Distributed Algorithm for the Actor Coverage Problem in WSN-based Precision Irrigation Applications

Abdelouahid Derhab Nourredine Lasla
Department of Theories and Computer Engineering
CERIST research center, Algiers, Algeria

***Abstract*—In this paper, we study the actor coverage problem with the goal of meeting the requirements of precision irrigation applications in Wireless sensor and Actor Networks (WSANs), which are: (1) the volume of water applied by actors should match plant water demand and (2) minimizing over-irrigation to the least extent. We take a novel approach to define and resolve the actor coverage problem. Based on this approach, we propose two algorithms: Centralized Actor-Coverage-IRRIG (CACI) and Distributed Actor-Coverage-IRRIG (DACI). The existing centralized and distributed approaches for the minimum cost actor coverage problem in WSANs are not optimal for all metrics. The communication scheme of DACI is designed in the way that it can keep the advantages of the centralized and the distributed approaches without inheriting their weaknesses. DACI constructs an actor cover set with the same optimality cost as CACI while incurring low signaling overhead. Complexity analysis and simulations results show that CACI and DACI are both better than the existing centralized algorithm in terms of cover set optimality. Also, DACI is better than the existing distributed algorithm in terms of message overhead.**

I. Introduction

A Wireless sensor and Actor Networks (WSAN) [3] consists of a network of sensor nodes that can detect events in the environment and a network of actuator nodes capable of taking action against the detected events. There are an increasing number of application areas that require WSANs like: disaster relief operations, intelligent building, home automation, and environmental monitoring. For example, in fire detection applications, sensors can relay the exact origin and intensity of the fire to water sprinkler actors so that the fire can easily be extinguished before it spreads. Typically, a WSAN consists of sensors which sense the phenomena, a sink that collects the data from the sensors to process and actors that act upon the command sent by the sink. In the literature, such architecture is known as semi-automated architecture. An architecture in which sensor nodes send information to the actor nodes directly without the involvement of sink node is called an automated architecture.

The Coverage problem has extensively been studied in Wireless Sensor Networks (WSNs). However, an exhaustive literature review find only few papers on the topic of *actor coverage* in WSANs [2], [1], [9], [10], where it is often required that every point of the deployment area is covered by at least one actor. Akkaya et al. [2], [1], [9] studied the problem of finding the best layout for the actor nodes that provides maximum coverage of the network area and minimize the data collection delay. Vedantham et al. [10] have tackled the problem of minimizing actor resources for responding to an event in a particular region. If multiple actors are required to cover an event region, it might be necessary to ensure that these acting regions are non-overlapping or mutually exclusive in order to ensure uniform acting behavior over the entire event region, but if the acting regions are circular, the overlapping between these regions is inevitable. They have defined the problem as follows: *Given a set of actors in an event region, what is the minimum subset of actors (i.e., actor cover set) that covers the entire event region such that there is minimal overlap in the acting regions?* Their goal is to maximize the non-overlapped acting regions of each actor within the event region in order to utilize the actor resources to the least extent (see Figure 1). They have proposed centralized and fully distributed approaches for the minimum actor coverage problem. In the centralized approach, the sink node, based on full knowledge of network topology, constructs the actor cover set. Then, it issues a command to each actor in the set. In the fully distributed approach, the sensor that detects an event, sends a *Request* message to each actor in its local region. After that, a coordination process is triggered. In this process, each actor waits for an amount of time before sending a *Notify* message to each actor in its local region. This message contains the benefit value (i.e., size of the non-overlapped region) that the actor can bring in case it is selected. Each actor might update its benefit value after receiving the benefit value of another actor. The waiting time at each actor is inversely proportional to its benefit function. In this way, only actors with the best benefit function will be first chosen to act in the region. Contrary to the centralized approach, the fully distributed approach has the advantage of incurring less message overhead and responding to events more quickly. However, since each actor makes autonomous decisions and cannot envision the precise state of the whole network, the cost of the actor cover set constructed by the distributed approach will be less optimal than that of the centralized one. On the other hand, centralized approaches show poor performance in terms of message overhead and event-to-action delay.

Our work is similar to [10] in the sense that we aim to minimize the actor resources. However, we mainly focus on studying the minimum coverage problem with the goal of

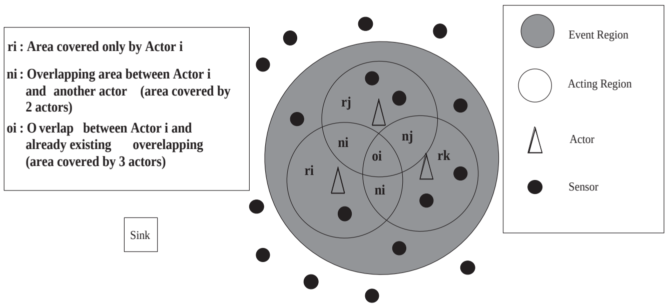

Fig. 1. Different types of regions used in Vedantham's algorithm

meeting the requirements of precision irrigation applications. Precision irrigation is defined as the accurate and precise application of water to meet the specific requirements of individual plants or management units and minimize adverse environmental impact. Hence, an important characteristic of a precision irrigation system is that the timing, placement and volume of water applied by actors (i.e., sprinklers) should match plant water demand. Over-irrigation invokes leaching of water and fertilizers affecting the environment. Hence, maximizing the non-overlapping area should be considered when designing a coverage configuration for precision irrigation applications.

In this paper, our original contributions are the following: First, we take a novel approach to define and resolve the actor coverage cost problem for WSN-based precision irrigation applications. Second, based on this new definition, we propose centralized and fully distributed algorithms. The distributed algorithm is designed in the way that it incurs the same actor coverage cost as the centralized one with lower message overhead. Third, the proposed protocols outperform the existing stat-of-the-art algorithms in terms of actor resources and size of non-overlapping area.

The rest of the paper is organized as follows: In Section II, we motivate and formally define the actor coverage problem for WSN-based precision irrigation applications. Section III proposes the centralized algorithm, while Section IV proposes the distributed and and the localized algorithm that address this problem. The analysis of complexity as well as the results of a simulation study are presented in Section V. Finally, Section VI concludes the paper.

II. Motivation and Problem Formulation

Vedantham et al. [10] intuitively suppose that maximizing the non-overlapped regions will lead to minimizing the number of actors selected to cover the event region, and hence there will be minimal overuse of actor resources. However, this claim is provided with no proof. They consider that the cost of an overlapped region, whether it is covered by two actors or more is the same and it depends only on its region size and they ignore the number of actors covering the region. As each new overlap incurs an additional amount of actor resources, this method cannot lead to an actor cover set with minimum cost. In addition, minimizing the number of actors implies also minimizing the water resources dissipated by the actor cover set selected to cover the event region. The actor's acting region might cover areas inside and outside the event region, and hence this method is not suitable for precision irrigation applications that only consider the plant demands, i.e., inside the event region. As the acting region of the sprinklers are circular, the overlapping between them is inevitable. Our problem focus on selecting the actor cover set that: (1) maximizes the non-overlapping area within the event region in order to match plant water demand, and (2) minimizes the overuse of water applied to the areas inside the event region. Although we do not consider the areas outside the event region but these areas, which contains plants and they are part of the agricultural field, are irrigated by sprinklers before reaching the required threshold to start irrigation. Our literature review [4], [8] found that the areas surrounding the event region as well the event region have very close values in terms of water availability (i.e., soil moisture), and hence their water demands are also very close. Figure 2 depicts the spatial distribution of soil moisture for a field [4]. If we want to irrigate the *very dry* area shown in the figure, the actor cover set that is going to cover the very dry area might irrigate its surrounding area, which is the *dry* area. The plants in the dry area will be subject to little over-irrigation as their water demands are also high.

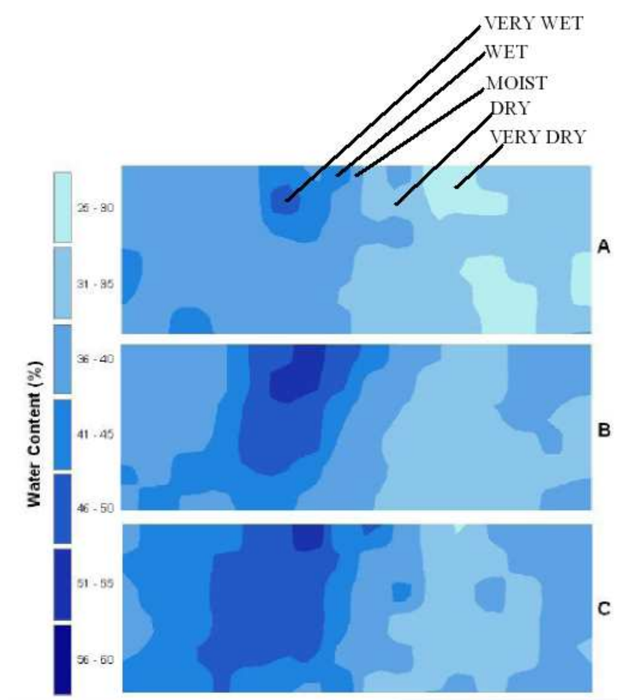

Fig. 2. Soil moisture maps for 3 different dates: A, B, and C. Colors indicate percentage of water in the soil, from 25% (light blue) to 60% (dark blue)

A. Network model

We consider a model of WSAN in which sensor and actor nodes as well as the sink node are deployed in the sensor/actor field. We assume that each node a_i is aware of its location l_i via a localization technique [5], and there is an underlying routing protocol for delivering messages to any sensor or actor [7]. The acting range and the acting region of each actor a_i, are denoted by AR_i and A_i respectively.

B. Definitions

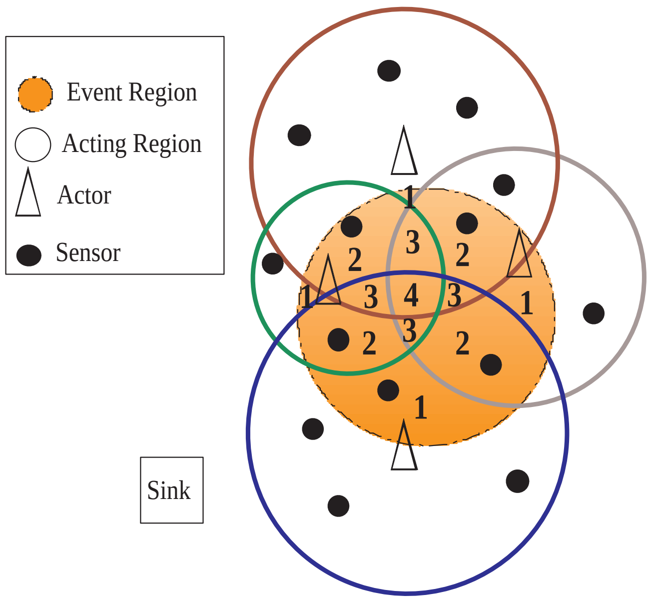

Fig. 3. Actor coverage configuration

A point in the event region R is said to be *K-actor-covered* ($K \geq 1$) if it is within the acting range of K active actors.

A region A is said to be *K-actor-covered* ($K \geq 1$) if every point in A is *K-actor-covered*. $\delta(A)$ denotes the actor coverage degree of A, which is in this case K.

Given an event region R, and a set of active actors whose acting range intersects with R, denoted by Ω, the set system $\xi = (R, \Omega)$ is called *actor-coverage configuration* if each point in R is within the acting range of at least one actor in Ω. Figure 3 shows an example of a configuration consisting of four actors and an event region. As shown in figure 3, a configuration ζ will partition the event region R into a set $\mathcal{P}$ of disjoint sub-regions. The actor coverage degree of each sub-region is shown on the figure.

We define $S_i = \{q \in \mathcal{P} | \delta(q) = i\}$ to be the set of sub-regions within the event region R whose actor coverage degree is i. The maximum actor coverage degree of a configuration ξ, denoted by $\Delta(\xi)$, is the largest coverage degree over all the subregions of the event region R. It is clear that $\cup_{i=1}^{\Delta(\xi)} S_i$ is $\mathcal{P}$.

Let us consider an actor a_i that covers a region $A \subseteq R$ whose size is $Size(A)$, the amount of resources dissipated by a_i on A is $Res(A, a_i) = r_i \times Size(A) \times T_i$, such that T_i is the time required by actor a_i to carry out an action and r_i is the amount of resources dissipated/time unit/space unit. For the sake of simplicity, we omit T_i and r_i in the rest of the paper. The amount of resources dissipated by a configuration $\xi = (R, \Omega)$ on the event region R is defined by:

$$Res(\xi) = \sum_{a_i \in \Omega} Size(A_i \cap R)$$

It can also be written as: $\sum_{i=1}^{\Delta(\xi)} \sum_{a \in S_i} i \times Size(a)$

$$\begin{aligned}
Res(\xi) &= \sum_{i=1}^{\Delta(\xi)} \sum_{a \in S_i} i \times Size(a) = \sum_{a \in S_1} Size(a) + \sum_{i=2}^{\Delta(\xi)} \sum_{a \in S_i} i \times Size(a) \\
&= \sum_{a \in S_1} Size(a) + \\
&\quad \sum_{a \in S_2} Size(a) + \sum_{a \in S_2} Size(a) + \\
&\quad \sum_{a \in S_3} Size(a) + \sum_{a \in S_3} 2 \times Size(a) + \\
&\quad \vdots \\
&\quad \sum_{a \in S_{\Delta(\xi)}} Size(a) + \sum_{a \in S_{\Delta(\xi)}} (\Delta(\xi) - 1) \times Size(a) \\
&= Size(R) + \sum_{i=2}^{\Delta(\xi)} \sum_{a \in S_i} (i-1) \times Size(a) \qquad (1)
\end{aligned}$$

We can notice that $Res(\xi)$ (Equation 1) is composed of two parts: (a) $Size(R)$, which is the exact amount of resources required by the event region R, and (b) the other part, which represents the extra amount of resources incurred by ξ.

The set system $\xi^M = (R_M, M)$ is called *partial actor-coverage configuration* if: $R_M \subseteq R$ and each point in R_M is within the acting range of at least one actor in $M \subseteq \Omega$. $\mathcal{P}^{\mathcal{M}}$ is the set of disjoint sub-regions within R_M.

We define $S_i^M = \{q \in \mathcal{P} | \delta(q) = i\}$ to be the set of sub-regions within the event region R_M whose actor coverage degree is i.

We define the minimum coverage cost problem as follows: *Given a set of actors in an event region, what is the set of actors $M \subseteq \Omega$ such that: the region covered by M within the event region R is $R_M = R$ and the extra amount of actor resources, $Res(\xi)$, is minimal?* We try to minimize the following function:

$$f^M = \sum_{i=2}^{\Delta(\xi^M)} \sum_{a \in S_i^M} (i-1) \times Size(a)$$

In order match the water demands of the plants inside the event region, we also need to maximize the non-overlapping areas covered by M. To achieve both goals, we try to maximize the following objective function:

$$Maximize(\alpha \times \sum_{a \in S_1^M} Size(a) - \beta \times f^M)$$

In this objective function, α and β are two constant weights such that $\alpha + \beta = 1$.

III. Centralized Actor-Coverage-IRRIG algorithm

In this section, we present the algorithm called Centralized Actor-Coverage-IRRIG (CACI). This algorithm adopts a centralized approach, in which the sink node constructs the actor cover set based on full knowledge of network topology.

When sensor nodes detect the existence of a new event, they report the event to the sink. Based on the reports, we assume that the sink is able to determine the location and the radius of the event. The execution of CACI is shown in Algorithm 1. In the algorithm, the sink node selects at each stage, the actor that incurs the maximum benefit function. The benefit function is defined as follows:

$(\alpha \times new\_area_i - \beta \times \Delta C_i)$, where: new_area_i is the size of the new covered area within R by actor a_i, and $\Delta C_i = (f^{M \cup a_i} - f^M)$ is the additional cost that a_i will bring in case it is selected.

The actor that does not bring new covered is not included in M. The selected actor is added to M (i.e., the set of actors selected so far), and the algorithm terminates when the event region R is totally covered by M. The sink then sends for each actor in M a *command* message to trigger the irrigation.

Algorithm 1 CACI Algorithm at node a_i

$M = \emptyset$;
$R_M = \emptyset$;
$\omega = \Omega$;
$f^M = 0$;
while $R \not\subseteq R_M$ **do**
    $max\_function = -\infty$;
    for each $a_i \in \omega$ **do**
        if a_i covers new area in R **then**
            $\Delta C_i = f^{M \cup a_i} - f^M$
            $function = \alpha \times new\_area_i - \beta \times \Delta C_i$
            if $function > max\_function$ **then**
                $max\_function = function$;
                $selected = a_i$;
            end if
        else
            $\omega = \omega - a_i$;
        end if
    end for
    $M = M \cup selected$;
    $R_M = R_M \cup (A_{selected} \cap R)$
    $\omega = \omega - selected$;
end while
Send command (start irrigation) to every actor M;

IV. Distributed Actor-Coverage-IRRIG

In this section, we present the distributed and localized version of Centralized Actor-Coverage-IRRIG called Distributed Actor-Coverage-IRRIG (DACI). We first provide an overview of Vedantham's distributed algorithm called the Neighborhood Back-off (NB), and then we describe our algorithm DACI.

A. Vedantham's distributed algorithm

In NB, the authors have defined the dependency region of a sensor D_S as the one with radius equal to the sum of sensing and acting range, while that of an actor D_A as twice the acting range. The algorithm is executed in two phases: (1) *notification phase*, where sensors nodes in the event region inform the actors about the existence of an event, and (2) *coordination phase*, in which the actors coordinate between each other to construct the actor cover set.

When a sensor detects the presence of an event, it reports the sensed information to all the actor in its dependency region by sending *REQUEST* message. When an actor receives such a message from a sensor, it first determines the additional event area covered by the sensor and adds that region to already existing event area. The actor defines the intersection of the event area with its acting region to be the the benefit function (i.e., the size the non-overlapped region within the event region covered by the actor). The actor waits for a given time to ensure reception of all *REQUEST* message within its dependency region. Then, it waits for an amount of time inversely proportional to its benefit function before sending a *Notify* message to each actor in its local region. When the waiting time expires, it means that the actor is ready to act on the region. Upon receiving the *Notify* message, each actor will update its benefit function and its waiting time. The new benefit function will be the size of the non-overlapped region covered by the actor.

The main drawback of this approach is that the actors involved in the algorithm are not within the dependency region of each other, and hence their coordination is imperfect. It is possible that several actors decide to to act at the same time, which leads to less optimal actor cover set.

B. DACI description

The communication scheme proposed by NB achieves less optimal actor cover set than the one proposed by the centralized algorithm. In order to address this issue, The communication scheme of the distributed algorithm should be designed in the way that the actors that share the same event region must know each other. To do so, every sensed event is disseminated within a radius of $(2 \times R_{max} + AR)$ from the sensor node, where R_{max} is the maximum radius for an event region. We assume that every actor is aware of its $(2(R_{max} + AR))$-distance neighborhood.

Every sensor node inside R disseminates the event it detects (i.e., the amount of moisture in the ground is below the required threshold) to all the actors in its $(2(R_{max} + AR))$-neighborhood. When an actor receives such information, it first determines the additional event area covered by the sensor and adds that region to already existing event area. The actor waits for a given time to ensure reception of all the events within the new formed event region. When this time expires, all the actors that intersect with the new event region know each other.

The main advantage of DACI is that it does not require any explicit coordination between the actors as NB does. By explicit coordination, we mean that signaling messages are exchanged between actors to coordinate actions and construct the actor cover set. As all actors in DACI can calculate the same Ω set, then implicit coordination can be exploited. Implicit coordination means that coordination information is not communicated explicitly by signaling messages, but is inferred from the local environment. It is obvious that this communication scheme incurs less message overhead than the one proposed in NB algorithm. The execution of DACI is shown in Algorithm 2.

Algorithm 2 DACI Algorithm at node a_i

$M = \emptyset$;
$R_M = \emptyset$;
$\omega = \Omega$;
$f^M = 0$;
while $R \nsubseteq R_M$ **do**
  $max\_function = -\infty$;
  for each $a_i \in \omega$ **do**
    if a_i covers new area in R **then**
      $\Delta C_i = f^{M \cup a_i} - f^M$
      $function = \alpha new\_area_i - \beta \Delta C_i$
      if $function > max\_function$ **then**
        $max\_function = function$;
        $selected = a_i$;
      end if
    else
      $\omega = \omega - a_i$;
    end if
  end for
  $M = M \cup selected$;
  $R_M = R_M \cup (A_{selected} \cap R)$
  $\omega = \omega - selected$;
  if $selected = a_i$ **then**
    START Irrigation;
  end if
end while

V. PERFORMANCE EVALUATION

A. Analysis of complexity

In this section, we analyze the communication complexity (CC) and time complexity (TC) of the proposed protocols CACI and DACI and compare it with Vendatham's centralized and distributed algorithm [10]. The communication complexity measures the number of one-hop transmissions required by the algorithm to perform the action. The time complexity, or event-to-action delay, measures the time difference between the occurrence of the event and the execution of its corresponding action. The worst-case comparison between these algorithms is shown in Table I.

TABLE I
COMPLEXITY COMPARISON OF ACTOR COVERAGE ALGORITHMS

	CC	TC
Vendatham's centralized	$O((K+I)\sqrt{N})$	$O(2\sqrt{N}+Y)$
Vendatham's distributed	$O(K\|\Omega\|D_S + \|\Omega\|^2 D_S)$	$O(D+W+Y)$
CACI	$O((K+I)\sqrt{N})$	$O(2\sqrt{N}+Y)$
DACI	$O(K(2R_{max}+AR)^2)$	$O(D+Y)$

We assume that N sensor and actor nodes are randomly distributed in the field A. The node density remains constant when the number of nodes increases, and the area A grows with N. Since the expected distance of two uniformly sampled points within a square of size $a \times a$ scales with a [6], it is expected that the number of hops between two random nodes increases proportional to N. We also assume that there are K sensors within the event region. The average distance between the event and the actor in Ω is denoted by D. W denotes the actor's waiting time in NB algorithm. Y denotes the waiting time to ensure reception of all the events within the event region. The size of the actor cover set constructed by the algorithm is denoted by I.

In the centralized algorithms, the K sensor nodes within the event region inform the sink node about a new event. The sink sends a command message to each actor in the actor cover set M. As the average number of hops between two random nodes is proportional to $\sqrt{N}$. This operation leads to a communication cost of $O((I+K)\sqrt{N})$ and a response time of $O(2\sqrt{N}+Y)$.

In NB algorithm, when a sensor detects a new event, it sends REQUEST message to all the actors in its dependency region. This notification phase incurs a communication cost of $O(K|\Omega|D_S)$. In the coordination phase, each actor sends a *NOTIFY* message to $|\omega|-1$ actors in its dependency region in order to construct the actor cover set, which leads to a communication cost of $O(|\Omega|^2 D_S)$. The event needs to travel a distance proportional to $O(D)$ to reach an actor in Ω. Each actor waits for $O(W)$ time to receive the rest of events in the event region, then waits for another $O(Y)$ time before acting, which leads to an event-to-action delay proportional to $O(D_S+W+Y)$.

In DACI, a sensor node disseminates the event within a zone of radius equals to $2R_{max}+AR$. This notification cost incurs a communication cost of $O(K(2R_{max}+AR)^2)$ and a time complexity of $O(D)$. After receiving the first event, each actor needs to wait for $O(Y)$ time before executing the algorithm. The main advantage of DCMI is that there is no need for signaling overhead to execute the coordination phase and its event-to-action delay is less than that of NB algorithm.

B. Simulation results

To evaluate the optimality of the actor cover set constructed, we compare the performance of Vendatham's centralized algorithm [10], CACI, and DACI. In our simulation, 225 actors with acting range 40m are deployed on an area of $100m \times 100m$ according to a uniform grid. α and β are set to 0.9 and 0.1 respectively. The following metrics are evaluated for the actor coverage algorithms.

1) *Extra cost:* is defined as the amount of extra resources dissipated on the event region.
2) *Number of actors:* is the number of actors selected by the algorithm to cover the event region.
3) *Size of non-overlapping area:* is defined as the size of areas within the event region that are covered by only one actor.

Figure 4 shows the performance results of the different actor coverage algorithms as a function of event radius. The first observation we draw from the figure is that CACI and DACI show the same results as they are able to construct the same actor cover set. The second observation is that for small event regions (less than 35), all the algorithms select the same actor cover set, and hence they show the same results. In this case, the actor cover set constructed is exactly one actor and is sufficient to cover the event region. CACI

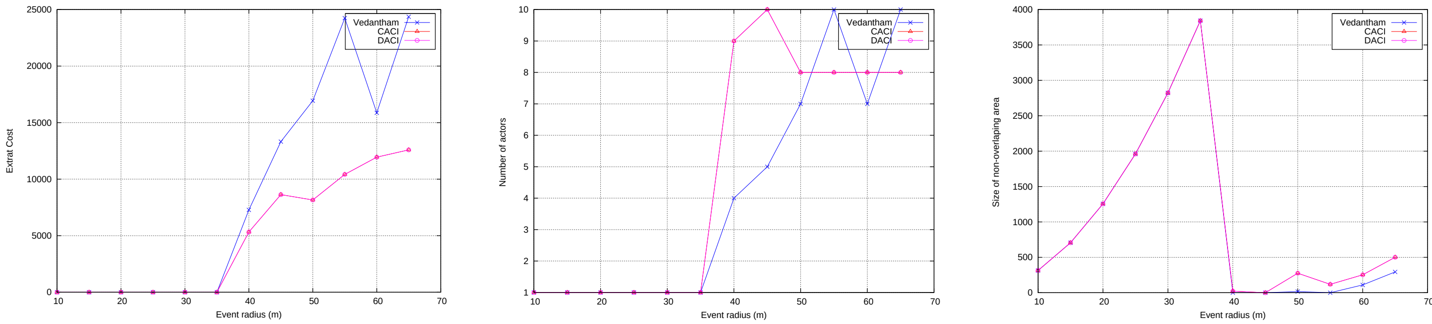

Fig. 4. Performance under different event size

and DACI outperforms Vendatham's centralized algorithm in terms of extra cost and size of non-overlapping area as they are designed to optimize the two metrics above. Vendatham's algorithm, on the other hand, is designed to minimize the number of actors required to cover the event region. So it outperforms CACI and DACI in some cases when the event region is larger than 35.

VI. Conclusion

In this paper, we have proposed a new approach to address the actor coverage problem in WSANs. The approach focuses on satisfying the requirements imposed by the precision irrigation applications, which are: (1) maximizing the non-overlapping area within the event region, and (2) minimizing the extra cost incurred by the actors when irrigating the event region. Based on this new approach, we have proposed two algorithms named: Centralized Min-Cost-IRRIG (CACI) and Distributed Min-Cost-IRRIG (DACI) respectively. The communication scheme of DACI is designed in the way that it can keep the advantages of the centralized and the distributed approaches without inheriting their weaknesses. Complexity analysis and simulation studies have shown that DACI constructs an actor cover set with the same optimality cost as CACI, and is better than NB algorithm in terms of communication complexity and time complexity as it uses an implicit coordination instead of an explicit one to select the actor cover set.

References

[1] K. Akkaya and S. Janapala. Maximizing connected coverage via controlled actor relocation in wireless sensor and actor networks. *Elsevier Computer Networks Journal*, 52(14), 2008.

[2] K. Akkaya and M. Younis. Coverage and delay aware actor placement mechanisms for wireless sensor and actor networks. *International Journal of Sensor Networks*, 3(3), 2008.

[3] Ian F. Akyildi and Ismail H. Kasimoglu. Wireless sensor and actor networks: research challenges. *Ad Hoc Networks Journal*, 2(4):351–367, 2004.

[4] L. Brocca, M.Galli, and M.Stelluti. Preliminary analysis of distributed in situ soil moisture measurements. *Advances in Geosciences*, 2, 2005.

[5] N. Bulusu, J. Heidemann, and D. Estrin. Gps-less low cost outdoor localization for very small devices. *IEEE Personal Communications, Special Issue on Smart Spaces and Environments*, 2834, 2000.

[6] S. R. Dunbar. The average distance between points in geometric figures. *College Mathematics Journal*, 28(3):187–197, May 1997.

[7] Arjan Durresi, Vamsi Paruchuri, and Leonard Barolli. Delay-energy aware routing protocol for sensor and actor networks. In *11th International Conference on Parallel and Distributed Systems (ICPADS'05)*, 2005.

[8] Alison C. McCarthy, Nigel H. Hancock, and Steven R. Raine. Variwise: a general-purpose adaptive control simulation framework for spatially and temporally varied irrigation at sub-field scale. In *Computers and Electronics in Agriculture*, volume 70, 2010.

[9] B. McLaughlan and K. Akkaya. Coverage-based clustering of wireless sensor and actor networks. In *IEEE International Conference on Pervasive Services (ICPS'07)*, 2007.

[10] Ramanuja Vedantham, Zhenyun Zhuang, and Raghupathy Sivakumar. Mutual exclusion in wireless sensor and actor networks. In *3rd Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON 2006)*, 2006.