This is the Pre-Published Version.

The following publication W. Zhu, J. Cao and M. Raynal, "Energy-Efficient Composite Event Detection in Wireless Sensor Networks," in IEEE Communications Letters, vol. 22, no. 1, pp. 177-180, Jan. 2018 is available at https://doi.org/10.1109/LCOMM.2017.2764458.

Energy-efficient Composite Event Detection in Wireless Sensor Networks

Weiping Zhu*, Jiannong Cao† and Michel Raynal‡
*International School of Software, Wuhan University, China
†Department of Computing, The Hong Kong Polytechnic University, Hong Kong
‡Institut Universitaire de France & IRISA-INRIA, Campus de Beaulieu, 35042 Rennes Cedex, France
Email: wpzhu@whu.edu.cn, csjcao@comp.polyu.edu.hk, raynal@irisa.fr

***Abstract*—Composite event detection is one of fundamental tasks for wireless sensor networks. In existing approaches, typically, a routing tree is used to enable information exchange among sensor nodes and collaborative detection of composite events. However, such a tree is not optimal in terms of energy efficiency because the relations included in composite events have not been fully utilized. In this paper, we propose a new type of routing tree called event detection tree (EDT) to achieve energy-efficient composite event detection. EDT reduces the amount of data to be transmitted by aggregating data into events, at the cost of an increased distance in the data transmission to achieve such aggregations. EDT achieves a trade-off of them to minimize the overall energy consumption. Simulation results show that our approach outperforms existing approaches and yields energy savings of up to 20%.**

***Index Terms*—Composite event detection, energy efficiency, WSN**

I. Introduction

Wireless Sensor Networks (WSNs) are often used to detect a change of a real-world state (e.g., a rise of the temperature)[1]. Such changes are called events. Events can be classified as primitive events and composite events. A primitive event can be directly detected by sensors, and a composite event is defined based on multiple correlated sub-events that are primitive events or other composite events. For example, a composite event that indicates a potential collision in a street is based on the sub-events that indicate the status of vehicles. The relations among the sub-events are defined by the semantics of application requirements. In the above example, it can be inferred that there would be a risk of vehicle collision if two vehicles move in the opposite directions in the same lane, and the distance between them decreases with time. In a typical composite event detection in WSNs, a routing tree is built for efficient information transmission among the sensor nodes, thus facilitating a collaborative detection of the composite event. Owing to the limited power supply of sensor nodes, the energy efficiency of the routing tree is a major concern.

Most of the existing studies on composite event detection [2], [3], [4], [5], [6], [7] utilize the opportunistic encounters of the sub-events in a routing tree to aggregate the sub-events into composite events. The routing trees built in these approaches are not optimized according to the events. Some data aggregation studies [8], [9], [10], [11], [12] can be modified to build the optimal routing trees, however these approaches support only non-decreasing aggregation functions, i.e., the amount of data for a composite event is not less than that for each of its sub-events. This assumption is not true in many event detection applications. For example, the amount of data for a potential vehicle collision event is much less than the amount of data for each of its sub-events.

In this letter, we propose a new type of routing tree for energy-efficient composite event detection; this routing tree is called *event detection tree* (EDT). EDT explicitly utilizes event relations to reduce the amount of data to be transmitted, at the cost of an increased distance in the data transmission. The overall energy consumption can be minimized based on a trade-off of them.

II. System Model and Problem Formulation

Event Specification Tree (EST) is a tree used to define a composite event of interest. The root of the tree corresponds to the specified event and the child nodes of the root correspond to its sub-events. This definition is iterative until the sub-events are primitive events. The relation is defined by a user-specific function that maps sub-events to the corresponding composite event. In particular, if we consider only the change in the amount of data after aggregation, the function is called an *aggregation function*. In an EST, the nodes whose data amount is less than any of its sub-events are called *checkpoints*.

In WSNs, we need to identify the sensor nodes required for event detection and connect them via an EDT. The primitive events can be detected directly by individual sensor nodes. Therefore, the corresponding sensor nodes (which are called *source nodes*) in the EDT can be determined in a straightforward manner. The composite events can only be detected by aggregating the partial detection results at intermediate sensor nodes. Therefore, new sensor nodes, in addition to the source nodes, are required in the EDT in order to facilitate the detection of composite events. In an EDT, a sensor node forwards its detection results to its parent node, and, based on the EST, the parent node checks whether a corresponding composite event occurs.

An EDT is denoted by $T(V, E)$, where V and E denote sensor nodes and communication links, respectively. r is the root. The amount of data for node $v \in V$ is denoted by $d(v)$, and the Euclidian distance between the two nodes that are incident to e is denoted by $l(e)$. Each edge $e \in E$ has a weight $c(e)$ that denotes the average unit cost of e for transmitting data [10]. In practice, $c(e)$ depends not only on $l(e)$, but also the technology employed in the MAC layer (e.g., handling transmission collisions and package loss). Our algorithm works on top of the MAC layer, and hence does not restrict the data transmission model to compute $c(e)$. To simply our computation, we use the following model:

$$c(e) = \delta \cdot l(e)^{\gamma} + \varepsilon \quad (1)$$

where ε is the energy consumption per bit to operate the transmitter/receiver circuit, and γ and δ are two parameters of radio transmission. The parameter setting follows [10]: $\gamma = 2$, $\delta = 100$ pJ/bit/m^2, $\varepsilon = 40$ nJ/bit. The energy consumption at the link e is $c(e)d(v)$ where v is the transmitter.

An energy-efficient EDT aims to reduce the amount of data and the distance in the data transmission. These two objectives are often contradictory. Some routing trees can reduce the transmission distance, but miss the opportunity to achieve early aggregation following the EST, thus resulting in increased energy consumption. Therefore, an proper routing tree is required to achieve the optimal energy consumption.

Composite event detection can be achieved by a two-phase solution. In the first phase, an EDT is built in the WSN. In the second phase, the composite event is detected based on the EDT, similar to data aggregation [8], [9], [10]. Building the EDT is critical to the entire process. Therefore, we formulate it as shown below and investigate it in this letter: Given a WSN that consists of a collection of sensor nodes and a sink node *r*, and a composite event defined by an EST *est*, assuming that according to *est*, all source nodes *srcNodes* are determined, the objective is to find an EDT connecting *srcNodes* to *r* such that the total energy consumption of the EDT is minimized.

III. The Algorithm

We propose the Energy-efficient Event Detection Tree Building Algorithm (EEDT) to solve the problem. In this algorithm, a central server stores the EST and the global information of the WSN. The EDT is computed at the central server, and then, it is disseminated to the sensor nodes.

A. Design Rationale

We modify data aggregation algorithms to implement composite event detection following two principles: supporting complex event specifications and supporting generic aggregation functions.

Following the first principle, the aggregation should be based on an EST, thus allowing the events to aggregate only if they satisfy certain relations. Following the second principle, if there is no checkpoint, data aggregation algorithms can be directly used to detect the events; if some checkpoints exist, further considerations are required to handle the aggregation based on event relations. Existing data aggregation algorithms only minimize the transmission cost; therefore, they may miss the opportunity of early aggregation that can reduce the amount of data to be transmitted (corresponding energy saving is called aggregation opportunity loss). One reasonable

Algorithm 1: Energy-efficient EDT Building Algorithm

$S = srcNodes \bigcup r, E^* = \varnothing$
foreach node $i \in srcNodes$ **do**
  $w(i) = d(i)$
endfch
$w(r) = 0$
while $S \neq \{r\}$ **do**
  foreach node pair (u, v) $(u \in S,\ v \in S)$ **do**
    if $v = r$ **then**
      $k(u, v) = w(u) \cdot l(u, v)$
    else
      $k(u, v) = \frac{w(u)w(v)(w(u)+w(v))}{w^2(u)+w^2(v)} \cdot l(u, v)$
    end
    $optl(u, v) = 0$
    compute *ch*(*u*) and *ch*(*v*), the nearest checkpoint of u and v, respectively
    if neither *ch*(*u*) nor *ch*(*v*) is an ancestor of both *u* and *v* **then**
      compute $ch(u).optl$ and $ch(v).optl$ based on equation 2
      /* if *ch*(*u*) or *ch*(*v*) does not exist, its *optl* is set to 0 */
      $optl(u, v) = \max(ch(u).optl, ch(v).optl)$
    end
    $tc(u, v) = k(u, v) + optl(u, v)$
  endfch
  find the minimum cost perfect matching of S based on tc; let the matched node pairs be $(u_1,\ v_1)....(u_m,\ v_m)$ $(m = \lceil |S|/2 \rceil)$
  add the shortest path between u_p and v_p $(1 \leq p \leq m)$ to E^*
  transmitter $= \varnothing$
  foreach node pair (u_p, v_p) $(1 \leq p \leq m)$ **do**
    if $u_p \neq r$ *and* $v_p \neq r$ **then**
      set $u_p.selected = 1$ with a probability of $\frac{w^2(u_p)}{w^2(u_p)+w^2(v_p)}$
      $v_p.selected = 1 - u_p.selected$
    else
      $s.selected = 1$ $(s = \{u_p, v_p\} - r)$
      $r.selected = 0$
    end
    let node $s \in \{u_p, v_p\}$ satisfy $s.selected = 0$ and t is the other node in $\{u_p, v_p\}$
    transmitter = *transmitter* $\cup \{s\}$
    node s sends its events to node t based on E^*
    node t performs aggregation based on the EST and updates the aggregated amount of data in $w(t)$
  endfch
  $S = S -$ *transmitter*
end
return E^*

routing selection is based on the comparison of the aggregation opportunity loss with the energy consumption due to increased transmission distance to achieve aggregation. However, the aggregation opportunity loss can be determined only after building the EDT, from which the increase in the amount of data and distance can be determined. We propose a method to estimate the aggregation opportunity loss. Assuming that early aggregation cannot be achieved at a checkpoint *c*, the upper bound of the potential loss is

$$c.optl = [\Sigma_{i \in children(c)} d(i) - d(c)] \cdot [\delta \cdot l(c, r)^{\gamma} + h(c, r) \cdot \varepsilon] \quad (2)$$

where $l(c, r)$ and $h(c, r)$ denote the distance and the number of hops, respectively, between c and r, and δ, γ, and ε follow our system model. The problem followed is how to compute $l(c, r)$ and $h(c, r)$, which would require knowledge of the location of *c*. We estimate its location as the weighted centroid of its child nodes. This catches the essences of early aggregation that when all the child nodes transmit their events to their parent

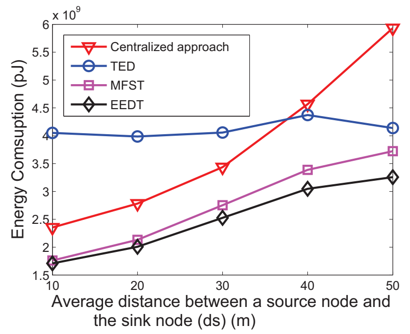

Fig. 1. Comparison of energy consumption for different approaches, by varying the distance between a source node and the sink node

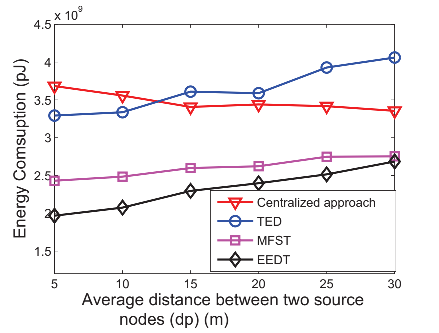

Fig. 2. Comparison of energy consumption for different approaches, by varying the distance between two source nodes

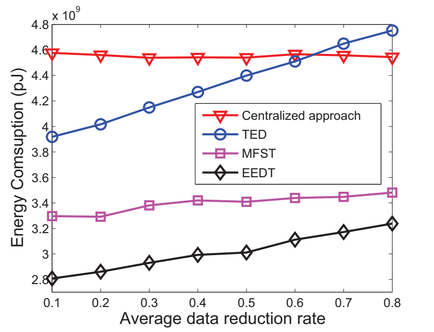

Fig. 3. Comparison of energy consumption for different approaches, by varying the data reduction rate

node, the total energy consumption is minimized.

B. The Algorithm

EEDT is shown in Algorithm 1. This algorithm is based on MFST [10]. First, all the source nodes *srcNodes* and the sink node *r* are added to S, and the EDT E^* is initialized to be empty. The process from line 6 to line 38 is an iterative one. In each iteration, the nodes pair up and transmit data from or to their partner nodes. From line 7 to line 20, the total cost of each node pair is calculated. The cost includes two components: transmission cost (lines 8-12), and aggregation opportunity loss (lines 13-18). The transmission cost is determined by the amount of data to be transmitted and the transmission distance. $l(u, v)$ denotes the distance of the shortest path between u and v. In this algorithm, data transmission is a random process (lines 25-35). Given the nodes u and v with weights of w_u and w_v, respectively, the probability that u transmits its data to v is $\frac{w^2(u_p)}{w^2(u_p)+w^2(v_p)}$ (line 26). It follows the rationale that a node with a larger amount of data should have a lower probability of being the transmitter. The study in [10] demonstrates that this random process can achieve good performance. During the calculation of the transmission cost, the amount of data to be transmitted is calculated using mathematical expectation (line 11). Aggregation opportunity loss is another type of cost. Early aggregation based on the EST is recommended. In line 21, the minimum cost perfect matching of nodes is calculated based on the total cost of all the node pairs. The detailed approach is described in [13]. The perfect matching achieves minimal overall cost of the selected node pairs. In line 22, the edges involved in the perfect matching are added to E^*. It is noted that some intermediate nodes are included in E^* in this step. For each node pair, we select one node as the transmitter and the other as the receiver (lines 25-33). Next, we compute the aggregated event and its corresponding amount of data (lines 34-35). Finally, the transmitters are removed for further processing (line 37), and the subsequent iteration begins if any node has not been processed (line 6).

In each iteration of Algorithm 1, the size of S is reduced by half; therefore, the algorithm terminates after $\log(n+1)$ iterations. Each iteration includes three basic operations: computing the total cost of the node pairs (lines 7-20), calculating the perfect matching of nodes (line 21), and performing aggregation (line 35). The complexity of them are $O(n)$, $O(n^2)$, and $O(n^2)$, respectively. Therefore, the complexity of the entire algorithm is $O(\log(n+1)\cdot n^2)$. This algorithm supports generic events that may have arbitrary complex relations. When restricted to the events having non-decreasing aggregation functions, the algorithm has an approximation ratio of $\frac{5}{4}\log(n)$ to the optimal energy-efficient EDT, following a proof similar with [10].

IV. Performance Evaluation

We perform simulations to compare EEDT with the centralized approach (composite event detection is performed at a central server after all primitive events are collected), MFST [10], and TED [5] in different situations. In the simulation, 100 sensor nodes are uniformly distributed in a 100 m $\times$ 100 m area. Each sensor node has a communication range of *rc* = 10 m. The EST is a complete binary tree with a depth of *L*=*5*. The amount of data for the primitive events is randomly chosen to be within the range of $[400, 500]$ bits. Eight intermediate nodes are randomly chosen as checkpoints. Given the amount of data for an event ap and the maximum amount of data for any of its sub-events as, the *data reduction rate* is defined as ap/as. The data reduction rate of checkpoints in the simulation follows a normal distribution with the center of *drr*. The average distance from a source node to the sink node is denoted by ds, and the average distance between two source nodes is denoted by dp. Simulations of 1000 runs are repeated to obtain each data point with the confidence level of 0.95.

We first vary *ds* to compare the energy consumption for different approaches; the results are shown in Fig. 1, where $dp = 20$ and $drr = 0.5$. In the centralized approach, the energy consumption increases linearly when ds increases owing to the linearly increasing transmission distance. TED performs worse than the centralized approach when $ds \leq 38$ because the transmission cost dominates the energy consumption in TED; however, the centralized approach performs worse than TED when $ds > 38$ because the benefit of aggregation in TED is sufficiently large. MFST and EEDT have fine-grained control of the routing tree; therefore, they perform better than the centralized approach and TED. EEDT outperforms MFST because in EEDT, the event relations are utilized to optimize the routing tree. As *ds* increases, the energy saved in EEDT increases compared with MFST.

The average distance between two source nodes (*dp*) is another important factor that affects energy consumption. We vary *dp* to compare the energy consumption for different approaches; the results are shown in Fig. 2, where $ds = 30$ and $drr = 0.1$. We observe that the energy consumption of TED is less than that of the centralized approach when $dr \leq 12$ because in TED, the events aggregate at the fusion nodes, thus reducing the amount of data to be transmitted. However, when $dr > 12$, the result is reversed because the transmission cost of reaching the fusion nodes increases in TED. MFST and EEDT always perform better than the centralized approach and TED. The advantage of EEDT is more obvious when dp is small and is unapparent when dp increases owing to the additional transmission cost. When $dp = 5$, EEDT achieves an energy savings of 20% energy when compared with MFST.

The last factor that we consider is event relation. First, we vary *drr* to analyze the performance of different approaches. The results are shown in Fig. 3, where $ds = 40$ and $dp = 20$. The centralized approach and MFST are not as sensitive to the change in *drr* as TED and EEDT. The centralized approach collects all primitive events to the sink node, without considering the reduction in the amount of data due to aggregation. MFST assumes a non-decreasing aggregation of any two events and does not explicitly utilize data reduction rate. TED utilizes this information for aggregation; therefore, TED shows better performance than the centralized approach when $drr \leq 0.62$. When *drr* is greater than 0.62, the benefit from aggregation is relatively limited when compared with the overhead of aggregation. TED is inferior to MFST because the transmission from the source nodes to the fusion nodes is not optimized. Owing to complete utilization of event relations, EEDT always consumes the least energy among these approaches. EEDT achieves an energy savings of 7%-15% when compared with MFST. In real applications, a simple predicate (yes or no) can be used to summarize a number of underlying events. In such a situation, *drr* can be considerably close to 0 (e.g., 0.001), and EEDT can save more energy.

Further, we vary the data reduction rates of the nodes of EST to investigate the performance. Half of the nodes at level 4 have a data reduction rate following a normal distribution $N(drr, 1)$. Half of the nodes at level 2 or 3 have a data reduction rate of 0.1. All other nodes have a data reduction rate of 1. The results are shown in Fig. 4. It is similar with previous figures, except that TED outperforms MFST when $drr \leq 0.3$. This behavior can be attributed to the utilization of event relations in TED. The energy saved in EEDT is between 8% ($drr = 0.8$, when compared with MFST) and 16% ($drr = 0.1$, when compared with TED).

V. Conclusion

In this letter, we built an event detection tree to perform data aggregation according to event relations. We achieved a trade-off between reducing the amount of data to be transmitted and reducing the distance in the data transmission in order to minimize the overall energy consumption. Simulation results showed that our approach achieves up to 20% energy savings when compared with other existing approaches.

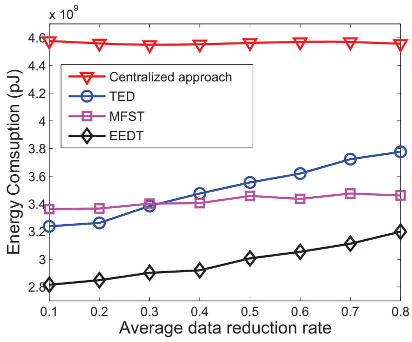

Fig. 4. Comparison of energy consumption for different approaches, by varying the data reduction rate (different data reduction rates for the nodes of EST)

Acknowledgment

This research is supported in part by National Natural Science Foundation of China No. 61502351, Luojia Young Scholar Funds of Wuhan University No. 1503/600400001, and Chutian Scholars Program of Hubei, China.

References

[1] K. Romer and F. Mattern, “Event-based systems for detecting real-world states with sensor networks: a critical analysis,” in *Proc. of ISSNIP*, 2004, pp. 389–395.

[2] Y. Li, C. Ai, C. T. Vu, Y. Pan, and R. Beyah, “Delay-bounded and energy-efficient composite event monitoring in heterogeneous wireless sensor networks,” *IEEE Trans. on Parallel and Distributed Systems*, vol. 21, no. 9, pp. 1373–1385, 2010.

[3] K. Ali, T. Anwaro, I. H. Naqvi, and M. H. Jafry, “Composite event detection and identification for WSNs using general hebbian algorithm,” in *Proc. of ICC*, June 2015, pp. 6463–6468.

[4] S. Zhang, S. Gong, Z. Cui, Q. Liu, and J. Fan, “An aggregation tree approach for event detection in wireless sensor networks,” *Journal of Software*, vol. 4, no. 8, pp. 899–906, 2009.

[5] Y. Lai, J. Cao, and X. Fang, “TED: Efficient type-based composite event detection for wireless sensor network,” in *Proc. of DCOSS*, 2011, pp. 1–8.

[6] J. Chen, L. Ramaswamy, D. K. Lowenthal, and S. Kalyanaraman, “Comet: Decentralized complex event detection in mobile delay tolerant networks,” in *Proc. of MDM*, 2012, pp. 131–136.

[7] C. A. Suescun and M. Cardei, “Event-based clustering for composite event detection in wireless sensors networks,” in *Proc. of IPCCC*, Dec 2016, pp. 1–8.

[8] Y. Zhu, R. Vedantham, S.-J. Park, and R. Sivakumar, “A scalable correlation aware aggregation strategy for wireless sensor networks,” *Information Fusion*, vol. 3, no. 9, pp. 354–369, 2008.

[9] A. Goel and D. Estrin, “Simultaneous optimization for concave costs: Single sink aggregation or single source buy-at-bulk,” in *Proc. of Algorithmica*, 2005, pp. 5–15.

[10] H. Luo, Y. Liu, and S. K. Das, “Routing correlated data with fusion cost in wireless sensor networks,” *IEEE Tran. on Mobile Computing*, vol. 5, no. 11, pp. 1620–1632, 2006.

[11] K. Choi and K. Chae, “Data aggregation using temporal and spatial correlations in advanced metering infrastructure,” in *Proc. of the International Conference on Information Networking*, 2014, pp. 541–544.

[12] Y. Lu, I.-S. Comsa, P. Kuonen, and B. Hirsbrunner, “Adaptive data aggregation with probabilistic routing in wireless sensor networks,” *Wireless Networks*, vol. 22, no. 8, pp. 2485–2499, 2016.

[13] A. Meyerson, K. Munagala, and S. Plotkin, “Cost-distance: Two metric network design,” in *Proc. of FOCS*, 2000, pp. 624–630.