector u . Each actor will have a decision vector q of the following parameters:

- distance to event (D),
- available energy (E),
- speed of actor (A),
- estimated time of arrival at event location (T),
- obstacles in place to reach event area (O), and/or
- event direction (if event moving) (M)

Each actor calculates time to arrive at an event location, determines whether there is sufficient energy to move to event location, perform necessary operations, and return to its original location. A careful analysis of the decision vector $q = \{D, E, A, T, O, M\}$ reveals that the elements in the decision vector all relate to energy and time. This is, distance to event and speed of actor is used to calculate the time to reach the event. If there are obstacles in place the actor may never reach the event area. The elements in the decision vector can be reduced to $q = \{E, T\}$.

The robustness is expressed as the maximum set of values of an uncertainty parameter α to avoid failure BenHaim (2006):

$$\hat{\alpha}(q) = \max \{ \alpha : \text{min. requirements always satisfied} \} (1)$$

$\hat{\alpha}(q)$ is the robustness of uncertainty α in the decision vector q while ensuring that minimum requirements are met.

The degree of robustness is assessed by a scalar reward function $R(q, u)$. $R(q, u)$ is the minimal requirement in the robustness function (equation (1)), that the reward $R(q, u)$ be greater than or equal to a critical value P_c . $R(q, u)$ depends on a vector q of actions or decisions as well as on an uncertain vector u , whose variations are described by an info-gap model $\bar{U}(\alpha, \bar{u})$, $\alpha \geq 0$.

Therefore, the system model can be expressed as a reward function $R(q, u)$, which determines the response to an event based on the actors' available energy and the time it will take an actor to react to an event.

$$R(q, u) = \frac{\text{Energy}}{\text{Time}} (2)$$

From equation (2) we can see that $R(q, u)$ has to meet a minimum requirement (the robustness requirement) to enable an actor to react to an event. If there are obstacles in an actor's path, assume T is ∞ (infinity).

then, $R(q, u) = 0$. Thus, the robustness of the decision algorithm $D(q, u)$ can be defined as the greatest value of the uncertainty parameter α for which the decision is the same for all events in the info-gap model $U(\alpha, \tilde{u})$. Formally the robustness will then be expressed as follows:

$$\hat{\alpha}(q, P_c) = \max \{ \alpha : R(q, u) \geq P_c, u \in U(\alpha, \tilde{u}) \} \quad (3)$$

Where P_c is the minimum critical reward function value that will ensure an actor can correctly respond to the event. Each actor calculates a decision vector for the other actors based on the actor's current knowledge of the other actors' values.

4.2.2 Performance Requirements

The performance requirements for robustness in the info-gap model specify the minimally acceptable values that the decision algorithm is required to achieve. These values may constitute success of the decision, or at least minimally acceptable values. To obtain the minimum performance requirements we assign values to the various elements in the decision vector, which are analysed to determine the critical value that the reward function has to achieve.

The first factor considered is time. Obstacles in the actor's path to the event are regarded as simple binary outputs, 0 for they exist, 1 for no obstacles. If there are obstacles, then an actor may not be able to reach an event. In this case, time-of-arrival is ∞ . If there are no obstacles, time-of-arrival of an actor at the event location is dependent on the distance to the event. An acceptable timeframe is also modelled as a simple binary output, where $t=1$ indicates an actor is able to reach an event within a specified time period and $t=0$ implies an actor cannot reach an event within the specified time period.

The second factor is movement. If an event is mobile, the choice of actor will have to consider the event velocity and direction. If the event is moving towards an actor, this actor is more likely to be chosen. If an event is moving at a fast speed away from an actor, then the actor closest to the location of where the event is first detected may not be the best actor to choose to respond to the event. Movement of event is mapped as a value:

- -1 event moving away from actor,
- 0 event not moving, and
- +1 event moving towards actor.

The third decision vector variable is energy. This is one of the critical elements because if an actor does not have sufficient energy, it will not be able to respond to any event, even if the event is located close to the actor. If the actor has sufficient energy to move to an event location, react to the event and return to its original location, then $e = 2$. If the actor has sufficient energy to move to an event location, react to the event but cannot return to its original location, then $e = 1$. If the actor has

sufficient energy to move to an event location, but cannot react to the event or return to its original location, then $e = 0$.

The key performance requirement is that an actor can reach the event location in real-time and have sufficient energy to move to the event location and adequately respond to the event. From equation [2] and based on the above criteria, then the critical value for the reward function is that the actor has sufficient energy to move to event location and react to event, i.e. $e = 1$, and that an actor is able to reach an event within a reasonable timeframe, i.e. $t = 1$. Therefore,

$$P_c = \frac{e}{t} = \frac{1}{1} = 1 \quad (4)$$

From equations [3] and [4], the critical value of the reward function is:

$$R(q, u) \geq P_c \geq 1 \quad (5)$$

4.2.3 Uncertainty Model

An info-gap model $U(\alpha, \tilde{u})$ represents uncertainty about each actors energy and time-of-arrival at the event location as the uncertain vector u . Each actor knows its current available energy, and can reasonably accurately calculate the time to reach the event location. During initialisation, it is assumed that each actor is also made aware of the other actors energy levels and location. It is also assumed that during infrequent intervals, each actor will send a message to the other actors via the central sink to inform them of the actor's current energy level and location.

Due to the soft real-time constraint, when an event occurs, it is not feasible to contact all other actors in the WSN and wait for their response to determine the actors' current energy and location data. An actor will be uncertain by how much the other actors current energy levels and location differs from the last updated status value. Equation [6] represents the uncertain vector consisting of uncertainty in actor energy (ϵ) and actor time-of-arrival at event location (τ).

$$\tilde{u} = (\epsilon, \tau) \quad (6)$$

This uncertainty about the degree of variability of an actor's current energy ϵ , and the expected time of arrival at the event location τ , is represented as the following info-gap model:

$$U(\alpha, (\tilde{\epsilon}, \tilde{\tau})) \{ (\epsilon, \tau) : |\epsilon - \tilde{\epsilon}| \leq \psi\alpha, |\tau - \tilde{\tau}| \leq \theta\alpha \}, \alpha \geq 0 \quad (7)$$

where, the value of the vector $\tilde{u} = (\epsilon, \tau)$ is uncertain. Uncertainty in energy and time vary independently of each other.

5 Simulation

An investigation of current simulators used for wireless sensor networks, such as the Network-Simulator (NS-2),

and OmNet++, revealed they would not meet our design requirements, without significant new development additions. Instead, a simulation was developed in Java. Although the programming language used is Java, the code could just as easily have been written using C or C++. The simulation had to consider various levels of uncertainty in the location, time-of-arrival and energy of each actor. The current simulation assumes that all events are fixed and not mobile. All actor's are aware at initialisation of each actor's location, time-of-arrival and energy. The uncertainty in an actor's location, time-of-arrival and energy is measured from these initial values which will change over time. For the worst case scenario, it is assumed that uncertainty in location, time-of-arrival and energy of the optimum set of actors as determined at initialisation will result in an increase in the location and time-of-arrival of the optimum actor(s) and a decrease in the energy levels of the set of optimum actor(s). All other actors location, time-of-arrival and energy are assumed to stay at constant values set at initialisation. The reason for this is to demonstrate that even under these conditions, using IGDT to coordinate which actor(s) are chosen results in a robust actor decision choice that can adequately react to the event within an acceptable timeframe. All the following results and discussions assume the above worst case scenario.

The simulated application area is a two-dimensional 700mx700m square grid. A 1000 sensor nodes are evenly placed around the application area. Nine actors are used in a configuration shown in Figures 1 and 2. The actors were placed mainly along the perimeter of the application area to ensure adequate response time could be achieved for an event anywhere within the application area. It is assumed that there are no obstacles in the application area that would prevent an actor from reaching an event location. The number of actors required in a MADD scenario is set to three. An event was simulated and the simulator run according to the decision algorithm. Various values for uncertainty α was chosen in the range $0 \leq \alpha \leq 500$ in increments of 20. An event was setup equidistant from at least three actors (Figure 1), and an event was setup close to an actor (Figure 2). These two events were chosen because they are the two worst case scenarios for any event to occur within the application area.

6 Results and Discussion

6.1 The info-gap model.

Uncertainty in time and energy vary independently. In our earlier work (Chinnappen et al., 2009), we showed the robustness of the optimum actor as uncertainty about its energy levels decrease and time to reach the event location increases. Two scenarios were considered, one for an event occurring approximately equidistant

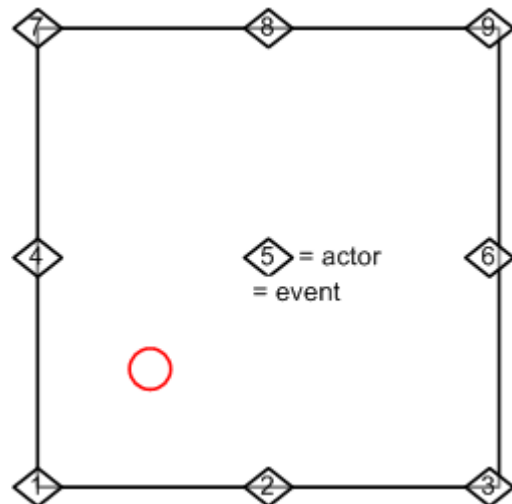


Figure 1 Event equidistant

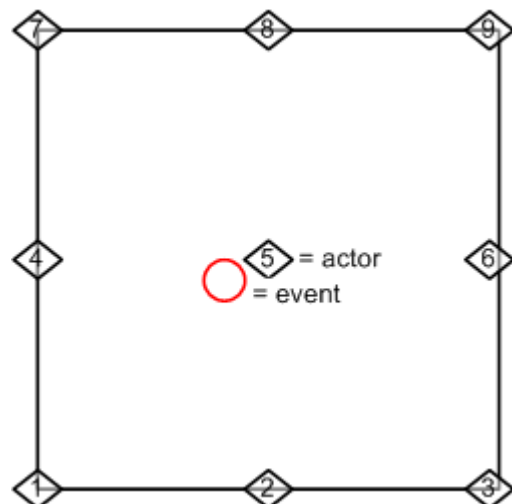


Figure 2 Event close to one actor

from four actors, (shown in Figure 1, and the other for an event occurring close to a specific actor is shown in Figure 2. As stated previously, the robustness is the maximum level of uncertainty that can be sustained while ensuring that an actor can reach an event in real-time with sufficient energy to react to the event. As energy and time are independent of each other, we show the robustness for uncertainty in energy (Figure 3), and uncertainty in time (Figure 4). Robustness needs to stay above 1, i.e. $P_c \geq 1$, to ensure that the chosen actor can respond to an event. From Figure 3 and Figure 4, it appears that uncertainty in the optimum actor's energy and time-of-arrival can be high before the critical value is reached. Robustness for an actor close to a specific event will be high initially because that actor will be the optimum actor to respond to an event until the uncertainty about an actor's location increases significantly. Note, that as the application area is set at 700mx700m, the uncertainty in an actor's time and energy has to exceed half the length of the application area before the robustness of the chosen actor declines

to less than 1. This means that the uncertainty of the chosen actor(s) energy and time-of-arrival must exceed more than 50% of the application area before IGD_T will return an actor choice that is incapable of reaching the event location and having sufficient energy to react to the event. In Figure [3] and Figure [4], we consider

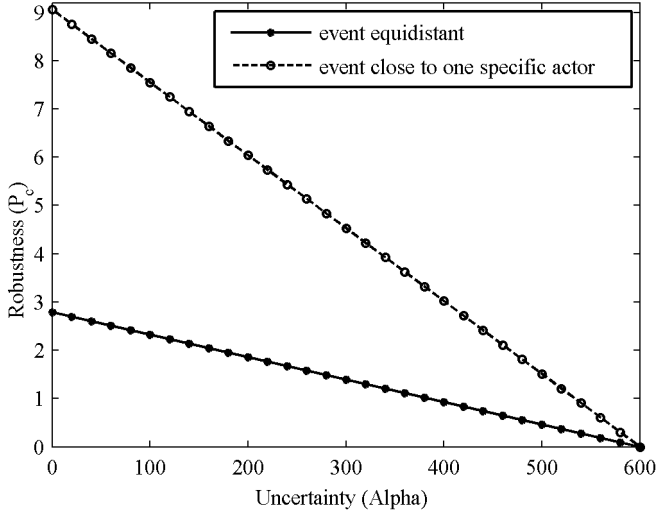


Figure 3 Robustness for uncertainty in actor **energy** to reach event location and react to event.

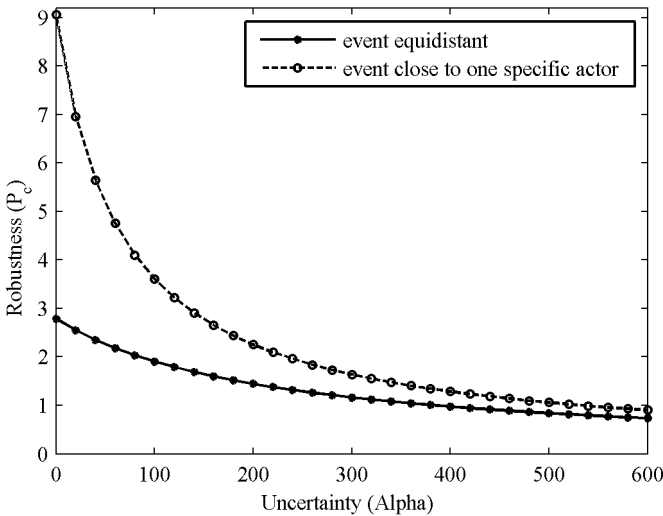


Figure 4 Robustness for uncertainty in actor **time** to reach event location.

robustness to uncertainty as energy decreases and time to respond to the event increases. This is the worst case scenario. If we consider uncertainty in distance to an event, and then calculate the time to respond to that event (Figure 5), then the robustness against failure ($P_c \geq 1$) is greater for the entire application area length. As expected the robustness gradient is much greater for an event close to a specific actor. From Figure 5, it can be seen that the robustness of the chosen actors tends to the same values as uncertainty increases. A possible reason is that as uncertainty increases, the uncertainty in

the location of the closest actor(s) increases irrespective of the event location. It should be noted that even as the robustness decreases, it still exceeds 1 which indicates that the chosen actor(s) are capable of adequately responding to the event.

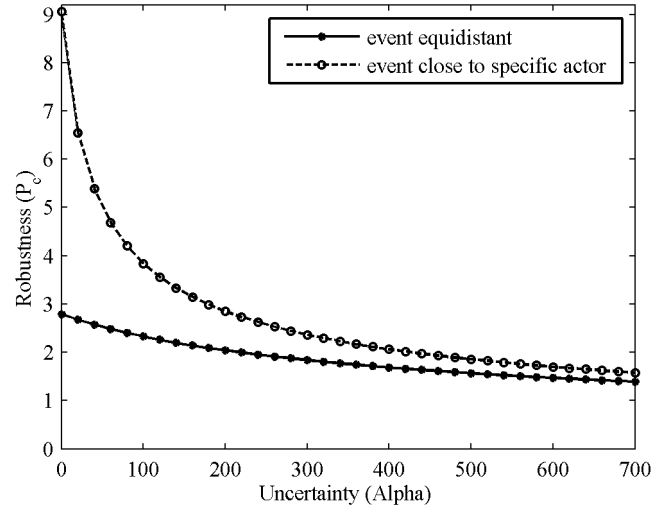


Figure 5 Robustness for uncertainty in actor **distance** to event

In our previous work (Chinnappen et al., 2009), we calculated the optimum set of actors to respond to an event as uncertainty in an actor's available energy decreases and time to reach the event location increases (Figures 6 and 7). The uncertainty of the optimum set of three actors' energy decreasing and distance from event increasing, while the other actors energy and location are known results in at least two actors from the correct set being chosen to respond. For example, for an equidistant event (Figure 1), actors 1, 2, 4 and 5 are almost equidistant from the event. If only three actors are in the set ($actor \in 1, 2, 5$), then the fourth actor (4) which is almost the same distance from the event as the optimum set of actors, will replace one of the optimum actors in the set as soon as uncertainty about any optimum actors' location exists. For an event close to one actor (Figure 2), actor 5 is located close to the event. The other two actors in the set are 2 and 4 with actors 6, 8 and 1's distance to the event relatively close to that of actors' 2 and 4. Thus, after a small amount of uncertainty about the location of actors 2 and 4, the actors' chosen in the optimum set will decrease to just the actor located close to the event.

The data depicted in Figures 6 and 7, (and also Figures 8, 9, 10, 11, 13, 15 and 17 later in this paper), show discrete states. In the simulation, when uncertainty is 0 and the energy and locations of all actors are known, the set of three actors best able to respond to an event at a specific location is calculated. This optimum set is used to compare against the other calculated sets as uncertainty about the optimum actors' energy, location and time-of-arrival is increased. The graphs depict how many of the actors from the initial optimum set are still

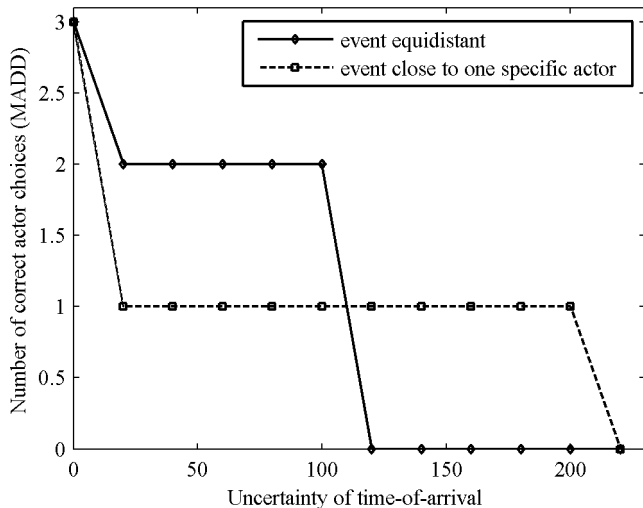


Figure 6 Uncertainty of actor time-of-arrival in number of actors chosen in optimum set

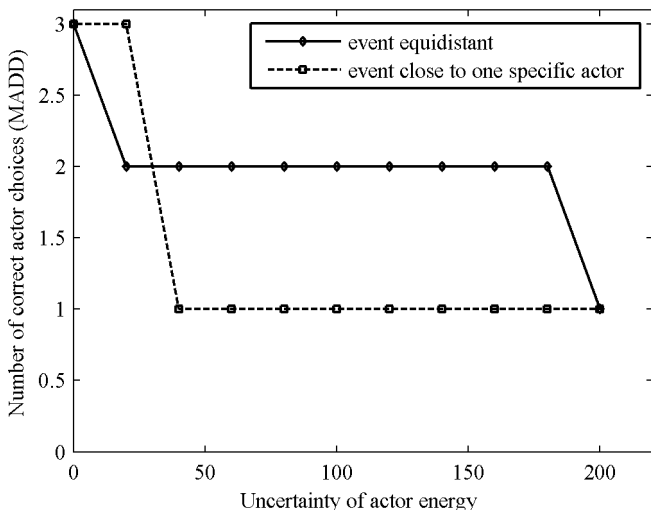


Figure 7 Uncertainty of actor energy in number of actors chosen in optimum set

part of any newly calculated sets at a specific level of uncertainty. This is the reason that the data in Figures 6 and 7, appear to suddenly change, because the figures show the number of actors out of the original set of optimum actors that are still part of the newly calculated set as the uncertainty in the original actors' energy, time-of-arrival and location varies.

The results shown in Figures 6 and 7 indicate the number of actors chosen in an optimum set as the distance of the optimum set of actors to the event is increased, while the distance to the event of the other actors stays the same. In Figure 6, the time to reach the event is calculated, based on the distance to the event. The uncertainty in time to reach the event is used to plot the results in Figure 6. To increase the number of optimum actors chosen as uncertainty about the actor's location increases, we assumed that an actor's energy can easily be replenished, and hence was not a factor in

the model. We also considered uncertainty in terms of the actual distance to an event (instead of uncertainty in the time it takes an actor to reach an event). All actors are assumed to have the same acceleration. The info-gap model with only distance as a factor is shown below.

$$U(\alpha, \tilde{\delta}) \left\{ \delta : |\delta - \tilde{\delta}| \leq \psi \alpha \right\}, \alpha \geq 0 \quad (8)$$

The distance to the event for the optimum set of three actors is increased by the given uncertainty and the time to reach the event is calculated. This resulted in slightly better results of the number of actors chosen from within the optimum set of actors that will respond to an event as shown in Figure [8] and Figure [9]. The correct number of actors or at least one or more actors from the optimum set over larger values of uncertainty is chosen.

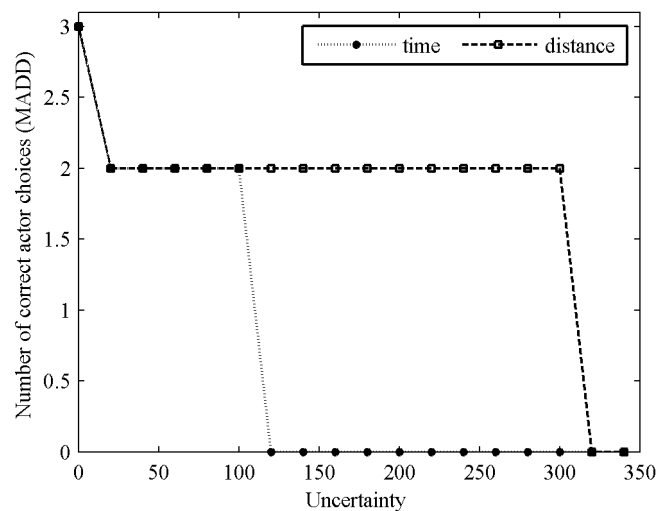


Figure 8 Uncertainty of actor distance to an equidistant event.

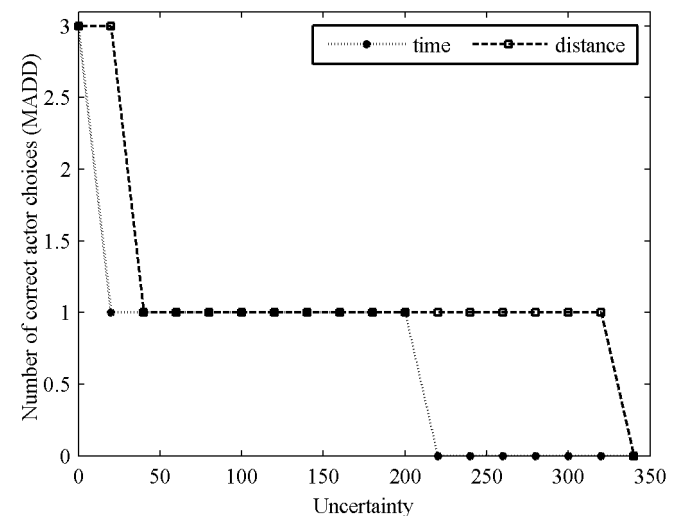


Figure 9 Uncertainty of actor distance to an event close to a specific actor.

6.2 The nearest neighbour info-gap model.

To improve the number of optimum actors chosen from the set as uncertainty about the actor's location increases, we considered a hybrid model based on the nearest neighbours of the optimum set when uncertainty is 0. For example, for an event close to say actor 5 (Figure 2), the nearest neighbours to actor 5 are 2, 4, 6 and 8. For an equidistant event, the optimum set will be the actors 1, 2, 4 and 5. The nearest neighbour for each actor will include at least two actors from the optimum set. So any other two actors that are part of this set are optimum actor's to respond to the event (assuming no obstacles in their path to the event). This can be expressed as follows:

$$\begin{aligned} A &\in \{a_i, \dots, a_n, i = 1, \dots, n\} \\ O &\in \{o_i, \dots, o_x, i = 1, \dots, x\} \\ P &\in \{p_i, \dots, p_x, i = 1, \dots, x\} \\ O &\subset A \text{ and } P \subset A \\ S &= O \cup P. \end{aligned}$$

A is the set of actors in the application area, O is the optimum set of actors to respond to the event and P is the nearest neighbours of each actor in the optimum set. O and S is the chosen set to respond to an event.

We obtained a significant improvement in the number of optimum actors chosen to respond to the event as uncertainty about the optimum actors location increased, as shown in Figure 10. For an equidistant event, the optimum set of actors is correct for a large amount of uncertainty. Even for an event close to one specific actor, the actors chosen to respond to the event stays within the optimum set of actors for a larger amount of uncertainty about the actors location.

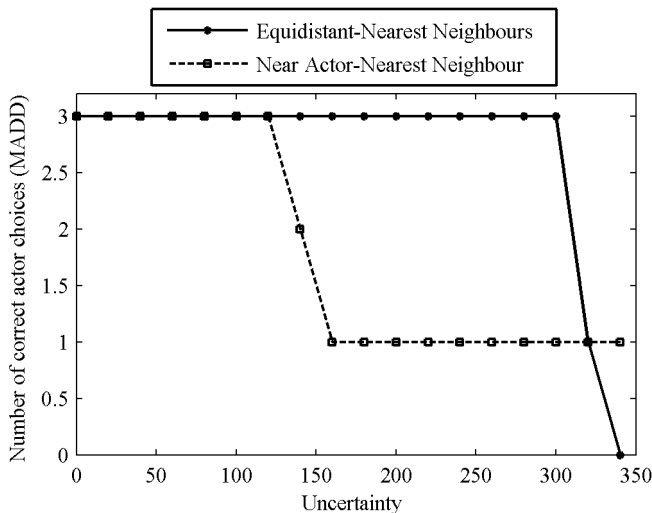


Figure 10 Hybrid nearest neighbour model

6.3 The info-gap model and the cluster model.

The division of the application area into clusters with one or more actors assigned to respond to events in a specific

cluster raises questions about the real-time response of the cluster system. The cluster model assumes that the actor(s) assigned to a cluster is the optimum actor placed to respond to an event within that cluster. In addition, knowledge about the exact location and energy resources of each actor is assumed to be always current. This is not always practical. To conserve energy in a WSN, the number of messages transmitted within the application area is kept to a minimum. Every time a mobile actor moves, it will have to continuously update the other actors of its new location coordinates and expended energy. The info-gap model will produce the best actor to respond to an event within real-time. In many instances the actor chosen using the info-gap model will be the actor assigned to a particular cluster. Figure 11 shows the IGDT set for three possible scenarios for event and actor location. The individual scenarios results are shown in Figure 13, Figure 15, and Figure 17. Consider the different scenarios described in Figures 12, 14, and 16.

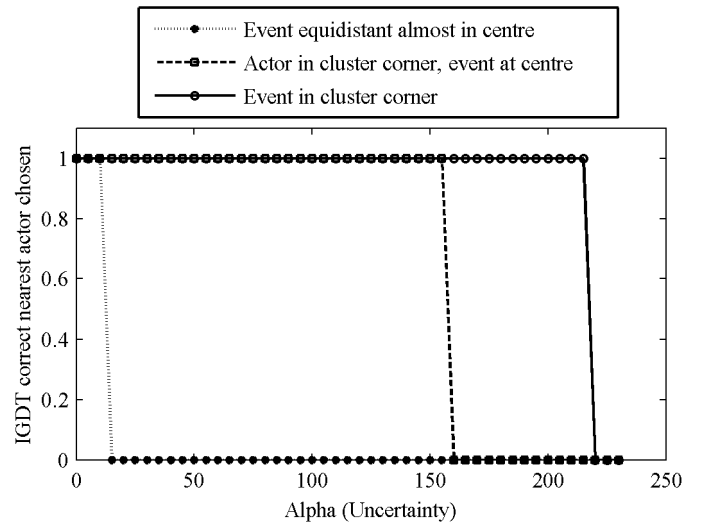


Figure 11 IGDT chooses correct cluster actor

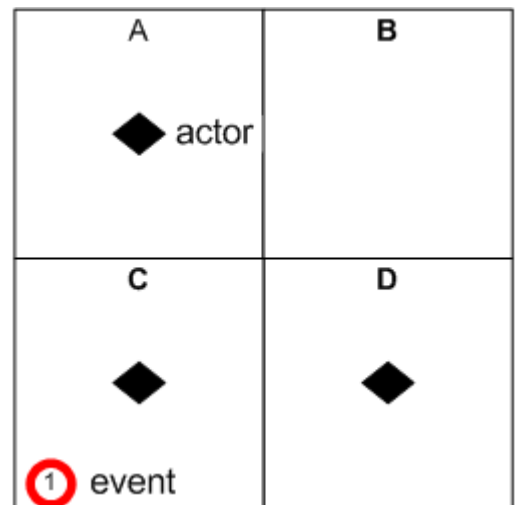


Figure 12 Cluster actor best placed to react

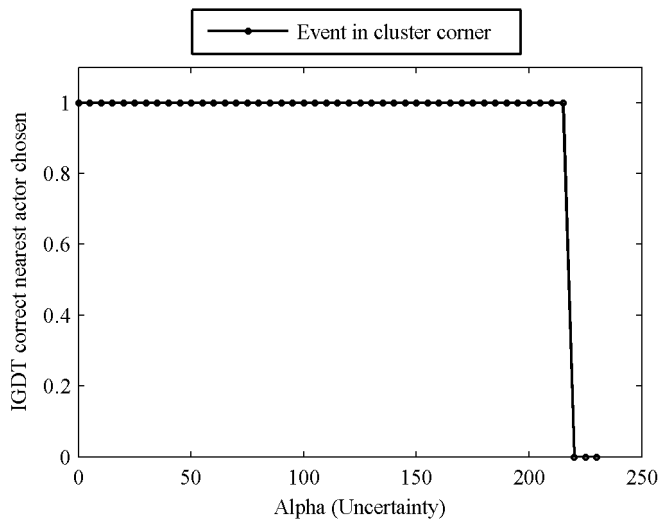


Figure 13 IGDT chooses correct actor for an event in cluster corner when uncertainty about the actor's location is large.

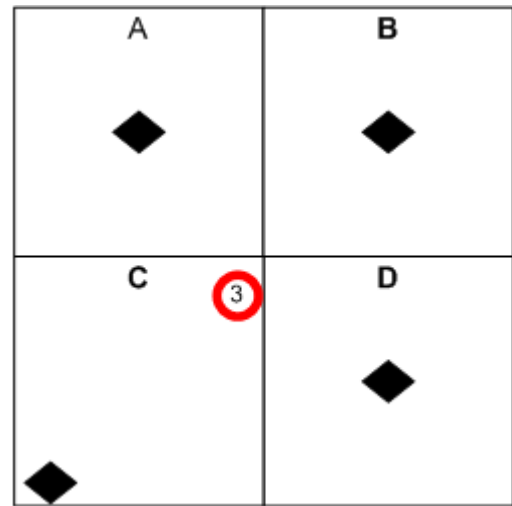


Figure 16 Event furthest from cluster actor

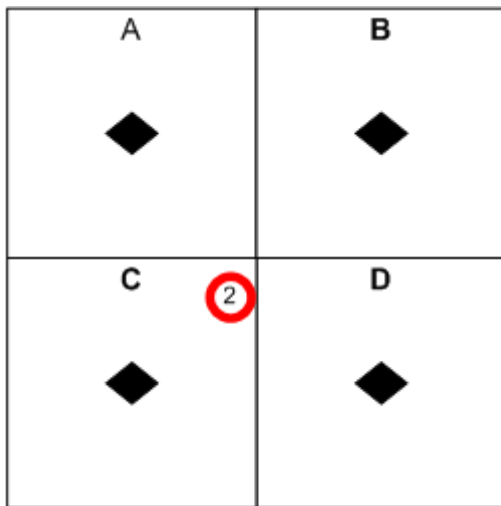


Figure 14 Event almost equidistant for all actors

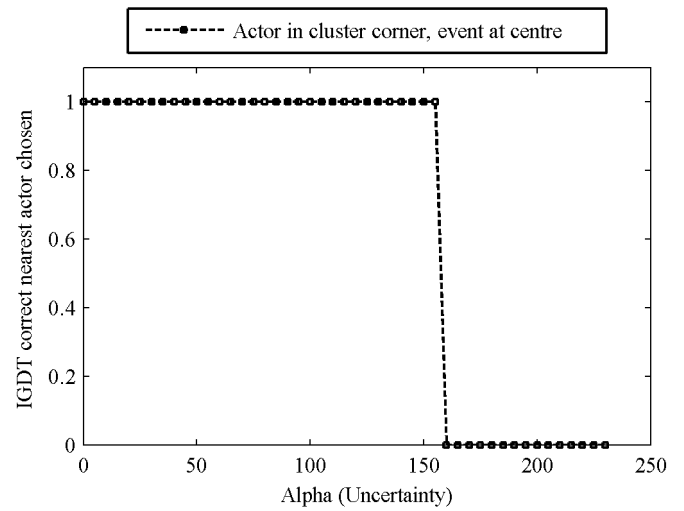


Figure 17 IGDT chooses correct nearest actor when event at centre and actor in cluster corner

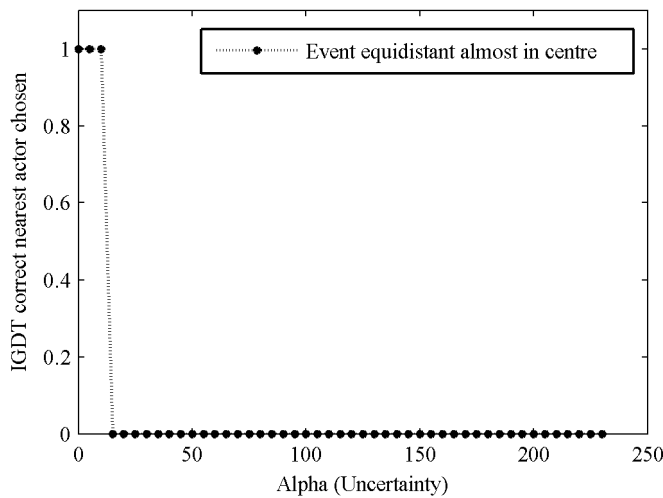


Figure 15 IGDT chooses correct actor for event equidistant from actors for only a small amount of uncertainty about the actor's location

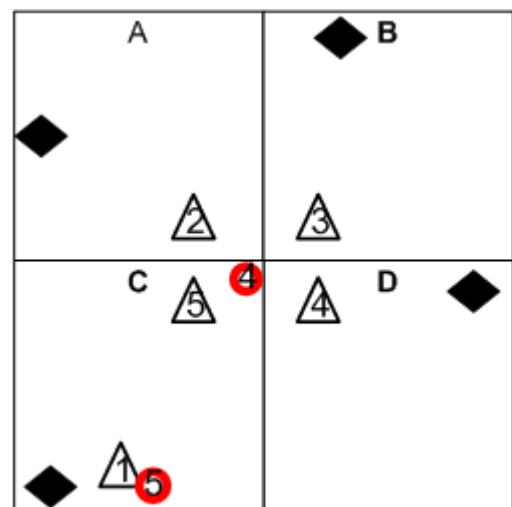


Figure 18 Localised nodes assist actor choice

In Figure 12, the actor assigned to cluster C is obviously best placed to respond to the event and the info-gap model correctly chooses this actor for a large amount of uncertainty about the actor’s location (Figure 13). As the actor moves further from the event, one of the neighbouring actors’ may be chosen depending upon the uncertainty about the distance to the event. In this example, the application area is assumed to be 500mx500m. Each cluster is 250mx250m. The cluster actor is located in the centre of the cluster at point (125,125). The event is located at point (0,0). If the cluster C actor moved further away then point (175,175) from the event and cluster actor A and D moved closer to points (0,250) and (250,0) respectively, than cluster actors A and/or D would be closer to respond to the event. Given the initial actor locations in Figure 12, the uncertainty about actor C’s location would have to move outside the cluster (to beyond point (280,280)) before IGDT chose a non-cluster actor to respond to the event.

In Figure 14, the event is within cluster C but is almost equidistant from each actor. If actor C moved a small distance away from the event or actors A, B or D, moved closer to the event location, than either of these three actors would be better placed to respond to the event. The info-gap model chooses the assigned cluster actor until the uncertainty about the location of an actor exceeds a small value (15) as shown in Figure 15. The robustness factor is greater than 1 (Figure 19), which indicates that when uncertainty about the location results in the actor not assigned to the cluster being chosen, then the chosen actor is still capable of reaching and reacting to the event in time. In fact, the actor chosen using the IGDT model would be the better placed actor to respond to the event instead of the required cluster actor.

In Figure 16, the event is farther from the assigned cluster actor than the other actors in the application area. In this scenario, the info-gap model correctly chooses the best actor to respond based on smallest distance to the event, unlike the cluster model which will require that the actor assigned to that cluster be required to respond to the event. The info-gap model chooses the actor closest to the event, until the uncertainty of this actor’s location exceeds 160 (Figure 17). It must be noted that as uncertainty about an actors location increases, as long as the robustness of the decision exceeds 1, then the chosen actor is still capable of reacting to an event (as shown in Figure 19).

6.3.1 Localised sensor computation.

To further improve the robustness of the actor decision choice, we consider using localised sensor information to enhance the choice of actor to respond to an event. Consider Figure 18, and assume that each actor patrols a specific section of the application area. This is similar to an an actor patrolling a particular cluster, but unlike the cluster model any actor can respond to an event

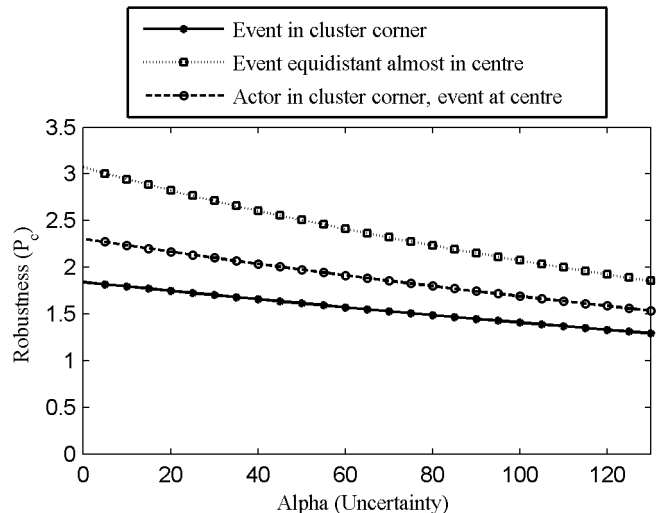


Figure 19 Robustness of IGDT decision

in the “cluster/section”. As the actor passes sensors it transmits a message with its ID, current location, timestamp and acceleration. The transmission range is short, i.e. typically a sensor node range. This is to limit the number of messages received by sensor nodes as the actor passes by.

When an event occurs, all sensor nodes near the event are informed of the event. Sensors 2,3,4 and 5 would each have received messages from actors A, B, C or D as they pass by at different intervals. We assume that the sensors perform data aggregation to reduce the amount of messages and message size of event messages. One method of data-aggregation that could be feasible for mobile actors (or sinks) in a WSN is directional-controlled fusion (Chen et al., 2009), which utilises a combination of multipath-converging and multipath-expanding to achieve load balancing, and improve network performance and lifetime.

When the sensors perform data aggregation to inform actors about the event, each sensor will inform the other sensors in the group about the last time a specific actor passed by and its acceleration. The factors can be used to determine the distance the actor has travelled and hence the nearest actor to the event. As the message is routed by other nodes to an actor, the node re-transmitting the message can add its actor data to the message. When an actor receives an event message, it can use this information to obtain a more robust decision about which actor or set of actors should respond to the event. This localised computation at the sensor node enhances the choice of correct actor(s) to respond and reduces the uncertainty of an actor’s location.

For events that occur only within a specific cluster, for example if an event occurs near sensor 1, then sensor 1 is only aware of actor C. The nodes routing the event message will also list actor C as the closest actor to respond to the event. The choice of optimum actor to respond in this scenario will continue to be actor C for

a large amount of uncertainty about actor **C**'s exact location.

For events close to a cluster intersection such as event **4** in Figure 18, sensors **2,3,4** and **5** would have received messages from actors **A, B, C** and **D**, as they pass by these sensors. The sensors will store the time the actors have sent messages when they pass by and the actor's acceleration and will be able to utilise this information to calculate the time cycle before the actor is due to pass by again. Using this information, the sensors located close to the event can estimate the current location of each actor and determine the closest actor to respond to an event. It could sent this data to the neighbouring actors. The actor's that receive this message can use this data to determine which actor will respond to the event. The message generated will have a higher priority than a message received from an actor which does not contain local sensor data. The use of localised sensor data about an actor's location reduces the uncertainty in the decision process. Used in conjunction with the info-gap model, it will ensure that the optimum actor decision set is always chosen to respond to an event.

6.4 Robustness, uncertainty and 95% confidence interval.

Figures 20 and 21 show confidence intervals at the 95% level for the robustness graph shown previously, namely, Figure 3. From the figure we can be 95% confident that the robustness of the decision will not exceed 1.6901 and not fall below a minimum value of 1.0957 for an event that is equidistant from all actors, and that the robustness of the decision will not exceed 5.4962 and not fall below a minimum value of 3.5632 for an event that is close to one specific actor. IGDT focuses on the robustness of decision making under uncertainty, i.e., by how much can an info-gap model differ under uncertainty and still produce a reasonably reliable result. From the figures it is clear that with increasing uncertainty, the robustness decreases but if the robustness stays above 1, the decision set (chosen actors) can still respond reliably to an event. The average robustness value is 1.3929 for an equidistant event and 4.5297 for an event close to a specific actor.

Similarly, Figures 22 and 23 show confidence intervals at the 95% level for the robustness graph shown previously, namely, Figure 4. From the figure we can be 95% confident that the robustness of the decision will not exceed 1.5452 and not fall below a minimum value of 1.1464 for an event that is equidistant from all actors, and that the robustness of the decision will not exceed 3.0803 and not fall below a minimum value of 1.7235 for an event that is close to one specific actor. The average robustness value is 1.3458 for an equidistant event and 2.4019 for an event close to a specific actor.

When the uncertainty in distance is considered as in Figure 5, the confidence intervals at the 95% level is shown in Figures 24 and 25. From the figure we can be 95% confident that the robustness of the decision

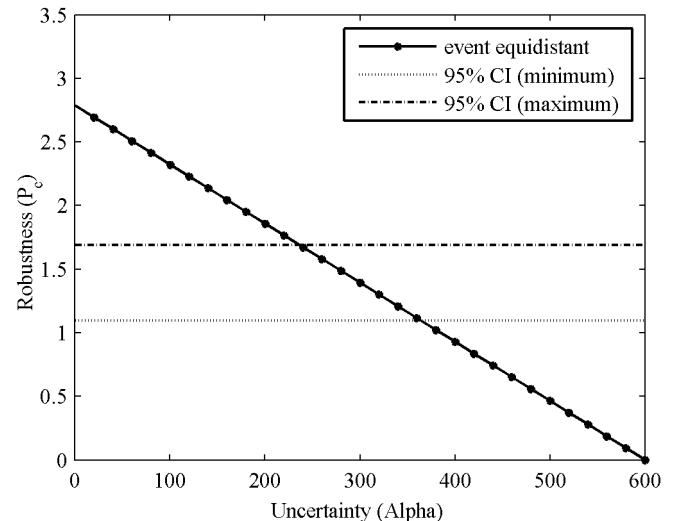


Figure 20 95% confidence interval for an equidistant event for uncertainty in actor energy

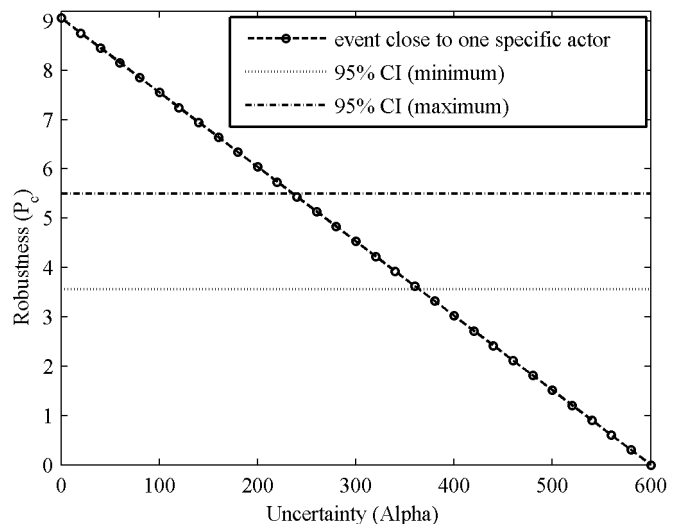


Figure 21 95% confidence interval for an event close to a specific actor for uncertainty in actor energy

will not exceed 1.9891 and not fall below a minimum value of 1.7332 for an event that is equidistant from all actors, and that the robustness of the decision will not exceed 3.2837 and not fall below a minimum value of 2.2592 for an event that is close to one specific actor. The average robustness value is 1.8612 for an equidistant event and 2.7715 for an event close to a specific actor, which indicates that on average the reliability of the IGDT model, (to choose suitable actors to respond to events relatively quickly, while reducing the number of interim messages transmitted within the WSN) is good. When the IGDT model is applied to actors' assigned to a specific cluster as in Figure 19, then the 95% confidence intervals for the three different scenarios can be seen in Figures 26, 27 and 28. There is a 95% confidence interval of (1.4087, 1.5468) for an event in a specific cluster corner, a 95% confidence interval of (2.0967, 2.3865), for an event almost equidistant from all

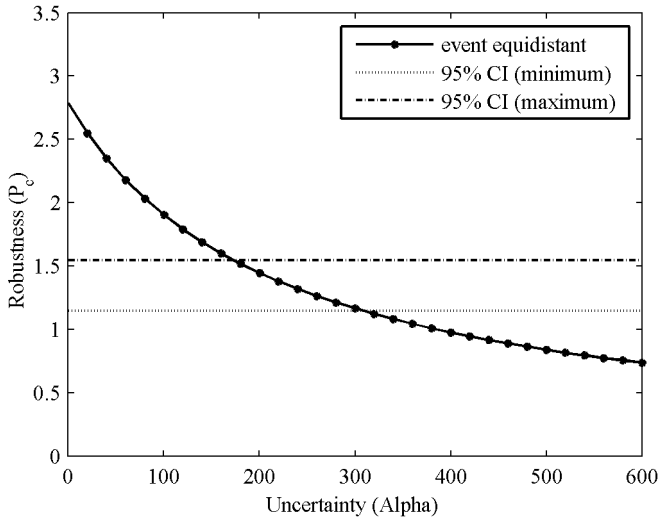


Figure 22 95% confidence interval for an equidistant event for uncertainty in actor time-of-arrival

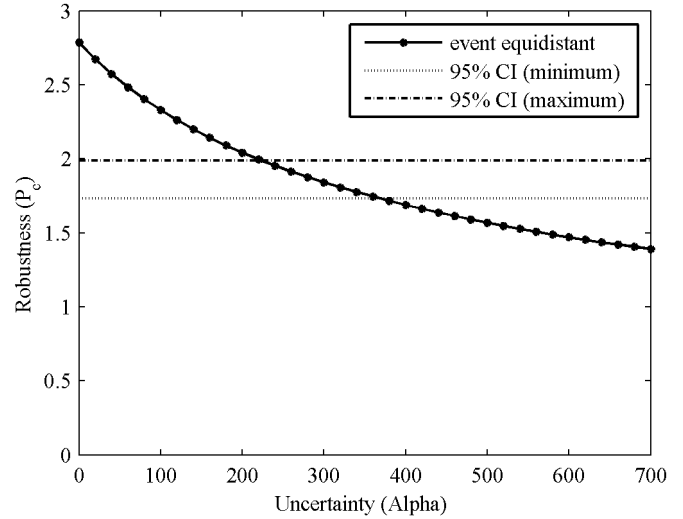


Figure 24 95% confidence interval for an equidistant event for uncertainty in actor location

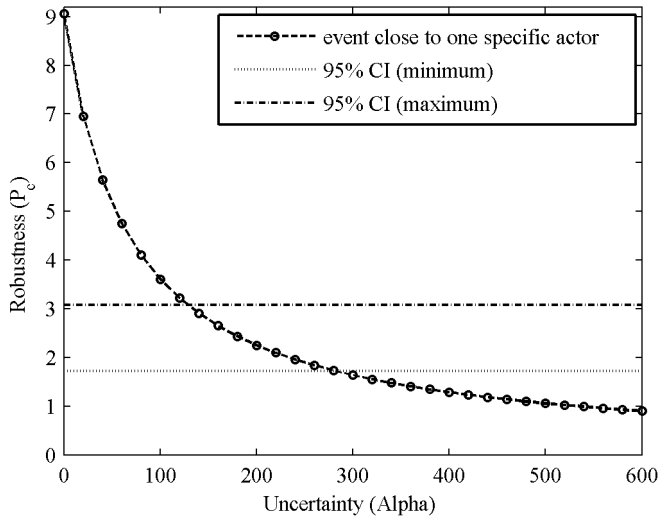


Figure 23 95% confidence interval for an event close to a specific actor for uncertainty in actor time-of-arrival

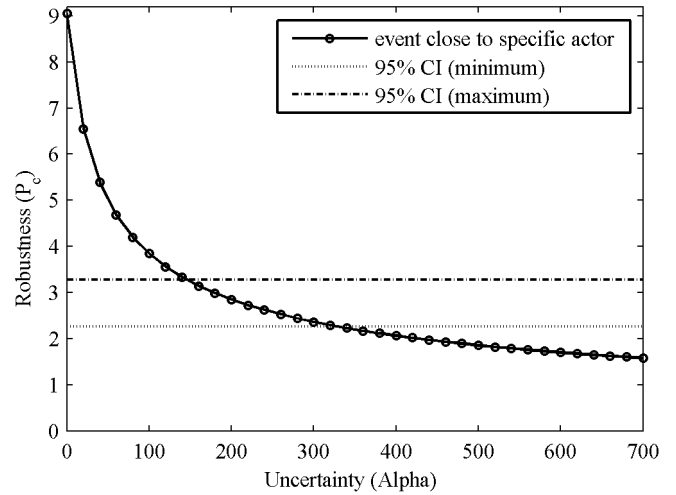


Figure 25 95% confidence interval for an event close to a specific actor for uncertainty in actor location

actors, and a 95% confidence interval of (1.6956, 1.8860) when a cluster actor is farthest away from an event in the centre of the application area. The average robustness value is 1.4777 for an event in a specific cluster corner, 2.2416 for an event almost equidistant from all actors, and 1.7908 for an event in the centre of the application area.

7 Conclusion

We have shown that info-gap decision theory can be used to choose actor(s) with sufficient resources to respond to an event when uncertainty of the actors' current energy levels and location exists. For swift response from the nearest actors with sufficient energy, a smaller level of uncertainty will be required. Where a response from any actor within a specified timeframe is required, then larger

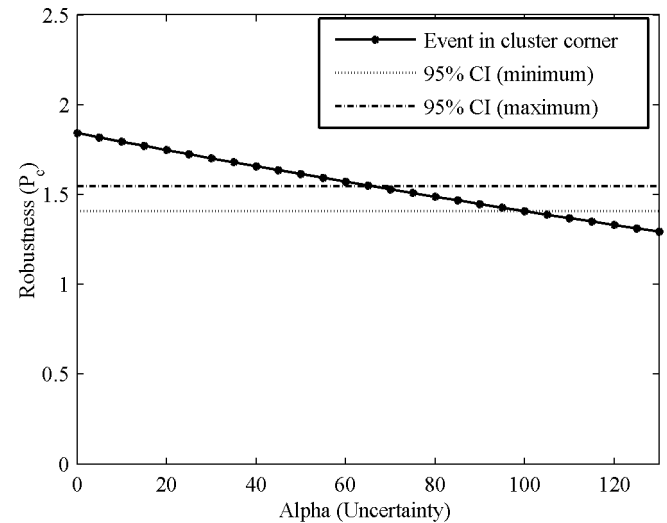


Figure 26 95% confidence interval for an event in cluster corner

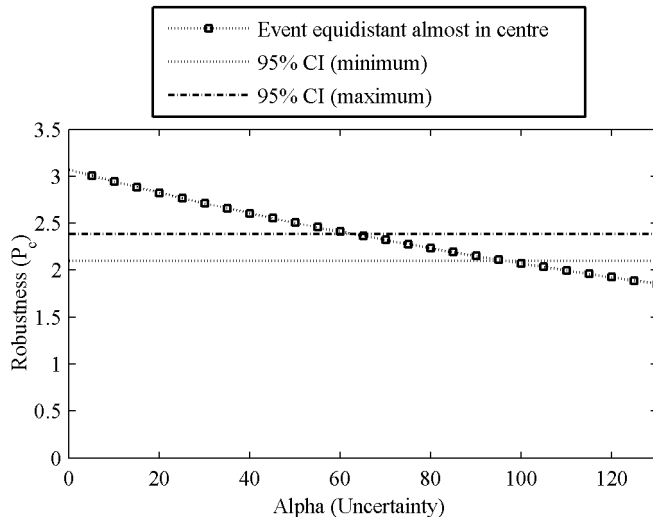


Figure 27 95% confidence interval for an event almost equidistant from all actors

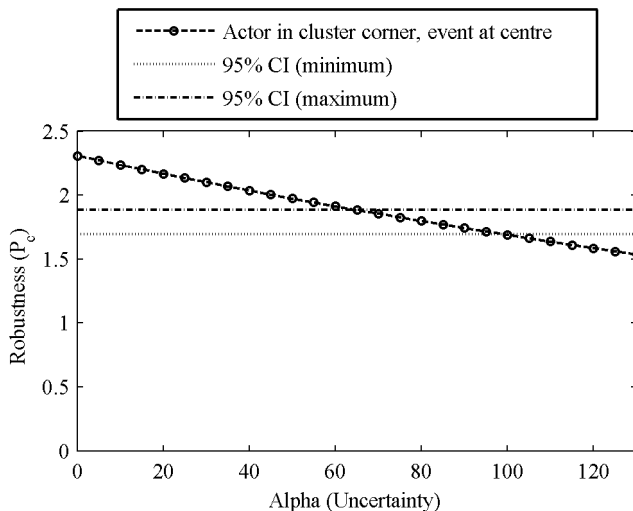


Figure 28 95% confidence interval for an event in centre of application area when cluster actor farthest away

uncertainty about other actors' energy and location information can be allowed as long as the robustness exceeds the critical value. When compared to clusters, the actor and event location within the cluster can lead to the assigned cluster actor not being the best actor to respond to the event. To improve the cluster approach, we propose the use of localised sensor information to determine the optimum actor(s) to respond to an event. This will increase the robustness of the decision as uncertainty about the actor's location increases.

The need to keep the amount of communication within a WSA to a minimum to preserve node battery lifetime as well as the need for a real-time response to an event means that actor coordination to determine which actor should respond to an event needs to be done as rapidly as possible. By using IGDIT we demonstrate that

a robust decision process to coordinate which actor(s) should respond to an event can be achieved reliably.

References

- Akyildiz, I., Su W., Sankarasubramanian Y. and Cayirci, E. (2002) 'Wireless sensor networks: A survey', *Computer Networks Journal*, Vol. 38, No. 4, pp:393–422.
- Akyildiz, I.F., Kasimoglu, I.H. (2004) 'Wireless sensor and actor networks: research challenges.', *Ad Hoc Networks*, Vol. 2, pp:351–367.
- Duncan, S.J., Bras, B., Paredis, C.J.J. (2008) 'An Approach to Robust Decision Making under Severe Uncertainty in Life-Cycle Design', *International Journal of Sustainable Design*, Vol. 1, No. 1, pp:45–59.
- Cheong, M. -P., Berleant, D., Shebl, G.B. (2008) 'Information Gap Decision Theory as a tool for Strategic Bidding in Competitive Electricity Markets', *In Proceedings of the 8th International Conference on Probabilistic Methods Applied to Power Systems*, Iowa State University, Ames, Iowa, September 12–16.
- Regan, H.M., Ben-Haim, Y., Langford, B., Wilson, W.G., Lundberg, P., Andelman, S.J., Burgman, M.A. (2005) 'Robust decision making under severe uncertainty for conservation management', *Ecological Applications*, Vol.15, No. 4, pp:1471–1477.
- 'Info-Gap Decision Theory: Decisions Under Severe Uncertainty', <http://www.info-gap.com>, Last accessed on 22 October 2010.
- Melodia, T., Pompili, D., Gungor, V.C. and Akyildiz, I.F. (2007) 'Communication and coordination in wireless sensor and actor networks.', *IEEE Transactions on Mobile Computing*, Vol. 6, No. 10, pp:1116–1129.
- Melodia, T., Pompili, D., Gungor, V.C. and Akyildiz, I.F. (2005) 'A distributed coordination framework for wireless sensor and actor networks.', *In Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, Urbana-Champaign, IL, USA, pp:99–110.
- Ngai, E.C.H., Lyu, M.R. and Liu, J. (2006) 'A real-time communication framework for wireless sensor-actuator networks.', *In Proceedings of the IEEE Aerospace Conference*, Big Sky, Montana.
- Demirbas, M. (2007) 'A transactional framework for programming wireless sensor/actor networks.', *In Proceedings of the 11th IEEE International Workshop on Future Trends of Distributed Computing Systems*, pp: 123–129.

- Rossi, L.F., Li, K., Yackoski, J. and Shen, C-C. (2007) 'Slime mold inspired coordinations for wireless sensor actor networks.', *In Proceedings of the First ACM workshop on Sensor and actor networks*, pp: 55-56.
- Bouhafs, F., Merabti, M. and Mokhtar, H. (2006) 'A coordination protocol for wireless sensor networks.', *In Proceedings of The 7th Annual PostGraduate Symposium on The Convergence of Telecommunications, Networking and Broadcasting, Computing and Mathematical sciences*, Liverpool John Moores university, UK.
- Ben-Haim, Y. (2006) 'Info-Gap Decision Theory: Decisions under severe uncertainty.', *Elsevier*, 2nd edition.
- Chinnappen-Rimer S. and Hancke, G.P. (2009) 'Actor Coordination in Wireless Sensor-Actor Networks', *2009 Annual IEEE India Conference (INDICON09)*, pp:1-4.
- Chen, M., Leung, V.C.M., Mao, S. (2009) 'Directional Controlled Fusion in Wireless Sensor Networks', *Mobile Networks and Applications (ACM/Springer)*, Vol. 14, No. 2, pp:220-229.