Design Document
ID: I

TOPGEN: A Network Topology Generator for Bisque

Hejun Wu Qiong Luo
Department of Computer Science
Hong Kong University of Science and Technology
{whjnn, luo}@cs.ust.hk

This document presents the design of TOPGEN, a tool that generates various network topologies for Bisque.

1 System Overview

Given the number of nodes in the network and the grid width, TOPGEN generates the grid sensor network topology. It describes the sensor network topology with an XML file that contains both the locations of nodes and the links between nodes.

2 Structure of TOPGEN

The structure of TOPGEN is shown in Figure 1. Because Bisque specifies that the nodes in a network be deployed in a grid topology, TOPGEN takes the number of nodes N and the grid width *a* as input. With this input, TOPGEN deploys the nodes, and writes the locations and the links into two tables. Finally, it outputs the tables into a network configuration file (*.xml).

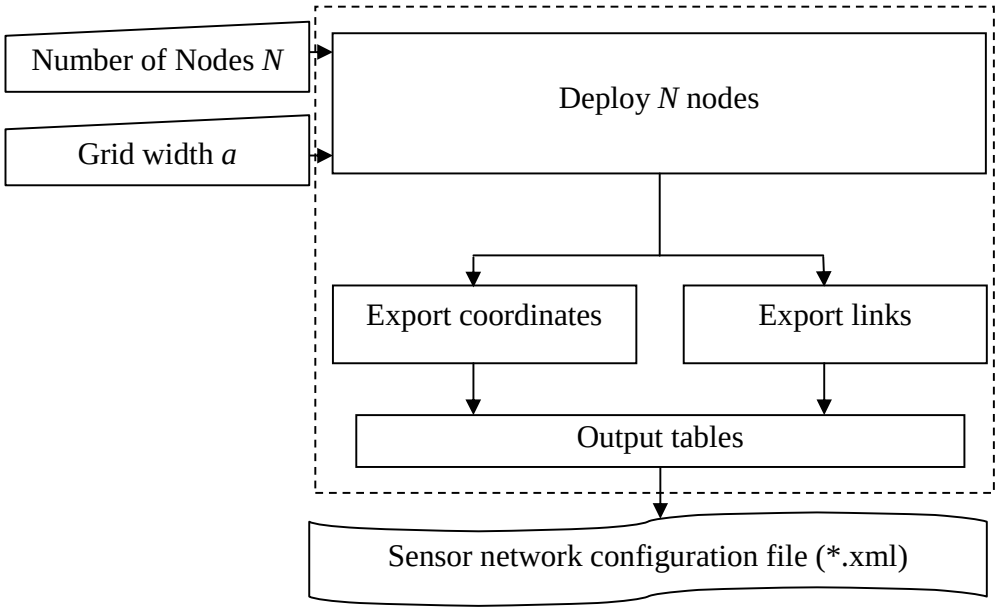

Figure 1. TOPGEN structure

3 Module Design

As shown in Figure 1, TOPGEN is composed of a user interface and four functional modules. The user interface gets the number of nodes and the grid width from a user. The deploying module assigns the locations of the *N* nodes. The remaining modules output the deployment as an XML file. These modules are described in the following.

- **User interface**

TOPGEN uses dialog windows to ask the number of nodes and the grid width of the deployment. An example dialog window is shown in Figure 2.

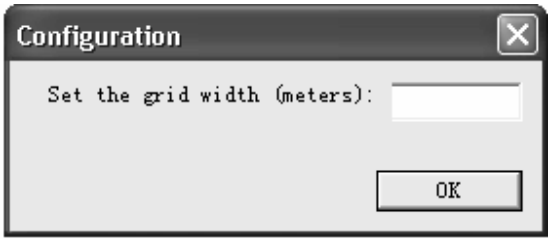

Figure 2. The dialog for setting the grid width

- **Deploying module**

This module reads the number of nodes *N* and the grid width *a*, and deploys the *N* nodes. As specified in Bisque, *N* is equal to *m*m*, where *m* is an integer. The *m*m* nodes (including a sink) are deployed at the cross points of an *(m-1)*(m-1)* grid with the cell width *a= w/(m-1)*. Without causing confusion, we call this grid an *m*m* grid. The position of each node is denoted as coordinate *(x, y)*, with the node at the lower left corner being *(0, 0)* and the node at the upper right corner *(w,w)*. The sink resides at or near the center of the area: if m is odd, it is at *(w/2, w/2)*; otherwise, it is at *((m-2)a/2, (m-2)a/2)*.

According to the current version specification of Bisque, a link between two nodes is determined given the transmission range and their distance. If the distance between two nodes is not larger than the transmission range, there is a link between the nodes. Otherwise, there is no link between them.

Figure 3 shows an example of a 9-node deployment. The transmission range TR is 10 meters. The sink node is put at the center *(10, 10)*. There are four links from the sink to the four nodes with *x = 10* or *y = 10*. The minimal number of hops from a corner node to the sink node is two.

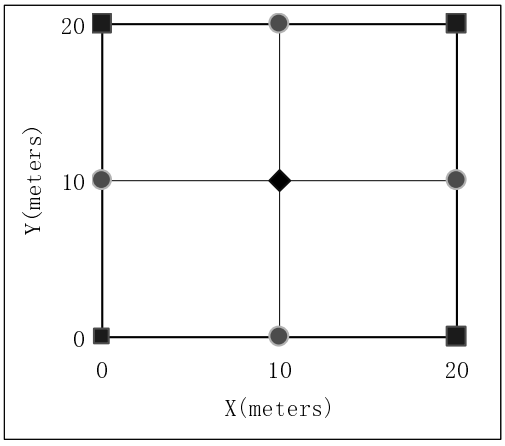

Figure 3. Deployment example

- **The two exporting modules**

The two exporting modules extract the coordinates and the links from the deploying module. They output the coordinates and the links as two tables.

- **Output module**

This module reads the two tables generated by the two exporting modules, and writes the coordinates and the links in the tables into the configuration XML file. A fraction of a configuration file is shown in Figure 4.

```
…
<mote id="1" model="mica2" state="on">
        <param name="position" value="0, 5" />
                <flash href="main.exe" />
</mote>
<mote id="2" model="mica2" state="on">
        <param name="position" value="0, 10" />
                <flash href="main.exe" />
</mote>
...
```

Figure 4. A fraction of network configuration file