

DEFENSE THREAT REDUCTION AGENCY



Making the World Safer

Optimizing the Placement of Chemical and Biological Agent Sensors

Daniel L. Schafer

Thomas Jefferson High School for Science and Technology

Defense Threat Reduction Agency TDOA OR

Mentor: George D. Gunn, PhD

Abstract

- A computer program was designed and implemented in Java to optimize the placement of chemical and biological (CB) agent sensors to best protect an installation. This was accomplished through the use of various optimization algorithms, combined with a Monte Carlo simulation method which determined relative utility functions for various sensor arrangements.

Introduction

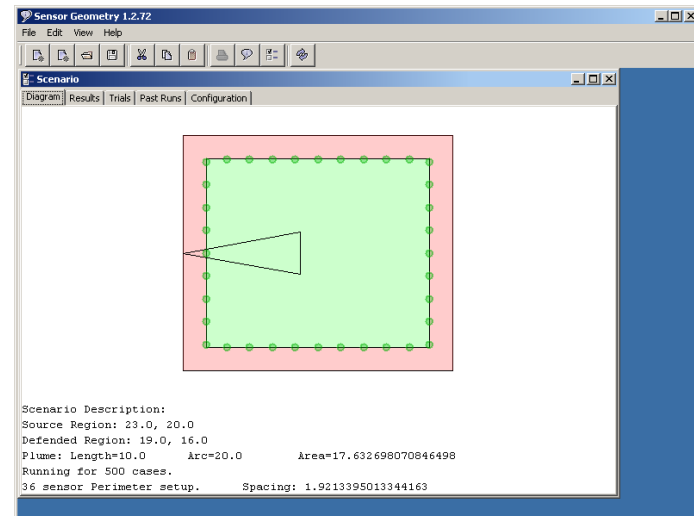
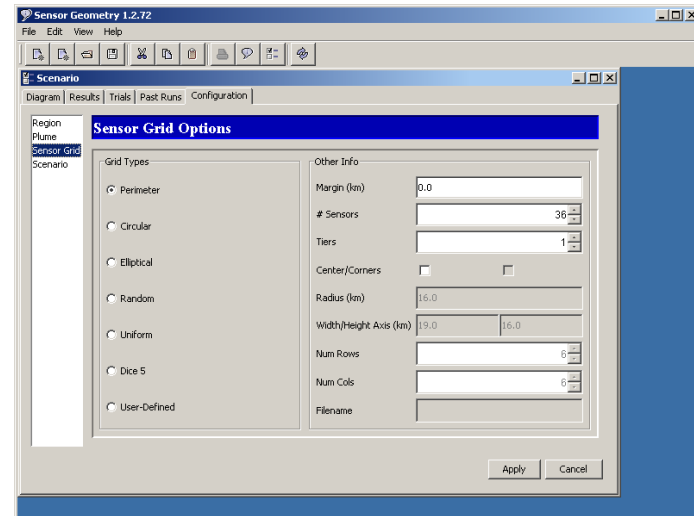
- WME – Major Threat
 - CB Weapon Detection
- Software Available
 - Difficult for optimization use
- New software needed

Thursday, August 11, 2005



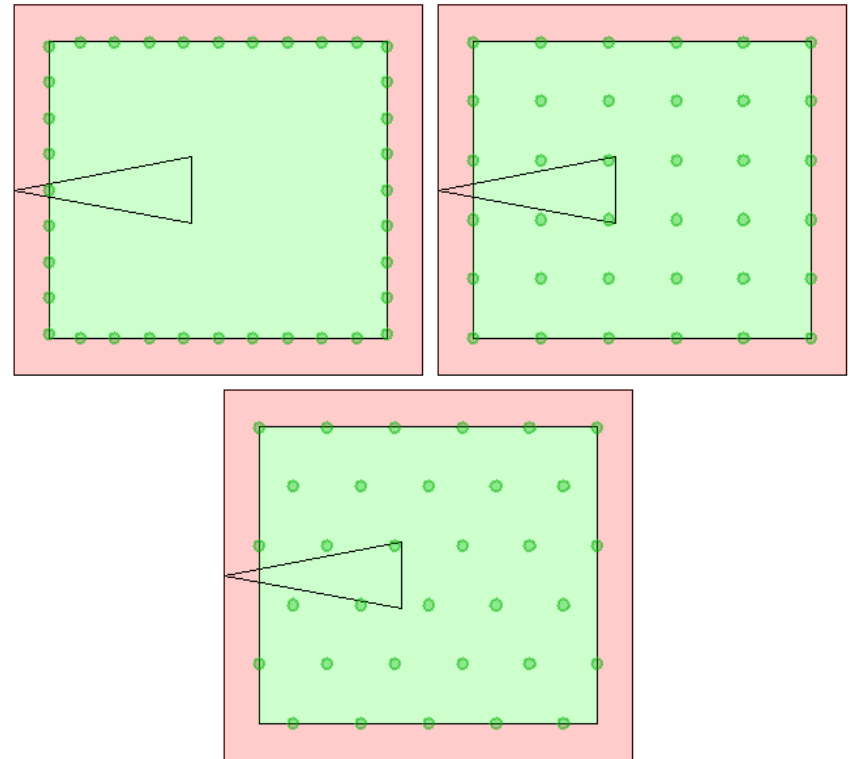
Equipment

- Java programming language
 - Support for all major Operating Systems
 - Potential web client extension
 - Native appearance



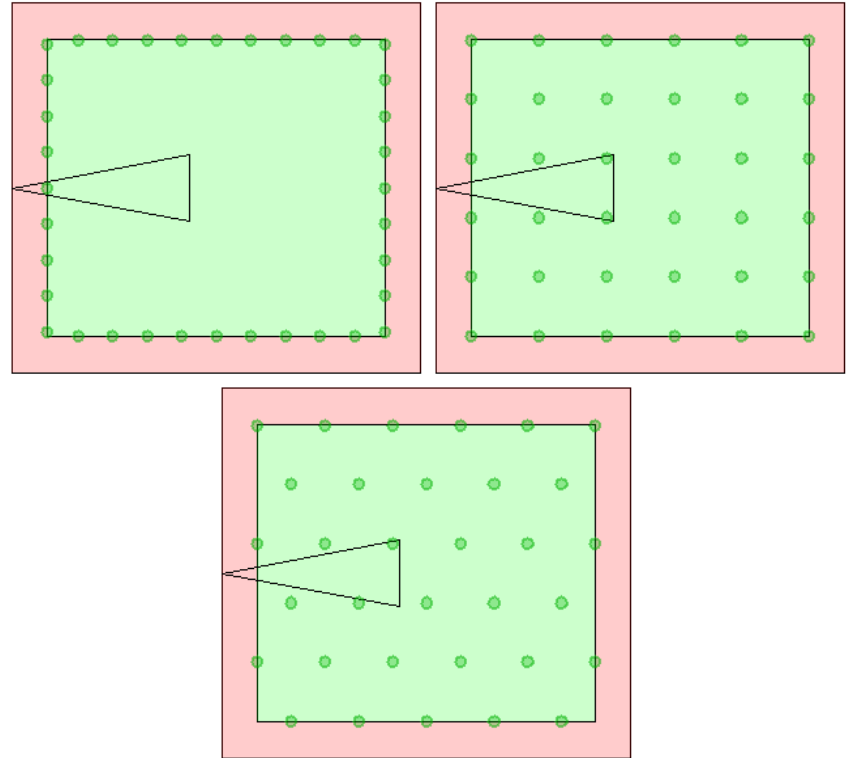
Procedures

- Base configuration
- Defended Area
 - Green
- Safe Area
 - Blue
- Plume Spawn Region
 - Green



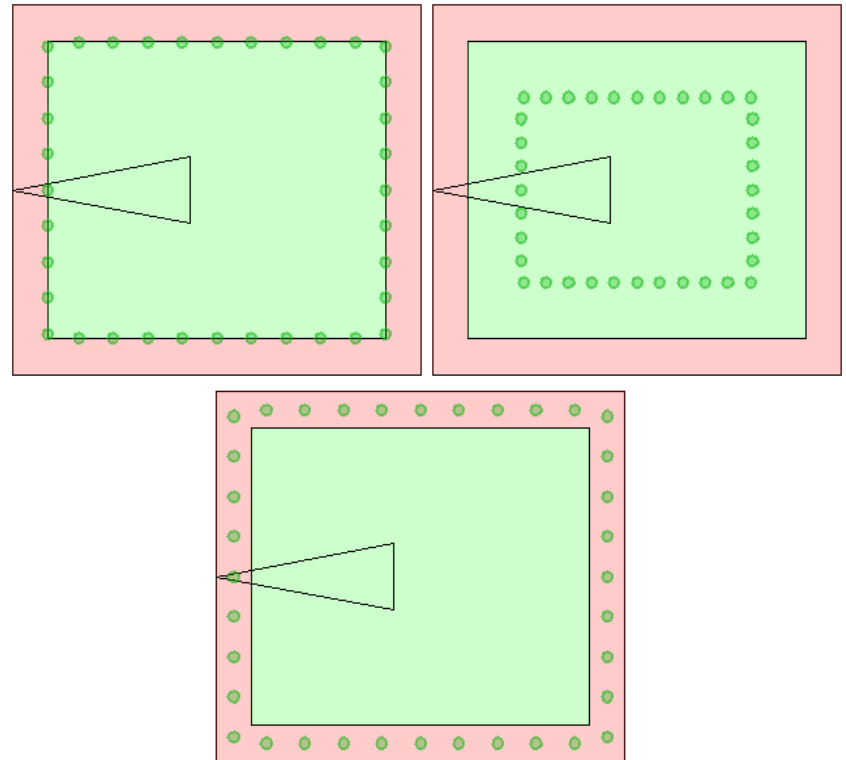
Procedures

- Sensor Grid Arrangements
 - Perimeter
 - Uniform
 - Dice 5



Procedures

- Margins
 - Positive – Sensors inside base
 - Negative – Sensors Outside base



Optimization Algorithms

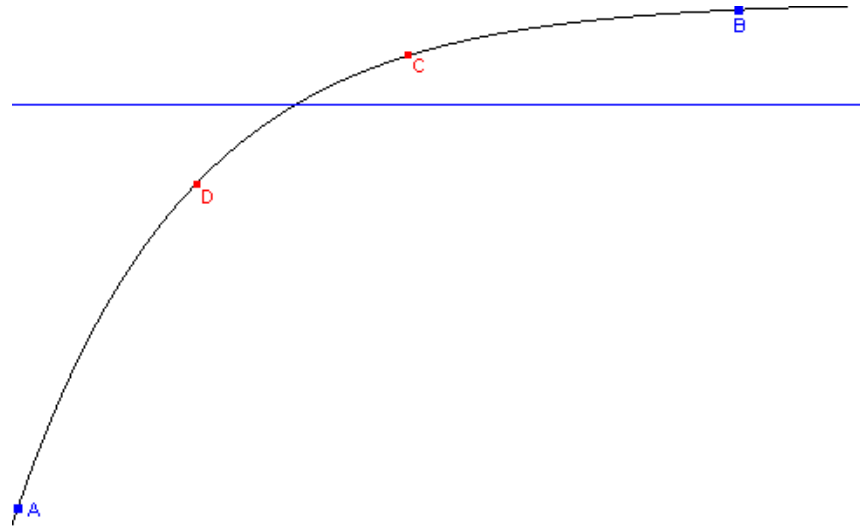
- “Placement”
 - Limited number of sensors
 - Find best configuration
 - Searches arrangements with many sensors

Optimization Algorithms

- “Allocation”
 - Specified Performance Threshold
 - Finds arrangement with fewest sensors.
 - Uses Binary Search and Golden Section Search

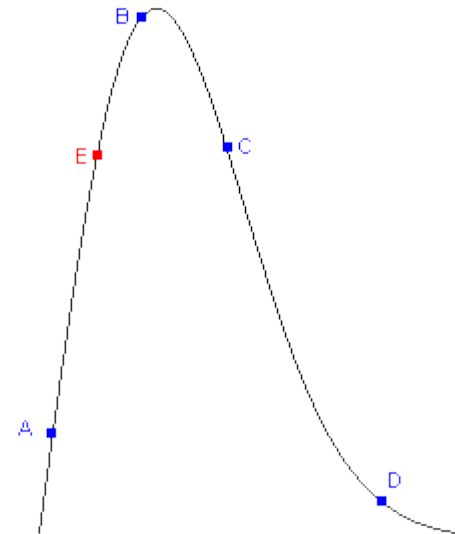
Binary Search

- Find intersection of 2 lines
- Divide Line in 2
- Determine which half contains intersection
- Repeat to specified accuracy



Golden section search

- Find maximum value of function
- Check 2 middle points, reduce region
- Only uses 1 new point per iteration
 - More efficient



Results

- Desired Program Traits
 - Fast
 - Effective
 - Modifiable
- Successful Optimizations

Number of Trials	Time (s)
500	≈ 100
1000	≈ 200
2500	≈ 480