MONASH University

ASSIGNMENT COVER SHEET

Student's name	(Family name) Diong	(Given names) Chen Xi	
ID number	32722656	**E-mail**	cdio0004@student.monash.edu
Unit code & name	FIT3143 Parallel Computing	**Unit code**	FIT3143

Title of assignment	**Assignment – 2 *Electric Vehicle Charging Grid: Operated by a Distributed Wireless Sensor Network (WSN)***
Lecturer/tutor	Dr. Shu Min Leong (Jessie)

Is this an authorised group assignment? ☐ **Yes** ☒ **No**

If this submission is a group assignment, each student must attach their own signed cover sheet to the assignment.

Has any part of this assignment been previously submitted as part of another unit/course? ☐ **Yes** ☒ **No**

Tutorial/laboratory day & time	Tuesday 8:00 a.m. – 10:00 a.m.
Due date Monday, 16 October 2023	**Date submitted** Wednesday, 11 October 2023

All work must be submitted by the due date. If an extension of time to submit work is required, a Special Consideration Application (In-semester Assessment Task) must be submitted.

Has an extension been approved? **Yes** ☐ **No** X **If yes, please give the new submission date ……./……/…….**

Please note that it is your responsibility to retain copies of your assessments.

Student Statement:

- I have read the University's Student Academic Integrity Policy and the University's Student Academic Integrity: Managing Plagiarism and Collusion Procedures.
- I understand the consequences of engaging in plagiarism and collusion as described in Assessment and Academic Integrity Policy
- I have taken proper care of safeguarding this work and made all reasonable effort to ensure it could not be copied.
- I acknowledge that the assessor of this assignment may for the purposes of assessment, reproduce the assignment and:
  i. provide to another member of faculty; and/or
  ii. submit it to a plagiarism checking service; and/or
  iii. submit it to a plagiarism checking service which may then retain a copy of the assignment on its database for the purpose of future plagiarism checking.
- I certify that I have not plagiarised the work of others or participated in unauthorised collaboration when preparing this assignment.

Signature ....Chen Xi Diong.. *Date…………11/10/2023………*

Privacy Statement

The information on this form is collected for the primary purpose of assessing your assignment. Other purposes of collection include recording your plagiarism and collusion declaration, attending to course and administrative matters and statistical analyses. If you choose not to complete all the questions on this form it may not be possible for Monash University to assess your assignment. You have a right to access personal information that Monash University holds about you, subject to any exceptions in relevant legislation. If you wish to seek access to your personal information or inquire about the handling of your personal information, please contact the University Privacy Officer: privacyofficer@adm.monash.edu.au

MONASH University

FIT3143 Semester 2, 2023
Assignment 2 – Report
Title – Electric Vehicle Charging Grid: Operated by a Distributed Wireless Sensor Network (WSN)

Include the word count here (for Sections A to C): 2013 words.

A. Methodology

A grid-based nearest architecture is used to simulate the Electric Vehicle Charging Grid operated by a Distributed WSN. This architecture ensures that the failure in one node does not affect the neighboring nodes, therefore mitigating catastrophic program failures (Gamage & Baskaran, 2020). According to Mazzer and Tourancheau (2008), the use of a Message Passing Interface (MPI) proves useful for implementing distributed communication and computation. In this assignment, the MPI library in C is used to achieve the simulation of exchanging information between Charging Nodes and the Base Station, and the POSIX threads library is used to achieve parallelism in simulating the charging ports and exchanging messages.

i) An Overview (Refer to Figure 2)

The link to the Flow Chart[1] illustrating the execution of the program is provided for reference.

Upon executing the program, the user will be prompted to input a few key metrics to be used within the simulation:

m – The number of rows of the charging grid

n – The number of columns of the charging grid

k – The number of charging ports per node

cycles – The number of iterations before shutting down for maintenance

free_thres – The number of available charging ports for a Charging Node to be free

full_thres - The number of available charging ports for a Charging Node to be full

After that, the program will initiate the setup procedures of the WSN, creating a Base Station, and *m* x *n* Charging Nodes, with each being an MPI process. Two MPI communicators are set up, one being the world communicator *MPI_COMM_WORLD*, and another being the 2-Dimensional Virtual Cartesian Grid Topology communicator *comm2D*. (Refer to Figure 1)

[1] Due to resolution constraints, uploading the whole diagram will cause a loss of information. The link to the online document (LucidChart) is provided instead. https://lucid.app/lucidchart/691112c5-8da9-4e09-99fb-37f416ad30a3/edit?viewport_loc=-1641%2C93%2C4863%2C2384%2CgJ7B.IwLe2pk&invitationId=inv_109ca7a4-3702-4f8c-b022-006595aaa061

ii) Simulating the Base Station (Refer to Figure 3)

The Base Station is responsible for responding to alerts from the Charging Nodes, as well as broadcasting the termination message at the end to shut them down.

The main process of the Base Station keeps track of the number of cycles/iterations, where each cycle is designed to simulate 1 hour of real-time. When the number of cycles has reached the user-specified threshold, the Base Station will notify all of the Charging Nodes to shut down for maintenance. It will then wait for confirmation messages from all of the Charging Nodes to indicate that there are no more pending communications within the nodes, and proceed to tell the Charging Nodes that it is safe to shut down. The proper shutdown procedure can be observed from the console output (Refer to Figure 4).

A POSIX thread is created for the Base Station, which is solely responsible for receiving messages from the Charging Nodes and responding to them. The messenger (Refer to Figure 5) is used for:

1. Receiving and responding to an alert from a Charging Node indicating that its quadrant is full.
2. Receiving a notification from a Charging Node that had previously sent an alert that it is now free.
3. Receiving confirmation messages from Charging Nodes indicating that they have no more pending communication.
4. Receiving messages from Charging Nodes stating that they have successfully shut down.

iii) Simulating the Charging Nodes (Refer to Figure 6)

Each Charging Node will spawn *k* POSIX threads, each simulating a Charging Port (Refer to Figure 7). Each Charging Port will periodically update its availability[2] in the shared array.

The main process of the Charging Node will periodically check for the availability of its ports (each interval simulates 10 minutes in real-time). If the availability of its ports is less than the specified threshold, the node will prompt its neighbors[3] for their availability. In the case where all of its neighbors are also full, the node will alert the base station and wait for its response with the nearest available[4] Charging Nodes.

Similar to the Base Station, each Charging Node will have its messenger thread (POSIX) created. The aim is to separate the concerns for the Charging Node and decrease the main node's workload. The messenger (Refer to Figure 8) is used for:

1. Receiving and responding to prompts from the Charging Node's neighbors asking for the node's availability.
2. Receiving and responding to prompts from the Base Station asking for the node's available neighbors.
3. Receiving the reply from the Base Station indicating the nearest non-neighbour Charging Nodes that are available if the node's quadrant is full.
4. Receiving the termination message from the Base Station.

All the nodes will continue simulating until it receives a termination message from the Base Station. The node will clean up and send a confirmation message indicating that it will not be sending any more requests. The node will then wait for the final shut-down message from the Base Station whilst responding to its neighbors to ensure that no communication requests are left hanging.

[2] Assuming EVs take anywhere from 1 to 10 hours to charge.
[3] The immediate top, bottom, right, and left adjacent nodes (if they exist).
[4] Here we are only concerned up to 2 levels of adjacency (i.e. neighbors of neighbors).

MONASH University

B. Results Tabulation

i) Summary of Events

A total of 5 simulations are run, with 2 on a local machine, and 3 on the CAAS platform. Due to hardware limitations, the local machine can only support small grid sizes (e.g. 2x2). Therefore, larger grid sizes are only tested on the CAAS platform. However, there is also a threshold[5] for the number of nodes a student is allowed to access, therefore the maximum grid size tested is a 2x5 grid with a total of 11 processes[6] running across 4 nodes.

The specifications (grid dimensions, number of ports, number of iterations, thresholds, and CAAS platform specifications) and results (total messages exchanged and total communication time) of the simulations are recorded in Table 1, Table 2, Table 3, Table 4, and Table 5. For each simulation, the number of events occurring at each iteration (hour) is plotted as a line chart (Refer to Figure 9, Figure 10, Figure 11, and Figure 12).

ii) Base Station Log File

The Base Station logs three events:

1) A free station alerting that it is full (Refer to Figure 13).
2) A previously reported station indicating that it is free (Refer to Figure 14).
3) An indication that the base station is shutting down for maintenance, along with a summary of events (Refer to Figure 15).

In event 1), the log consists of the time the report is received, the station that sent the report, the station's coordinates, the station's availability, the nearest available stations that are sent by the base station (None if all stations are full), the communication time, and the number of messages exchanged between the Base Station and the reporting station.

In event 2), the log consists of the time the report is received, the station that sent the report, the station's coordinates, the station's availability, and the number of messages exchanged between the Base Station and the reporting station.

In event 3), the log consists of the total number of alerts received, the total number of free reports received, the total number of messages exchanged, and the total communication time established throughout the program runtime.

[5] This value is assumed to be 4.

[6] This is including the Base Station, for a total of m x n + 1 processes.

MONASH University

iii) Charging Station Log File

A Charging Station logs four events:

1) Alerting the Base Station that it is full (Refer to Figure 16).
2) When it is full, but one or more of its neighboring stations are free (Refer to Figure 17).
3) Reporting to the Base Station that it is free (Refer to Figure 18).
4) An indication that the station has shut down (Refer to Figure 19).

In event 1), the log consists of the time the report is sent, the station's availability, the available neighbors (None in this case), and the nearest available stations that are received from the base station (None if all stations are full).

In event 2), the log consists of the time the report is sent, the station's availability, and the available neighbors.

In event 3), the log consists of the time the report is sent and the station's availability.

In event 4), the log consists of the number of requests (alerts) that it has sent to the Base Station throughout its runtime.

C. Analysis & Discussion

i) Significant Observations and Explanations

Two hypotheses are defined in this section:

Hypothesis 1 - Square WSN grids have better communication efficiency compared to rectangular grids.
Hypothesis 2 - The communication time within the CAAS platform is longer than that within a local machine.

From the results of the average communication time in Table 6, we can observe that square (i.e. n x n) WSN grids have better communication efficiency than rectangular grids, which aligns with the first hypothesis. From the calculations of the average time taken to exchange a message[7] for each simulation in Table 6, we can observe this pattern on both a single computer and on the CAAS platform. This may be due to the average distance between each node and the base in a square grid distribution is lesser than the average distance between each node and the base in a rectangular grid, resulting in lesser transmission time and messaging latency.

There is a noticeable difference in the communication time recorded by the Base Station and the Charging Nodes, whereby the communication time from the Charging Node's point of view far exceeds the communication time from the Base Station's point of view. As an example, in the first local simulation (Refer to Table 1), the total communication time logged by the Base Station is 819103 nanoseconds. However, the communication time with the Base Station recorded by a single Charging Node (Refer to Table 6), say Node 0, is 2515122 nanoseconds, which is more than double the total time spent by the Base Station communicating with all the Charging Nodes.

The reason for this may be due to the Base Station serving requests from the other nodes. For example, in the first local simulation (Refer to Table 1), we can see that the number of alerts received by the Base Station is equal to the number of Charging Nodes. Since the Base Station is only able to respond to one node at a time, the other reporting nodes have to wait for their turn. Furthermore, aside from the communication between the Base Station and the reporting node, the other stations are requesting availability from one another as well. As such, there is a drastic increase in the amount of communication happening at that time frame, which leads to a delay in the exchanging of messages.

[7] Messages between the base station and the charging node.

MONASH University

ii) Performance Comparison and Analysis

Comparing the results in Table 6, for the same grid size, we can see that to exchange a message between the Base Station and a Charging Node, a single computer takes 8622 nanoseconds on average; whereas the CAAS platform takes 20446 nanoseconds on average. Additionally, for the same grid size, the total communication time taken for each Charging Node is also considerably longer on the CAAS platform (Refer to Table 7). This may be due to the distance between the computing nodes on different platforms. The communication time between nodes on a single computer is faster than nodes on the CAAS platform, which adheres to the second hypothesis.

For a single computer, the processes are run on cores in the same CPU, which are relatively close to one another. On the CAAS platform, the processes may be run on different computing nodes, which may be physically far apart from one another. Therefore, the messaging distance on the CAAS platform is higher than that of a single computer, explaining the longer communication time taken on the platform.

iii) Possible Limitations

There is a case where occasionally, some processes hang while running on a single computer. It is important to note that some parts of the program are still actively running (i.e. printing to the console and logging to files), which shows that only parts of the program are affected. This situation does not occur when the program is executed on the CAAS platform. A possible reason for this is due to the much more limited number of processors present in a single computer. The computer may be running tasks in the background, causing some of the MPI processes to be put in wait by the system’s scheduler and to be off-sync with the other processes, thus waiting indefinitely in some MPI_Send or MPI_Recv calls. A suggestion is to execute the program while minimizing the tasks running in the background for best performance.

Due to a generic seed being set for the random number generators in each simulation, the simulations do not differ much from one another. It is recommended to use a more dynamic seed value to observe a wider variety of interactions between the processes.

D. References

Gamage, B. M. S. V., & Baskaran, V. M. (2020). Simulation and Analysis of Distributed Wireless Sensor Network using Message Passing Interface. *arXiv preprint arXiv:2007.00754*. Retrieved from [2007.00754] Simulation and Analysis of Distributed Wireless Sensor Network using Message Passing Interface (arxiv.org)

Mazzer, Y., & Tourancheau, B. (2008). MPI in wireless sensor networks. In *Recent Advances in Parallel Virtual Machine and Message Passing Interface: 15th European PVM/MPI Users' Group Meeting, Dublin, Ireland, September 7-10, 2008. Proceedings 15* (pp. 334-339). Springer Berlin Heidelberg. Retrieved from Title (springer.com)

E. Appendix

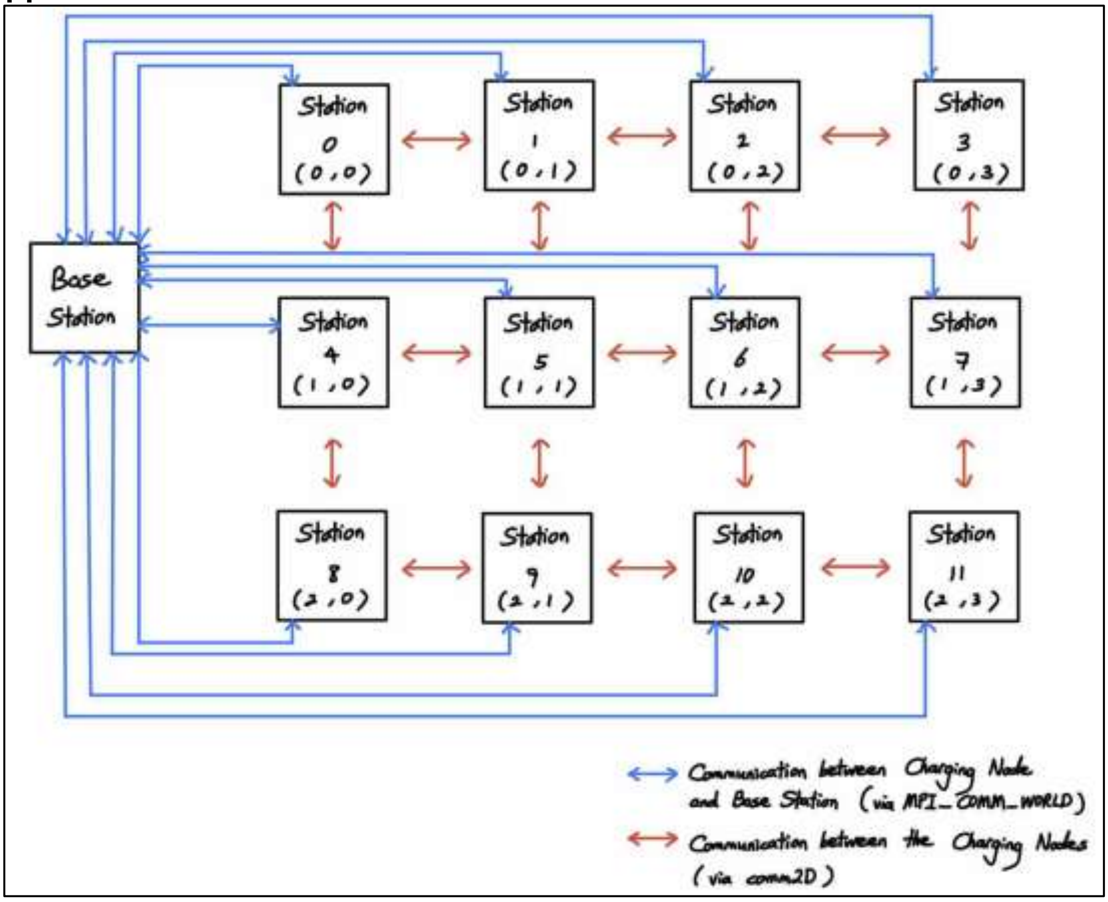

Figure 1. A 3x4 Grid Configuration of the Wireless Sensor Network as the Electric Charging Grid.

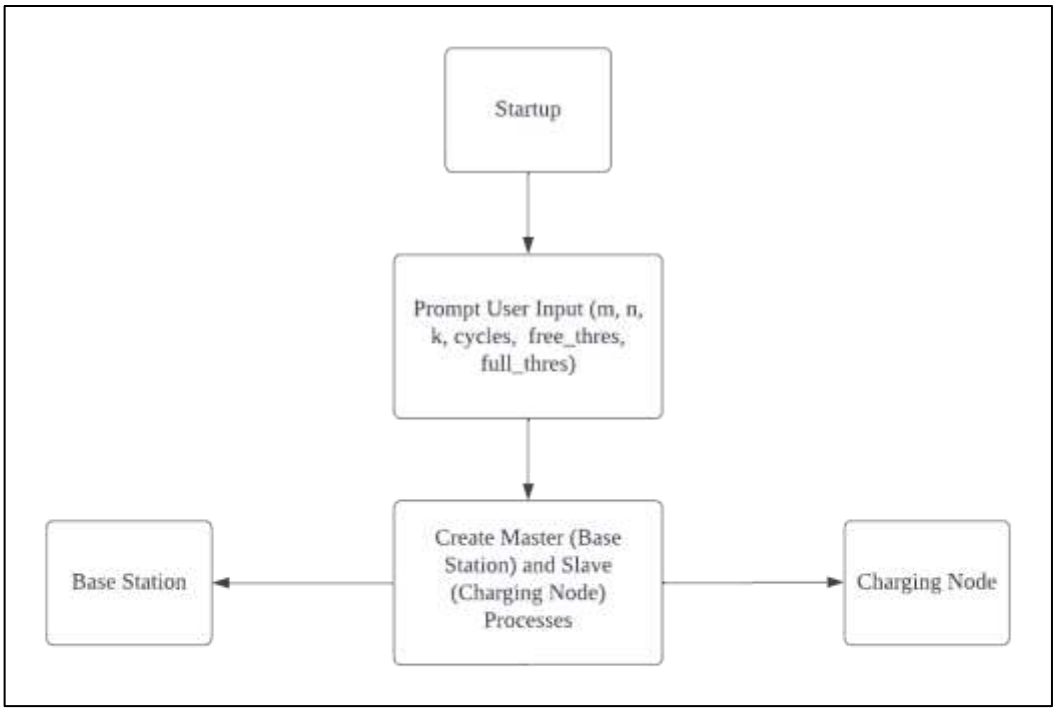

Figure 2. Initialisation of the WSN.

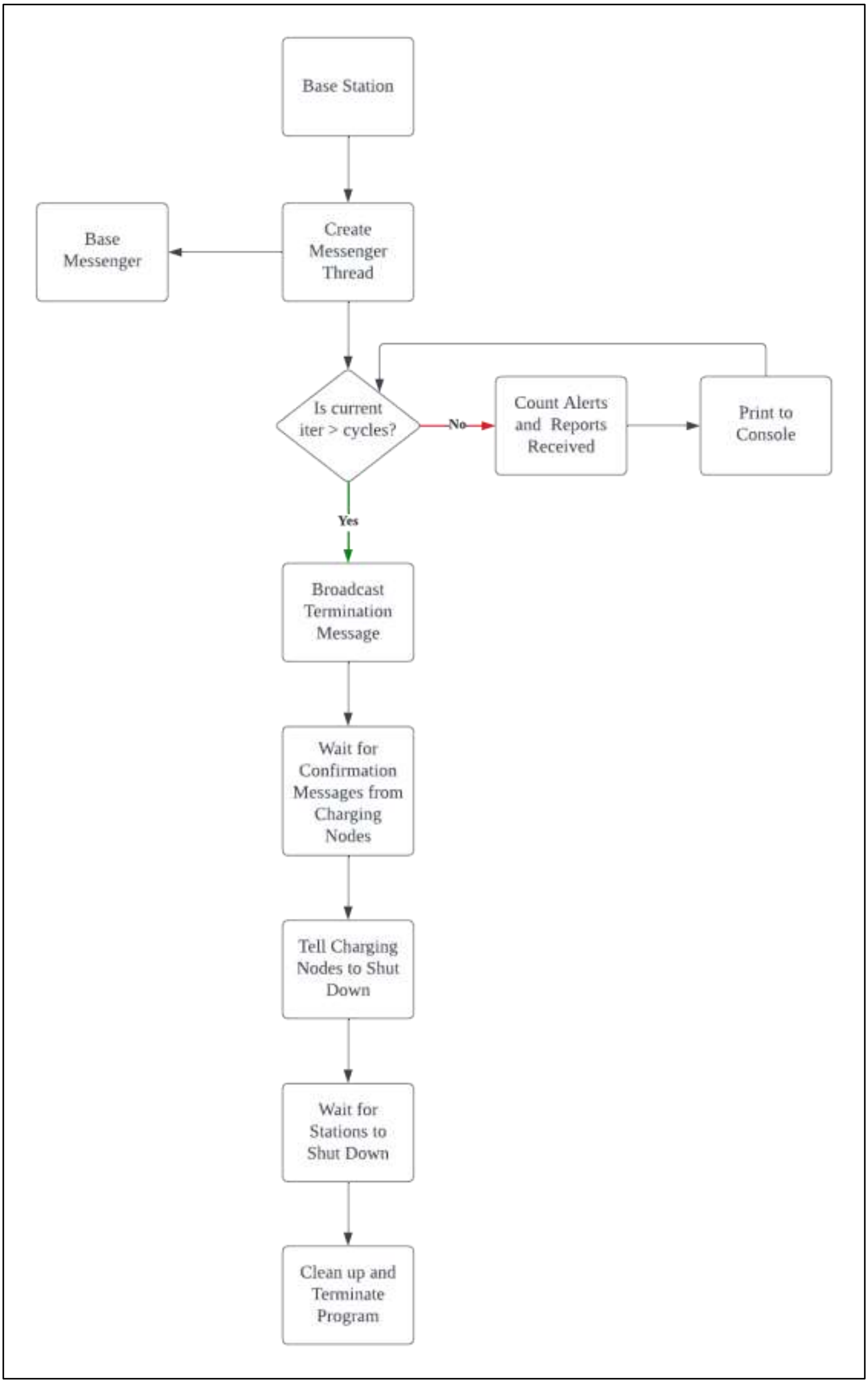

Figure 3. Procedure of Base Station

```
Shutting down for maintenance
End flag broadcasted
Station 0 sending confirmation to base
Station 2 sending confirmation to base
Station 1 sending confirmation to base
Goodbye from Station 0's Messenger
Station 3 sending confirmation to base
Goodbye from Station 3's Messenger
All stations have sent confirmation messages
Goodbye from Station 1's Messenger
Goodbye from Station 2's Messenger
Charging Station 0 has shut down
Charging Station 2 has shut down
Charging Station 1 has shut down
Charging Station 3 has shut down
All stations have shut down
Goodbye from Base Station
```

Figure 4. Snapshot of Proper Termination Messages.

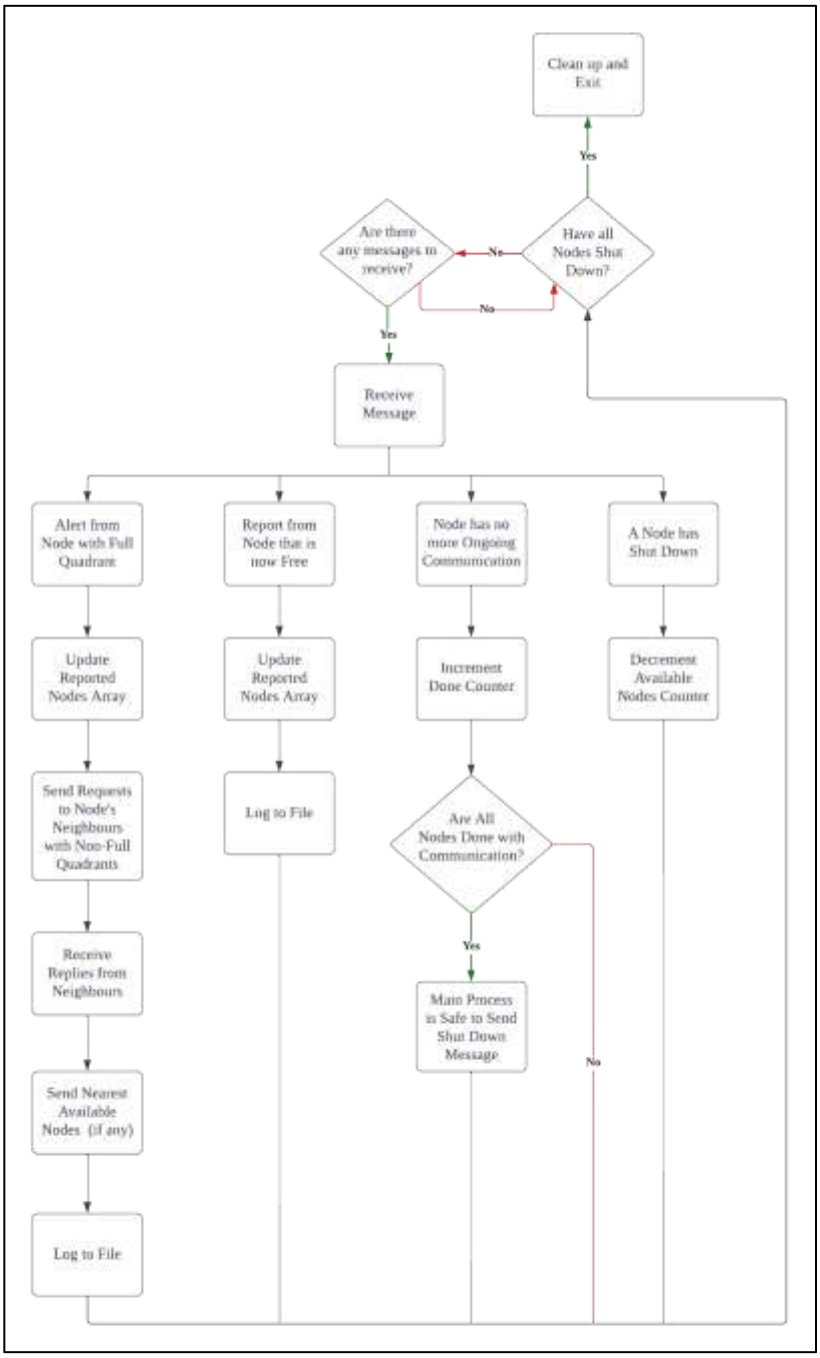

Figure 5. Procedure of the Base Messenger

MONASH University

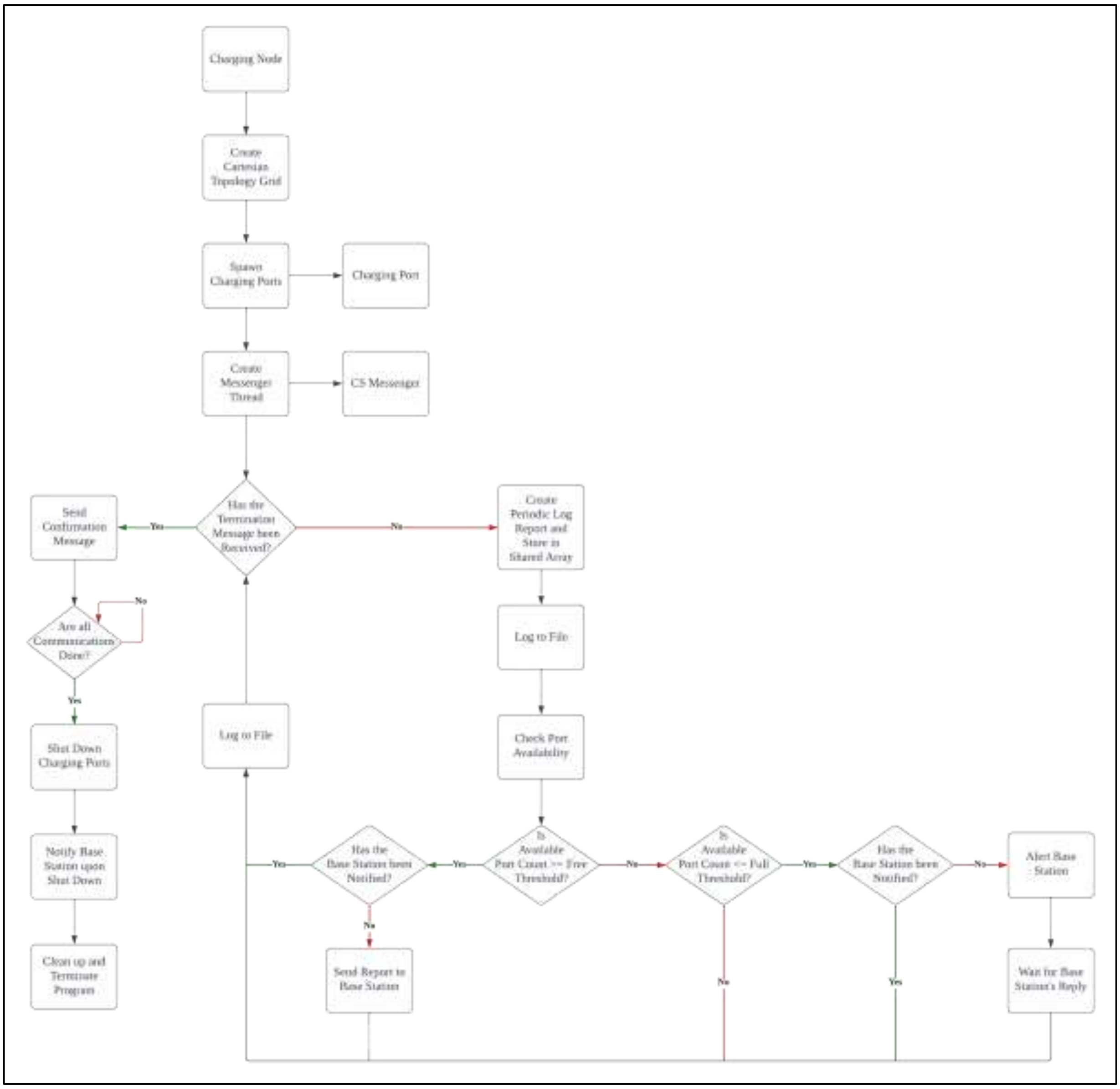

Figure 6. Procedure of a Charging Node

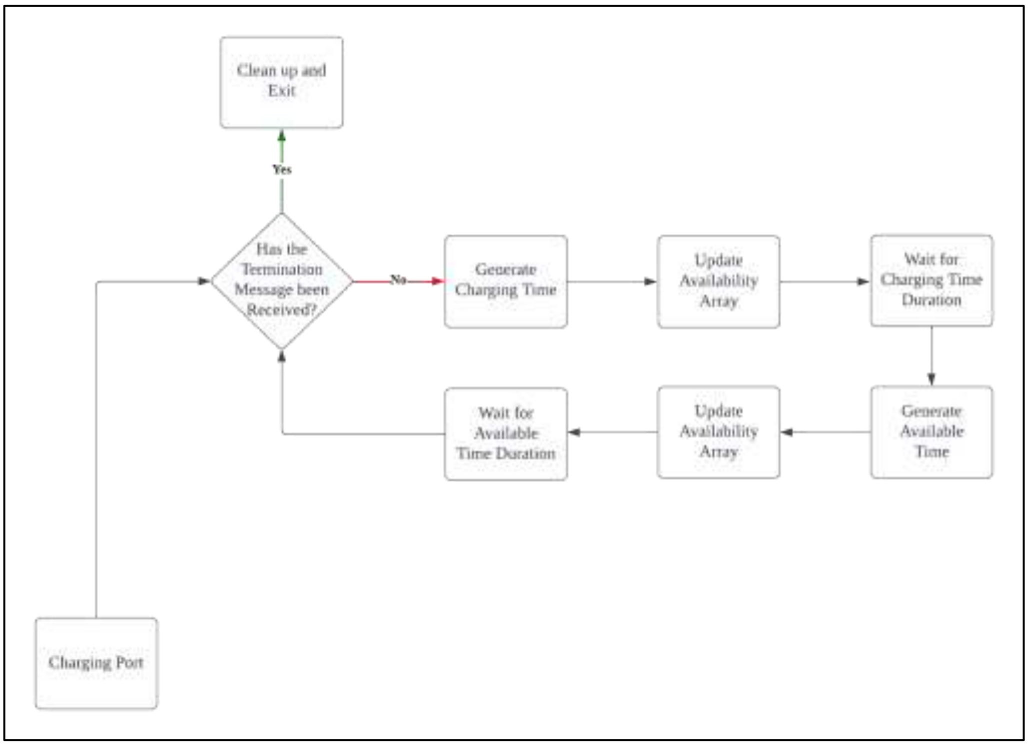

Figure 7. Procedure of a Charging Port

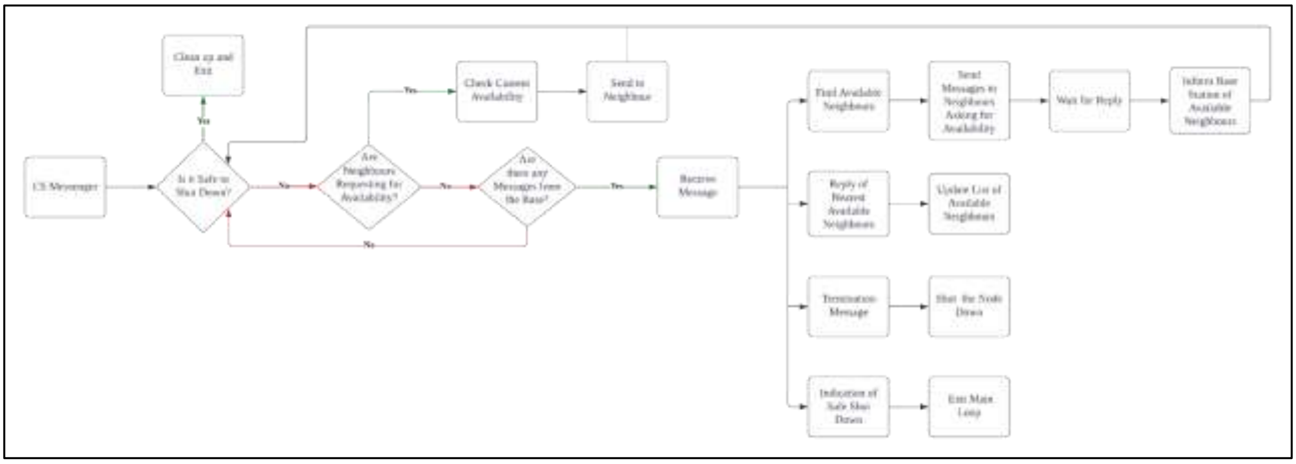

Figure 8. Procedure of a CS Messenger

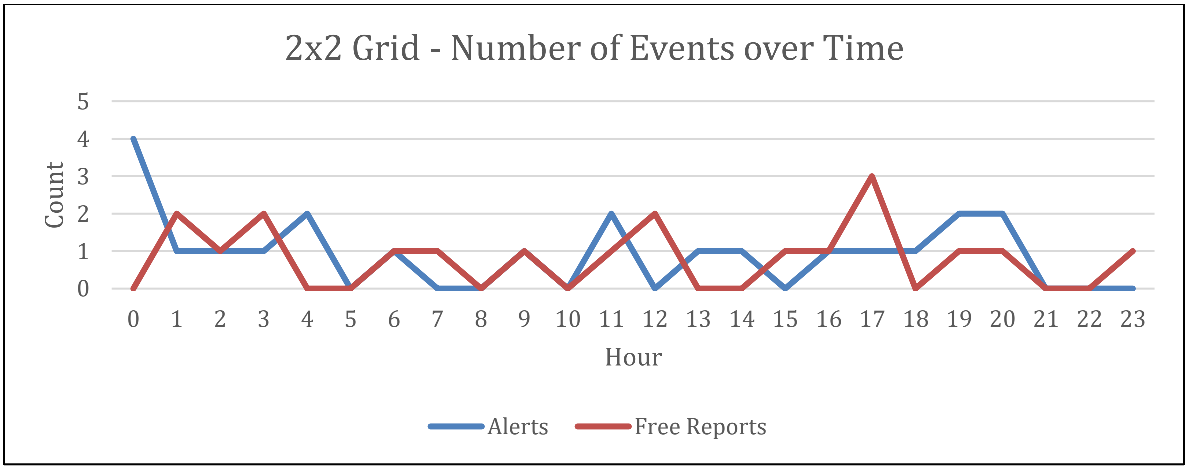

Figure 9. Chart of Triggered Events in Simulation 1 (Local Machine).

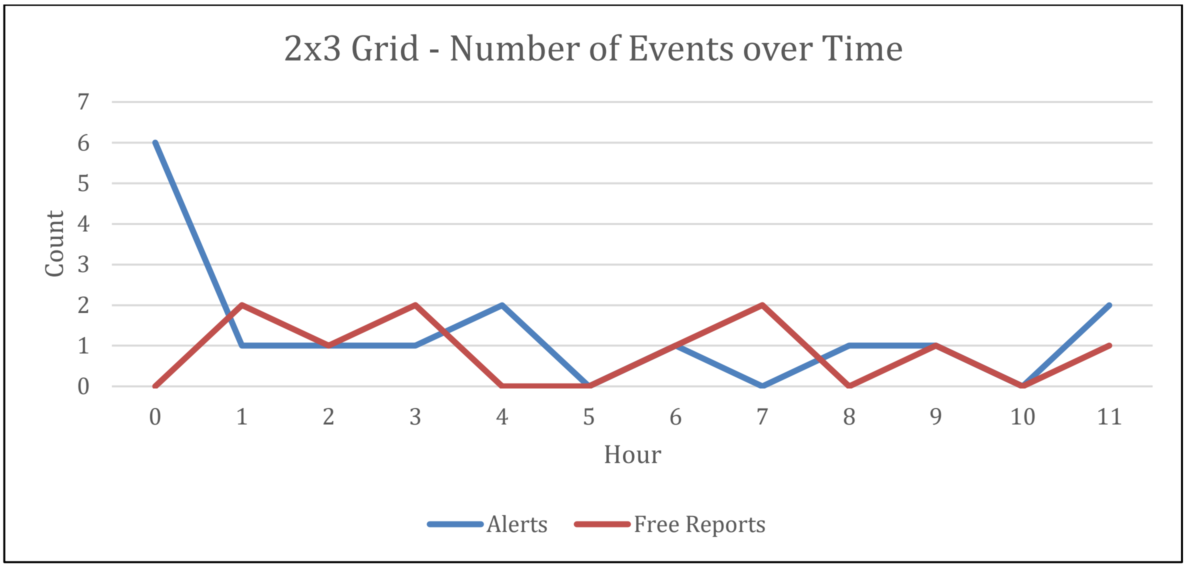

Figure 10. Chart of Triggered Events in Simulation 2 (Local Machine).

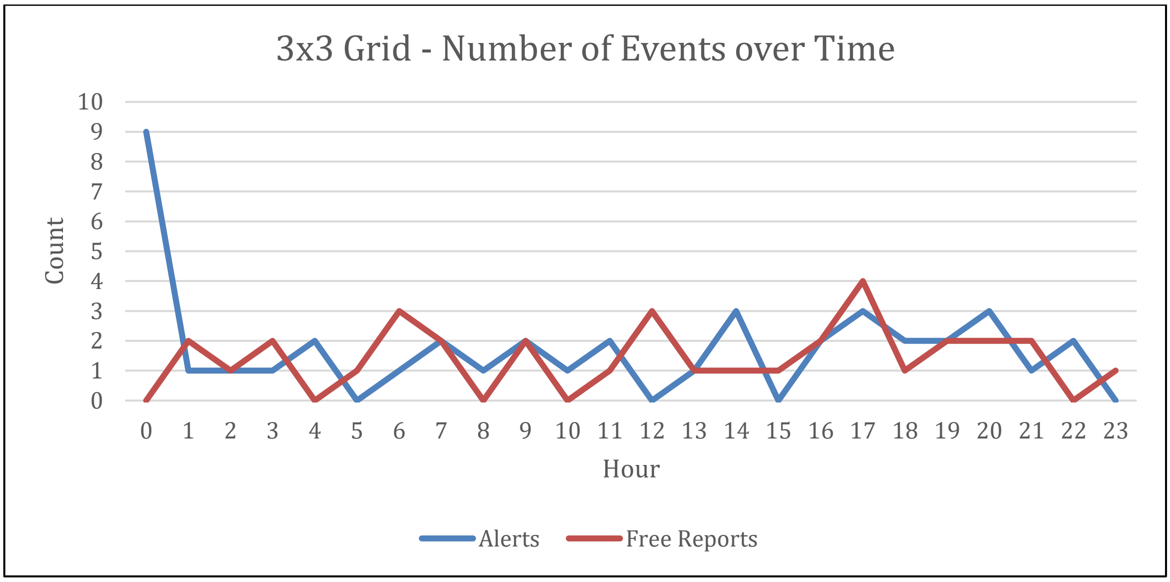

Figure 11. Chart of Triggered Events in Simulation 1 (CAAS)

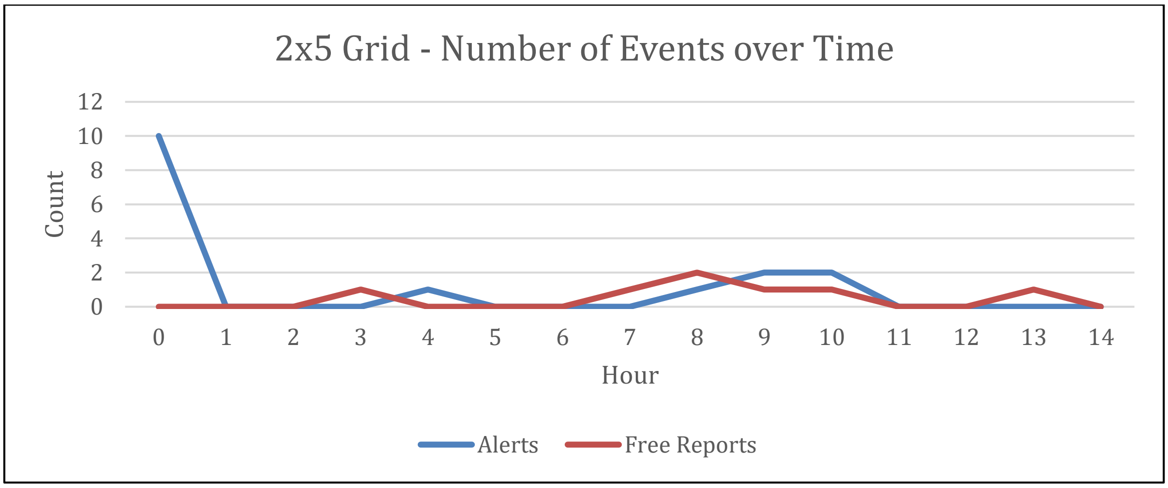

Figure 12. Chart of Triggered Events in Simulation 2 (CAAS)

```
--------------------------------------------------------------------------------
Full Report 1
Time is 0:10:0
Received report from heavily-in-use station 2 at (1,0), current availability is 0 port(s)
Sent available neighbouring stations nearby:
None
Communication Time (nanoseconds): 12100.00
Number of messages exchanged: 5
--------------------------------------------------------------------------------
```

Figure 13. A Base Station Log of Receiving an Alert from a Full Station.

```
--------------------------------------------------------------------------------
Free Report 1
Time is 1:40:0
Received report from free station 1 at (0,1), current availability is 2 port(s)
Number of messages exchanged: 1
--------------------------------------------------------------------------------
```

Figure 14. A Base Station Log of Receiving a Free Report from a Previously Reported Station.

```
--------------------------------------------------------------------------------
Base Station has shut down
Total Full Reports (Alerts) Received: 22
Total Free Reports Received: 19
Total Messages Exchanged: 95
Total Communication Time (nanoseconds): 330801.00
```

Figure 15. A Base Station Log of Termination and Summary of Events.

```
|  Hour  | Minute | Second | Availability |
|       0|      10|       0|             0|
--------------------------------------------------------------------------------
Station 0 is heavily in use at 0:10:0, current availability is 0 port(s)
Available neighbouring stations: None
Nearest available stations from base: None
--------------------------------------------------------------------------------
```

Figure 16. A Charging Station Log with a Full Quadrant.

```
|       1|      50|       0|             1|
--------------------------------------------------------------------------------
Station 0 is heavily in use at 1:50:0, current availability is 1 port(s)
Available neighbouring stations: 1 2
--------------------------------------------------------------------------------
```

Figure 17. A Charging Station Log with Free Neighbouring Nodes.

```
|      23|      40|       0|             2|
--------------------------------------------------------------------------------
Station 0 is free at 23:40:0, current availability is 2 port(s)
--------------------------------------------------------------------------------
```

Figure 18. A Charging Station Log when it is Free.

```
Station has shut down
Total number of requests sent: 7
Total communication time with base(nanoseconds): 2515122.00
Total communication time with other stations(nanoseconds): 9313686.00
```

Figure 19. A Charging Station Log Indicating Successful Shut-Down.

Local Machine Simulation 1		
Specifications		
Charging Grid Size (m x n): 2 x 2 Number of Charging Ports per Node (k): 3 Number of Iterations: 24 Full Threshold: >= 2 Ports Free Threshold: <= 1 Port		
Results		
Total Messages Exchanged: 95 Total Communication Time: 819103 ns		
Hour	**Alerts**	**Free Reports**
0	4	0
1	1	2
2	1	1
3	1	2
4	2	0
5	0	0
6	1	1
7	0	1
8	0	0
9	1	1
10	0	0
11	2	1
12	0	2
13	1	0
14	1	0
15	0	1
16	1	1
17	1	3
18	1	0
19	2	1
20	2	1
21	0	0
22	0	0
23	0	1
Total	22	19

Table 1. Specifications and Results Tabulation of Simulation 1 on Local Machine

Local Machine Simulation 2		
Specifications		
Charging Grid Size (m x n): 2 x 3 Number of Charging Ports per Node (k): 3 Number of Iterations: 12 Full Threshold: >= 2 Ports Free Threshold: <= 1 Port		
Results		
Total Messages Exchanged: 75 Total Communication Time: 1239900 ns		
Hour	**Alerts**	**Free Reports**
0	6	0
1	1	2
2	1	1
3	1	2
4	2	0
5	0	0
6	1	1
7	0	2
8	1	0
9	1	1
10	0	0
11	2	1
Total	16	10

Table 2. Specifications and Results Tabulation of Simulation 2 on Local Machine

CAAS Platform Simulation 1		
CAAS		
Nodes: 2 Tasks per Node: 5 CPUs per Task: 3		
Specifications		
Charging Grid Size (m x n): 3 x 3 Number of Charging Ports per Node (k): 3 Number of Iterations: 24 Full Threshold: >= 2 Ports Free Threshold: <= 1 Port		
Results		
Total Messages Exchanged: 208 Total Communication Time: 3408698 ns		
Hour	**Alerts**	**Free Reports**
0	9	0
1	1	2
2	1	1
3	1	2
4	2	0
5	0	1
6	1	3
7	2	2
8	1	0
9	2	2
10	1	0
11	2	1
12	0	3
13	1	1
14	3	1
15	0	1
16	2	2
17	3	4
18	2	1
19	2	2
20	3	2
21	1	2
22	2	0
23	0	1
Total	42	34

Table 3. Specifications and Results Tabulation of Simulation 1 on CAAS Platform

CAAS Platform Simulation 2		
CAAS		
Nodes: 4 Tasks per Node: 3 CPUs per Task: 5		
Specifications		
Charging Grid Size (m x n): 2 x 5 Number of Charging Ports per Node (k): 5 Number of Iterations: 15 Full Threshold: >= 4 Ports Free Threshold: <= 2 Port		
Results		
Total Messages Exchanged: 85 Total Communication Time: 2291882 ns		
Hour	**Alerts**	**Free Reports**
0	10	0
1	0	0
2	0	0
3	0	1
4	1	0
5	0	0
6	0	0
7	0	1
8	1	2
9	2	1
10	2	1
11	0	0
12	0	0
13	0	1
14	0	0
Total	16	7

Table 4. Specifications and Results Tabulation of Simulation 2 on CAAS Platform

CAAS Platform Simulation 3		
CAAS		
Nodes: 2 Tasks per Node: 3 CPUs per Task: 3		
Specifications		
Charging Grid Size (m x n): 2 x 2 Number of Charging Ports per Node (k): 3 Number of Iterations: 24 Full Threshold: >= 2 Ports Free Threshold: <= 1 Port		
Results		
Total Messages Exchanged: 101 Total Communication Time: 2065108 ns		
Hour	**Alerts**	**Free Reports**
0	4	0
1	1	2
2	1	1
3	1	2
4	2	0
5	0	0
6	1	1
7	0	1
8	0	0
9	1	1
10	0	0
11	2	1
12	0	2
13	1	0
14	1	0
15	0	1
16	1	1
17	1	3
18	1	0
19	2	1
20	2	1
21	0	0
22	0	0
23	0	1
Total	22	19

Table 5. Specifications and Results Tabulation of Simulation 3 on CAAS Platform

Grid Size (m x n)	Platform	Average Time Taken (ns/message)
2 x 2	Local	$\frac{819103\ ns}{95\ messages} \approx 8622$
2 x 2	CAAS	$\frac{2065108\ ns}{101\ messages} \approx 20446$
2 x 3	Local	$\frac{1239900\ ns}{75\ messages} \approx 16532$
3 x 3	CAAS	$\frac{3408698\ ns}{208\ messages} \approx 16387$
2 x 5	CAAS	$\frac{2291882\ ns}{85\ messages} \approx 26963$

Table 6. Average Message Exchanging Time on Different Grid Sizes.

Local Simulation 1				
Node	**Coordinates**	**Messages to Base**	**Communication Time with Base (ns)**	**Communication Time with Neighbours (ns)**
0	(0,0)	7	2515122	9313686
1	(0,1)	6	2579726	3519620
2	(1,0)	5	130500	2338904
3	(1,1)	4	2422626	9637792
Local Simulation 2				
Node	**Coordinates**	**Messages to Base**	**Communication Time with Base (ns)**	**Communication Time with Neighbours (ns)**
0	(0,0)	4	6971102	1976505
1	(0,1)	3	7910803	3091303
2	(0,2)	4	118900	1246503
3	(1,0)	2	224601	2552000
4	(1,1)	1	16899708	6461251
5	(1,2)	2	2342301	1666806
CAAS Simulation 1				
Node	**Coordinates**	**Messages to Base**	**Communication Time with Base (ns)**	**Communication Time with Neighbours (ns)**
0	(0,0)	7	3080766	2520081
1	(0,1)	6	2289047	15523953
2	(0,2)	5	7315349	18156871
3	(1,0)	4	1215402	19632383
4	(1,1)	5	7350643	13360120
5	(1,2)	3	5249936	16605708
6	(2,0)	3	3351685	17035391
7	(2,1)	3	6376674	3456227
8	(2,2)	6	9459240	2842214
CAAS Simulation 2				
Node	**Coordinates**	**Messages to Base**	**Communication Time with Base (ns)**	**Communication Time with Neighbours (ns)**
0	(0,0)	2	2335797	10011683
1	(0,1)	1	3219806	17765570
2	(0,2)	2	3263467	11755414
3	(0,3)	1	5591810	10388996
4	(0,4)	1	5597921	7771409
5	(1,0)	1	7330494	7821225
6	(1,1)	2	9977055	9643783
7	(1,2)	2	948414	13090286
8	(1,3)	3	1304851	16785174
9	(1,4)	1	13477607	10542077
CAAS Simulation 3				
Node	**Coordinates**	**Messages to Base**	**Communication Time with Base (ns)**	**Communication Time with Neighbours (ns)**
0	(0,0)	7	3063692	14876993
1	(0,1)	6	383331	14625094
2	(1,0)	5	1911380	14803552
3	(1,1)	4	3318580	12744005

Table 7 Communication Times for Each Simulation.