Borel Cayley Graph-based Topology Control for Power Efficient Operation in Ad Hoc Networks

Abdellatif Serhani
RAI2S, INPT
University of Mohammed V
Rabat, Morocco
abdollatiif@gmail.com

Najib Naja
RAI2S, INPT
University of Mohammed V
Rabat, Morocco
naja@inpt.ac.ma

Abdellah Jamali
RI2M, ENSA
University of Hassan I
Settat, Morocco
abdellah.jamali@uhp.ac.ma

***Abstract*—In this paper, we focus on the design of network topology control algorithm in wireless ad hoc networks. We approach topology control with the well known Borel Cayley Graphs (BCGs), a family of pseudo-random graphs. BCGs have been shown to be an efficient candidate topology in interconnection networks due to their small diameter, short path length and low degree. We consider the problem of adjusting the transmit powers of nodes before assigning IDs of Borel Cayley Graphs as logical topologies in wireless ad hoc networks. We compare the performance of our algorithm with other existing topology control algorithms. Our simulation indicates that the proposed ID assignment has a better performance compared with other assignment methods.**

***Index Terms*—Distributed algorithms, power management, Borel Cayley Graph, Topology Control**

I. Introduction

A multihop wireless networks, such as radio networks, ad hoc networks, and sensor networks, are networks where a packet may have to traverse multiple consecutive wireless links in order to reach its destination. Mobile Ad hoc NETwork (MANET) is a mobile, multihop wireless network which is capable of autonomous operation. It is characterized by energy-constrained nodes, bandwidth-constrained, variable-capacity wireless links and dynamic topology, leading to frequent and unpredictable connectivity changes.

The absence of a central infrastructure implies that an ad hoc network does not have an associated fixed topology. Indeed, topology control, defined by how to determine an appropriate topology over which high-level routing protocols are implemented, has emerged to be one of the most important issues in wireless multi-hop networks [1]. Topology control is important in wireless ad hoc networks for at least two reasons: (*i*) It affects networks spatial reuse and hence the traffic carrying capacity, and (*ii*) It impacts on contention for the medium and hence mitigates MAC-level interference.

The topology depends on ”uncontrollable” factors, such as node mobility, weather, interference, noise, as well as on ”controllable” parameters, such as transmit power and antenna direction [2]. Then, several topology control algorithms lie in the fact that each node can potentially change the network topology by changing the transmit powers to control its set of neighbors [2] [4] [5]. Whereas, other techniques react on routing mechanisms, based on predefined metrics, that efficiently deal with changes in the topology due to uncontrollable factors.

According to [3], topology control algorithms can be classified as location-based, direction-based, or neighbor-based approaches. In each of these approaches, a resulting topology is returned by the topology control algorithm in an ad hoc manner. However, since these approaches do not generate a predefined graph, there is no guarantee on the topology’s overall properties, such as a bounded diameter and average path length.

In this paper, we approach topology control with predefined structured graphs. We react on adjusting transmission power to reduce interference and hence expanding the network’s lifespan. Thus, we achieve higher throughput as compared to schemes that use fixed transmission power [6]. Structured graphs have been studied for a long time and are good candidates for interconnection networks [7]. Various graph-based interconnection networks have been studied and used as good physical topologies for wavelength routing lightwave networks [7]. In peer-to-peer overlay network schemes, various predefined structured graphs are used to form a P2P overlay. The proposed systems utilize the properties of the graph to form the overlay network and develop a routing algorithm. Examples of P2P overlay networks include Arrangement-Graph Overlay (AGO) to reduce system overhead and bind routing hops [8], k ring lattices with Chords [9], and PeerStar by utilizing the star interconnection network as a peer to peer topology [10]. Graph-based wireless sensor networks have also been explored by other researches [11] [12]. In general, a predefined graph topology with its deterministic connection rule facilitates performance analysis. In addition, some offer symmetry, hierarchy, and hamiltonicity, and can have a constant low degree and existing distributed routing protocol, all preferable properties for ad hoc networks.

This paper features three major contributions.

Fistly, we focus on Borel Cayley graphs (BCG), a member of Cayley graph family [13], and we are interested in the problem of imposing and overlying the physical topology, defined by the transmission range of each node, by the logical topology dictated by our BCGs. We focus on this theoretic structure graphs because of their relatively small diameter

 ISBN: 978-1-61208-920-1

and low constant degree. Furthermore, these graphs also have a small average hop count between nodes and a simple distributed routing algorithm [14]. Let ***V*** denote the collection of nodes and let ***G*** denote the physical graph on ***V*** in which there is an edge from node *u* to node *v* if and only if *u* can directly reach *v*. when running our topology control algorithm, each node in ***V*** can be assigned an ID of the entire BCGs, and in result returning an undirected subgraph T_{BCG}, such that (*i*) T_{BCG} consists of all the nodes in ***G*** but has fewer edges, (*ii*) if *u* and *v* are connected in ***G***, they are still connected in T_{BCG}, and (*iii*) a node *u* can transmit to all its neighbors in T_{BCG} using less power than is required to transmit to all its neighbors in ***G***. Secondly, in our topology control algorithm, each node *u* in the network constructs its neighbor set $N(u) = \{v|(u,v) \in \boldsymbol{T}\}$ in a distributed fashion. Finally, if T_{BCG} changes to T_{BCG}**'** due to node failures or mobility, our topology control algorithm reconstructs automatically a connected T_{BCG}**'** without global coordination.

The outline of this paper is as follows. Section 2 reviews the definition and properties of BCGs and gives preliminaries on outdoor radio propagation and the power consumption model. In Section 3, we present our distributed topology control algorithm. Section 4 describes network simulation results that show the effectiveness of the algorithm. Finally, Section 5 summarizes our contributions and conclusion.

II. Related work

Topology Control (TC) is defined as communication nodes having the ability to modify or select their neighbors or connections (i.e. active links) at their will. This allows each node to be part of the network, based on its own judgment of constraints and resources for optimum operation. A network topology is formed in an ad hoc manner based on nodes' location, direction, or some sort of neighbor ordering. Topology control also aims at achieving specific design goals such as energy efficiency and interference mitigation by selecting logical neighbors and adjusting the transmission power accordingly.

In [15], Blough et al provided a $K\text{-}Neigh$ topology control algorithm in which a node chooses k closest physical neighbors as logical neighbors. $K\text{-}Neigh$ is a basic neighbor-based topology control protocol based on the construction of k-neighbor graph as logical communication graph that guarantees an interference bounded topology connected with high probability, provided the maximum power topology (i.e., the graph obtained when all the nodes transmit at maximum power) is connected. $K\text{-}Neigh$ is a simple, fully distributed, asynchronous, and localized protocol. The overall number of messages exchanged by $K\text{-}Neigh$ is exactly two times of the number of nodes and the execution time is strictly bounded.

In [16], Wattenhofer et al. described the XTC topology control algorithm which is one of the few topology control protocols which are location-free. XTC is based on ranking information between nodes. Their algorithm try to remove long links while preserving network connectivity to force nodes to use multiple short hops, which saves energy and prolongs network lifetime. Their technique does not assume location information, neither does it require the network to be a Euclidean graph. The XTC algorithm consists of three steps. In the first step, each node broadcasts once at the maximum power and then ranks all its neighbors according to its link quality to them (from high to low). Each node transmits its ranking results to neighboring nodes during the second step. In the final step, each node locally examines all of its neighbors in the order of their ranking and decides which one needs to be directly linked. The XTC algorithm features the basic properties of topology control such as symmetry and connectivity while running faster than most previous algorithms.

Ning Li et al. [17] presented a Minimum Spanning Tree (MST) based topology control algorithm, called Local Minimum Spanning Tree ($LMST$). Nodes in $LMST$ topology control algorithm construct a local minimal spanning tree with a bounded logical node degree and establish bidirectional link with a guaranteed connectivity. Each node builds its local MST independently and only keeps on-tree one-hop nodes as its neighbors. Their approach tries to minimize the overhead to maintain a connected topology in a dynamic wireless ad hoc network which the degree of any node in the resulting topology can be bounded by 6. The bounded degree on each node is desirable because a small degree reduces the MAC level contention and interference.

Ramanathan et al. [2] proposed to adjust incrementally node transmit powers in response to topological changes so as to maintain a connected topology using minimum power. Using (bi)connectivity as their objective, They described optimal centralized algorithms and distributed heuristics for transmit power control. In a mobile ad hoc network, they proposed a two distributed heuristics for topology control, namely Local Information No Topology ($LINT$) and Local Information Link-State Topology ($LILT$). In the the former, each node checks periodically the number of active number of neighbors (degree) in its neighbor table (built by the routing mechanism) and adjust the transmit powers based on the formula 1 to reduce the power.

$$p_d = p_c - 5.\mathcal{E}.log(\frac{d_d}{d_c}) \tag{1}$$

where d_d, d_c, p_c and p_d denote, respectively, the desired degree, the current degree, the current transmit power of a node, and the targeted power. The propagation loss function varies as some $\mathcal{E}$ power of distance.

In these techniques and in many other TC techniques, such as Common power ($COMPOW$) protocol [19], cone based topology control ($CBTC$) [4], extended topology control (XTC) [16], $K\text{-}Neigh$ topology control and adaptive neighbor-based topology control ($ANTC$) [18], a network topology is formed in ah hoc manner to retain a minimum number of interconnections among the nodes that can communicate by expending very little energy. However, since these approaches do not generate a predefined graph, there is no guarantee on the topology's overall properties such as a bounded diameter and average path length. Furthermore, producing a degree bounded network topology that preserves connectivity in the worst-case is a challenge [15]. In other words, the goals of preserving worst-case connectivity and having a nontrivial upper bound on the physical node degree inherently conflict with each other.

Motivated by this observation, we tackle the TC problem with the goal of generating a network topology by imposing a predefined well-known $BCGs$ graph topology with a constant low-degree d_{BCG}. More precisely, we produce a network topology in which the physical node degree as well as the logical node degree are upper bounded by exactly the same BCG degree d_{BCG} by adjusting the transmit power for each node and its d-neighborhoods.

III. Preliminaries

In this section, we provide a brief summary of the definition and important properties of Cayley graphs, and Borel Cayley graphs [13].

A. Cayley Graph

A Cayley graph is a special family of pseudo-random graphs constructed from a finite group of elements which correspond to the nodes of the graph [14]. Connections between nodes of Cayley graphs are defined by the group operation and a set of generators. The formal definition of Cayley graphs is as follows:

Definition 1 (*Cayley graph* [14]). A graph $C = \mathbb{C}(V, G)$ is a Cayley Graph with vertex set V if two nodes $v_1, v_2 \in V$ are adjacent $\Leftrightarrow v_1 = v_2 * g$ for some $g \in G$, where $(V, *)$ is a finite group and $G \subset V \backslash \{I\}$. G is called the generator set of the graph and I is the identity element of the finite group $(V, *)$.

The definition of Cayley graph requires vertices to be elements of a group but does not specify a particular group. Thus, a Cayley graph can be generated over an arbitrary finite group, and there are many varieties of Cayley graphs.

B. Borel Cayley Graph

The Borel Cayley graphs (BCGs) are Cayley graphs constructed from Borel subgroups. The BCGs are regular, vertex transitive, and pseudo-random graphs [14]. The definition of a Borel subgroup is given below.

Definition 2 (*Borel subgroup* [14]). Let V be a Borel subgroup, $BL_2(\mathbb{Z}_p)$, of the nonsingular upper triangular 2×2 matrices $GL_2(\mathbb{Z}_p)$ with a parameter a such that $a \in \mathbb{Z}_p \backslash \{0, 1\}$, then

$$V = \left\{ \begin{pmatrix} x & y \\ 0 & 1 \end{pmatrix} : x \equiv a^t (\mathbf{mod}\, p),\, y \in \mathbb{Z}_p,\, t \in \mathbb{Z}_k \right\} \quad (2)$$

where p is prime, and k is the smallest positive integer such that $a^k \equiv 1 (\mathbf{mod}\, p)$.

Definition 3 (*Borel Cayley graph (BCG)* [14]). Let V be a Borel subgroup, and let G be a generator set such that $G \subseteq V \backslash \{I\}$; then $B = \mathfrak{B}(V, G)$ is a Borel Cayley graph with vertices for the 2×2 matrix elements of V and with directed edge from v to u if $u = v * g$, where $u \neq v \in V, g \in G$, and $*$ is a modulo-p multiplication chosen as a group operation.

Note that $n =\mid V \mid = p \times k$, where k is a factor of $p - 1$.

C. Node representation

Generalized Chordal Rings (GCR) [20] and the more specialized Chordal Rings (CR) [21], on the other hand, are two existing topologies that are defined in the integer domain and have a systematic and regular structure.

Definition 4 (*Generalized chordal ring (GCR)* [20]). A graph is a generalized chordal ring if its nodes can be labeled with integers mod n, the number of nodes, and there is a divisor q of n such that node i is connected to node j iff node $i + q$ is connected to node $j + q$ $(mod\, n)$.

According to this definition, nodes of GCR are classified into q classes, each class with $^n/_q$ elements. The classification is based on modulo q arithmetic. Two node having the same residue $(mod\, q)$ are considered to be in the same class. Since i connects to j implies $i + q$ connects to $j + q$ $(mod\, n)$, nodes in the same class have the same connection rules defined by the *connection constants* or *GCR constants*. When the GCR constants for the different classes are known, connections of the entire graph are defined.

A chordal ring (CR) is a special case of GCR, in which every node has $+1$ and -1 modulo n connections. In other words, all nodes on the circumference of the ring are connected to form a Hamiltonian cycle. A Hamiltonian cycle is a graph cycle through a graph that visit each node exactly once.

This GCR representation is useful for routing because nodes are defined in the integer domain and there is a systematic description of connections. An interesting proposition, given in [14], provided an explicit algorithm to transform any Cayley graph into GCR. We restate this proposition as follows:

Proposition 1 For any finite Cayley graph, $\mathbb{C}$, with vertex set V, and any $T \in V$ such that $T^m = I$, there exists a GCR representation of $\mathbb{C}$ with divisor $q = n/m$ where $n =\mid V \mid$.

The element T is refereed to as the transform element and it can be any element in the vertex set V. The elements of the group are then partitioned into q classes by premultiplying the transform element with the representing element of each class a_i.

Any vertex $v \in V$ is represented with T and a_i as follows [13]:

$$v = T^j * a_i \quad (3)$$

 ISBN: 978-1-61208-920-1

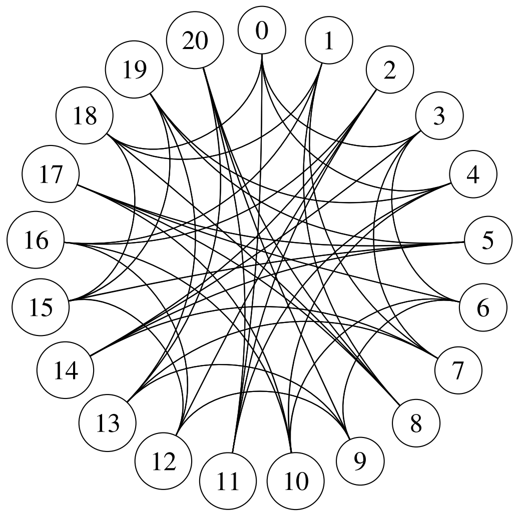

Fig. 1. Borel Cayley Graph with $p = 7, a = 2,$, and $k = 3$

Thus, there is a function that maps the Borel subgroup onto integer domain of GCR numbered between 0 and $n-1$, where n is the number of elements in the Borel subgroup:

$$\begin{aligned} f\colon \quad V &\to \quad \mathbb{N} \\ T^j * a_i &\to q * j + i \end{aligned} \tag{4}$$

D. Example (BCG with $p = 7, a = 2,$, and $k = 3$)

Figure 1 shows an example of Cayley graphs over $BL_2(\mathbb{Z}_p)$ with $a = 2$ and $p = 7$. for degree-4 graphs, there are four generators: $\mathbf{A}$, $\mathbf{B}$, $\mathbf{A^{-1}}$ and $\mathbf{B^{-1}}$. Let $\mathbf{A} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$ and $\mathbf{B} = \begin{pmatrix} 2 & 1 \\ 0 & 1 \end{pmatrix}$. Note that in this case, we have $k = 3$, $p = 7$, $n = p \times k = 21$ and the diameters is 3. We arbitrarily choose the transform element $\mathbf{T} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$ with $\mathbf{T}^7 = \mathbf{I}\ (mod\ p)$ to produce a GCR representation. Furthermore, we choose the representing element of class i to be $\mathbf{a_i} = \begin{pmatrix} a^i & 0 \\ 0 & 1 \end{pmatrix}$, $i = 0, ..., q-1$. Since $m = 7$, the divisor $q = n/m = 3$. Let $\mathbf{V} = \{0, 1, ..., 20\}$. For any $i \in \mathbf{V}$, if $i\ mod\ 3 =$:
”0”: i is connected to $i+3$, $i-3$, $i+4$, $i-10$ $(mod\ 21)$;
”1”: i is connected to $i+6$, $i-6$, $i+7$, $i-4$ $(mod\ 21)$;
”2”: i is connected to $i+9$, $i-9$, $i+10$, $i-7$ $(mod\ 21)$;

E. The power consumption model

Transmit power control has been recently proposed as a technique to increase network capacity and to reduce energy consumption in ad hoc network. Decreasing the nodes’ transmission power with respect to the minimum level potentially has two positive effects: (*i*) increasing the nodes’ energy consumption, and (*ii*) increasing the spatial reuse, with a positive overall effect on network capacity.

Although, the current literature on topology control has focused attention on energy consumption, trying to adjusting the transmit powers of nodes or to minimizing the energy cost in order to create a desired topology [2] [5]. In these works, the general idea is that increasing spatial reuse by providing upper bounds on the node degree in the final network topology. The rationale for considering node degree is that, if a node has relatively small degree, it will experience relatively low contention when accessing the wireless channel. Thus, the spatial reuse is increased, as well as network capacity. However, since these approaches do not generate a predefined graph, there is no guarantee on the final topologys bounded degree. Furthermore, producing a degree bounded network topology that preserves connectivity in the worst-case is impossible [15]. In other words, the goals of preserving worst-case connectivity and having a nontrivial upper bound on the physical node degree inherently conflict with each other.

Ramanathan and Rosales-Hain [2] propose an ingenious distributed topology control algorithm for transmit power control. For mobile networks, they present two distributed heuristics that adaptively adjust node transmit powers in response to topological changes and attempt to maintain a connected topology using minimum power. The idea in [2] inspires us to let each node adjusting their transmit powers before running our TC algorithm, and thus assuring a minimum network lifetime.

As in [2], each node is configured with two parameters, the ’desired” node degree that is the BCG’ degree d_{BCG}, and the current degree d_c that is the number of active neighbors in its neighbor table (built by the routing mechanism). The node in the network attempts to adjust its operational power periodically. If the current degree is greater than d_{BCG}, the node reduces its operational power. If the degree is less than d_{BCG}, the node increases its operational power. If neither is true, no action is taken.

Based on the well-known generic model for propagation [22], by which the propagation loss function varies as some $\mathcal{E}$ power of distance, Ramanathan et al. [2] derive the formula to reduce the power. Specifically, if d_d, d_c, p_c and p_d denote, respectively, the desired degree, the current degree, the current transmit power of a node, and the targeted power. Equation 1 computes transmit power p_d so that the node has the desired degree d_d.

In our system, we use $\mathcal{E} = 4$ and $d_d = d_{BCG}$. Equation 1 can thus be used to calculate the new power periodically. We note that the formula applies for both power increase and decrease to bring the degree close to d_{BCG}.

IV. Distributed network protocol

Mobility is a prominent characteristic of MANETs, and is the main factor affecting topology changes and route invalidation. Thus, node ID assignment becomes an important issue. An effective assignment allows most connections to be imposed and hence having most of the communication links of the original theoretic graph while preserving a small diameter and average path lengths.

A. Terminologies

We denote the logical node ID of node u in BCG by u_{id} and define the following.

1) $\mathcal{N}(u)$: Set of physical neighbors of node u.
2) $\mathcal{D}(u_{id})$: Set of logical neighbors of node u_{id} in BCG graph.
3) $\mathcal{T}(u_{id})$: Set of logical neighbors of node u_{id} in BCG graph that aren't yet assigned to u's physical neighbors.
4) $\mathcal{L}(u)$: Set of logical IDs of node u's physical neighbors.
5) $\mathcal{F}(v, u_{id}) = |\mathcal{D}(u_{id}) \cap \mathcal{L}(v)|$: the number of logical neighbors of u_{id} that are also logical IDs of node v's physical neighbors.

We define three different packet types.

1) **Token Assignment** packet: forwarded by nodes and used to initiate the distributed node ID assignment algorithm. the node that starts the operation sets an assignment counter A_{Cnt} to N. As the Token Assignment reach node v, A_{Cnt} decreased by one before forwarding the token to the next node, and assignment algorithm ends once A_{Cnt} reaches 0.
2) **Request Info** packet: Used by the node u holding the Token Assignment to broadcast $\mathcal{T}(u_{id})$ to its physical neighbors so as to identify those who do not yet have their ID.
3) **Reply Info** packet: if a physical neighbor node v, without any ID assigned yet, receiving a Request Info packet from node u, it sends a Reply Info packet to node u. The Request Info packet contains a list of $\mathcal{F}(v, x_{id})$ for each x_{id} proposed by u.

B. Algorithm

The distributed node ID assignment involves the following algorithm:

(*i*) Periodically, each node checks the number of active neighbors (degree) in its neighbor table built by routing mechanism. Equation 1 can thus be used to adjust transmit power to bring the degree close to d_{BCG}.

(*ii*) Assignment operation after node u received a Token Assignment packet.

Step 1: Node u received a Token Assignment packet.

1) If any ID u_{id} not yet assigned to u it takes randomly an u_{id} ranging from 0 to $N-1$.
2) Send a Request Info packet to all its physical neighbors so as to identify those who do not yet have their IDs.

Step 2: Node u collects the Reply Info packets.

1) If there is no Reply Info packet received, forward the Token packet randomly to one of its physical neighbors.
2) Having received a list of $\mathcal{F}(v, x_{id})$, $v \in \mathcal{N}(u)$ and $x_{id} \in \mathcal{T}(u_{id})$, from their physical neighbors. Node u grouped them for each proposed $x_{id} \in \mathcal{T}(u_{id})$ and sort each group in descending order. and assign to node v the ID x_{id} that have a larger number of $\mathcal{F}(v, x_{id})$ among all $\mathcal{F}(w, x_{id})$, $v \neq w \in \mathcal{N}(u)$.
3) Node u send to those neighbors their respective new $x_{id} \in \mathcal{T}(u_{id})$ values.

Step 3: Node u sends Token Assignment packet randomly to one of the node newly assigned.

(*iii*) The operation after node v receives a Request Info packet.

Step 1: A node v receives a Request Info packet.

Step 2: Node v calculates $\mathcal{F}(v, x_{id})$, $x_{id} \in \mathcal{T}(u_{id})$ that represent, for each proposed x_{id}, the number of logical neighbors of x_{id} that are also IDs of node v's physical neighbors.

1) if node v already has its ID v_{id}, the Request Info packet is ignored.
2) Otherwise, A node v send those calculated numbers piggybacked in Reply Info packet.

V. Simulation

In this section, we describe the setup for our considered scenario. Then, we present our estimation results.

A. Setup

Our method was designed to monitor wireless ad hoc mobile network, thus, we evaluated our topology control algorithm through a discrete-event network simulator (ns-3). The evaluation was done in terms of the following topological and spectral metrics:

- $Diameter$: The longest distance between any pair of nodes of a graph. A shorter diameter is a desirable property for large network.
- $Average\ path\ length$: The average of the shortest paths between all pairs of nodes in a graph.
- R_c: The percentage of the number of edges of the resultant graph over that of the original BCG. That is,

$$R_c = \frac{Number\ of\ edges\ in\ resultant\ graph}{Number\ of\ edges\ in\ original\ BCG} \tag{5}$$

In the simulation, we used 506, 1081 and 2265 nodes networks for BCGs. Table I summarizes the parameters values used for simulations. Parameter p and a determine N (the number of nodes) and BCG parameter k. Parameters t_1 and y_1 were used to define the first generator. Parameters t_2 and y_2 were used to define the second generator. For each network size, 100 distinct samples were collected. Nodes of network models were uniformly and randomly distributed in an area of $100 \times 100\ m^2$. For each radio range from $5m$ to $150m$.

We compared our topology control algorithms with existing topology controls: K-Neigh [15] , which the number of neighbors of node bounded by a giver value k, with $k = 6$, Max topology which all nodes within the maximum radio range are logical neighbors, and Dist-swap topology control based on BCGs [23] with a distributed node ID assignment algorithm (Dist-swap). in this section we name our assignment method: Assignment ID (Assign-ID).

TABLE I
PARAMETERS OF BOREL CAYLEY GRAPHS

N	p	k	a	t_1	t_2	y_1	y_2
506	23	22	7	1	7	1	1
1081	47	23	2	1	7	1	1
2265	151	15	2	1	7	1	1

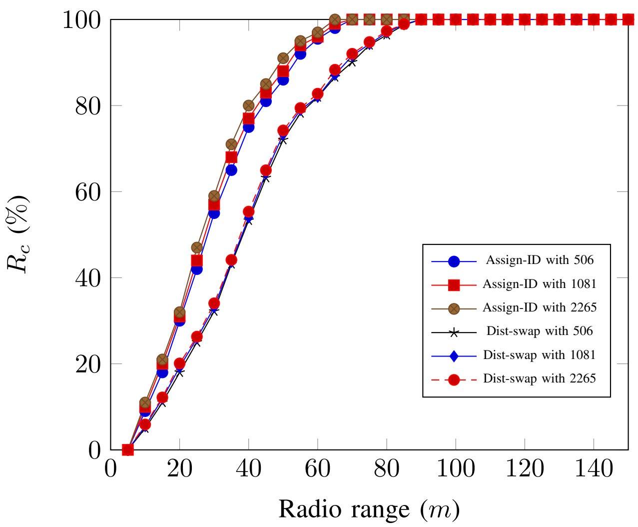

Fig. 2. Edge construction percentage with distributed node ID assignment.

We compute the energy consumed for all the nodes in the network during the multi-hop packet routing environments. We only consider the energy consumptions from data transmission (E_{tx}) and reception (E_{rx}).

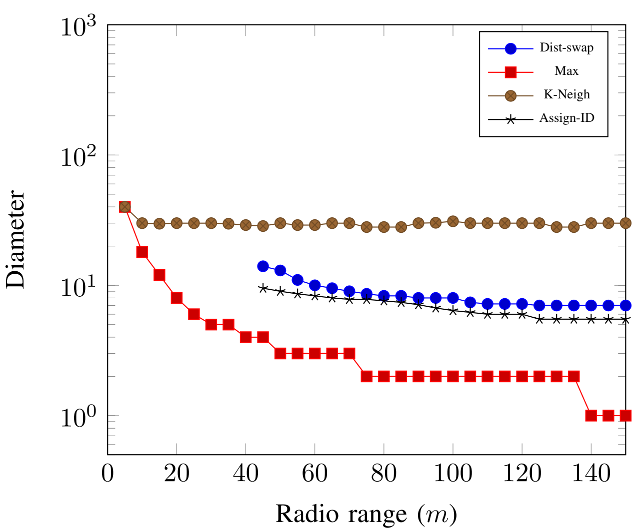

Fig. 3. Comparison of diameters

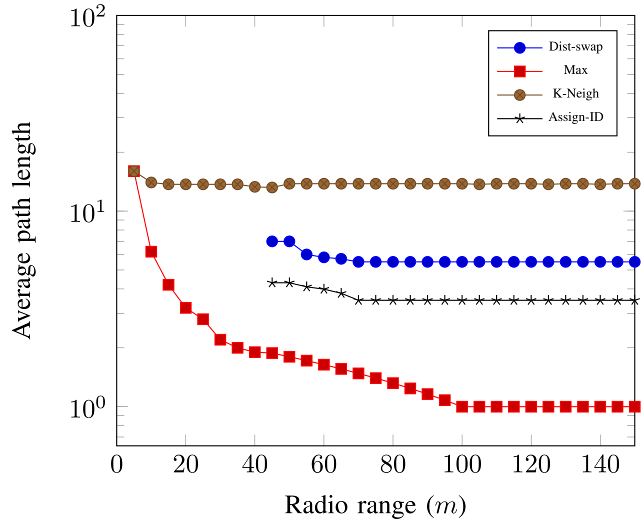

Fig. 4. Comparison of average path lengths

B. Simulation Results

Figure 2 shows R_c of Assign-ID and Dist-swap assignment methods. The Assign-ID shows that R_c is a slightly better than that of the Dist-swap assignment algorithm when the radio range is less than $80m$. For $R_c = 95\%$, the Dist-swap and Assign-ID needed $55m$ and $70m$ respectively. Overall, he Assign-ID showed better performance of R_c.

Figures 3 and 4 show the diameter and average path length versus the transmission range, respectively. We only compared the cases where the resultant graph is a connected graph. Thus, Dist-swap and Assign-ID both require $45m$ radio range to produce a connected network. From these figures, we confirm again that the Assign-ID have the smallest diameter and the shortest average path length for a given radio range except Max topology. This is because as a radio range increases, BCGs have a constant number of logical neighbors, while Max topology has much more logical neighbors than others.

In Figure 5, an interesting observation is that the energy consumed at each node is determined by the average path length and the diameter of the network topologies. Another observation is that the nodal energy consumption for Assign-ID is consistently smaller that that of the k-Neigh with $k = 6$.

Even though Max topology has the smallest diameter and the shorter average path length for a given radio range, due to its large number of logical neighbors, it consumed 60 times more nodal power than Assign-ID in Figure 5.

Max topology has a larger number of logical neighbors and hence larger power is consumed at the receiver modules.

VI. CONCLUSION

In this paper, we proposed the BCG-based topology control algorithm to impose the predefined BCG topology on wireless ad hoc networks. We proposed a new ID assignment method to overly our host network and have in return a new topology with a number of logical connections following predefined graph connection rules in a distributed manner.

 ISBN: 978-1-61208-920-1

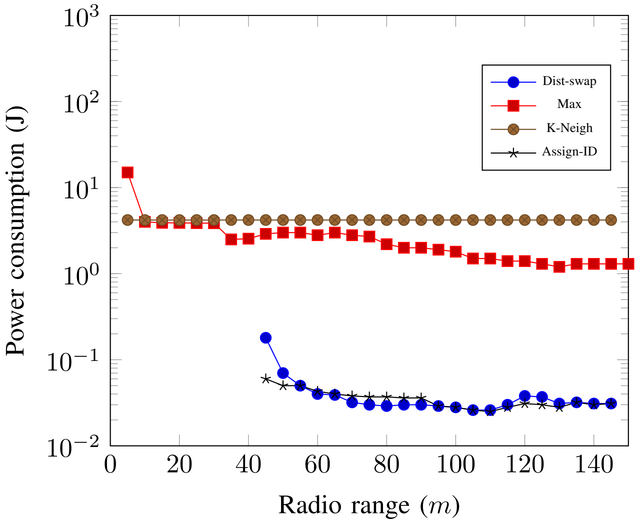

Fig. 5. Comparison of power consumption

Before assigning logical IDs to physical nodes in the network, our algorithm adjust node transmit powers in order to maintain a connected topology using minimum power and produce a topology with physical node degrees that are upper bounded by d_{BCG}. Our simulation results over a range of network sizes 500 and 2300 nodes showed that our node ID assignment method requires a small radio range $45m$ to produce a connected network with a small diameter and a small average path length.

In the future, we plan to expand our investigation to more realistic network environments involving topology control and routing protocol. we will also investigate the effects of interference on BCG-based networks.

References

[1] R. Rajaraman, "Topology control and routing in ad hoc networks: a survey," SIGACT News 33, 2, pp. 6073, June 2002.

[2] R. Ramanathan, and R. Rosales-Hain, "Topology control of multihop wireless networks using transmit power adjustment," Proceedings IEEE INFOCOM 2000, Tel Aviv, Israel, pp. 404-413 vol.2, March 2000.

[3] P. Santi, "Topology control in wireless ad hoc and sensor networks," ACM Computing Surveys, vol. 37, no. 2, pp. 164-194, 2005

[4] Li Li, J. Y. Halpern, P. Bahl, Yi-Min Wang, and R. Wattenhofer, "A cone-based distributed topology-control algorithm for wireless multi-hop networks," in IEEE/ACM Transactions on Networking, vol. 13, no. 1, pp. 147-159, Feb. 2005

[5] S. Narayanaswamy , V. Kawadia , R. S. Sreenivas , and P. R. Kumar, "Power Control in Ad-Hoc Networks: Theory, Architecture, Algorithm and Implementation of the COMPOW Protocol," in Proc. of European Wireless 2002, Next Generation Wireless Networks: Technologies, Protocols, Services and Applications, Florence, Italy, pp. 156-162, February 2002.

[6] V. Rodoplu, and T. H. Meng, "Minimum energy mobile wireless networks," IEEE J. Selected Areas in Communications, vol. 17, no. 8, pp. 1333-1344, August 1999.

[7] K. N. Sivarajan, and R. Ramaswami, "Lightwave networks based on de Bruijn graphs," in IEEE/ACM Transactions on Networking, vol. 2, no. 1, pp. 70-79, Feb. 1994.

[8] SH. Lu, KC. Li, KC. Lai, and YC. Chung "A scalable P2P overlay based on arrangement graph with minimized overhead," Peer-to-Peer Netw. Appl. 7, pp. 497510, 2014.

[9] I. Stoica, R. Morris, D. Karger, M. Kaashoek, and H. Balakrishnan, "Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications," ACM SIGCOMM Computer Communication Review, vol. 31, no. 4, pp 149160, 2001.

[10] H. Shafiei, Z. Aghazadeh, A. Khonsari, and M. Ould-Khaoua, "PeerStar: An attractive alternative to existing peer-to-peer topologies," 2009 IEEE Symposium on Computers and Communications, pp. 128-134, 2009.

[11] A. Abu Taleb, J. Mathew, and D. K. Pradhan, "Fault diagnosis in multi layered De Bruijn based architectures for sensor networks," 2010 8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops), pp. 456-461, 2010.

[12] A. A. Al-Mamou, and H. Labiod, "ScatterPastry: An Overlay Routing Using a DHT over Wireless Sensor Networks," The 2007 International Conference on Intelligent Pervasive Computing (IPC 2007), pp. 274-279, 2007.

[13] K. Tang, and B. Arden, "Representations of Borel Cayley graphs," in The Fourth Symposium on the Frontiers of Massively Parallel Computation, McLean, VA, USA, pp. 194-201, 1992.

[14] B.W. Arden, and K.W. Tang, "Representations and routing for Cayley graphs," (computer networks), IEEE Trans. Commun. 39 (1991), pp. 15331537.

[15] D. M. Blough, M. Leoncini, G. Resta, and P. Santi, "The k-Neighbors Approach to Interference Bounded and Symmetric Topology Control in Ad Hoc Networks," in IEEE Transactions on Mobile Computing, vol. 5, no. 9, pp. 1267-1282, Sept. 2006.

[16] R. Wattenhofer, and A. Zollinger, XTC: a practical topology control algorithm for ad-hoc net-works, in 18th International Parallel and Distributed Processing Symposium, 2004. Proceedings, 2004, pp. 216223.

[17] N. Li, J. Hou, and L. Sha, Design and analysis of an mst-based topology control algorithm,IEEE Transactions on Wireless Communications, vol. 4, no. 3, pp. 11951206, 2005.

[18] Z. H. Mir, and Y. Ko, Adaptive neighbor-based topology control protocol for wireless multi-hopnetworks,EURASIP J. Wirel. Commun. Netw., vol. 2012, p. 97, 2012.

[19] S. Narayanaswamy, V. Kawadia, R. Sreenivas, and P. Kumar, Power control in ad-hoc networks:Theory, architecture, algorithm and implementation of the compow protocol,Proc. of theEuropean Wireless Conf., 01 2002.

[20] J.-C. Bermond, C. Delorme, and J.-J. Quisquater, "Strategies for interconnection networks: Some methods from graph theory," Journal of Parallel and Distributed Computing, Vol. 3, no. 4, 1986, pp. 433-449.

[21] Arden, and Hikyu Lee, "Analysis of Chordal Ring Network," in IEEE Transactions on Computers, vol. C-30, no. 4, pp. 291-295, April 1981.

[22] T. S. Rappaport, Wireless Communications, Principles and Practice, Prentice-Hall, 1996.

[23] R. Junghun, Y. Jaewook, N. Eric, and K. W. Tang, "Borel Cayley Graph-Based Topology Control for Consensus Protocol in Wireless Sensor Networks," ISRN Sensor Networks, vol. 2013, pp. 1-15, 2013.

ISBN: 978-1-61208-920-1