ENMSJ : An Efficiency Filtering Technique using Bitmap Vectors for n-way Joins in Wireless Sensor Networks

Boubekeur DJAIL
Ecole nationale Supérieure d'Informatique (ESI)
Algiers, Algeria
Email: b_djail [AT] esi.dz

Abstract— **In wireless sensor networks, join queries execution introduces a high energy consumption. While energy is an important factor for sensors survival, several techniques were developed to reduce it. Sensors energy is affected by the number of transferred messages whereas query is performed. The aim of the proposed techniques was then to decrease the communicated data volume. So, the exchanged data volume is soaring when joins are performed between many data tables. This joins type is called: n-way join query. In this paper, we present an efficiency technique to treat n-way join queries in wireless sensor networks. This technique is named: Enhanced N-way Mediated Semi-Join (ENMSJ). ENMSJ is an improvement of a precedent strategie that we proposed: N-way Mediated Semi-Join (NMSJ). ENMSJ uses bitmap tables to more reduce transferred messages quantity. We compared the two techniques to test their performance. Obtained results are very hopeful.**

***Index Terms*— communication cost, in-network join, n-way join, wireless sensor networks.**

I. Introduction

A wireless sensors network is composed by a high number of wireless sensors which are dispersed in large zones. These sensors are limited in memories capacities and in processing abilities. Data exchange between these sensors is done by wireless support.

A sensor gets a data and records it in a local data table. Data tables of all sensors with the same type consist a distributed database table [1]. To ask this table, relational queries can be used, such: selections, projections, joins, ...

In wireless sensors networks, joins queries are executed to access data logged in many distributed database tables. Such queries introduce a high-energy consummation, which can paralysis sensors in first and all the network thereafter.

The main aim of the researchers in this field is to decrease the consumed energy quantity. Many of algorithms are so proposed. Most of them are dedicated to binary joins. N-way joins are rarely invoked.

For n-way joins queries, it is necessary to determinate the execution order of the intermediate joins whom it consists. We note that the number of the execution orders grow exponentially with the number of the used tables.

We present in this paper an efficiency energy technique based on semi- join principle to treat n-way join queries in wireless sensor networks. The proposed technique adopts the same principles than NMSJ (N-way Mediated Semi-join) [2]. Additionally, ENMSJ uses bitmap tables to more reduce data transferred size.

In follows, we introduce, at first, section 2 which provides an overview of joins queries in wireless sensor networks. Related work is described in Section 3. Section 4 details the proposed technique. Section 5 presents realized tests and establishes a discussion about the obtained results. Finally, section 6 concludes the paper.

II. Joins queries in wireless sensor networks: an overview

A. Definitions

A join is a relational query, which is performed at least between two tables. The result is a new table that contains tuples determined by the concatenation of the tuples of each table, based on a definite condition: the join predicate.

A join query is called equi-join if it has only equality operators in its join predicate.

A binary join considers only two tables. An n-way join is performed between three or more tables.

B. Implementation of join queries in wireless sensor networks.

In wireless sensors network, a join query can be performed by two different ways. Hence, we define two diverse implementations [3, 4]:

Extern join: The tuples of jointed tables are first transferred to the sink from the nodes. After that, the join query is performed. This implementation requires an excessive energy consumption. Its energy cost is so soaring.

In-network join: This execution is adopted by most of recent proposed techniques. It permits reducing the number of transmitted messages during join accomplishment. In-network join implementation engenders the best energy cost.

C. Join types in wireless sensor network

Considering the spatial aspect during joins queries execution in wireless sensors networks, we can define two types of joins[3, 4]:

Unique region joins: The join query is performed between data tables in a single region of a set of sensors.

Inter-region joins: Two or more data tables tack part to the join query. Each data table is recorded in a distinct region of a wireless sensors network.

Considering the temporal aspect of a join query execution in wireless sensors network, we distinguish three classes of joins queries [3]:

One shot joins: That if join queries are performed between static tables. Each table is defined by a fixed window corresponding to a limited range of time or tuples.

Continuous joins: There are the joins that are executed continuously by using sliding windows to delimit set of tuples that are concerned by join operation at each step.

Periodic joins: These joins are performed repetitively at interval of times. They can be considered as a particular case of continuous joins.

III. RELATED WORKS.

The techniques proposed to treat join queries in wireless sensors networks can be classified in two great classes: those without filtering (non-joinable tuples) and those with filtering. These techniques are presently the most adopted because of the best costs which they permit.

The non-filtering techniques are the first proposed techniques. Yao and Gehrke [1]. discussed the difference of the transmission cost between an extern join and an in-network join. They concluded that in in-network execution the energy consumption is low for low values of low join selectivity. Bonfils and Bonnet [5] studied the optimal node where a join query must be executed considering an in- network execution. They determined that the optimal node is on the shortest path between the two nodes of the join regions, and this is the node which has more data to transmit. Coman and al. [6] presented local join and mediated join techniques to treat an inter-region join. Local join executes the query at one of the two regions of a binary join, whereas Mediated join performs the join at a region situated between the two zones. Coman and al. noted that no precise technique shows the best results for all queries.

Relating to the techniques with filtering, Coman and al. [6] used semi-join principle and proposed Local Semi-Join technique as a variant of Local Join technique. Yu and al. [7] proposed Synopsis Join technique to perform an inter- region equi-join query between statics tables. The authors adopted a distributed alternative of the semi-join technique. Min and al [8] studied various plans to perform a join query, and they presented a cost model to determine the optimal plan under various conditions.

Other techniques were proposed to address specific joins queries. Mo and al. [9] studied spatial queries in wireless sensor network. Kang and al [10] proposed techniques for iceberg join queries, where only tuples whose cardinality surpasses a given threshold are admitted to the join operation.

The techniques that are described above addressed mainly binary joins. Few works addressed n-way joins in wireless sensors networks.

Stern and al. [11] presented SENS-join, a technique that was considered n- way join queries. SENS-join executes join queries at the sink after determining filters by treating the join attributes values transmitted by the nodes.

This technique causes high energy consumption due to the high number of transmitted messages from nodes to the sink.

NLJ (N-way Local Join) [12] is the technique that we proposed in aim to more reduce energy consumption by adopting an in-network execution. NLJ has the inconvenient to be a non-filtering technique.

NLSJ (N-way Local SemiJoin) [13] with its variant NMSJ (N-way Mediated SemiJoin) [2] offer a filtering solution to improve NLJ performances [14].

NLSJ runs in three steps:

- The query is diffused by the sink to the root nodes of the regions.
- An intermediate join is performed for each nodes pair (S_i, S_{i+1}). S_{i+1} is the site chosen as the nearest site to S_i. This execution is done as follows:

The S_{i+1} site transmits the join attribute to S_i.

At Si, a semi-join is performed between the join attribute and the local table.

The determined result is communicated to S_{i+1}.

At S_{i+1}, the final result of the intermediate join is determinate.

- The product of the last intermediate join, which corresponds to the final result of the join query, is transmitted to the base station.

NMSJ a variant technique of NLSJ improve NLSJ by chosen for each intermediate join between two sites: S_i and S_{i+1}, the best site which to be first site to transmit join attribute to another. This site is that contain less tuples to transmit.

IV. Enhanced N-way Mediated Semi-Join (ENMSJ).

A. ENMSJ description

In this paper, we describe Efficiency N-way Mediated Semi-Join (ENMSJ), a technique for n-way join queries in wireless sensor networks. We consider performing a one-shot inter-region join with syntax as:

```
SELECT <List of attributes from R1, R2, ...
and Rn>
FROM  R1, R2, …, Rn
WHERE  pred(R1)  AND pred(R2)  … AND pred(Rn)
AND join-express (R1.join-attrib , R2.join-
attrib ,…, Rn.join-
attrib)
```

With:
R1, R2, …, Rn: Static tables which participate to the query.
Pred (Ri): is the selection predicate of a table Ri.
Join-express: is the join condition expression.

In wireless sensors networks, join queries are used to accumulate data from many tables over a lot of controlled regions. An example for applications that manipulate such queries: the traffic control of vehicles, monitoring of birds' immigration, etc.

For illustration, to track immigrant birds over three controlled zones, we can write the following query:

```
SELECT    B1.BId, B1.time, B2.time, B3.time
FROM      B1, B2, B3
WHERE      (B1.time IN reg1) and (B2.time IN
reg2)
and (B3.time IN reg3) and (B1.BId = B2. BId)
and (B2. BId= B3. BId)
```

Where:
B1, B2, B3: represent the tables in different regions.
Id, time: are the attributes of each table.
reg1, reg2, reg3: define the respective times intervals during which birds go respectively over the three regions.

Enhanced N-way Mediated Semi-Join (ENMSJ) is an improvement of NMSJ [2] that we already presented for the same join query kind. ENMSJ reduces significantly the number of exchanged messages during each intermediate join operation. To transfer the join attribute, the source region of an intermediate join communicates a vector bit, that represent the values at this attribute. Like NMSJ, ENMSJ uses also the left linear trees technique [15] to decide the choice of the execution order.

ENMSJ executes an n-way join query in three phases:

Phase 1. Query dissemination.

The query is transmitted from the sink to the root nodes of all concerned regions. A GPSR protocol [16] is used to guarantee a correct delivery of the query.

Moreover, the nodes of each region refer their tuples to the root node, by using GPS or localization algorithms [17] to fix their positions and their neighbors positions.

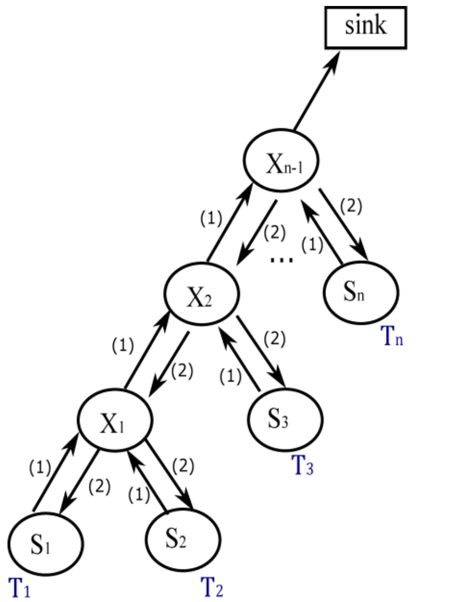
Phase 2. Query execution

The join query is realized by a succession intermediate join. Each one is performed by a pair of nodes (S_i, S_{i+1}). The node S_{i+1} is selected as the nearest node to the precedent site Si.

Five steps are completed by each intermediate join (Figure1):

i. The nodes (S_i, S_{i+1}) choose the site X_i where the intermediate-join must be performed. X_i is selected from the two sites (S_i, S_{i+1}) as the site which have lower number of tuples number to participate in the join operation. This to reduce the tuples number to transmit.
ii. The bitmap of the table at the node X_i is transmitted to the second site (the non-chosen site).
iii. A semi-join is performed at the site which receive the bitmap.
iv. The result of semi-join operation is communicated to X_i.
v. The intermediate join is confirmed at this site.

Phase 3. The result at the last site is transmitted to the sink.



The source site transmits the bit vector to the destination site
The source sit receipts the result of the semi-join

Fig. 1. ENMSJ principle

B. Illustrative example

To better explain the ENMSJ principle, the following example is done, where a join query is performed between three tables.

The first join operation is executed by the nodes pair (S_1, S_2). The site S_1 having a low number of tuples to transmit is selected as X_1 to perform the intermediate join. X_1 (S_1) determine and communicate the bitmap to S_2, which realizes the semi-join with its tuples. The result is then transferred to X_1 where the final result of T_1 join T_2 is calculated (Figure 2 (a)). At the second step, the intermediate join between the nodes pair (S_2, S_3) is achieved in the same manner as previously (Figure 2 (b)). At the end, the final result is sent to the sink (Figure 2 (c)).

C. Cost calculation

The cost of ENMSJ technique is determined by the number of transmitted messages during a query execution.

For n tables, the cost is done by the following formula:

$$nT + (n-1)\frac{nb\_j\_att\_val}{8} + \frac{1}{4}f_1T^2 + \frac{1}{4}f_1f_2T^3 + \cdots + \frac{1}{4}f_1f_2 \cdots f_{n-1}T^n + f_1f_2 \cdots f_{n-1}T^n \quad (1)$$

If we assume that all selectivity joins are equals, we have the following formula:

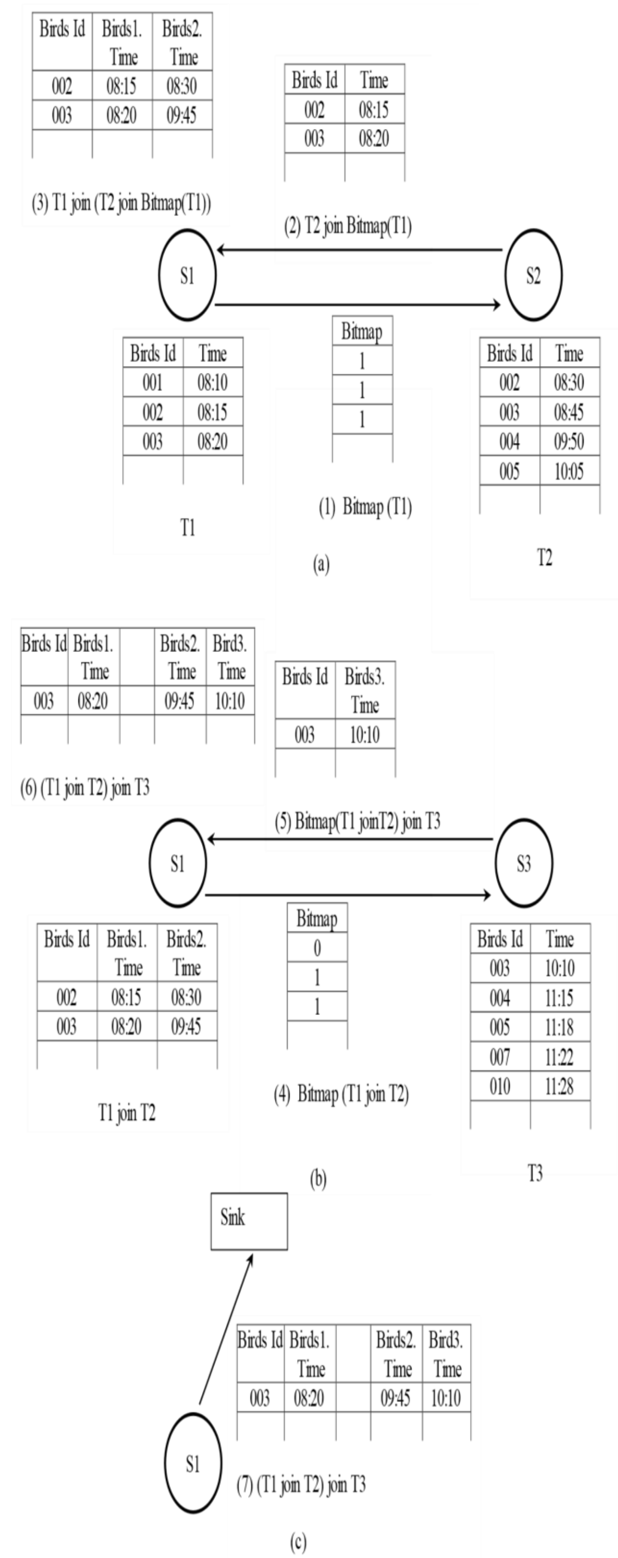


Fig. 2. Example of ENMSJ execution

$$nT + (n-1)\frac{nb\_j\_att\_val}{8} + \frac{1}{4}fT^2 + \frac{1}{4}f^2T^3 + \cdots + \frac{1}{4}f^{n-1}T^n + f^{n-1}T^n \quad (2)$$

Where: : $nT + (n-1)\frac{nb\_j\_att\_val}{8}$ is the first cost transmission before the first intermediate join.

$\frac{1}{4}fT^2$ is the cost transmission of the result of the first intermediate-join.

$\frac{1}{4}f^2T^3$ is the cost transmission of the result of the second intermediate-join.

$\frac{1}{4}f^iT^{i+1}$ is the cost transmission of the result of the ith intermediate-join.

V. Performance analysis

A. Experimentation environment

We realize the experimentation tests of ENMSJ technique on the NS3 simulator. We assume, for this, the following:

- Each region is structured in arborescence, with one node designated as root.
- The table size is 2000 tuples.
- Join attribute range is: 1 – 10000.
- The tuple size is 40 bytes.
- The message size is the same than the tuple size.
- The column size is 10 bytes.
- The result tuple size is 30 bytes.

In the simulation, two cases are considered: a simulation with three tables and another with five tables. In each case, we calculate the communication cost of the join execution by calculating the number of transmitted messages according to selectivity factors values of intermediates joins.

Two value ranges of selectivity factors are envisaged for each case: the [10^{-5},10^{-4}] range and [10^{-4},10^{-3}] range. The first interval denotes the low values, the second the high values.

B. Experimentation results

In all cases of the performed simulation, the ENMSJ technique performs better than NMSJ (Figures 3,4,5,6).

With three tables, for low values of selectivity factor, the difference between the generated cost by the two techniques is about 937 messages.

For high values of selectivity factor, for three tables, ENMSJ send the same number of messages lower than NMSJ.

For five tables, for low values and high values of selectivity factor, 1875 messages are transmitted by NMSJ higher than ENMSJ.

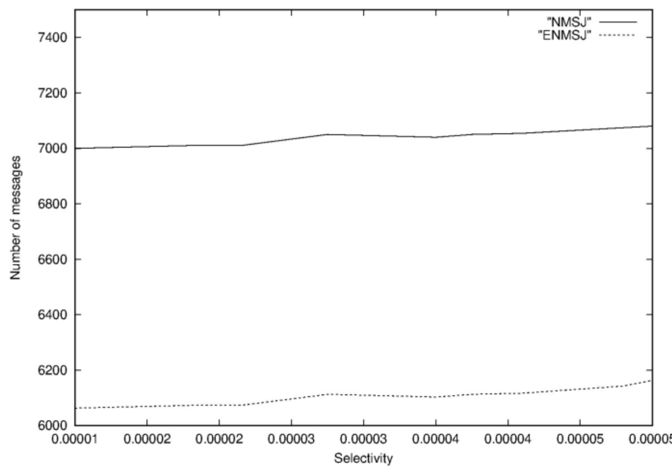


Fig. 3. Communication cost for 3 tables in the interval [10^{-5} , 10^{-4}]

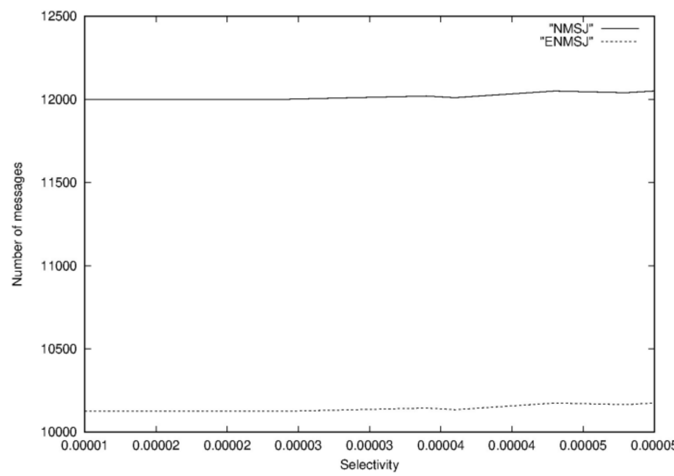


Fig. 4. Communication cost for 5 tables in the interval [10^{-5} , 10^{-4}]

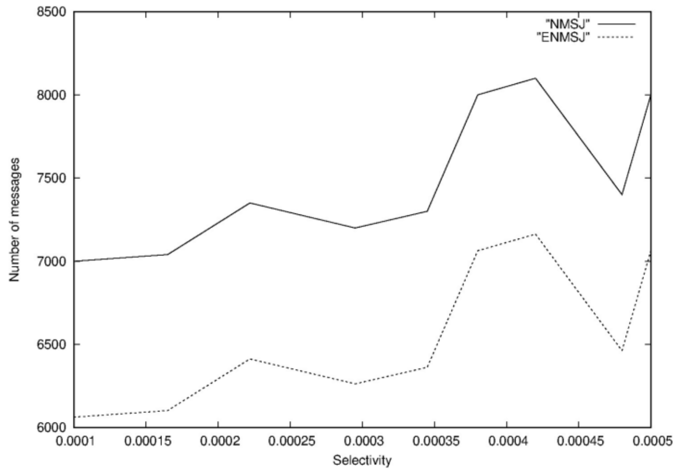


Fig. 5. Communication cost for 3 tables in the interval [10^{-4} , 10^{-3}]

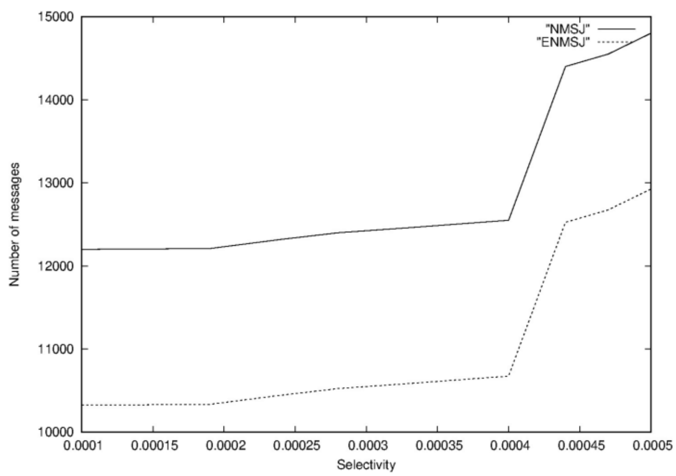


Fig. 6. Communication cost for 5 tables in the interval [10^{-4} , 10^{-3}]

C. Discussion

Like NMSJ, ENMSJ adopts two advantageous principles to decrease considerably the number of transmitted tuples during a query effecting. It practices the in-network principle to reduce messages volume transmitted between internal nodes and to the sink. ENMSJ also uses the semi-join technique to remove the non-joinable tuples and so to more minimize quantity of transmitted messages.

Additionally, ENMSJ replaces the transmission of the join attribute by the emission of a bitmap which represents the values of the join attribute. This permit a high gain in the cost of the join execution. ENMSJ performs better than NMSJ, for every tested value of selectivity and for every number of used tables.

VI. CONCLUSION

In this paper, we presented a new technique called ENMSJ, to perform n- way join queries in wireless sensors networks. ENMSJ is a filtering technique, which is considered as an extension of NMSJ, a best technique in this way.

ENMSJ adopts the same principles than NMSJ, and additionally uses the bitmap vector to more reduce the number of transmitted messages during a join query execution.

Therefore, ENMSJ shows better performances than performed NMSJ in the all simulations.

In future work, we propose to do more improvements to further reducing the number of transmitted messages while a join query is executed in wireless sensors networks.

REFERENCES

[1] Y. Yao and J. Gehrke, "Query Processing in Sensor Networks," in CIDR, 2003, pp. 233-244.
[2] B. Djail, W.-K. Hidouci, and M. Loudini, "NMSJ: A FILTERING TECHNIQUE FOR N-WAY JOINS IN WIRELESS SENSOR NETWORKS," UNIVERSITY POLITEHNICA OF BUCHAREST SCIENTIFIC BULLETIN SERIES C-ELECTRICAL ENGINEERING AND COMPUTER SCIENCE, vol. 80, pp. 23-34, 2018.
[3] H. Kang, "In-network processing of joins in wireless sensor networks," Sensors, vol. 13, pp. 3358-3393, 2013.
[4] B. Djail and K. W. Hidouci, "Les jointures dans les réseaux de capteurs sans fil " in Conférence Nationale sur les Technologies de l'Information et les Télécommunications CNTIT'13, 10-11 Décembre 2013, 2013.
[5] B. J. Bonfils and P. Bonnet, "Adaptive and decentralized operator placement for in-network query processing," Telecommunication Systems, vol. 26, pp. 389-409, 2004.
[6] A. Coman, M. A. Nascimento, and J. Sander, "On join location in sensor networks," in 2007 International Conference on Mobile Data Management, 2007, pp. 190-197.
[7] H. Yu, E.-P. Lim, and J. Zhang, "On in-network synopsis join processing for sensor networks," in 7th International Conference on Mobile Data Management (MDM'06), 2006, pp. 32-32.
[8] J.-K. Min, H. Yang, and C.-W. Chung, "Cost based in-network join strategy in tree routing sensor networks," Information Sciences, vol. 181, pp. 3443-3458, 2011.
[9] S. Mo, Y. Fan, Y. Li, and X. Wang, "Multi-attribute join query processing in sensor networks," Journal of Networks, vol. 9, pp. 2702-2712, 2014.
[10] H. Kang, "In-Network Processing of an Iceberg Join Query in Wireless Sensor Networks Based on 2-Way Fragment Semijoins," Sensors, vol. 15, pp. 6105-6132, 2015.
[11] M. Stern, E. Buchmann, and K. Böhm, "Towards efficient processing of general-purpose joins in sensor networks," in 2009 IEEE 25th International Conference on Data Engineering, 2009, pp. 126-137.
[12] B. Djail, K. W. Hidouci, and M. Loudini, "A technique for n-way joins in wireless sensor networks," Database Systems Journal, vol. 7, pp. 3-9, 2016.
[13] B. Djail, K. W. Hidouci, and M. Loudini, "N-way Local SemiJoin : A Filtering Technique for N-Way Joins in Wireless Sensors Networks," Journal of Electronic Systems, vol. 6, pp. 7-16, 2016.
[14] B. Djail, W. K. Hidouci, and M. Loudini, "A comparative evaluation of techniques for N-way joins in wireless sensors networks," Pollack Periodica, vol. 15, pp. 13-24, 2020.
[15] M. Steinbrunn, G. Moerkotte, and A. Kemper, Optimizing join orders: Citeseer, 1993.

[16] B. Karp and H.-T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in Proceedings of the 6th annual international conference on Mobile computing and networking, 2000, pp. 243-254.
[17] A. Savvides, M. Srivastava, L. Girod, and D. Estrin, "Localization in sensor networks," in Wireless sensor networks, ed: Springer, 2004, pp. 327-349.