

Chemical Biological Sensor Rapid Simulator

by
Jeffrey J. Scott
Senior Project Manager
and
Quintin Stotts
Senior Consultant, Measurement & Automation
Data Science Automation, Inc.
USA

Category:
Life Sciences

Products Used:
NI LabVIEW 2009

The Challenge:
Quickly develop an application to simulate a customer's standoff chemical agent sensor. The application would need to communicate with a main command center to simultaneously read gas concentration values from a running model using previously acquired measurements and also relay that concentration along with the azimuth and elevation positions.

The Solution:
Use LabVIEW to interface with external technologies (.NET and TCP/IP) with minimal effort. Create a user friendly interface that displayed the current scan position (azimuth and elevation) of the simulated sensor and the gas concentration read from the model also made LabVIEW a perfect choice for this project.

Abstract:
A LabVIEW application was developed to serve as a simulator for a standoff chemical agent sensor which operates as part of a large network of sensors used to scan and track chemical clouds over a wide scale area. The LabVIEW application will generate data that simulates the manner in which the sensor scans its field of view to detect chemical or biological agents. This data, along with gas concentration from an external running model, is transferred to a central command system for processing along with other sensor data in the network to triangulate moving chemical clouds.

Large System, Small Lab

The overall tracking system implements five sensors to accurately triangulate a chemical cloud over a large scale square area. One sensor resides at each corner and the fifth sensor is used for determining elevation of the cloud. The system is implemented in a remote area therefore making the ongoing testing of the overall system difficult. The decision was made to develop simulators for the sensor so that the overall system could be tested in the lab prior to traveling to the field for exhaustive testing. One or two real sensors in the lab could be put in operation while the rest of the network could be comprised of simulators and the central command system could operate as normal. The command system software was ongoing constant tweaks and changes based on end-user feedback, so it would be very impractical to make continuous trips to the field test site to verify new changes that occur every week.

Simulator Operation

The simulator needed to perform three main operations: Generate positional data, connect via TCP/IP to a running model providing gas concentration data based on the simulators current position, and send its current position and gas concentration readings to the central command system via .NET calls. All this information would also be conveyed to the user interface for operator verification. See Figure 1.

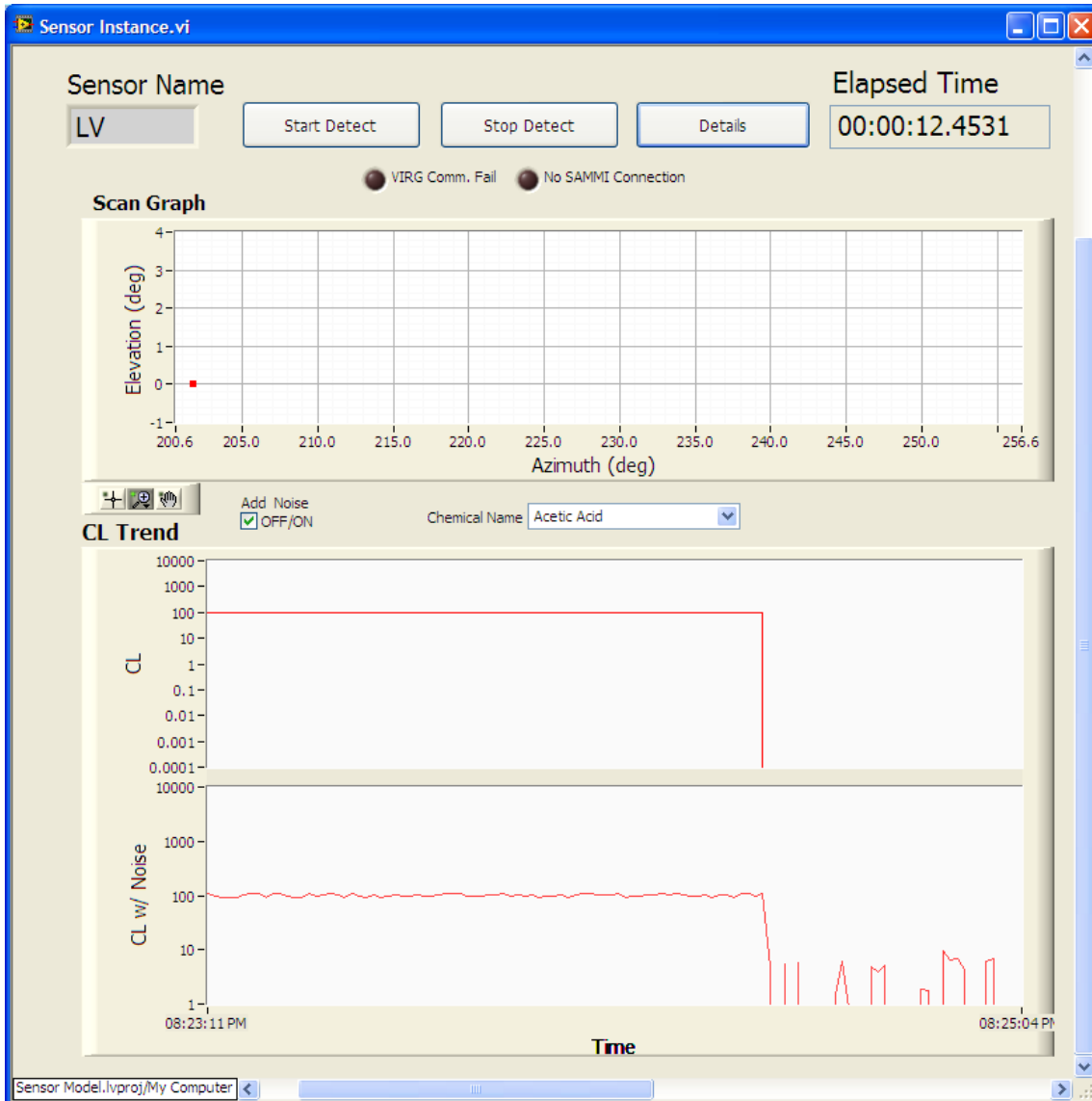


Figure 1. Simulated Sensor User Interface

The sensor’s position is in terms of azimuth and elevation. A real sensor, when idle, performs raster scanning, rotating left to right and right to left between certain a certain azimuth minimum and maximum. Once it reaches a minimum or maximum azimuth it increments in elevation and scans right to left. It continues this until it reaches its elevation maximum, then it decrements in elevation scanning left to right and right to left at each elevation until it reaches its elevation minimum, and then repeats all over again. A sensor can also operate in a detect tracking mode in which it quickly moves in diagonal motion within its azimuth and elevation minimum and maximum ranges. If it detects a gas concentration that exceeds a certain threshold it begins finely scanning the area in which it detected a threat to determine the overall area that the gas resides in. So the LabVIEW simulator had to mimic this exact operation, keeping track of its current position, calculating what the next position should be and displaying the current position in an XY Graph on the user interface.

This current position was sent to an externally running model to determine and retrieve a gas concentration, using acquired data from a previous field test, which brings real-world factors such as wind and temperature into play. The running model would connect with sensors using a TCP/IP connection to allow the sensors to tell the model where they are scanning and the model to provide back gas concentration values based on that scan position.

The gas concentration values along with the current position of the sensor were then simultaneously transmitted to the central command system to piece together data from all five sensors and triangulate the exact area and movement of a chemical cloud. The command system expected the data in a specific format and a .NET call already existed to create the data message and also transmit the message to the command system, again via TCP/IP. So instead of battling the challenges of making sure the command format was exactly right and redeveloping TCP/IP code again, LabVIEW was able to easily interface with the existing .NET library to handle the formatting and transmission of the position and concentration data.

Summary

Thanks to LabVIEW, the simulator application was developed and deployed in less than 2 weeks time in order to conduct critical testing in the lab prior to making trips to the actual test site for overall system testing. Multiple simulators could be easily deployed as executables running as separate instances to test out the entire network without the need for any real, physical sensors in place.