Optimizing VHF Repeater Coordination Using Cluster Analysis

2011 MCM Problem B

Evan Menchini, Will Frey, Patrick O'Neil

Virginia Tech

August 5, 2011

Outline

1. Introduction
2. The Models
   - The “Bender” Snake Model
   - The Branching Model
3. Model Analysis
   - Case Studies
   - Sensitivity Analysis
   - Conclusion

Problem Restatement

2011 MCM Problem B

Find the minimum number of radio repeaters required to support 1,000 simultaneous users. The users inhabit a flat circular area with a 40 mile radius and are permitted to broadcast between 145-148 MHz. The repeaters transmit frequency is 600 kHz above or below the received frequency and 54 different CTCSS tones are available.

Available Technology

Repeaters

Repeaters are stationary devices that pick up weak signals, amplify them, and retransmit them on a different frequency. To avoid inteference with the incoming signal, the repeater rebroadcasts the new signal 600 kHz above or below the received signal. To avoid repeaters interfering with one another the Metropolitan Coordination Association states that repeaters must be at least 10 miles apart.

Available Technology

Repeaters

Repeaters are stationary devices that pick up weak signals, amplify them, and retransmit them on a different frequency. To avoid inteference with the incoming signal, the repeater rebroadcasts the new signal 600 kHz above or below the received signal. To avoid repeaters interfering with one another the Metropolitan Coordination Association states that repeaters must be at least 10 miles apart.

Line-of-Sight Transmission

Note that the range of a repeater is directly correlated to its height. The line-of-sight calculation to determine the effective distance is given by $d = 1.5\sqrt{A_f}$ where d is the distance in miles and A_f is the height of the repeater in feet.

Available Technology

Continuous Tone-Coded Squelch System

Continuous Tone-Coded Squelch Systems (CTCSS) mitigates interference by associating a subaudible tone with signals being received/transmitted by the repeater. In order to communicate through a private line repeater, users must also broadcast this tone.

Assumptions and Justifications

- Geometry is euclidean

Assumptions and Justifications

- Geometry is euclidean
- The system is closed

Assumptions and Justifications

- Geometry is euclidean
- The system is closed
- Antennas are isotropic

Assumptions and Justifications

- Geometry is euclidean
- The system is closed
- Antennas are isotropic
- Each user is a “low-power” user

Assumptions and Justifications

- Geometry is euclidean
- The system is closed
- Antennas are isotropic
- Each user is a “low-power” user
- Each user “plays nice”

Assumptions and Justifications

- Geometry is euclidean
- The system is closed
- Antennas are isotropic
- Each user is a “low-power” user
- Each user “plays nice”
- There are more than 1,000 potentials users

Assumptions and Justifications

- Geometry is euclidean
- The system is closed
- Antennas are isotropic
- Each user is a “low-power” user
- Each user “plays nice”
- There are more than 1,000 potentials users
- The geographic distribution of users is known

Assumptions and Justifications

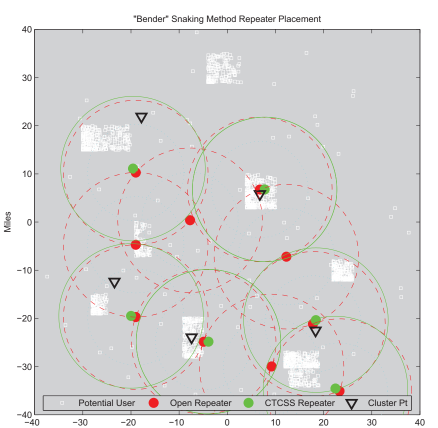
- Geometry is euclidean
- The system is closed
- Antennas are isotropic
- Each user is a “low-power” user
- Each user “plays nice”
- There are more than 1,000 potentials users
- The geographic distribution of users is known
- Users and repeaters are distinct entities

Assumptions and Justifications

- Geometry is euclidean
- The system is closed
- Antennas are isotropic
- Each user is a “low-power” user
- Each user “plays nice”
- There are more than 1,000 potentials users
- The geographic distribution of users is known
- Users and repeaters are distinct entities
- VHF signals are not affected by physical entities in the area

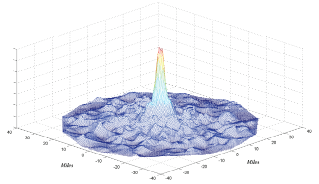
k-means Clustering

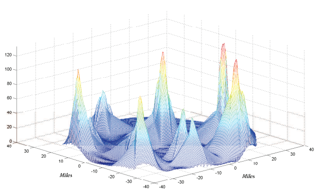
k-means clustering is a method of cluster analysis which aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean. The algorithm used for k-means clustering is readily available online and in literature. For this project we utilized Matlab's built in k-means cluster analysis.



Population Distributions

We developed two likely population distributions to test the models on: a city/suburbs distribution and a rural/small-town distribution.



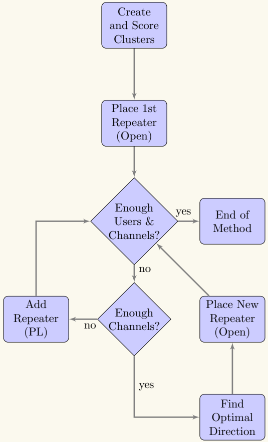


Outline

1. Introduction
2. The Models
   - The "Bender" Snake Model
   - The Branching Model
3. Model Analysis
   - Case Studies
   - Sensitivity Analysis
   - Conclusion

The "Bender" Snake Model

The "Bender" Snake Model seeks to maximize the number of connected users by efficiently creating a snake-like chain of open repeaters across the given area.



The “Bender” Snake Model

We first place a repeater so that the greatest number of users are covered. We then create another repeater along the perimeter of the newly established network area. Each new repeater is created in the direction of a cluster point so that **the model essentially snakes from one cluster point to another**. A problem arises as there are only a small number of available frequency bands on the open repeaters. To solve this problem, **we place CTCSS lines whenever we have a deficiency in the number of available bands.**

The “Bender” Snake Model

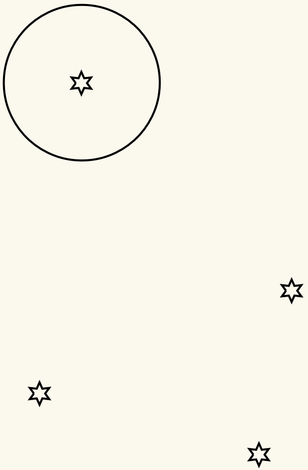
We first place a repeater so that the greatest number of users are covered. We then create another repeater along the perimeter of the newly established network area. Each new repeater is created in the direction of a cluster point so that **the model essentially snakes from one cluster point to another**. A problem arises as there are only a small number of available frequency bands on the open repeaters. To solve this problem, **we place CTCSS lines whenever we have a deficiency in the number of available bands.** By design, network growth tends toward establishing connectivity near and between cluster points. As more users around each cluster are accomodated, the score associated with their respective cluster should reflect that change. To account for this, cluster scores may be reevaluated to encourage intelligent networking in the model. This is known as **reclustering**.

The “Bender” Snake Model

How does the model work?

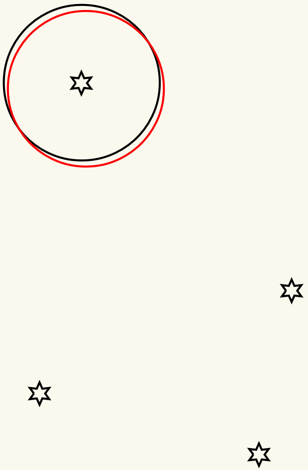
The "Bender" Snake Model

How does the model work?



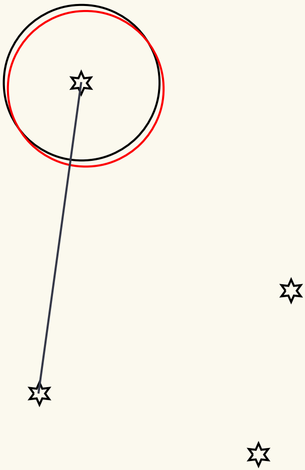
The "Bender" Snake Model

How does the model work?



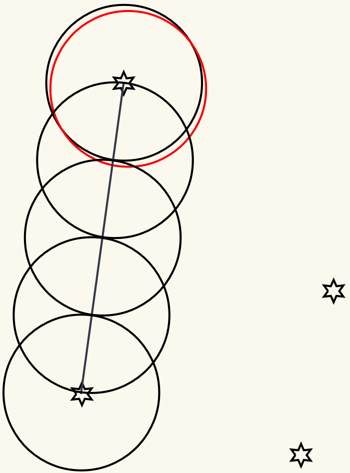
The “Bender” Snake Model

How does the model work?



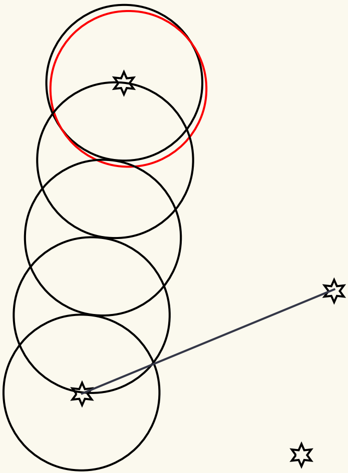
The "Bender" Snake Model

How does the model work?



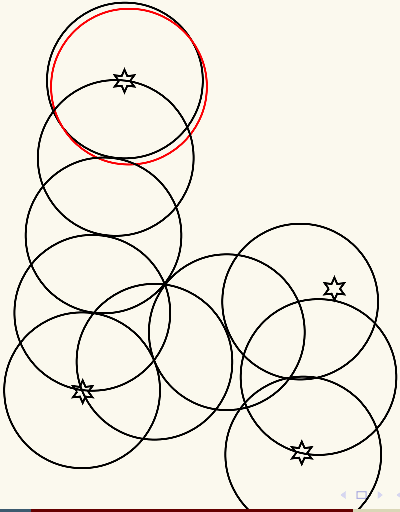
The “Bender” Snake Model

How does the model work?



The “Bender” Snake Model

How does the model work?



The “Bender” Snake Model

One advantage of our models is that they are both highly versatile. The “Bender” Snake model uses the following parameters.

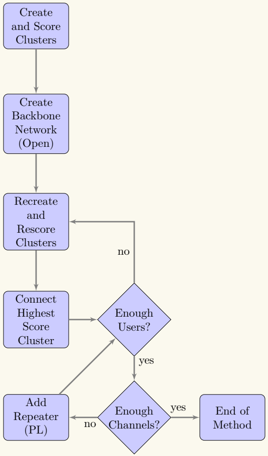
Parameter	Description
n	Number of users within 40 miles
k	Number of k-means cluster points
d_s	Maximum distance for user-to-repeater communication
h_r	Height of repeater towers
d_h	Repeater output distance
Δf	Frequency separation / channel width

Outline

1. Introduction
2. The Models
   - The “Bender” Snake Model
   - The Branching Model
3. Model Analysis
   - Case Studies
   - Sensitivity Analysis
   - Conclusion

The Branching Model

The Branching Model creates a backbone network of open repeaters that supports a number of branch connections.



The Branching Model

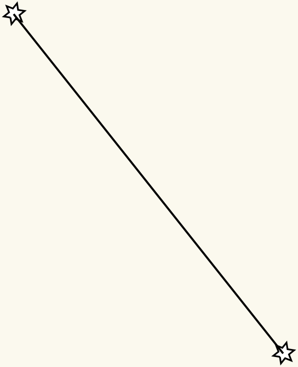
The Branching Method uses k-means cluster analysis and scoring to determine the optimal placement of repeaters. The model **creates a backbone network of open repeaters between the two highest scoring clusters**. After the backbone has been established, the model reclusters and rescores the remaining users and creates a branch of open repeaters between the existing network and the highest scoring cluster. After the entire network has been established, the model places CTCSS repeaters to ensure channel availability is not a concern in user-dense locations so that all users may be supported simultaneously. **This model requires reclustering after every iteration.**

The Branching Model

How does the model work?

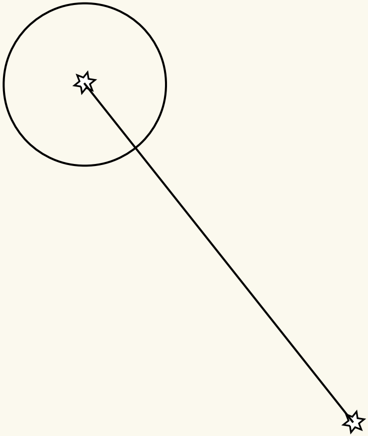
The Branching Model

How does the model work?



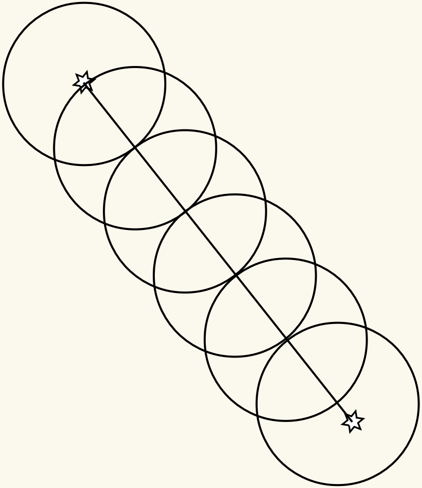
The Branching Model

How does the model work?



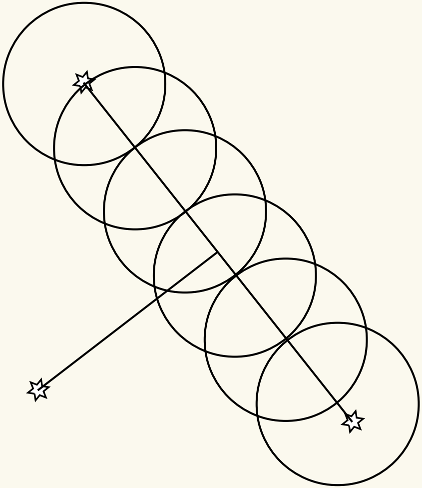
The Branching Model

How does the model work?



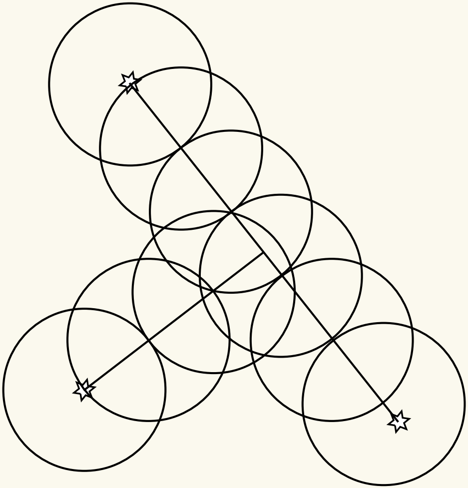
The Branching Model

How does the model work?



The Branching Model

How does the model work?



The Branching Model

The Branching Model supports even more customization than the first model. The parameters relevant to this model are listed below.

Parameter	Description
n	Number of users within 40 miles
k	Number of k-means cluster points
d_s	Maximum distance for user-to-repeater communication
h_r	Height of repeater towers
d_h	Repeater output distance
Δf	Frequency separation / channel width
l_n	Number of Long Distance Lines
l_c	Number of Locations in Long Distance Connections

Model Comparison Summary

Similarities

- *k*-means clustering is prevalant in both models.
- Variable-strength repeaters may be employed in both models.
- The change of frequency from a repeater ($\pm$ 600 kHz) is resolved last in both models.

Model Comparison Summary

Similarities

- k-means clustering is prevalant in both models.
- Variable-strength repeaters may be employed in both models.
- The change of frequency from a repeater ($\pm$ 600 kHz) is resolved last in both models.

Differences

- The models generate the network differently.
- Reclustering is required for the branching model.
- The method in which private lines are introduced differs between the models.

Outline

1. Introduction
2. The Models
   - The “Bender” Snake Model
   - The Branching Model
3. Model Analysis
   - Case Studies
   - Sensitivity Analysis
   - Conclusion

Case Studies

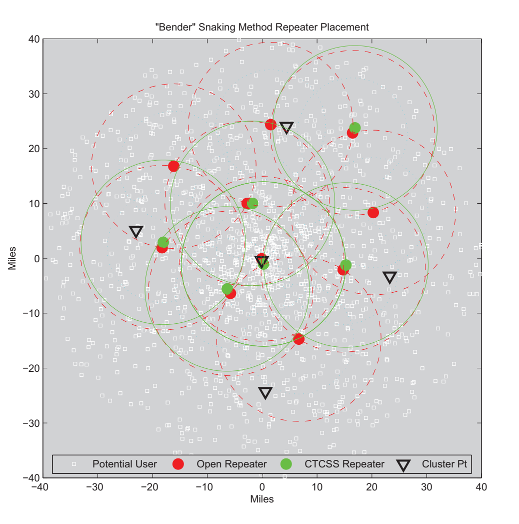
- When we discuss "parity" we refer to the $\pm$600 kHz difference in the recieving and broadcast frequencies of the repeaters. The graphs show these assignments by representing each open line repeater as a node, each labeled with "$+$" or "$-$" accordingly.
- The parameters for the case studies were set to values we deemed reasonable based on our research from our referenced sources.

Snaking Model: City Distribution

The model places the first open repeater slightly north of the city, creating a large deficit. This requires the placement of two CTCSS repeaters on this iteration to compensate. The process spirals counterclockwise around the city, placing 9 open line and 8 CTCSS repeaters for a **total of 17 repeaters**.This creates a closed loop of 9 repeaters, so the two connected “step-up” repeaters cannot necessarily communicate directly, however the signal will travel in the opposite direction around the loop and be recieved.

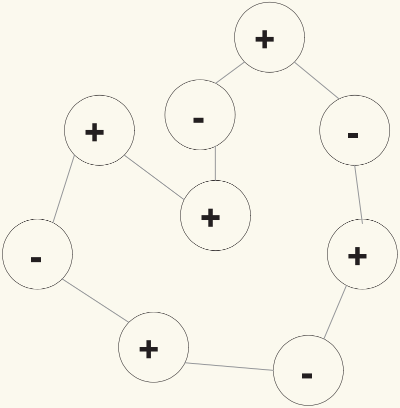
Snaking Model: City Distribution

Below are the results of running the snaking model on the city distribution.



Snaking Model: City Distribution

Repeater Pairity: There will be some signal leakage when the two step-up repeaters communicate directly but the signal will travel in the opposite direction around the entire network and be recieved.

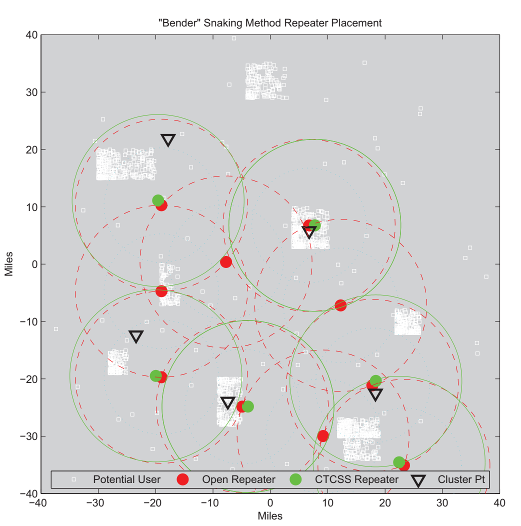


Snaking Model: Rural Distribution

The model starts near the northwestern-most town and works its way south and then east before heading north again as it gravitates towards the cluster points. Again, the model places CTCSS repeaters on each iteration as necessary. The model places 10 open and 8 CTCSS repeaters for a **total of 18 repeaters**. Again we have two connected step-up repeaters, but they are both connected to a step-down repeater, allowing for full connectivity.

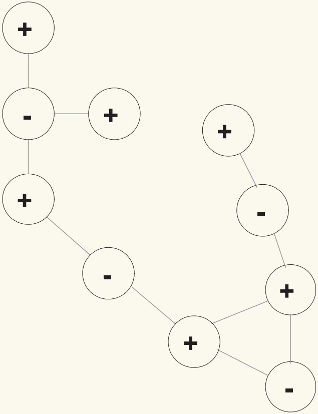
Snaking Model: Rural Distribution

Below are the results of running the snaking model on the rural distribution.



Snaking Model: Rural Distribution

Repeater Partiy: Note the southeastern node group where there are two step up repeaters connected. While this will result in some signal leakage outside of the available spectrum, the path that involves the step down node allows these two step up nodes to communicate without signal loss.

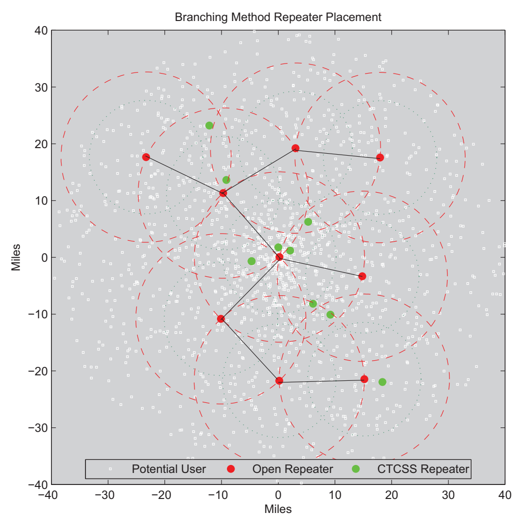


Branching Model: City Distribution

The model places the first open repeater in the city and creates three main branches to cover the surrounding suburbia, placing 8 repeaters. This structure is created first, and CTCSS lines are placed later. (The network structure has been highlighed with the black lines). Parity assignment is trivial.

Branching Model: City Distribution

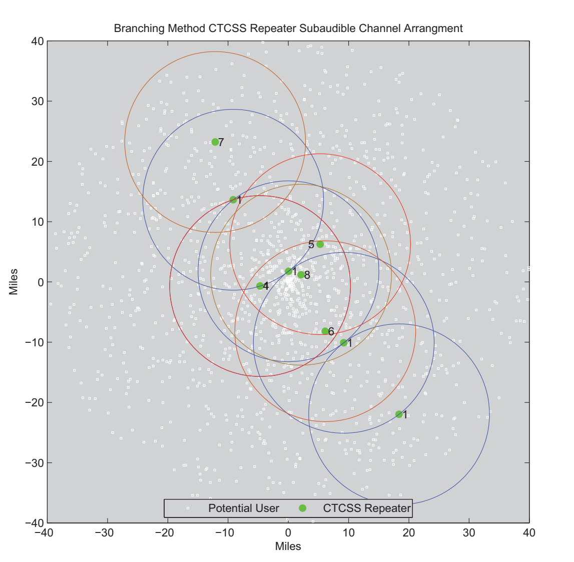
Below are the results of running the snaking model on the rural distribution.



Branching Model: City Distribution

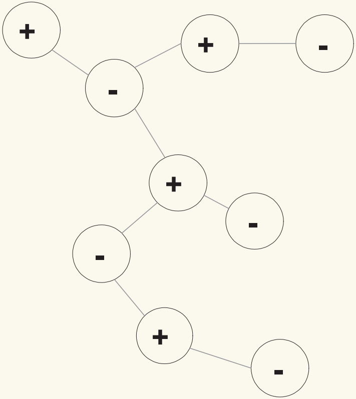
CTCSS repeaters are utilized to provide better overall connectivity than in the Snaking model. The blue circles form a separate network of CTCSS repeaters of a three tones, a “long distance line” (each blue circle is actually three repeaters at the same location). The other repeaters provide local lines in a manner similar to the Snaking model. This places 17 CTCSS repeaters; with the 9 open repeaters this is a **total of 26**.

Branching Model: City Distribution



Branching Model: City Distribution

Repeater Pairity: The parity assignment is quite simple here.

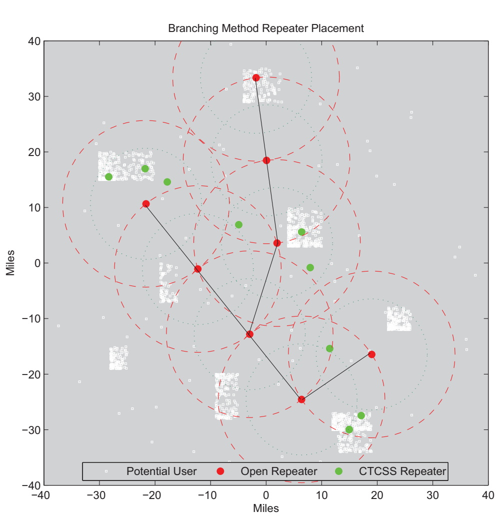


Branching Model: Rural Distribution

The branching structure is quite apparent here. The model connects the two main towns with the four-long straight spline, and the other node structures branch off of this. This places 8 open repeaters. This network is designed to efficiently cover the area rather than simply rushing from one population center to the next linearly. Again, parity assignment is trivial.

Branching Model: Rural Distribution

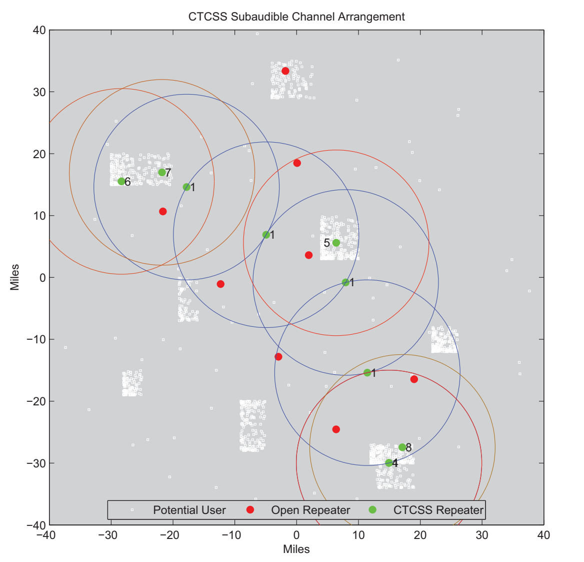
Below are the results of running the branching model on the rural distribution.



Branching Model: Rural Distribution

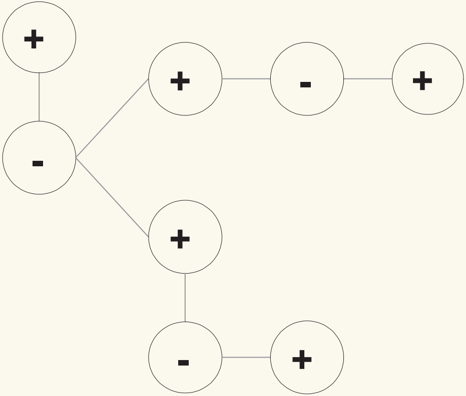
The CTCSS repeaters are placed afterwords. Again, the blue repeaters form the long distance line, and all others are local only. The model places 17 CTCSS repeaters; with the 8 open this is a **total of 25**.

Branching Model: Rural Distribution



Branching Model: Rural Distribution

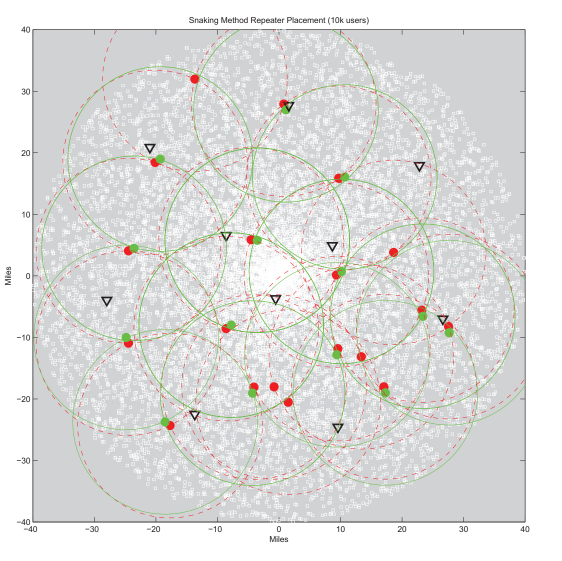
Repeater Pairity: Parity assignment for this model is fairly trivial.



10,000 Simultaneous Users

The prompt required that we stress test our model by attempting to accomodate 10,000 users. To accomplish this, we choose our frequency separation to be 10 kHz (this was 25 kHz in the earlier case studies) and run the Snaking model on a city distribution of 12,000. This places 19 open and 33 CTCSS repeaters for a **total of 42**. We conclude that the model is robust and can create an efficient network even for large populations.

10,000 Simultaneous Users



Mountainous Terrain

While VHF radio signals are blocked by large land features, the line-of-sight propogation method permits an increased range when the antenna height is increased. As a result, mountains could be used as an advantage by placing repeaters on top of them rather than around them (in which case the mountain would block the signal). However, **the effective range is proportional to the square root of the height**, providing diminishing returns, so this is not a trival fix.

In the case where there are many peaks spread out over a large area (a mountain range), the solution is to build the open-repeater network that snakes through the valleys formed by the mountains. However, this eliminates the line-of-sight advantage that the mountain could provide and thus would likely require a large number of repeaters.

Outline

1. Introduction

2. The Models
   - The “Bender” Snake Model
   - The Branching Model

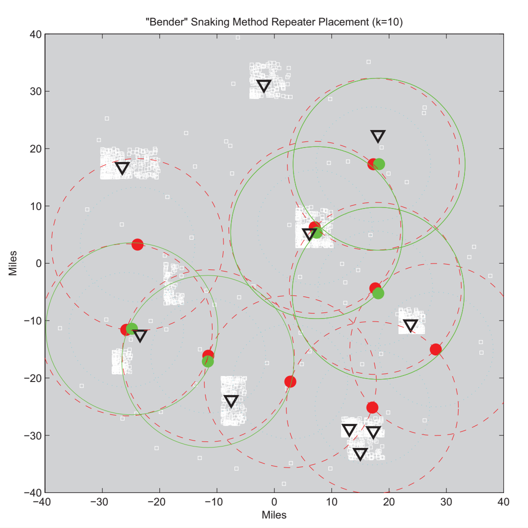
3. Model Analysis
   - Case Studies
   - Sensitivity Analysis
   - Conclusion

Sensitivity: Number of Cluster points

The model relies heavily on k-means clustering, so it is important to determine the sensitivity to the number of clusters chosen initially (k). In the earlier case studies, we used $k = 5$, and here we test the Snaking model on a Rural distribution using $k = 10$ and $k = 20$ as well. When $k = 5$ or $k = 10$, the model places nine open line repeaters; $k = 20$ places eight. In all cases, the model places eight CTCSS lines. We conclude that significant changes in the number of cluster points does not impact performance.

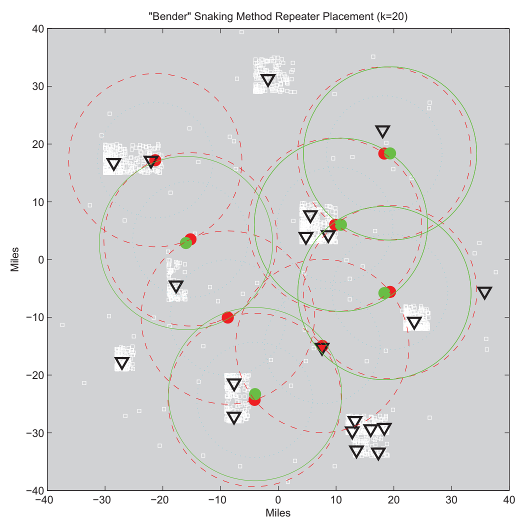
Sensitivity: Number of Cluster points

Number of Cluster Points = 10



Sensitivity: Number of Cluster points

Number of Cluster Points = 20

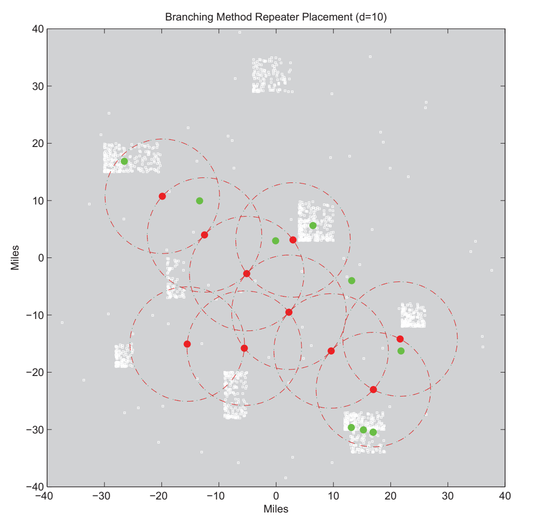


Sensitivity: Separation Distance

Varying tower height (and hence repeater separation distance) results in a significant change in model performance. As the height is increased, a repeater can cover a larger area, and thus fewer are required to create a network. The pitfall of this is the increased cost associated with a taller tower. The user must find a reasonable compromise.

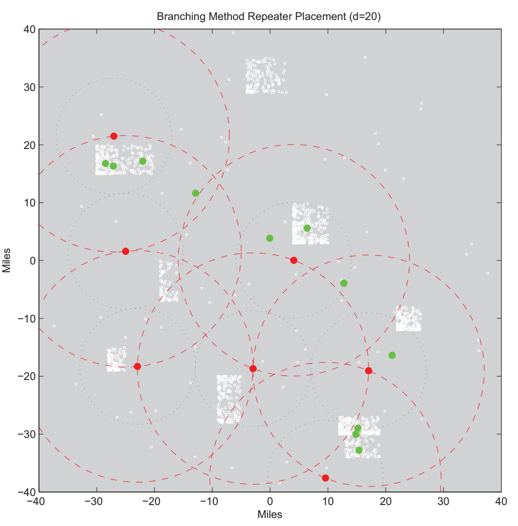
Sensitivity: Separation Distance

Separation Distance = 10mi



Sensitivity: Separation Distance

Separation Distance = 20mi

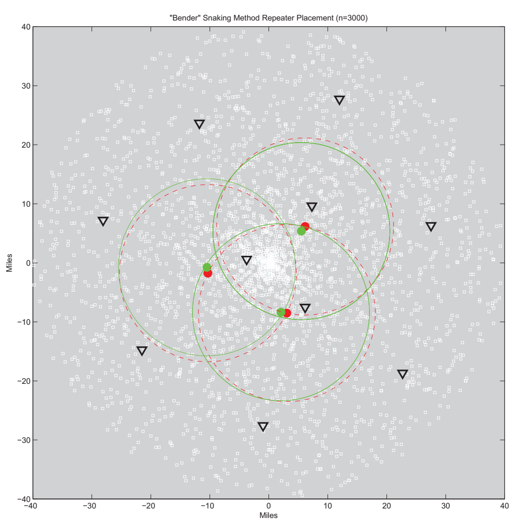


Sensitivity: Initial Population

Our algorithms run until the desired number of people are connected to the network and there are enough channels for those people. Changing the starting population on the map drastically impacts model performance. This is not suprising as a higher population allows the model to capture the desired number of people faster.

Sensitivity: Initial Population

Initial Population = 3000



Strengths

- **Versatility of the models.** Both models incorporate a variety of user determined parameters. We were very impressed that our model accomodates 10,000 users under the established requirements.
- **Smart Clustering.** Implementation of reclustering creates a smarter algorithm that targets the highest priority populations *at that moment*. This updating allows the model to make the “best” decision at every iteration.
- **Efficient Use of CTCSS Lines.** Both models, even with 10,000 users, do not exhaust available CTCSS tones. The unused tones could be used to accomodate more network traffic if it were desired.

Weaknesses

- **Large reliance on *k*-means cluster analysis.** Other clustering methods exist and the choice to use *k*-means exclusively does limit the effectiveness of our model. Future work would include the implementation of other clustering methods in an attempt to improve efficiency. In particular, we believe that Quality Threshold clustering (a method in which a distance threshhold, not the number of clusters, is set) could have improved model performance.
- **Difficulty with populations close to target.** We found that if only 1,000 users were present, the algorithm would circle around itself trying to hunt down the last few remaining users. We justified this with our assumption.

Outline

1 Introduction

2 The Models
- The "Bender" Snake Model
- The Branching Model

3 Model Analysis
- Case Studies
- Sensitivity Analysis
- Conclusion

Conclusion

- The minimum number of repeaters required to support 1,000 users under our assumptions is 17
- Better connectivity requires more repeaters
- 54 CTCSS lines are not necessary
- CTCSS lines have multiple applications