Usman Mahmood Khan

Muhammad Saad Qureshi

Muddassir Ahmed Khan

AERO2

ECD Semester Project Report

Aer02 maps global regions that are ideal for athletes’ training and performance.

Table of Contents

[Abstract: 2](#_Toc439685629)

[1.0: Literature Review: 2](#_Toc439685630)

[2.0: Block Diagram: 3](#_Toc439685631)

[2.1: Circuit Encapsulation Diagram: 3](#_Toc439685632)

[2.1.1: Circuit Diagram 4](#_Toc439685633)

[2.1.2: Components Used: 4](#_Toc439685634)

[2.2: STM Program: 6](#_Toc439685635)

[2.2.1: File Structure Hierarchy: 6](#_Toc439685636)

[3.0: Communication Protocol: 6](#_Toc439685637)

[4.0: Android Application: 7](#_Toc439685638)

[4.1: Overview: 7](#_Toc439685639)

[4.1.1: Record Mode: 7](#_Toc439685640)

[4.1.2: Map Mode: 8](#_Toc439685641)

[4.2: File & Class Hierarchy: 9](#_Toc439685642)

[5.0: Cloud / Database Overview: 9](#_Toc439685643)

[5.1: Database Tables: 9](#_Toc439685644)

[5.2: Computation Engine: 9](#_Toc439685645)

[5.3: Class Structure: 9](#_Toc439685646)

[5.4: APIs: 10](#_Toc439685647)

[5.4.1: Google Place API: 10](#_Toc439685648)

[5.4.2: Google Maps API 10](#_Toc439685649)

[5.5: Future Horizons: 10](#_Toc439685650)

[6.0: Use cases: 10](#_Toc439685651)

[6.1: Volunteer Mode: 10](#_Toc439685652)

[6.2: User Mode: 11](#_Toc439685653)

[7.0: Conclusion: 11](#_Toc439685654)

[8.0: Appendices: 12](#_Toc439685655)

[8.1: Appendix 1 [STM File Structure Hierarchy]: 12](#_Toc439685656)

[12](#_Toc439685657)

[8.2: Appendix 2 [Communication Protocol]: 12](#_Toc439685658)

[8.3: Appendix 3 [Android File Structure Hierarchy]: 15](#_Toc439685659)

[9.0: References: 17](#_Toc439685660)

# Abstract:

The project maps global regions that are ideal for athletes training and performance. The objective of Aer02 is to enhance the training and performance of athletes by suggesting regions around them that have ideal air content suited to their health needs. Athletes can use Aer02 application with or without hardware. With hardware, they have the added advantage of real time smog data and an alarm system. The system works in three main steps: First, specialized sensors are used to map air pollution and smog concentrations over certain regions (such as sectors of Islamabad) using surveillance carried out by our participants. Second, the information is fed to a machine learning algorithm to develop a relationship between oxygen concentration and various demographic and industrial factors such as population density, energy consumption, automobiles’ quantity and et cetera. Third, AerO2 application suggests athletes certain places in a city which are best suited for performance, in form of a heat map.

# 1.0: Literature Review:

Smog is a type of air pollutant that affects visibility in air. According to a recent survey, smog affects a wide subset of population. Recently, the over-intensity of smog has become a particularly strong problem especially in the populous cities [1].

Smog is caused by various factors: both natural and human. In this project, our focus was on man-made factors alone. Recent studies show that man-made smog is derived from coal emissions, vehicular emissions, industrial emissions, forest fires and the like [2]. Modern smog is a type of air pollution that is derived vehicular emissions from internal combustion engine [3].

Smog is highly toxic to humans and is a major problem in cities like Los Angeles, Beijin, Delhi et cetera.

Based on above findings, it is worth determining smog concentration. However, the challenges is to make a smog for all locations in the globe due to limited resources. Therefore, the goal of this project is to employ machine learning algorithms to develop a model between smog concentration and traffic index and location index.

# 2.0: Block Diagram:

The block diagram is shown below:

Azure SQL Storage

Python Based Azure-hosted Computation Engine

Cloud Module

SQLite Local Cache Storage

Mobile Application

Android Mobile Application

Sensor Circuit

Hardware Module

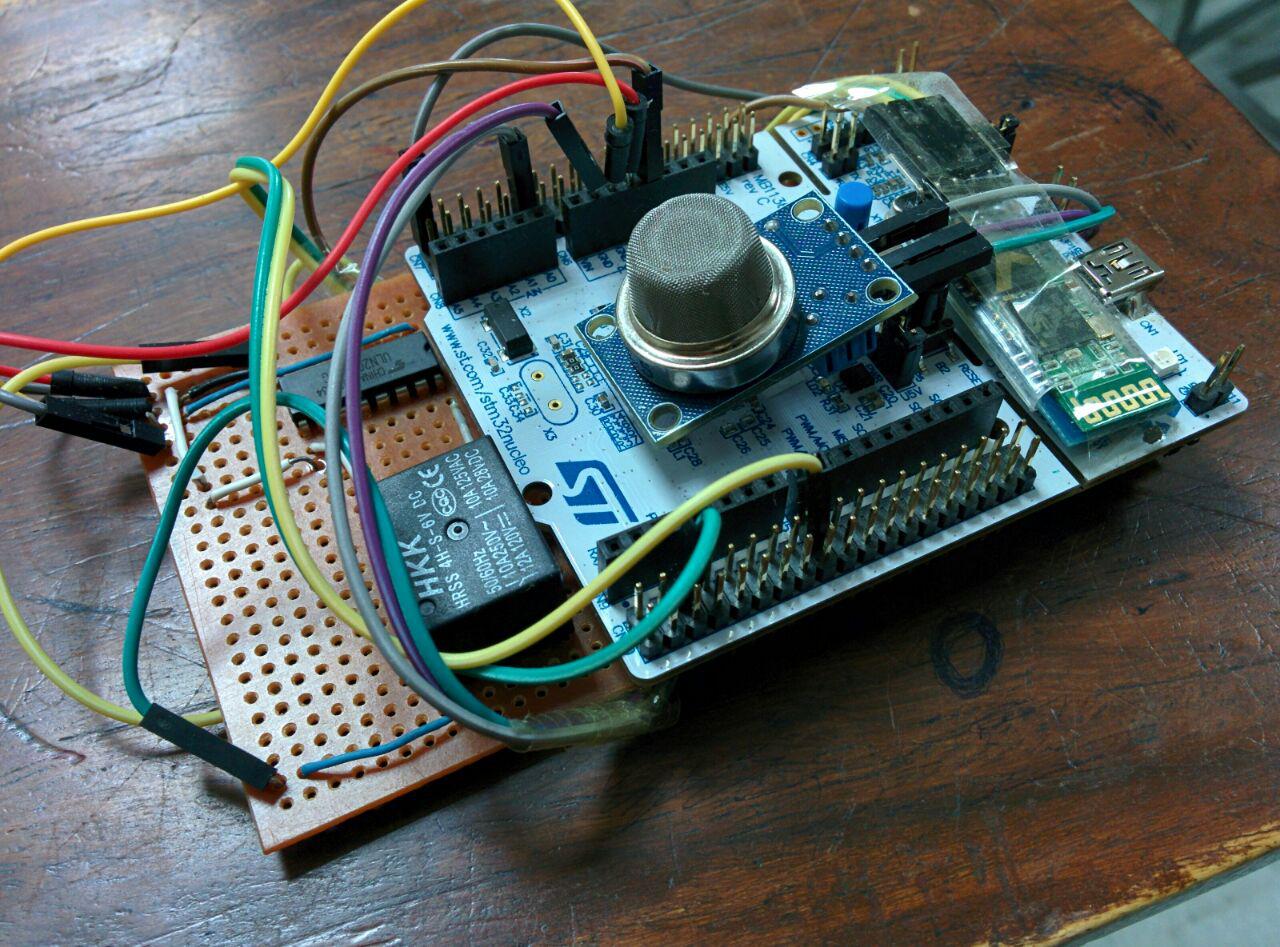
STM32 Board

## 2.1: Circuit Encapsulation Diagram:



### 2.1.1: Circuit Diagram

The PCB diagram of the circuit is shown in figure below:



### 2.1.2: Components Used:

Following components were used for this project:

-*STM32F103 Nucleo Board:*

It is an evaluation board which contains state-of-the-art ARM Cortex M3 MCU. The specifications are given in the link below:

<http://www.st.com/web/en/resource/technical/document/datasheet/CD00161566.pdf>

-*MQ-5*:

MQ-5 is a smog sensor with high sensitivity to LPG which is one of the major causes of man-made smog.

Sensitive material of MQ-5 gas sensor is SnO2 [Indium Tin Oxide], which has lower conductivity in clean air. When the target combustible gas exists, the sensor has higher conductivity and increases with increase in concentration of target gases.

The target gases are:

1- Methane

2- Butane

3- Propane

4- Other combustible gases

MQ-5 gas sensor has:

- High sensitivity to LPG, natural gas , town gas

- Low sensitivity to smoke & alcohol

<http://www.dfrobot.com/image/data/SEN0130/MQ-5.pdf>

-*HC-05:*

HC-05 is a Bluetooth module whose default baud rate is set at 9600. Although the baud rate can be changed, but it was enough for our requirement.

-*ULN2003A:*

ULN2003A is an op-amp module which is used to drive relay and provide sufficient current for its operation.

-*Relay:*

Relay was used to add switch functionality to our device so as to save power as well as enable user to switch it at their ease.

-*Vero Board:*

Vero board was used to solder different components together in a component manner.

## 2.2: STM Program:

### 2.2.1: File Structure Hierarchy:

The file structure hierarchy is provided in *Appendix 1*.

# 3.0: Communication Protocol:

An extensive list of communication protocol was used in order to prevent security breach of data and develop an industry-compliance project.

Communication protocol consists of authentication, enable and access stages. In authentication, user first sends a command to ask system to start authentication process. STM responds by telling user if it is ready to start authentication process. Afterwards, user sends their username and password and STM responds by letting them know if the credentials are valid.

In enable stage, user can enable sensor(s). For this project, we used just one sensor but in future multiple sensors can be appended. The communication protocol affords room for that ability.

In access stage, STM sends smog data to the user. It sends 0 if it was unable to retrieve data in a rare case. At this stage, user has the option to send disable command at any time to disable smog sensor.

A comprehensive overview of communication protocol is provided in *Appendix 2*.

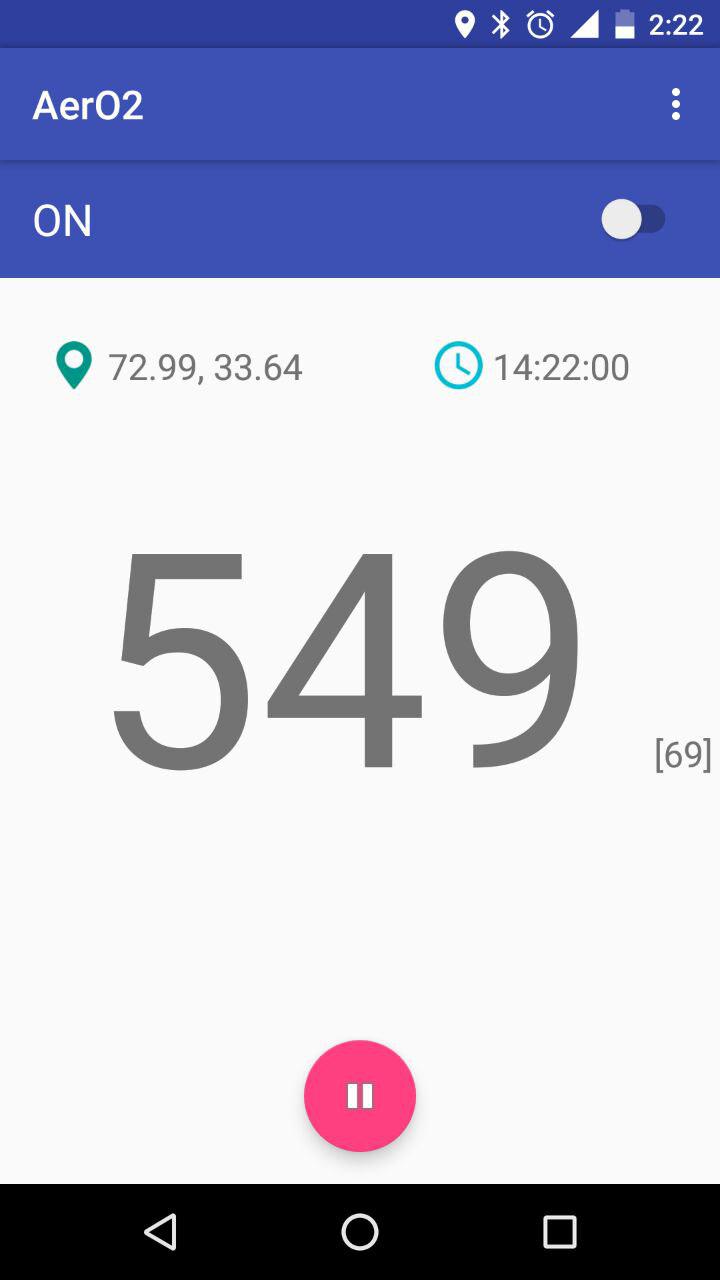
# 4.0: Android Application:

## 4.1: Overview:

Android application runs in two modes which are explained below:

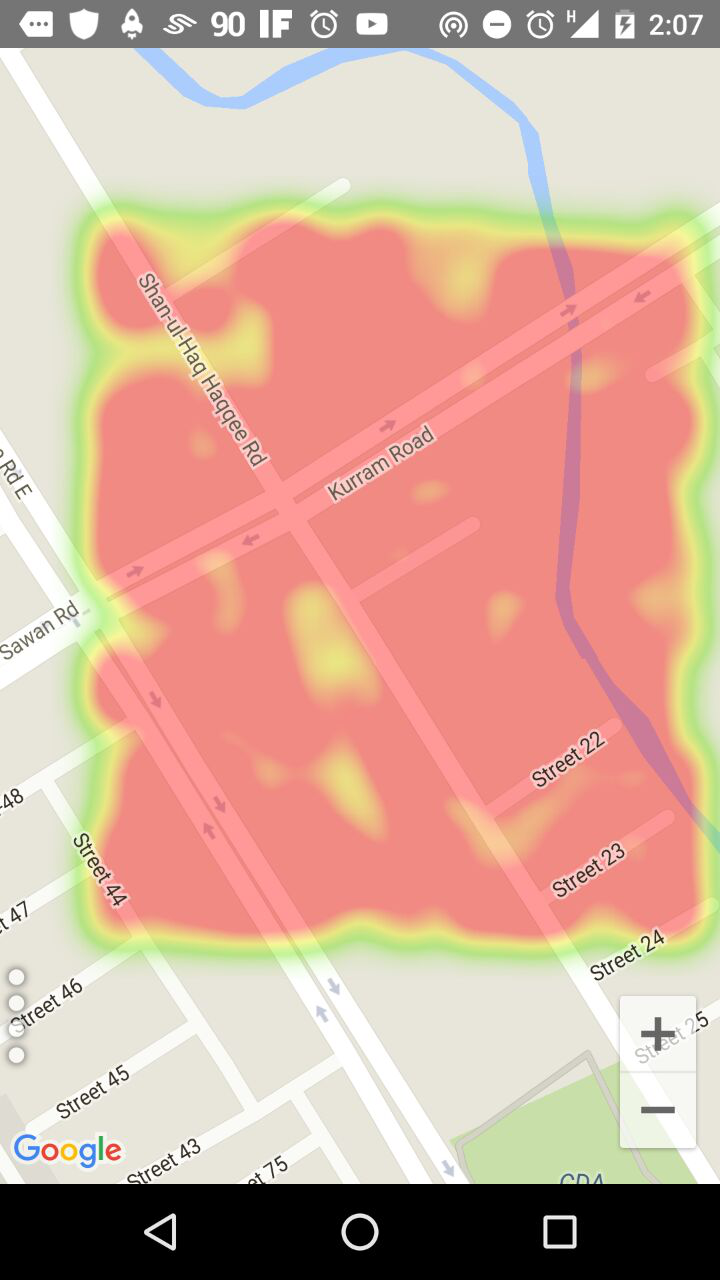
### 4.1.1: Record Mode:

In record mode, athletes see real-time values provided by smog sensor and contribute to overall learning algorithm. A screenshot of this activity is shown below:



### 4.1.2: Map Mode:

In map mode, runners get a whole range of heat maps to see the areas in a city where smog values are minimal. Furthermore, they get routes where smog value is minimal and where they can perform in a good manner. A screenshot of this activity is shown below:



## 4.2: File & Class Hierarchy:

In general, the theme of the application is to collect values from user and record them immediately in local SQLite cache storage. Before storing, application performs various checks to ensure that:

1. Communication protocol is followed
2. Smog values are valid
3. Location values provided by mobile GPS are valid (not null)

A comprehensive overview of Android file hierarchy is given in *Appendix 3.*

# 5.0: Cloud / Database Overview:

## 5.1: Database Tables:

Our project employed three data tables.

1. *SampleDataTable:* All values retrieved from hardware are sotored in SampleDataTable. The column structure of SampleDataTable is shown in *Appendix 3*.
2. *ResultDataTable:* After the machine learning process and calculation of traffic and air index, the results are fed back to ResultDataTable as shown in *Appendix 3*.
3. *PropertiesDataTable:* The properties of each city are shown in PropertiesDataTable as shown in *Appendix 3*.

A screenshot of the three tables in Azure is shown in figure below:

## 5.2: Computation Engine:

A virtual machine is run in Azure that runs once every 24 hours, runs the machine learning algorithm and feeds results back to ResultDataTable.

## 5.3: Class Structure:

-*AzureSQLHandler :* Azure SQL Database Handler uses SQL login credentials to connect and Interact with the Azure SQL DB

-*DataBaseLayer:* Database Layer allows functions for accessing the SQL Database at a higher abstraction level and easy-access

-*AzureRoutine:* Azure Routine Schedules the tasks for running on the Computation engine

-*DataTraining:* Data Training allows the intialization of the Properties Table and runs the Machine Learning Algorithms

-*Maps:* Maps utilizes the google maps api to calcuate certain properties of each node location

## 5.4: APIs:

### 5.4.1: Google Place API:

We tracked industries information using Google Places API. The api is flexible to track specific industries that contribute to pollution.

The implemented function called Google Place API and returned the nearest place within the radius provided by user.

The supported list of places can be found here: <https://developers.google.com/places/supported_types>

The API receives following arguments:

:param radius:

:param ptype:

:param key:

:param origin:

### 5.4.2: Google Maps API

We use Google Maps API to track distance. The implemented function called Google Distance API and returned the duration/km.

departure\_time is number of seconds in int from December, 1970.

traffic model can be 'best\_guess', 'optimistic', or 'pessimistic'

## 5.5: Future Horizons:

The machine learning was not entirely completed due to time constraints. However, correlation coefficients were determined between smog values and traffic & industries’ data. This gave us hints about strong relationship between the two. This is shown in figures below:

In future, we plan to use regression trees which are ideal for developing such types of models.

# 6.0: Use cases:

## 6.1: Volunteer Mode:

This mode allows our volunteers to samples air data from their location of runs. They will be provided free arm bands (with our hardware embedded inside) for this purpose and will be rewarded with premium services depending on their contributions.

## 6.2: User Mode:

This is the regular user mode which will allow general user base to view locations on a heat- map. In addition, following are the potential uses:

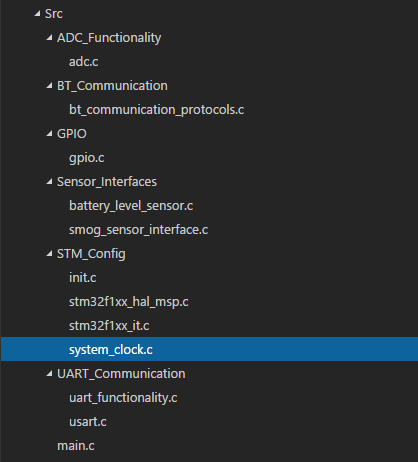
1. Suggesting spots in city where smog values (and consequently pollution values) are minimal.
2. Suggesting running routes which have least averaged smog value.
3. Incentivizing local committees to take actions to minimize smog values by giving them an accurate description.

# 7.0: Conclusion:

The project provided us an opportunity to combine our learnings from various domains into one concrete product. We hope to continue this project in future as we see real-use case for runners and athletes.

# 8.0: Appendices:

## 8.1: Appendix 1 [STM File Structure Hierarchy]:



## 8.2: Appendix 2 [Communication Protocol]:

STM Communication Protocols with Mobile Application

This software implements a very simple message queue framework of STM with Android Mobile Applications.

Applications include:

-Secured P2P communication over Bluetooth

-Authentication with the MCU directly from the Mobile Application

-Enabling and Disabling the sensors remotely

-Requesting the required Data and Receiving it in the desired format

INSTALLATION:

The user end API will not be made available for installation as it is only meant to be utilized by our application.

USAGE

This extension is based on the idea of 3 Digit I/O commands appended after the Prefix which is either I or O depending on where the command is coming from. 'I' representing the Command by the MCU and 'O' being the commands from Application.

Following is the list of all the available commands along with their format:

'O' Commands:

-Availablility Enable / Disable

OAUT:1 / OAUT:0

-Set Username

OUSR:[Username (String)]

-Set Password

OPAS:[Password (String)]

-Enable / Disable Smog Sensor

ONSG:1 / ONSG:0

-Request Smog Sensor Data

OSSG:1

-Request Battery Percentage

OBAT:1

'I' Commands:

-Availablility Status

IAUT:[Enabled or Disabled (0 or 1)]

-Username Status

IUSR:[Authenticated or Disabled (0 or 1)]

-Password Status

IPAS:[Authenticated or Disabled (0 or 1)]

-Smog Sensor Status

INSG:[Enabled or Disabled (0 or 1)]

-Smog Sensor Data

ISSG:[Data Value]

-Battery Percentage

IBAT:[Percentage Value]

Standard Operating Procedure (SOP) Examples

-------------------------------------------

// Availablility

OAUT:1 -> IAUT:1

OAUT:0 -> IAUT:0

// Username Enter Correct AND IAUT:1

OUSR:saad -> IUSR:1

// Username Enter Incorrect OR IAUT:0

OUSR:hacker -> IUSR:0

// Password Enter Correct AND IUSR:1

OPAS:correct\_password -> IPAS:1

// Password Enter Incorrect OR IUSR:0

OPAS:incorrect\_password -> IPAS:0

// Request for Smog Sensor Enable AND IPAS:1

ONSG:1 -> INSG:1

ONSG:0 -> INSG:0

// Request for Smog Sensor Enable AND IPAS:0

ONSG:1 -> IPAS:0

ONSG:0 -> IPAS:0

// Request for Smog Sensor Data AND IPAS:1

OSSG:1 -> ISSG:334

// Request for Smog Sensor Data AND IPAS:0

OSSG:1 -> IPAS:0

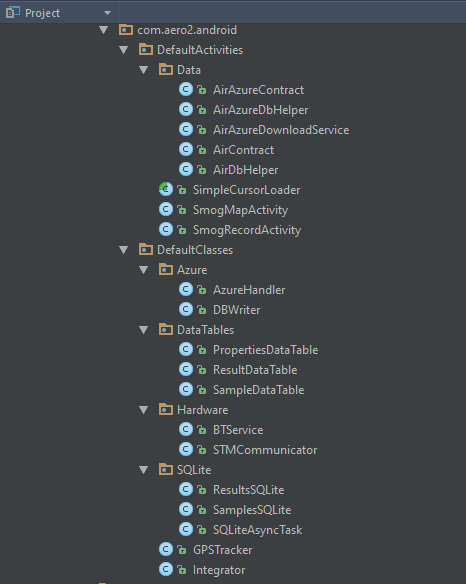
// Request for Battery Percentage AND IPAS:1

OBAT:1 -> IBAT:98

// Request for PPG Sensor Data AND IPAS:0

OBAT:1 -> IPAS:0

## 8.3: Appendix 3 [Android File Structure Hierarchy]:



*Azure:* Dbwrite implements a mobile service to write and retrieve data from Azure. AzureHandler builds a upper level interface that communicates between Android and Azure.

*DataTables:* The attributes of each class in this package are shown below:

1) SampleDataTable

-Attributes:

0) time (number)

1) lat (number)

2) long (number)

3) alt (number)

4) smog (number)

5) normalized (number)

2) ResultsDataTable

-Attributes:

0) time (number)

1) lat (number)

2) long (number)

3) air\_index (number)

3) PropertiesTable

-Attributes:

0) sampled (number)

1) lat (number)

2) long (number)

3) r\_index (number)

4) i\_index (number)

*Hardware:* This package implements classes that deal with hardware components.

*SQlite:* This package contains classes that implement various sqlite functions on different abstraction levels to store and retrieve data from local storage.

# 9.0: References:

### [1] Watkins et al. (2013). [Air pollution and arrhythmic risk: the smog is yet to clear](http://www.sciencedirect.com/science/article/pii/S0828282X12013311).

[2] Saitoh et al. (2002). *Chemical Characterization of particles in winter-night smog in Tokyo.*

### [3] Shipchandler et al. (2008). [Estimating Smog Precursor Emissions from Idling Vehicles in the Chicago Metropolitan Area](http://www.cleanaircounts.org/documents/Estimating%20Emissions%20from%20Idling%20Vehicles.pdf). *Clean Air Counts*