# Planning for Communication Resources

Brett Browning and Manuela Veloso
May 2003
CMU-CS-03-120

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA, 15213

This research was sponsored in part by the Defense Advanced Research Projects (DARPA) and the Air Force Research laboratory (AFRL) under agreement No. F30602-00-2-0549.

The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing official policies or endorsements, either expressed or implied, of DARPA or AFRL.

**Keywords:** Communication planning, resource allocation, frequency assignment

**Abstract**

For many human team activities, ranging from military operations through to emergency rescue or large entertainment events, communications resources must be assigned to different teams or team members. These assignments must reflect the capabilities of the available communication devices and avoid conflicting use of communications channels already in use in the local environment. In general, finding and assigning available communication channels for short-term use is a task performed manually by human operators. Operators, using generic tools, such as spreadsheets and database manipulation programs, access government databases to obtain information on frequency usage and then manually attempt to locate suitable unused channels. This process is time intensive, prone to error, and 'mechanistic' in nature. In this paper, we describe the CommPlanner, a new fully implemented system developed to automate this assignment procedure and thereby speed up and make more reliable the process. We describe the algorithms used by the CommPlanner, and the underlying issues that, while not always obvious, must be addressed in the processes of assigning frequency usage.

## 1 Introduction

In many human team tasks, assigning open and available communication channels to each agent in the team is critical to the team success. Traditionally, for both civilian and military applications, such channel assignment is performed by hand in a painstaking, laborious process. The task, however, is a mechanistic one that is ripe for automation. In this paper, we describe CommPlanner, a prototype software package for automating the assignment of free communication channels to teams operating in a known geographic area.

As technological progresses, finding free communication channels for a given device becomes increasingly complex. Usage of the available spectrum is becoming increasingly dense as new devices, which typically require less bandwidth than older devices, 'chop up' the available bandwidth. With the advent of spread spectrum devices, which are more immune to jamming but make more complex use of the spectrum, the situation becomes even more convoluted. Secondly, across a geographic area, channels can be reused or overlap, and can vary in power significantly. Typically, there exist government databases that store frequency usage, or allotment, information. Thus, finding and assigning free channels to a team of agents operating in an area is a process of accessing the databases to retrieve the relevant frequency usage, and determining what parts of the spectrum are available and reachable by the communication devices to be used.

Invariably, an expert operator using spreadsheet and database access tools performs this task manually. It is both a time intensive, laborious operation, and a mechanistic one that could conceivably be automated by specialized software. Indeed, in this work we describe the CommPlanner, a prototype software package developed to perform exactly this task. The CommPlanner makes only modest use of state-of-the-art planning technology, indeed the task is performed manually by humans in a mechanistic way that is procedural rather than inventive, however it fulfils a need that is currently unfulfilled by any commercial or Open Source package that we are aware of.

There are many potential advantages to be gained from automating the frequency assignment task. The assignment process, which normally takes on the order of days, can vastly hastened. Furthermore, as it is a laborious task performed by a computer rather than by a human, the potential for errors to creep into the process is significantly reduced. Graphical visualizations of frequency usage, and frequency usage over space, can impart the viewer with an easy understanding how the spectrum is being used. This knowledge can illuminate project decisions and policy decisions for governing bodies.

The potential advantages are more profound than this, however, particularly if the approach is taken to its natural conclusion by automating the entire communications resource allotment problem. To begin with, even very large projects involving hundreds or thousands of channels could be assigned automatically within minutes thereby greatly easing the planning requirements for such projects. For high-risk endeavors in dynamic environments, for example disaster rescue, communications resources can be replanned on the fly within the constraints of the already assigned resources as the controlling constraints evolve over time (e.g. more people are required, certain devices do not work in the conditions etc.). Within a project, or across multiple projects if the data is combined, the usage of frequencies and devices can be monitored and analyzed to aid in determining how devices are best used, and what devices will be required in the future. If centralized communication planners are used in a client/server format, then multiple organizations can conduct operations in the same area without any explicit coordination between the planning bodies of the organizations. Finally, in situations where a particular transmitter needs to be suppressed the request could be automatically generated and sent to the network operators. The work we describe here is addresses the fundamental task of frequency assignment, but promise for more exciting developments

This paper is structured in the following way. The following section describes the frequency assignment problem and also reviews some of our early work on graphical visualization techniques. We then discuss the CommPlanner, our prototype server software for assigning frequency bands using government furnished databases. We then discuss the results of the CommPlanner performance, and conclude with a discussion of the future directions we are pursuing to achieve the goal of complete communication planning.

## 2 The Problem

In this section, we describe specifically the problem of communication channel assignment. Although many scenarios are possible, we focus on the task of finding free channels meeting device specifications for a team of agents operating in a geographical area.

As mentioned in the introduction section, the ability to graphically visualize represents a very useful step towards understanding the frequency assignment problem, and how the frequency spectrum is used over a geographical area. In our early work, we developed a tool using OpenGL to perform just this task. Figure 1 shows the output of this visualization for some (around 200 transmitters) frequency usage in the Pittsburgh, Pennsylvania area.

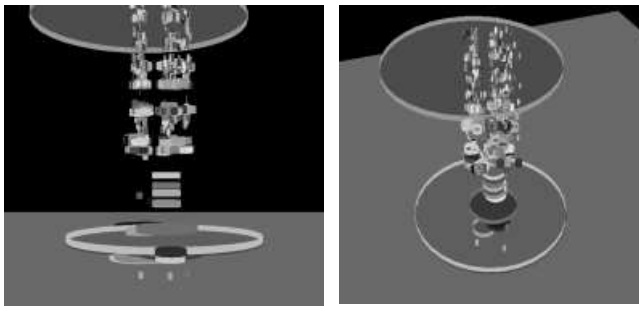

**Figure 1. Graphical visualization of the frequency usage over area. The *x*-*y* axes cover geographic space, while the *z* axis is the frequency dimension.**

Essentially the process works by plotting in 3D space the use of frequencies (the *z* axis) over a geographic area (the $x - y$ axes, respectively). The colors in this diagram are representative of different transmitter types. Note that this tool makes no account of line-of-sight limitations. Rather, it estimates the range of each transmitter from the power of that transmitter. As it runs within OpenGL, the user can 'fly' around the frequency space to examine it. Figure 2 shows an overlay of the transmitter location and range on a 2D map. In this case, color represents the transmitter frequency.

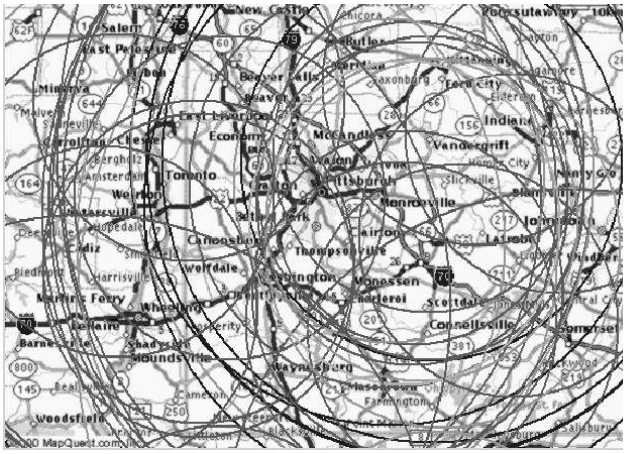


**Figure 2. A 2D plot over a map of the frequency usage.**

Although this example only shows part of the frequency usage, it is readily apparent that channels accumulate in bands and overlap in a haphazard fashion. The frequency assignment task translates to finding open gaps in this 3D frequency-area space within the area of interest and the device capabilities. In short, we are looking for a cube or cylinder of the frequency-area space that is unused. Thus, we must consider what transmitters are in the local vicinity, over what range and with what frequencies they operate. Within the area of interest and device capabilities we must then attempt to find unused portions and assign those channels to the users as required.

More concretely, for each type of device used we must assign a number of useable channels as requested by the user. Each device has a list of properties that determine what parts of the frequency space are relevant along with

usage rules for that device in different environments. In particular, we consider devices to have a limited effective transmission range, limited bandwidth, frequency centers, and a range of settable frequencies within the device bandwidth. To assign useable channels for the device, we must first determine what channels are freely available for use in the environment. In short, our program must consult an oracle of known transmitters that impact on the area of interest and then determine what channels are useable for the device in question.

Within the US, and in most countries around the world, permanent transmitters must be registered with the local government authorities. Thus, there are usually commercial and government databases for known transmitters. These databases provide a set of records encoding the transmitter geographic location (usually in latitude and longitude coordinates), its frequency usage, power, owner, and sometimes purpose (e.g. commercial, military, emergency and so on).

Each agent within the team needs to be able to communicate to various networks of agents. Typically, there will be a network for each agent (so that one-to-one communication is possible) along with a hierarchy of larger networks for wider broadcasts. The hierarchy of larger networks will often represent the hierarchical structure physically present within the team (ie. sub-teams) as well as several other requests, including in particular the need for additional wide broadcast channels for emergency announcements.

## 3 CommPlanner Implementation

In this section we describe our approach to the communication assignment problem. Specifically, we describe the details of the CommPlanner program that we have developed to perform automatic frequency and call sign assignment.

### 3.1 Overview

Given the resource management nature of the problem, our approach to the CommPlanner uses a client-server model. The CommPlanner program operates as a server. It manages database information, accepts and executes requests for new communication plans from clients, and sends the results back to the client in a standardized format. Figure 3 shows a block diagram of this approach.

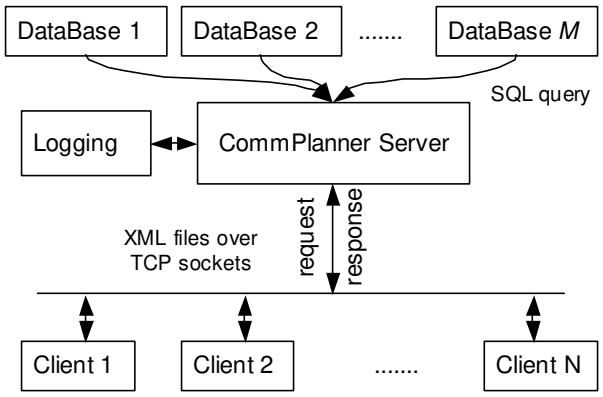


**Figure 3. The block diagram of the CommPlanner.**

With the client-server format, following the usual approach, the CommPlanner communicates to its clients via TCP sockets. Requests, and responses are both sent over the sockets as XML files where tokens in the file delimit the information for the request and the resulting frequency assignments that are returned.

The main loop of the CommPlanner server consists of waiting for clients and then receiving their XML file requests as they connect. In our current implementation, each request is processed independently, although this need not be the case and is a simple extension of our current work to allow multiple requests for the same or overlapping frequency-area space. Upon receiving each request, the CommPlanner requests information on transmitters that

operate in the local frequency-area space to the request area and device from its database resources. Additionally, it will log all transactions to log files for documentation and later analysis. Upon receiving the results from the database queries, the CommPlanner will generate the list of available frequencies of suitable bandwidth for each device in question. An allocation process follows this where bands are assigned using some algorithm. We have used both random assignment and bottom up assignment but other alternatives are possible. Upon assigning the bands, a call sign is attached to each band. An XML file encoding the results is then transmitted back to the client. If any errors occur during the process, then an error signal is sent back instead.

In the following paragraphs we describe the details of each portion of the CommPlanner system.

## 3.2 File Formats

There are three XML file formats used for the CommPlanner: request files, result files, and the configuration file. Request files encode a request for frequency usage through a series of rectangular geographic areas with a list of devices, and number of channels per device, to be used in that area. Table 1 shows an example portion of a file. Here the area is delimited by two latitude and longitude coordinates that create a bounding box (i.e. upper left corner and lower right corner). The list of devices must be known by the server, which obtains this information through its configuration process.

```
<?xml version="1.0"?>
<inputData>
   <area>
      <upperLeft> 39.800 -78.8185 </upperLeft>
      <lowerRight> 40.000 -80.000 </lowerRight>
      <device>
         <name> UHF handheld </name>
         <numOfBands> 50 </numOfBands>
      </device>
      <device>….
   </area>….
```

**Table 1. An example request XML file. (Users may specify graphical regions that are translated into XML or input XML files are provided by user applications.)**

In the current formulation, the frequency-area space for each device-area request must be disjoint and non-overlapping. This is because each device-area request is performed independently of the others. In practice this is the approach followed by human planners.

The output format is somewhat similar. Each assigned band is given a unique integer identifier, a call sign, and the assigned frequency. The assigned bands are grouped into the device-area that they resulted from as shown in Table 2.

```
<outputData>
   <area query="0">
      <device name=”UHF handheld”>
         <net id="0" callsign="Charlie">137.295 M</net>
         <net id="1" callsign="Omega">126.215 M</net>
         …
      </device>…
```

**Table 2. The output file. (M denotes MHz.)**

Likewise configuration information is specified via an XML format. The configuration file contains database locations, the file location for the callsign list, default values for unspecified fields such as transmitter power, debugging configuration options, as well as the listing of devices and their parameters.

The device list defines the properties of each assignable device. The current implementation includes the unique device name used to recognize the device, its operating frequency range (device bandwidth), the frequency range for each available channel on the device (channel bandwidth), and the frequency centers the channels operate on (channel centers). Most devices have a number of pre-assigned channels on set frequency divisions rather than being capable of arbitrary frequency within the device frequency range. Thus, an operating frequency for the device is $f$ *modulo* $f_c$, where $f_c$ is the frequency center division. The number of channels for the device can of course be reconstructed as: $N = f_{device} / f_c$.

This current work focuses primarily on narrow spectrum transceivers and does not specifically address the issue of spread spectrum transceivers. In such a situation, the level of interference will vary depending upon the method of spread spectrum communication – direct sequence, frequency hopping, chirp, or time hopping. Future work will address this issue to provide probability of jams, which will depend upon the transmission technique for the device and the known transceivers in the area.

## 3.3 The Assignment Process

For each request a number of assumptions are made. As described above, we assume that different devices-area is disjoint meaning the device operates in a unique portion of the frequency-area space without overlaps. Thus, devices used in separate device-areas are essentially non-conflicting. We also ignore line of sight considerations. Following the usual procedure, we assume each device is capable of operating over the area it is assigned to. In future work, we will remove this assumption and generate the operational range of the device using line-of-sight calculations.

As a consequence of these assumptions, each area and device request can be processed separately without consideration for the whole. Hence, these assumptions, although apparently limiting, greatly simplify the processing steps and in practice are not a great hindrance. Indeed, frequency planning which is done by hand has evolved in such as way to make these assumptions viable.

Table 3 shows psuedo code for the assignment process. The key to doing the frequency assignment is to determine what frequencies are open and available in the selected portion of frequency-area space and then using some assignment policy to select frequencies. To obtain this information, we make use of a frequency database (or databases). These databases are available for commercial enterprises as well as for military ones. They essentially contain a multitude of information regarding frequency usage (ie. transmitter/receiver geographic location, power/sensitivity, bandwidth, frequency centers etc.). The task for the CommPlanner is then to query these database(s) to retrieve the records relevant to the device and area in question and then to determine what parts of the spectrum are then free for use.

```
For each area
   For each device
      Generate SQL query
      Retrieve records from databases
      Clear open band list (O = {0})
      For each retrieved record F
         Expand F by device bandwidth
         If overlapping ( O ⋂ {F} ≠ {0} )
            Remove F from list
      End
      Generate Assignments (O)
   End
End
```

**Table 3. The open band generation process.**

To access a database the CommPlanner follows the usual Microsoft Windows approach and uses the data access objects (DAO) library. The DAO library provides a simplified interface to access and query a range of different database formats (DBase, ODBC, etc.). Using the Microsoft Foundation Classes (MFC) wrappers that encapsulate the DAO access methods, we can generate an SQL query to retrieve the relevant records and process them accordingly. There is a catch that needs to be accounted for, however. Although the essential content of the query is independent of the database in question, the specifics of the query are not. In particular, different databases, while containing essentially the same information (or some subset thereof), may make use of different tables as well as different table and column names. To account for this, the major components of the SQL query, which remain constant, are stored in the configuration XML file and it is only the query values (the latitude/longitude coordinates etc) that are modified.

Using the DAO MFC class, the database is opened and queried to retrieve the records relevant to the area and frequency of interest. The area is derived directly from the area specification in the request. The frequency of interest is derived from the device bandwidth, which is widened by a default value to account for boundary cases. This default value, which increases the bandwidth at both limits, is also stored in the configuration file.

Upon retrieving the records of interest the core of the assignment algorithm comes into play. The key idea is to identify which frequencies are free and then to make the channel assignments from within these "open bands". To identify the open bands, we first assume that the entire frequency range of the device is available for use. As we iterate on the retrieved records, we remove the portions of the open band(s) that overlap with retrieved record. At the end of this process, we have a list of ranges that are viable.

Concretely, an open band is a frequency range

$$o_k = [f_{low}, f_{high}]$$

Thus, the set of non-overlapping open bands $O$ is

$$O = \{o_1, o_2 ... o_N\} \text{ where } o_i \cap o_k = \{0\} \quad \forall i,k \in 1..N$$

The set of open bands encodes the partitioned range of available frequencies. Graphically, $O$ corresponds to a series of non-overlapping ranges as shown in Figure 4, which are vertical slices of frequency-area space.

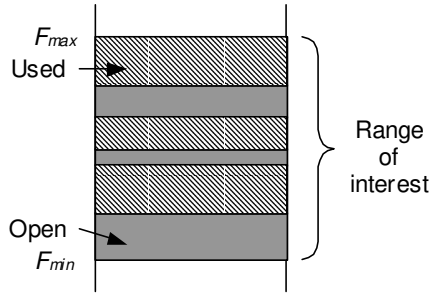


**Figure 4. Graphic visualization of open bands**

Each retrieved record encodes a used frequency range as:

$$F_j = [f - \tfrac{1}{2} f_{bandwidth}, f + \tfrac{1}{2} f_{bandwidth}]$$

where $f$ and $f_{bandwidth}$ are the frequency center and bandwidth, respectively. In some cases the bandwidth may not be defined. In such situations a nominal default value is used instead, which is obtained from the configuration file. We first expand the range of $F_j$ to account for the device channel bandwidth, thereby saving a later check. Thus:

$$F'_j = [f - \tfrac{1}{2}(f_{bandwidth} + f_{channel}), f + \tfrac{1}{2}(f_{bandwidth} + f_{channel})]$$

Now, the set of open bands is modified to remove any ranges where $F'_j$ occurs. The first step is to identify which bands, if any, overlap with $F'_j$. If any open bands are identified, then these must be sub-divided or removed so that they no longer overlap with $F'_j$. To test for overlapping bands, we check for an overlapping range of each open

band in $O$ and $F'_j$, ie. we check whether $O_k \cap F'_j \neq \{0\}$ for an overlap. If an overlap exists, four situations arise corresponding to the amount of overlap. $F'_j$ may be a subset of $O_k$, or vice versa, or there may be partial overlap in either direction meaning $O_k$ must be shortened from the bottom or the top. Figure 5 shows an example of the split situation where $F'_j$ is a subset of $o_k$ producing two sub-ranges as a result.

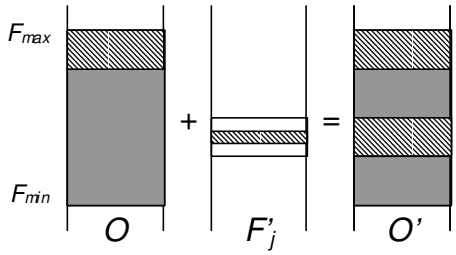


**Figure 5. An example of the split situation.**

To speed up this operation we store $O$ as a prioritized list keyed on the lower frequency bound (given that the open bands are guaranteed non-overlapping either frequency bound is appropriate). Although a tree structure (or perhaps a KD-Tree) would provide a faster mechanism for storing the open bands, the list proves sufficiently fast for most considerations and hence the extra overhead was not warranted. Typically, for a city sized area and a normal device, only a few thousand transmitters need to be considered. The open bands can be maintained as a sorted list during their construction quite easily. Assuming the list is already ordered by frequency (either range can be used as the ranges are non-overlapping by construction), the only addition operation results from when the used frequency is a subset of an open band (see Figure 5). All other cases result in the removal of an open band, or modification of its upper or lower bound. For the subset case, the open band $o_k$ must be split into two sub-bands. As they are strict sub-sets of the original open band they occupy the same place in the ordering. Thus, the two new open bands replace the original parent $o_k$. The ordering and non-overlapping constraints are therefore maintained and the process can continue.

At the end of this process, all open bands of sufficient width within the frequency-area space of the device are identified. The resulting open bands may consist of multiple channels and are not in general aligned on the frequency centers of the device. The latter is taken care of during the assignment stage. Channel assignment can be performed in a number of ways: linearly starting from the top or bottom of the device capabilities, by largest (or smallest) free area, or randomly. All methods share the same approach:

1. Choose an open band
2. Choose a frequency center from within that band
3. If there is sufficient bandwidth when aligned on the frequency centers then assign this band

The band assignment itself merely requires creating a new assigned band $B_k$ and adding it to the list of assigned bands $B$. In addition to determining the frequency center of the band, we need to assign it an identifier and also a call sign. The CommPlanner performs call sign assignment randomly using a list of useable call signs. Alternative methods, based on generation rules of some kind, are possible but are not the focus of the current work. Thus each new band is:

$$B_k = (id, f, callsign) \text{ with } B = \bigcup_k B_k$$

Once generated it is an easy process to generate the XML output, which is just an enumeration of the set $B$ for each area/device token. If successful, the results are sent back to the client via the TCP socket.

## 4 Results and Discussion

The CommPlanner performs its required task as designed as will shortly enter in the field user testing. In its current form it is able to satisfy requests of say a few hundred channels for a city-sized area in a matter of seconds (the actual response time varies depending upon the machine and version of Microsoft Windows). Additionally, the

CommPlanner maintains access logs of all its activities for monitoring purposes. Figure 6 shows the debugging output of an example planning process for 150 channels for a UHF device in the greater Pittsburgh area. Note the frequency output here is plotted on a logarithmic scale.

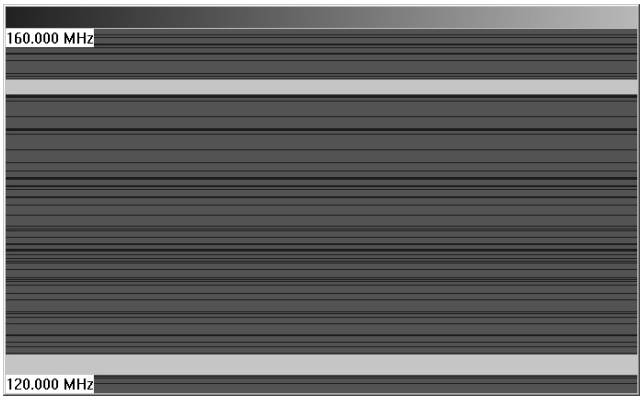


**Figure 6. Assignment results for greater Pittsburgh area for a UHF device with 150 channels. Darker bands are assigned, lighter bands are used, light gray is unused. Note logarithmic scale for frequency.**

It is readily apparent that the technology used here is systematic, and yet, to our knowledge, this is the first type of application of this kind. To date, all communication planning is performed manually, if mechanistically, using simple tools such as spread sheets and manual database accesses. Given the amount of data handled, it is a time consuming and error prone task. In many cases, communication planning is a task that requires specialized training (although it should be noted that this training covers more than the specific task described here). Thus, the entire communication planning process, which is a critical part of many team activities, hinges on the operations and knowledge of a few individuals. Clearly, automating this process is of paramount importance to reduce the chance for errors, speed up the allocation process, and also provide a mechanism for wider distribution.

As mentioned above, the CommPlanner is about to enter user testing and should hopefully be in practical operation in the near future. Future enhancements to the CommPlanner include adding security measures to ensure only valid clients can request frequencies, adding line-of-sight considerations to boost the area based approach, and allowing for multiple requests in overlapping portions of frequency-area space.

## 5 Conclusions

In conclusion, this paper has described the CommPlanner program, a tool for automating the frequency assignment process for teams operating in areas with known static frequency usage. To date, this task is performed manually in a tedious and error prone process by operators with specialized training. The process used by these operators is mechanistic, and therefore ripe for automation. We have presented our working prototype solution to this problem, its details and inherent problems and limitations therein. The CommPlanner, which is about to enter user testing, represents a first step towards the ultimate goal of completely automating communication planning for human teams performing mission-like tasks.

## Acknowledgements

Thank you to James Bruce, who helped in an initial phase, in particular generating the visualization graphs illustrated in Figure 1. Thank you also to Gal Kaminka for helping with a clear definition of the problem through several interactions with the real users. Finally, we would like to thank our funding sources, who supported and guided this research and development to address this specific problem.