QUALITY ESTIMATION FOR SCARCE SCENARIOS WITHIN MOBILE CROWD SENSING SYSTEMS

By

SHERIF B. AZMY

A thesis submitted to the Graduate Program in Department of Electrical and Computer Engineering in conformity with the requirements for the Degree of Master of Applied Science

Queen’s University
Kingston, Ontario, Canada
November, 2020

Copyright © Sherif B. Azmy, 2020
Dedication

FOR THE SAKE OF ALLAH

to my parents
Abstract

Mobile Crowd Sensing (MCS) is a paradigm that exploits the presence of a crowd of moving human participants to acquire, or generate, data from their environment. As part of the Internet of Things (IoT) paradigm, MCS serves the quest for more efficient operation of a smart city. Big data techniques employed on this data produce inferences about the participants’ environment, the smart city. However, sufficient amounts of data are not always available. Sometimes, the available data is scarce as it is obtained at different times, locations, and from different MCS participants who may not be present. As a consequence, the scale of data acquired maybe small and susceptible to errors. In such scenarios, the MCS system requires techniques that acquire reliable inferences from such limited data sets. To that end, we resort to small data techniques that are relevant for scarce and erroneous scenarios. In this thesis, we discuss small data and propose schemes to tackle the problems associated with such limited data sets, in context of the smart city. We propose two novel quality metrics, MAD-Q and MADBS-Q, to deal with small data, focusing on evaluating the quality of a data set within MCS. We also propose an MCS-specific coverage metric that combines the spatial dimension with MAD-Q and MADBS-Q. We show the performance of all the presented techniques through closed-form mathematical expressions, with which simulations results were found to be consistent.
List of Publications

Journal Publications


Conference Publications


Book Chapters

Acknowledgment

First of all, I would like to express my gratitude to Prof. Nizar Zorba and Prof. Hossam Hassanein for their unrelenting support, supervision, mentorship, and guidance in teaching me the intricacies of research. I also would like to thank Prof. Boualem Boashash for his inspiring influence that compelled me to embark on the academic journey in hopes of becoming an engineering educator, and Dr. Mohammed Ben Haj Rhouma for teaching me about hard work and rigor.

I thank my family for their support and encouragement, as without them I would not have been able to have the perseverance to remain on this path nor the patience to learn. My family has taught me the notion of self-reliance and self-learning throughout my childhood, and it is that notion that had led me - by the support of Allah - to where I currently am.

I also thank my dear friend, Sharief Saleh, whom has been in my company throughout our studies, for Allah has fated that not only do we remain in each other’s company throughout our undergraduate studies in Qatar, but also - by sheer unplanned coincidence! - throughout our post-graduate studies at Queen’s University in Canada. I also thank my dear friend, Salman Ismail-jan, for his wise presence, sincere friendship, and honest advice.

I thank my friend and flatmate, Marwan Rashed, for the food he makes, the jokes he says, the trivialities we fight over, and for making life in Kingston bearable (or unbearable). Thanks to him and Omar, Osama, Magdy, and Amr, Kingston has come to feel more of a home than a place in a strange land.

I also extend my thanks to the TRL members for their support, especially Amir for his sound advice, Adel for his kindness (and the amazing dark TeXStudio trick), Rawan for her wise insight throughout our academic journey from Qatar to Canada, and Nagib for his active diligence and thirst for learning.
# Table of Contents

Abstract ii

List of Publications iii

Acknowledgment iv

Table of Contents v

List of Figures vii

List of Abbreviations viii

List of Symbols ix

1 Introduction 1

1.1 Overview and Motivation ........................................ 1

1.2 Challenges ...................................................... 1

1.3 Objective and Contribution ..................................... 4

2 Mobile Crowd Sensing: Description, Architecture, and Literature Review 6

  2.0.1 Entities of an MCS System .................................. 7

  2.1 MCS Layered Architecture .................................... 8

  2.1.1 The Application Layer ...................................... 8

  2.1.2 The Data Layer ............................................. 12

  2.1.3 The Communication Layer ................................ 14

  2.1.4 The Sensing Layer ......................................... 16

  2.2 Quality in MCS ............................................... 19
### 3 Mathematical Essentials and the Spatiotemporal MCS Model

3.1 MAD-Trimmed Mean .......................... 24
3.2 The Non-Parametric Bootstrap ............... 28
3.3 The Bootlier ................................ 29
3.4 MCS Spatiotemporal Model .................. 30

### 4 Cell-Specific Quality Metrics

4.1 MMTM Statistic, and the MAD-Q Quality Metric ................. 36
4.2 The MADBS-Q Bootstrap-based Quality Metric ................. 38
4.3 The Impact of Control Parameters ....................... 39

### 5 MCS-Specific Coverage Metric

5.1 Overall Coverage Quality ...................... 43
5.2 Relative Coverage Quality ...................... 45

### 6 Simulation and Results

6.1 Cell-Specific Quality .......................... 47
6.2 Coverage Quality ............................... 52
6.3 Discussion .................................. 53

### 7 Conclusions

7.1 Summary and Conclusion ....................... 54
7.2 Recommendations and Further Work .............. 55

References ................................. 57

Appendix: Bootstrap Probability Analysis 69
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>General MCS Architecture</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>Layered MCS Architecture</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>Population with Abnormal Outliers (lower and upper)</td>
<td>25</td>
</tr>
<tr>
<td>4</td>
<td>Spatiotemporal Diagram</td>
<td>31</td>
</tr>
<tr>
<td>5</td>
<td>Description of MCS-Specific Coverage Metric over a $C \times C$ square grid</td>
<td>44</td>
</tr>
<tr>
<td>6</td>
<td>$Q_{MAD}(N)$ and $Q_{BS}(N)$</td>
<td>48</td>
</tr>
<tr>
<td>7</td>
<td>$Q_{MAD}(\bar{\Theta})$ and $Q_{BS}(\bar{\Theta})$</td>
<td>48</td>
</tr>
<tr>
<td>8</td>
<td>$Q_{MAD}(\lambda)$ and $Q_{BS}(\lambda)$</td>
<td>49</td>
</tr>
<tr>
<td>9</td>
<td>$Q_{MAD}(N, \bar{\Theta})$</td>
<td>49</td>
</tr>
<tr>
<td>10</td>
<td>$Q_{MAD}(\bar{\Theta}, \lambda)$</td>
<td>50</td>
</tr>
<tr>
<td>11</td>
<td>$Q_{MAD}(\lambda)$ for different $\beta$</td>
<td>50</td>
</tr>
<tr>
<td>12</td>
<td>$Q_{MAD}(\beta)$ for different $\lambda$</td>
<td>51</td>
</tr>
<tr>
<td>13</td>
<td>Spy plot of the adjacency matrix, $Q_{rel}$</td>
<td>52</td>
</tr>
<tr>
<td>14</td>
<td>Venn Diagram of the Bootstrapped Sample Space and its subsets</td>
<td>69</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>--------------</td>
<td>--------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>4G</td>
<td>Fourth Generation communication technology</td>
<td></td>
</tr>
<tr>
<td>5G</td>
<td>Fifth Generation communication technology</td>
<td></td>
</tr>
<tr>
<td>AODF</td>
<td>Average Outlier Deviation Factor</td>
<td></td>
</tr>
<tr>
<td>BD</td>
<td>Big Data</td>
<td></td>
</tr>
<tr>
<td>GPS</td>
<td>Global Positioning System</td>
<td></td>
</tr>
<tr>
<td>IMU</td>
<td>Inertial Measurement Unit</td>
<td></td>
</tr>
<tr>
<td>IoT</td>
<td>Internet of Things</td>
<td></td>
</tr>
<tr>
<td>LTE</td>
<td>Long-Term Evolution</td>
<td></td>
</tr>
<tr>
<td>MAD</td>
<td>The median absolute deviation</td>
<td></td>
</tr>
<tr>
<td>MAD-Q</td>
<td>MAD Quality Metric</td>
<td></td>
</tr>
<tr>
<td>MADBS-Q</td>
<td>MAD Bootstrap Quality Metric</td>
<td></td>
</tr>
<tr>
<td>MCS</td>
<td>Mobile Crowd Sensing</td>
<td></td>
</tr>
<tr>
<td>MMTM</td>
<td>Mean-MAD Mean-Trimmed Mean</td>
<td></td>
</tr>
<tr>
<td>ODF</td>
<td>Outlier Deviation Factor</td>
<td></td>
</tr>
<tr>
<td>PoI</td>
<td>Point of Interest</td>
<td></td>
</tr>
<tr>
<td>SD</td>
<td>Small Data</td>
<td></td>
</tr>
<tr>
<td>WLAN</td>
<td>Wireless Local Area Network</td>
<td></td>
</tr>
</tbody>
</table>

viii
List of Symbols

$B$  
Number of bootstrap resamples.

$b_\theta$  
The breakdown point of a statistic $\theta$.

$C$  
Grid size.

$f_{jl}$, $f_{ju}$  
Frequencies required for the $j^{th}$ median to be selected.

$N$, $N_o$  
Size of a sample set. The subscripts $N(\cdot)$, $N(\cdot,i)$ specify the size of a modified sample set.

$N(\mu, \sigma)$  
A normal distribution centered at $\mu$ with spread $\sigma$.

$Q_{BS}$  
MADBS-Q Quality Metric.

$Q_{MAD}$  
MAD-Q Quality Metric.

$Q_{map}$, $\hat{Q}_{map}$  
Quality matrix, containing quality for the corresponding $(i,j)$ cell. $\hat{Q}_{map}$ is its singleton variant.

$Q_{\min}$  
Minimum quality threshold.

$Q_{rel}$  
Relative quality between two adjacent cells.

$Q_{tot}$  
Overall quality metric.

$r_{M1}$, $r_{M2}$  
Samples that contribute to the median, in case of an even sample size.

$r_U$, $r_L$  
Samples greater than, or less than, $x_n$, in case of an odd sample size.

$X$, $x_i$  
A sample set $X = \{x_1, x_2, \ldots, x_N\}$, $x_i$ for $i = 1, 2, \ldots, N$.
The subscripts $X(\cdot)$, $x(\cdot,i)$ specify a modified sample set.

$\bar{x}$, $\bar{x}_k$, $\bar{x}_{MAD}$  
Mean, $k$-trimmed mean, and MAD-mean of a sample set $X$. 
\( \bar{x} \) The median of a sample set \( X \).

\( \beta \) MMTM sensitivity, also mixing parameter.

\( \delta \) Multiples of \( \hat{\sigma}_{MAD} \) non-outlier samples are away from the median.

\( \gamma \) Scaling factor for a corresponding quality metric.

\( \lambda \) Multiples of MAD non-outlier samples are away from the median.

\( \mu \) Truth of a sensed quantity.

\( \mu_{MMTM} \) Mean of the MMTM bootstrap distribution.

\( \hat{\mu} \) Estimate of the true value of a sensed quantity. The subscript \( \hat{\mu}_{(\cdot)} \) indicates an estimate by a specific statistic.

\( \Phi^{-1}(p) \) The quantile function, also known as the inverse cumulative distribution function, for probability \( p \).

\( \hat{\sigma} \) An estimate of the standard deviation of the sensed population. The subscript \( \hat{\sigma}_{(\cdot)} \) indicates an estimate by a specific statistic.

\( \bar{\sigma}_\theta \) The standard error of a statistic \( \theta \).

\( \sigma_{MMTM}^2 \) Variance of the MMTM bootstrap distribution.

\( \theta_{MMTM} \) Mean-MAD Mean-Trimmed Mean statistic.

\( \theta, \theta_b, \theta^* \) An arbitrary statistic; \( \theta_b \) refers to a bootstrap instance of that statistic; and \( \theta^* \) is the set of \( B \) bootstrap statistics.

\( \Theta, \bar{\Theta} \) An outlier’s deviation from the truth. \( \bar{\Theta} \) indicates average deviation for all outliers in a set.

\( \xi \) Estimation error. The subscript \( \xi_{(\cdot)} \) indicates the estimation error by a specific statistic.
\( \mathbb{E}(\cdot) \) Expectation operator.

\( \mathbb{I}(\cdot) \) Indicator function.

\( \mathcal{U} \) Bootstrapped sample space.
Chapter 1

Introduction

1.1 Overview and Motivation

Recent breakthroughs within the Internet of Things (IoT) paradigm have led to an unprecedented integration of sensors across various fields that relate to human life. As a result, infrastructures incorporating sensors are becoming more complex as the quantity and variety of sensors increase, making infrastructures more demanding for schemes that improve the efficiency of a smart city’s operation. Such infrastructural demands are a concern for smart city administrators, as more data could enhance the city’s operational efficiency, such as crisis response and transportation. One notable paradigm is Mobile Crowd Sensing (MCS), which exploits the availability of smartphones within a crowd, to benefit from their smart devices’ embedded sensors. MCS is capable of transforming a crowd of smartphone users into an extended instrument, MCS participants, for the benefit of the smart city [1].

1.2 Challenges

MCS has a wide range of applications that permit observing the social and physical dynamics of a smart city, which makes MCS a useful tool for smart city management [2]. In MCS, administrators assign tasks to participants in an autonomous manner [2]. This autonomy of sensing can be classified into participatory sensing, or opportunistic sensing [2]. In participatory sensing, the MCS system requests the participant to voluntarily engage in the execution of a task, for example, using the phone’s microphone to acquire noise levels [3] or the phone’s camera to take a photo [4]. Opportunistic sensing, on the other hand, waits until the participant satisfies the conditions necessary for the task (such as time, place, sensor capabilities, etc.), executing the task with no active intervention from the participant [5]. So far, MCS management frameworks such as 4W1H [6], and platforms, such as ParticipAct [4] were developed to address the real-time implementation of MCS. Others have developed
MCS models with the objective of reduced mathematical complexity, such as modeling the MCS sensing problem over both space and time, i.e., *spatiotemporal cells* [7].

Previous research efforts [4], [8] focus on large data sets, proposing techniques that assume abundant data is always available, often in the Big Data (BD) scale. While BD techniques are generally superior when sufficient data is available, Small Data (SD) techniques are provide insight when the data is scarce, insufficient, and erroneous, in spite of the challenge posed by these three characteristics. Nevertheless, SD can build upon the output of BD which is an inference resulting from the analysis of a large amount of data, for example, a categorical classification. SD can also complement BD by ensuring the resilience of the data sources at the beginning of the data collection pipeline. In other words, SD techniques aim to maintain the quality of the data collected at a local scale, whereas BD techniques analyze at a global scale. The combination of both SD and BD allows the exploitation of the global vs. local contrast [9].

In MCS, if the participants’ sensor quality in a spatiotemporal cell is low, then only a subset of the measured values can be considered, in such a case SD is to be used. SD also serves MCS from an economic point of view, as the administrator may be asked to reduce costs by reducing the number of sensors, in that case, a deliberate choice of a small number of sensors in combination with SD techniques would reduce the MCS sensing cost, by trading-off quality. In order to optimize such trade-off, the MCS administrator needs a metric that relates sensing quality to the number of participants for each spatiotemporal cell in order to minimize the cost, further motivating the need for SD.

MCS system costs can have a few general forms such as the provision of a service [2], incentive payments [5], or data consumed [10] to name a few. An example of an MCS system is one that asks users to perform a certain task, e.g., measure the noise pollution [11]. An incentive payment type of MCS cost within such a system could be a discount coupon to a specific shop or a restaurant, redeemable points in a loyalty program, credit for an online store (such as Google Play credits). Another type of MCS costs is the provision of services,
such as tailored recommendations, entertainment, or even a rank in a gamified MCS scoring system. From an MCS administrator’s point of view, costs could be data traffic, or the collaboration with entities that provide services to the participants. Ultimately, the costs scale up with the number of participants from the point of view of the system.

The definition of quality [12,13] depends on several parameters, and on the stakeholders’ needs, where for a small number of MCS participants, the accuracy of reports is affected by the presence of heterogeneity among the sensing devices in an MCS system [2]. Such heterogeneity is a source of discrepancy in the readings reported for a specific area of interest, as the accuracy and the precision of the embedded sensors vary from one smart device to another. A sensor estimates the true value of the sensed quantity, however its reading belongs to a sample distribution whose mean is an estimate of the true value (e.g., the normal distribution that is often observed in physical quantities) [14]. The resulting heterogeneity renders the system vulnerable to inference errors. With assumptions of a large enough data scale, these errors have a minuscule impact. However, in a spatiotemporal space, or a cell, that may contain a small number of participants, the impact of a single error could be detrimental. Therefore, specific techniques for SD scenarios are needed for MCS implementation in IoT systems with heterogeneous devices.

In a practical MCS scenario, the MCS system is blind to the true value and thus relies on MCS participants as a proxy to the true value. Furthermore, the system is also blind to how accurate the participants’ sensors are. In combination with the scale of the data in SD for a spatiotemporal cell, the truth estimate’s susceptibility to errors is greatly increased as a single incorrect measurement, an abnormality or an outlier could impair the inference, and thus the MCS system [13]. In such SD scenarios, the MCS administrator needs a metric that permits proper classification of readings under the stringent conditions of SD, and characterizes the reliability, and thus the quality, of an MCS cell’s readings as a whole. It is imperative that any MCS system is capable of automatic detection and isolation of a potential error before any analysis. In this regard, we propose novel quality metrics for MCS systems under the
1.3 Objective and Contribution

SD scenarios, as well as their extension to an MCS-specific coverage metric for the whole MCS area of interest. Such an MCS-specific coverage metric allows better characterization of MCS participants’ presence and quality over the area of interest.

1.3 Objective and Contribution

This work aims to formulate a controllable quality metric for scarce scenarios in MCS systems. The formulation is based on using the logarithm of the reciprocal of the weighted difference between centrality estimates of different robustness to evaluate the accuracy of reports in a data set in an absolute, population-agnostic manner, for readings whose populations are symmetric. Such a formulation provides the MCS administrator with the ability to control the extent to which anomalous values are excluded or included, as well as a relationship between the number of participants and the perceived quality. This formulation is designed for scarce scenarios in which sample sizes are small to rely on statistical generalizations.

The contributions of this work address the event when the amount of data is too small for proper inference, or some readings come from poor or erroneous sensors. The first metric, the MAD-Q quality metric, allows the discernment of quality for a small number of participants as low as 11. The second metric, the MADBS-Q quality metric, is more complex, but it can discern quality for an even smaller number of participants as small as eight. Lastly, we address the coverage problem, in which the quality of each cell is extended to define an overall regional quality, allowing the MCS administrator to target the recruitment process towards specific cells to achieve coverage uniformity over the whole region.

The proposed quality metrics are developed mathematically in order to show the impact of the involved parameters. Such quality metrics give the administrator a degree of control over the outlier-sensitivity and outlier-tolerance of the MCS system. This ultimately controls the MCS system’s outlier sensitivity. The control on both parameters, outlier sensitivity and tolerance, is important for commercial systems, as their tuning impacts the relation between
the desired quality, the allowed cost, and the number of participants needed within an MCS system. Moreover, this method is applicable for MCS applications whose readings come from a symmetric distribution.

The thesis is organized as follows: Chapter 2 provides a comprehensive literature review of MCS, in particular its description and architecture, as well as the concept of quality in MCS and relevant works; Chapter 3 provides a primer to the robust MAD-Trimmed Mean, the statistical bootstrap, and the spatiotemporal MCS system model considered throughout this thesis; Chapter 4 introduces the MMTM statistic and consequent MAD-Q and MADBS-Q quality metrics, as well as an analysis of the MMTM statistic; Chapter 5 describes an MCS-specific coverage metric based on either of the quality metrics; Chapter 6 evaluates the applicability and function of the MAD-Q, MADBS-Q, and the MCS coverage metric via simulations; and finally, Chapter 7 concludes and sheds light on future work.
Chapter 2

Mobile Crowd Sensing: Description, Architecture, and Literature Review

The taxonomy and architecture of MCS have assumed different variations as the literature has progressed. Such evolution can be traced since MCS’s nascence in 2010 till recent day in the works: [6, 15–25]. Nevertheless, the most comprehensive architecture defined till now is that in [24], as it assumes a top-down layered approach to MCS architecture. In this section, we provide an overview of MCS as per the architecture described in [24] with modifications based on our review of other works in the MCS literature.

It is important to note that MCS is unique and that “sensing” in MCS is different from the traditional definition of sensing. To acquire a better picture of the sort of sensing in MCS, it is important to look at the factors that led to the rise of MCS. MCS is influenced by various predecessor paradigms such as Wireless Sensor Networks (WSNs), Crowdsourcing, Internet of Things (IoT), distributed, cloud, and edge computing, as well as the advances in hardware that led to the appearance of wearable devices and smartphones over the last two decades [21, 22, 24, 25]. The main element that has completely revolutionized the definition of “sensing” in “Mobile Crowd Sensing” is the presence of the famous “wisdom of the crowd” in addition to human intelligence [21]. In MCS, “Sensing” is not only limited to transduction of physical phenomena or relying on virtual sensors that perform inferences on such measurements, it also involves the intelligence, mobility, and heterogeneity in the sensing process. Sensing in MCS includes using mobile devices as an extended instrument to infer about the environment of the device as well as crowdsourcing human-generated data from the cyber-physical space, which also includes humans’ interactions on social media. This paradigm-shift has made MCS an invaluable tool for public safety, emergency, and crisis response, as well as unlocked a lot of potential applications that range from environmental monitoring to recognizing incidents.
2.1 MCS Layered Architecture

2.0.1 Entities of an MCS System

A Mobile Crowd Sensing (MCS) campaign generally consists of three main entities: an MCS administrator; an MCS system (or platform); and participants [2]. An MCS administrator is an entity that designs and publishes tasks in an MCS campaign. Such an entity is closely related to the management of smart cities and aims to cater to the purposes of the smart city for its efficient operation. An MCS administrator may be interested in data collection about a specific phenomenon at a specific location and time.

The MCS Participants, on the other hand, are members of the crowd who participate in an MCS system to execute MCS tasks set by the administrator, in exchange for a service or a monetary reward (incentives). They can execute tasks participatorily, being actively involved in the task execution, or opportunistically, by passively being present at the time and place at which the task is automatically executed.

Both the MCS administrator and the MCS participants interact together via the MCS system, or platform, through which the campaign is deployed. The MCS system is an automated service that connects the participants to the administrator, recruiting and assigning tasks to them in an automatic manner while benchmarking and evaluating their performance, reliability, and quality. The MCS system, generally, is a centralized point beyond which the majority of the data is collected, processed, and analyzed. Figure 1 illustrates the three entities involved in an MCS campaign.

![Figure 1: General MCS Architecture](image)
2.1 MCS Layered Architecture

The MCS layered architecture, proposed and detailed in [24], consists of four layers: application, data, communication, and sensing. The advantage of such layered architecture is that it permits independence in the development of these layers without sacrificing their dependency upon each other. High level functionalities such as the MCS task and its properties, and the participants as well as their compensation are decided by the MCS administrator in the topmost layer, the application layer. This is followed by the more technical data layer in which aspects relating to the management of the collected data and its processing are performed by the MCS system. The communication layer involves the communication technologies upon which the MCS system and the MCS participants communicate as well as the techniques that allow the timely delivery of the collected data. The activities of the final layer, the sensing layer, take place on the MCS participant’s sensing device. Such a device is usually a sensor-laden internet-enabled mobile device that is casually used by its owner who is asked to, or consents to, performing the MCS tasks as part of their participation in the MCS campaign. Figure 2 illustrates the layered MCS architecture. In each of the following subsections, we shall provide an overview of each layer’s functionality, the entities involved in each layer, and the main elements of which each layer is comprised, related to relevant examples from the literature.

2.1.1 The Application Layer

The application layer involves the high-level aspects of the MCS campaign. In this layer, aspects concerning the task’s nature and the participant involvement strategy are decided. Mainly, the entities involved in the application layer are the end-users of MCS, namely the MCS administrator, the MCS consumer (if any), and the MCS participants. The MCS administrator outlines the design of the MCS campaign that serves their purpose. An MCS
2.1 The Application Layer

Human-in-the-loop
MCS Layered Architecture
Application Layer
Task Scheduling
Proactive
Reactive
Assignment
Centralized
Decentralized
Execution
Single Task
Multi-task
User Recruitment
Voluntary
Incentivized
Selection
User-centric
Platform-centric
Type
Contributor
Consumer
Data Layer
Management
Storage
Centralized
Distributed
Format
Structured
Unstructured
Dimension
Single Dimension
Multi-dimensional
Processing
Pre-processing
Filtering and denoising
Analytics
Machine Learning & Data Mining
Real-time/Online
Post-processing
Statistical
Predictive
Communication Layer
Technologies
Infrastructure-based
Cellular
WLAN
Infrastructure-less
WiFi-Direct
LTE-Direct
Bluetooth
Reporting
Upload mode
Relay
Store and Forward
Methodology
Individual
Collaborative
Timing
Synchronous
Asynchronous
Sensing Layer
Elements
Deployment
Non-dedicated
Dedicated
Activity
Always-on
On-demand
Acquisition
Homogeneous
Heterogeneous
Sampling Process
Frequency
Continuous
Event-based
Responsibility
Mobile Devices
Central Collector
User Involvement
Participatory
Opportunistic
The work of this thesis is about evaluating the quality of participants from a platform-centric point of view.

Ad-hoc topologies are often used in collaborative sensing.
Post processing involves the evaluation of participants.

Collaborative sensing could require different incentivization.
Between collaborating participants, participatory sensing is heavily on-demand.
Both participatory and opportunistic employ context-awareness.
Event-based is often based on context-awareness.
Context-awareness is heavily on-demand.
Participatory sensing in heavily on-demand.

Figure 2: Layered MCS Architecture
2.1 The Application Layer

consumer, if exists, is a party to whom an MCS service is provided. For example, Google Maps, as an MCS system, provides traffic information to its users, the MCS consumers [22]. Additionally, the MCS participants are incentivized and are assigned tasks via the application layer.

At this layer, the first element is the **MCS task**. The design of the MCS task involves three main aspects: scheduling, assignment, and execution. **Scheduling** is divided into proactive and reactive scheduling. **Proactive scheduling** [27–29] gives the freedom for the participants to spontaneously decide to contribute data, where proactive scheduling is generally impractical for traditional sensing scenarios due to its spontaneity and lack of objective. Nevertheless, it is invaluable in emergency situations and in the context of public safety as its inherent serendipity has the potential to capture several aspects of unexpected events, in particular events of emergency, crisis, and disasters. This would allow prompt emergency response, especially in the context of a smart city. **Reactive scheduling** [30–32], on the other hand, an MCS participant is assigned a task then proceeds to accomplish it. Reactive scheduling is more common for MCS application as it is often the case that the objective of the sensing is known beforehand.

The second aspect of task design is the centrality of the **task assignment**, which is either centralized or decentralized. In **centralized assignment** [28, 30–32], a centralized entity, often the MCS platform, is the one that performs the task assignment. In MCS, centralized assignment is more common as the participants are a diverse set of individuals from a crowd. **Decentralized assignment** [27, 29], on the other hand, is a niche in MCS literature as it borders other topics such as Intelligent Transportation Systems and Mobile Social Networks, which are networks formed on the fly [21, 33–35]. The final aspect of task design is the type of **execution**. This is classified into single- and multi-task execution. The MCS participant could be assigned a **single task**, such as the acquisition of a noise sample [3], or they could be assigned **multiple tasks** [5, 36, 37]. This classification can be thought of dedicating the crowd for a specific objective, or utilizing their heterogeneity.
2.1 The Application Layer

The last element of this layer relates to users involved in the MCS campaign from three perspectives: participant recruitment, participant selection, and user type. Out of a crowd of smart device users, a subset of MCS participants are recruited. By being a part of an MCS campaign, participants consent to have their smart devices’ resources consumed to perform MCS tasks. However, such resources are limited. Thus, participants ought to have a good reason, or an incentive, to participate in such an MCS campaign.

Participant recruitment is divided into two types: voluntary and incentivized recruitment. In voluntary recruitment [4], the participants are intrinsically motivated to join the MCS campaign out of a desire to improve the welfare of the community or a similar motivator. Incentivized recruitment [38–40] relies on incentives to extrinsically motivate participants to join the MCS campaign. Incentives have three main forms: a service to be provided, entertainment, or monetary credit. The service provided by Google Maps is an example of an incentive, in exchange for traffic information, participants provide GPS and WiFi fingerprints [41]. Entertainment incentives could involve a form of gamification, for example the authors in [42] and [43] devised gamification as an incentive for MCS (For example, the geographical allocation of Pokèmon in Pokèmon Go could be an attractive incentive). Ultimately, the MCS campaign could rely on providing monetary incentives which involve the crediting of points or money to the participants (e.g., Google Play Credits such as in the Google Opinion Rewards app [44,45]).

While the recruitment process could involve the task assignment process, it is not always the case. Sometimes, the set of participants to whom tasks are assigned needs to be narrowed down to achieve certain criteria. Participant selection [46] is the process in which tasks are assigned to participants. Selection could be “user-centric” (e.g., [47–50]), in which the task is assigned to everyone, by default, and those who successfully execute their tasks and contribute the data are those who are “selected”. The other type of participant selection is “platform-centric” (e.g., [30–32, 51]). In platform-centric selection, the MCS system selects the subset of participants that are in the best interest for the MCS campaign. Such
“best interest” could be decided by specific metrics, such as coverage, participant distribution, participant’s reputation, or reading’s quality, to name a few. Despite the fact that MCS consumers are indeed a part of the MCS campaign, the majority of MCS literature - including this thesis - do not address the presence of “consumers” and use the words user and participant interchangeably. This is due to the fact that the consumer is an end-user whom the MCS administrator aims to service, which is beyond the scope of MCS.

2.1.2 The Data Layer

The data layer is responsible for the management and processing of data. An MCS campaign can be promptly designed to deploy the functionality of the data layer in the cloud or closer to the network edge [52]. Most data processing, mining, and inference is done by the MCS system over the MCS participants’ contributions.

Data management deals with the storage, format and the dimensionality of the collected data. Data storage could be maintained in a centralized cloud location when the data requires significant processing [4, 6, 53], or it can be stored in a distributed fashion [21, 22]. In terms of format, data could be structured or unstructured. Structured data is data that follows a consistent format whose processing is straightforward. Tables are examples of structured data. Unstructured data, on the other hand, require significant effort to be analyzed. Images, for example, are a form of unstructured data. Objects inside an image are not straightforward to detect, as such detection often requires the usage of high-level techniques or even learning. Dimension refers to the number of attributes in the contributed data, which could be a single attribute, i.e., the value of a sensed quantity, or multiple attributes, e.g., contextual information in addition to a reading.

Processing is a fundamental element of the MCS campaign, as it is the means by which data becomes useful, and by which inferences - and thus insights - about the sensed phenomena are obtained. Pre-processing [54–57] refers to the preliminary treatment of the contributed data. The purpose of the pre-processing step is to ensure that the data is
consistent in later steps. The contributed data is reported by different devices with different sensors. The inherent heterogeneity of the crowd does not allow direct processing of the data. The preprocessing step provides a common ground upon which all contributions could be considered. For example, raw data might be reported directly to the following steps without any preprocessing, but it can also be filtered, denoised, and normalized to ensure consistency and minimize storage space [22]. Analytics is the core process by which inferences are mined and extracted [21, 22, 25]. Machine Learning techniques [26, 54, 55] could identify patterns in contributed data, however, the majority of these techniques assume that the data is readily present, which makes them inappropriate for real-time applications. Real-time and online [46, 58, 59] techniques, on the other hand, analyze the data as soon as it is acquired. Such analysis requires high responsiveness (i.e., short delay) and significant computation.

Post-processing refers to further insight that can be obtained from further analysis of the data. While seemingly similar to analytics, the difference between post-processing and analytics is in terms of the locality of the processing. Analytics are employed to obtain inferences that relate to the phenomena on a global scale. Post-processing, on the other hand, relates to the phenomena on a local scale; the difference between analytics and post-processing lies in the contrast between global and local [9]. Sensed trends can be recognized, aggregated, and analyzed to obtain global insights. Post-processing can be in the form of statistical analysis in which correlations, averages, among other statistical features are obtained. It can also be predictive, seeking to forecast or predict a trend. However, we would like to introduce a new aspect to the architecture in [24], and it is that post-processing also involves managerial aspects that relate to user selection. For example, the trustworthiness of an MCS participant [60], the quality of the readings provided [13, 61], as well as the incentive reward paid to the participants [62], all rely on post processing of the contributed data [63]. We would like to emphasize that the work of this thesis overlaps both the application and the data layer in the aspects of user selection and post-processing from
2.1 The Communication Layer

a platform-centric perspective.

2.1.3 The Communication Layer

The communication layer is responsible for the delivery of the MCS contributions among the involved entities: the MCS participants and the MCS system. In it, wireless communication technologies and data delivery methodologies are outlined.

Technologies in the communication layer mainly refer to the wireless technologies that establish communication links between the MCS participants and the platform, and the MCS participants themselves. It is important to note that network topology plays a role in whether a technology is infrastructure-based (centralized) or infrastructure-less (ad-hoc) [25,34,35].

Infrastructure-based cellular technologies [64] such as LTE, 4G, and the nascent 5G, are generally accessible and timely making them reliable for real-time applications. 5G, in particular, is perceived as a true enabler of MCS, as the prospective employment of Software-defined Networks (SDNs) and the millimeter wave technology would greatly improve the performance of cellular networks [24]. Moreover, the low latency requirements for tactile internet [65] would enhance the reliability of real-time MCS applications. However, a disadvantage of the cellular network is the associated cost [10]. Cellular data traffic is costly as it requires a cellular subscription. Moreover, this restricts the majority of the sensing done to cellular-enabled devices, which are mostly smartphones, cellular-enabled tablets, and a niche selection of smartwatches (e.g., Apple Smartwatch). It is important to note here that the majority of MCS literature treats cellular connectivity as the defacto MCS communication technology [63,64,66].

Wireless Local Area Network (WLAN) [59,67] is a cheaper infrastructure-based alternative to the cellular network. By utilizing WLAN access points, mobile devices can achieve connectivity at the cost of longer delay. The reliance on WLAN for the transmission of contributed data is often suitable for delay-tolerant applications as WLAN coverage is not
always guaranteed, unlike cellular coverage.

There are also other communication technologies that could be employed such as the long range LoraWAN, the ultra narrowband Sigfox technology, Zigbee, and Bluetooth. Unfortunately, very few devices support them as of now, which is their usage in MCS literature is scarce. Nevertheless, it is expected that consumer devices will eventually support these technologies, enabling their incorporation in MCS.

Infrastructure-less, or ad-hoc, communication is often peer-to-peer and proximity-based (on link strength, not distance). Such infrastructure-less topologies are similar to Mobile-WSNs, where MCS participants could collaborate with each other over the sensing process or the delivery of their contributions [68]. Mostly, device-to-device technologies such as WiFi-Direct, LTE-Direct, and Bluetooth are used. There are numerous ad-hoc topologies that could be assumed (clusters, hierarchies, etc), and MCS participants can assume different roles in such topologies (which could influence the incentivization and the selection process) [38,40].

Reporting is concerned with the management of data transmission between the MCS participants and the MCS system, and it has three main elements: upload mode, reporting methodology, and report timing. Upload mode refers to whether the data is transmitted as soon as it is received by the MCS participant (i.e., real-time MCS) or if the data is stored then forwarded (i.e., delay tolerant MCS) [25].

Reporting methodology refers to the behavior of the MCS participant in the vicinity of nearby devices, and how is the sensing executed with respect to others. MCS participants could pursue the execution of MCS tasks individually, or they could execute them collaboratively [68,69] with other MCS participants. Collaborative sensing heavily overlaps with the earlier mentioned infrastructure-less communication technologies.

While reporting timing indeed spans both the communication and sensing layers, we believe that the authors [24] were sound in choosing to represent it in the communication layer. This is due to the fact that transducing natural phenomena (or sensing), or computing
high level inferences locally consumes less energy than wireless communication; making it cheaper to sense than communicate. Its position is properly under the communication layer due to the more significant impact of timing on communication rather than on sensing. Nevertheless, we believe that the definition of reporting timing that has been introduced in [24] should be placed under collaborative sensing as it involves collaborating with other participants, as reporting timing for individuals is already included under sampling process in the sensing layer (to be explained in the next subsection).

2.1 The Sensing Layer

The sensing layer is the main driver behind MCS, as the disruption that permitted MCS to appear was the prevalence of sensors in mobile devices. This layer is concerned with two main aspects that describe the sensing process: the sensing elements, and the sampling process [24]. MCS participants are the main entity involved in the sensing layer.

**Sensing elements** refer to the sensors that perform the sensing process, and they are treated in this layer from three perspectives: deployment, activity, and acquisition.

**Sensing deployment** refers to the types of sensors deployed. While most MCS participants carry sensor-laden devices, some phenomena cannot be sensed except by means of a dedicated sensing device. For example, a smartphone comes equipped with microphones, camera, proximity sensors, luminescence sensors, magnetic field sensors, accelerometers and gyroscopes (i.e., inertial measurement units), and others [21, 24, 25, 64]. An MCS campaign can deploy these “omni-sensing” non-dedicated devices to sense a specific phenomena that can be observed using the plethora of sensors on a smartphone. Nevertheless, some phenomena cannot be sensed by these sensors and require the deployment of specific sensors. For example, a food gluten sensor is a standalone sensor that is dedicated to sensing gluten [24]. Indeed, such sensors lack connectivity and computational ability, however, they are smartened by being connected to an accompanying smart device. Although this type

---

Indeed, there are MCS applications in which sensing is more time-critical than communication (e.g., self-driving vehicles), however the majority of MCS applications are complacent with regards to timing.
of sensing is regarded as a part of MCS, very few works in the MCS literature have been
catered to them.

**Sensing activity** refers to the behaviour of the sensors, is the sensing always-on, or is it on-demand. While this aspect overlaps with the user involvement aspect (discussed under sampling process), it refers to the sensing behaviour of the sensing elements themselves rather than the process. **Always-on sensors** sense continuously while requiring little energy for their operation. An Inertial Measurement Unit (IMU) is an example of such a sensor that can be used for dead reckoning positioning [41, 67], as well as the detection of the context of the device user’s activity [22, 54, 70] (i.e., idling, walking, running, etc.). Such sensors allow the detection of the context, the awareness of which is another class of MCS applications on its own [54, 70].

**On-demand sensors** are event-triggered, or context-aware. Such sensors detect a conditional event, or a context, that needs to be satisfied prior to sensing. Sensors in this category often consume a lot of resources. A smartphone’s camera [4], for example, senses the light that is then processed into an image. Objects within such an image cannot be detected right away, and camera features that rely on such object detection, such as face recognition, or depth-of-field simulation cannot be performed immediately; such features need a degree of local processing. Similarly, the GPS sensor samples the position periodically at a low-power state, but when it is employed for navigation its operation mode changes, consuming a lot of energy [71].

The data produced by sensors could assume different forms. **Acquisition** refers to the form of data that is acquired from a sensor. For example, an accelerometer can measure the acceleration in three dimensions. These measurements can be provided as a set of **homogeneous** raw data. However, some inference could be done locally on these devices to detect patterns. For example, different modes (walking, cycling, driving) can all be detected by means of an accelerometer and a gyroscope. Such modes are also sensed data, however the type of sensor that senses them is based on a **heterogeneous** set of sensors, making it
2.1 The Sensing Layer

A virtual sensor [21, 22].

The sampling process involves the lower-level description of the sensing task, particularly in terms of Sampling frequency, execution responsibility, and the user involvement [24].

Sampling frequency refers to how often the sensing task is executed. It is synonymous to the sampling rate employed in signal processing, but it also extends to include non-signal-based such as surveys and statistical sampling. Generally, the term “sensing cycle” is comprehensive to both signal- and non-signal-based crowdsensing tasks [38]. Such sampling could be continuous, i.e., sensing tasks are executed at a regular rate. Once the task is assigned, contributions are collected until certain criteria is met. Such criteria could be the achievement of the required sensing goal, the exhaustion of a specific set of resources, a threshold on the incentives paid to the user, etc. There is a trade-off between energy consumption and accuracy in selecting the sampling frequency. Oversampling could induce needless redundancies in the data, ultimately wasting both energy and storage. On the other hand, undersampling could lead to aliasing among other problems that limit the acquired benefit from the data [24]. Sampling can also be event-based, which is triggered by the detection of an event, making it adaptive to the context of the participant or the platform.

Sampling responsibility refers to the entity that triggers the execution. Such entity could be the MCS participants themselves, or it could be the MCS system. By having a task assigned, the MCS participants receive instructions in the task description. This refers to the definition of the task, which varies from MCS campaign to another. For example, an MCS campaign might use the 4W1H framework [6], however, the task execution could be pushed by the MCS system, with no intervention, with regards to the execution responsibility, from the MCS participant. It is important to note that responsibility is distinct from user involvement. Some tasks could be triggered by the MCS system, yet require the active participation of the user for the execution. It is rather about whom is responsible for managing the collection of the data [24].

User involvement has been the popular type of classification in the majority of MCS
literature. User involvement can be either participatory, requiring active intervention from the user for proper execution, or opportunistic, in which the sensing task is done passively when the opportunity is presented [21, 24, 25, 72]. There is a strong relation between user involvement and on-demand sensing element activity, as shown in Figure 2, as the execution of participatory sensing tasks requires human intelligence to be involved. It is argued that human intelligence is centric to participatory sensing, as it is the aspect upon which the wisdom of the crowd [21], the characteristic that makes MCS superior to its predecessors, transpires. For example, a human might be asked to move around the city, take out the smart device and take a photo [4], or record a noise sample [3]. On top of that, the majority of participatory sensing involves time and resource consuming tasks, thus demanding stronger incentives [38].

Opportunistic sensing [21, 57, 73], on the other hand, is an approach in which sensing is done passively. As a result, opportunistic sensing often employs context-awareness to evaluate the conditions required to execute the task [21, 54]. For example, a task could be restricted by a geofence within which the sensing task is executed opportunistically [4]. The advantage of opportunistic sensing is that it involves low user involvement [25], and thus is cheaper than participatory as less incentives, in particular monetary incentives, need to be provided to the participant. However, it could compromise the privacy and security of the participants or consume their resources [73, 74]. Thus, opportunistic tasks need to be designed taking privacy, security, and resource consumption into consideration.

2.2 Quality in MCS

In MCS literature, quality can be loosely defined as the reliability of data collected through an MCS campaign to the campaign’s objective, however such definition is subjective. Indeed, numerous authors define quality differently as: timeliness, accuracy, and confidence [72];

- Timeliness, accuracy, and confidence [72].
• The radius of a data pollutant’s confidence interval, which is rigorously estimated using maximum likelihood [75].

• A given quantity, such as the resolution of an image [76].

• A participant’s ability to cover a Point of Interest (PoI)’s location and the importance of that PoI, which is computed as a weighted average [77].

• A sensor’s accuracy [71].

• Reputation rank, a value derived based on trustworthiness\(^3\) that is validated by peer participants [78].

• Metrics that evaluate the quality of a participant’s contribution, such as accuracy for numeric readings or measurements of an image’s blur, lighting, and haze [43].

• Extent of cross-validation, in which two sets of participants are recruited to collect data where one collects data and the other validates [79] (similar to [78], but in a more rigorous manner).

2.2.1 Related Work: Quality in MCS

Various works have investigated quality, mainly for the purpose of reducing costs and providing a means to see the impact of incentives. Throughout this subsection, we investigate the closest works in the MCS literature to our work.

MeLoDy, an incentive mechanism that is proposed in [49], is based on the long-term performance of a participant in satisfying certain criteria dictated by each task. With regards to quality, the authors introduce a framework for quality inference based on treating the MCS participant a linear dynamical system. Quality is assumed to vary with every sensing cycle, and it is modeled as a latent (unobserved) time-series that is to be estimated. Quality, as a random variable, has a transition probability between every two sensing cycles in the series.

\(^3\)Trust [78] is another term, like quality, that is subjectively used and similarly needs a rigorous definition.
The estimation, and the *prediction* of participant quality is done by learning the time-series by inferring the posterior transition probability using Expectation-Maximization. This work treats quality from a user-centric point of view.

The authors in [50] suggest a quality-based incentive mechanism that employs expectation maximization for the estimation of quality. The range of the sensed quantity is initially quantized into discrete bins, in which an MCS participant’s contributed data falls in. Then, for each user, an effort matrix is defined whose entries are defined as the probability that a participant’s reported value matches the ground truth. Indeed, the ground truth cannot be observed which makes the values in this effort matrix latent (i.e., unobservable, or hidden). The procedure then is to employ expectation maximization to infer each participant’s effort matrix and estimate the probability that the reported value is correct. Quality is then defined as the mean of the estimated effort matrix. Similar to the work in [49], this work treats quality from a user-centric point of view.

A dynamic quality-enhanced incentive mechanism for MCS tasks is presented in [43]. Their work treats quality from a user-centric view too, as it is defined used two main participant-based quantities: completion ratio, and quality indicator. Completion ratio refers to the proportion of task “aspects” that have been contributed out of all needed aspects. The authors mention that Completion ratio is a global metric of task completion that could be extended to the system as a whole, for all participants, called “whole completion ratio”. The quality indicator is a quantity that is defined based on the data. For numerical data, the quality indicator, $QI_u$, is defined as:

$$QI_u = \frac{\sum_{i=1}^{\mid D_u \mid} \left( 1 - \frac{d_i - G(d_i).center}{MaxDiss(G(d_i))} \right)}{\mid D_u \mid}$$  \hspace{1cm} (1)$$

where $G$ is a group of an individual participant’s reported readings. It can be seen from Equation 1 that the quality indicator is defined as a weighted average of how far is each reading from the center of all readings by a single participant, divided by the maximum
distance for normalization. This definition is very close to the definition of a normalized standard deviation, however the centrality estimate is not clear. The quality a specific participant then is defined as the product of the completion ratio and the quality indicator.

The authors in [80] have proposed a statistical quality model for evaluating the overall quality of the participants’ contributed data. Their definition of quality defines quality as a linear regression of four variables: sensing duration, coverage, user experience, and device context. By duration, they mean the entire sensing duration from the publishing of the task to fulfilling it. Coverage is generally defined as a percentage in their scheme, yet it is unclear what that percentage signifies. User experience is the proportion of operations (or subtasks) within a task that a participant can successfully execute, which is a quantity that is specific to participatory sensing. Device context refers to the device’s current state, as previously explained in the subsection regarding the sensing layer.

The work in [79] attempted to ensure quality by means of recruiting a cross-validating crowd, to check the crowdsensed data’s validity and rate the participants [79]; by recruiting participants based on their social profiles and technical expertise, for a crowdsourcing campaign on top of a crowdsensing campaign. The framework and mechanism proposed in [79], while mildly susceptible to the well-known mainstream bias, are capable of increasing data quality by means of using the cross-validating crowd, to update the crowdsensed data distribution and evaluating the participants’ contributions, and thus affecting the received incentives. The contribution of [79] is interesting and practically applicable, however it does not cater to situations where the SD problem is present, in addition to it relying on experts’ subjective opinions in the evaluation of the crowdsensed data.

Authors in [81] tackle the same problem as our work, however they tackle it from the perspective of the task, rather than the perspective of the participants and their reported data, and with the assumption that quality can only be ensured by increasing the number of participants. The approach proposed in [81] aims to solve the problem of selecting prices in a posted price model using chance-constrained optimization, to minimize the total cost while
maintaining robustness. However, their approach requires an abundance of data as it necessitates the presence of multiple participants performing a task, which limits its applicability to scarce scenarios.

While most works are interesting, they are treating quality from an individual, user-centric, point of view for the purpose of incentive-based quality control. The work done in this thesis is distinct with regards to that aspect as it treats quality from a platform-centric point of view for the sake of number of participants-based quality control, particularly for scarce scenarios. While the objective of both user-centric and platform-centric quality control approaches is the same, i.e., reducing the cost incurred by the system and increasing the sensed quality, this work treats quality from a holistic, point of view, rather than defining for each individual participant. Moreover, this work provides a general definition for quality that is application-agnostic, absolute, prior-independent, and versatile.
3.1 MAD-Trimmed Mean

For a sample, \( X = \{x_1, x_2, \ldots, x_N\} \), the dispersion of the population can be estimated via the standard error. However, it is recommended for a large sample set, and cannot be used for smaller samples, as the sample standard error is a non-robust measure of dispersion which is vulnerable to outliers. The vulnerability of a statistic, \( \theta \), to outliers is expressed in terms of the breakdown point, \( b_\theta \). The breakdown point is the proportion of outliers within a sample at which the statistic becomes blind to outliers and not robust anymore. The standard error, \( \bar{\sigma}_{\bar{x}} \) is based on the mean, \( \bar{x} \), which has a breakdown point \( b_{\bar{x}} = 0 \), at which both measures, \( \bar{\sigma}_{\bar{x}} \) and \( \bar{x} \), are vulnerable to outliers unless the sample is large enough [82]. Therefore, when the sample is small, we resort to the usage of robust statistics, such as the sample median and the Median Absolute Deviation (MAD), which both have a breakdown point of \( b_{\tilde{x}} = 0.5 \); a point beyond which the outliers are populating the sample itself. In practical systems, it is non-realistic to have more than 50\% outliers. This implies that the MAD is a reliable practical measure of dispersion. The MAD is defined as the median of the absolute deviations from the sample median, expressed as:

\[
\text{MAD} = \text{median}\{ |x_i - \tilde{x}| \} \tag{2}
\]

where \( x_i \) is the \( i^{th} \) sample in \( X \), and the median is denoted by \( \tilde{x} \).

The MAD observes the deviations from the median, unlike the standard error which considers the square of the deviations from the mean. However, they are related as the MAD is a robust consistent estimator of the standard deviation [82], whereas the standard error is a non-robust consistent estimator of the standard deviation. The MAD can be used
3.1 MAD-Trimmed Mean

to estimate the standard deviation as:

\[ \hat{\sigma}_{\text{MAD}} = \frac{1}{\Phi^{-1}(3/4)} \text{MAD} = \frac{1}{1.4826} \text{MAD} \mid_{f(x)=N(\mu,\sigma)} \]  

(3)

where \( \hat{\sigma}_{\text{MAD}} \) is the MAD-based standard deviation estimator, \( f(x) \) is the probability distribution followed by the population, \( N(\mu,\sigma) \) is a normal distribution centered at \( \mu \) with a spread of \( \sigma \), and \( \Phi^{-1}(3/4) \) is the quantile function (which is the inverse cumulative distribution function) at 75%.

The MAD covers the median distance from the sample median, \( \tilde{x} \), which is located at the center having it span the portion between 25% and 75%, and thus the reason why the quantile function is evaluated at that point. For any symmetric distribution, the MAD covers 50% spanning from the left to the right of \( \tilde{x} \). For any normal distribution \( N(\mu,\sigma) \), \( \Phi^{-1}(3/4) = 1.4826 \). As a result, the relation of the MAD and the standard deviation was derived, nevertheless this relation is valid for any well-defined distribution \( f(x) \) [82].

![Population with Abnormal Outliers (lower and upper).](image)

The MAD inherits the robustness of the median, which makes it useful for the detection and the removal of outliers. However, the definition of outliers is vague, and depends on the application in hand. For MCS systems, physical sensor measurements tend to follow an even
3.1 MAD-Trimmed Mean

symmetric distribution (i.e., defined by an even function), such as the normal distribution. The normal distribution includes *normal* outliers that rise from the extremities of the distribution. However, there are *abnormal outliers* which lie far outside the three sigma range, defined by the three sigma rule \((\pm 3\sigma)\) which is valid for all symmetric unimodal distributions [83]. In case the distribution is an even multimodal distribution, other criteria would be required for the definition of expected outliers, such as those based on the interquartile range. These abnormal outliers come from an *unexpected phenomenon* that is not modeled by the normal distribution such as the tail modalities in Figure 3. The distribution in Figure 3 is a multimodal distribution which is a superposition of abnormal distributions representing an unexpected phenomenon and an expected distribution of a sensed physical measurement.

In an MCS system, these abnormal outliers need to be isolated as they pose a hindrance that impairs the MCS system. The MAD allows the removal of outlier samples present in \(X\), producing a MAD-trimmed sample, denoted \(X_{\text{MAD}}\), where only values from within the range of the normal distribution are considered [84], expressed as:

\[
X_{\text{MAD}} = \{ X : x_j \in [\bar{x} \pm \lambda\text{MAD}] \} \tag{4}
\]

where \(\lambda\) is how many multiples of MAD far from the median are the non-outlier samples. The product \(\lambda\text{MAD}\) can be related to the consistent estimation of the standard deviation \(\delta\text{MAD}\), to define it as multiples of \(\sigma\) deviations from the mean, where \(\delta = \lambda\Phi^{-1}(3/4)\). \(\lambda\) is an important parameter as it controls the range for which values, even if outliers, are indeed in \(X_{\text{MAD}}\).

Nevertheless, normal outliers are *expected*, unlike *abnormal outliers* which are beyond the three-sigma range. Thus, for the administrator to ensure proper estimation of the true value, \(\mu\), that considers normal outliers, a \(\lambda\), such as \(\lambda = 4\) corresponding to the three sigma range for example, could be selected. As a result, only values that belong to the interval

\[\text{Normal here refers to expected outliers coming from a distribution itself, not necessarily a normal distribution.}\]
\[ \bar{x} \pm \lambda \text{MAD} \] will be considered in \( X_{\text{MAD}} \). Therefore, the MAD-mean can be defined as:

\[
\bar{x}_{\text{MAD}} = \frac{1}{N - N_{\text{MAD,o}}} \sum_{i=1}^{N - N_{\text{MAD,o}}} x_{\text{MAD},i}
\]

\[
= \frac{1}{N_{\text{MAD}}} \sum_{i=1}^{N_{\text{MAD}}} x_{\text{MAD},i}
\]

(5)

where \( N \) is the sample size, \( N_{\text{MAD,o}} \) is the number of outliers outside the \( \bar{x} \pm \lambda \text{MAD} \) range, \( x_{\text{MAD},i} \) is the \( i \)th element in the MAD-trimmed sample, \( X_{\text{MAD}} \), and \( N_{\text{MAD}} = |X_{\text{MAD}}| \). Algorithm 1 shows the steps in which the MAD-mean is computed.

**Algorithm 1** Computing the MAD-mean.

**Input:** A sample: \( X = \{x_1, x_2, ..., x_n\} \)

**Output:** MAD-mean: \( \bar{x}_{\text{MAD}} \)

1. Initialize \( \lambda \)
2. for all \( x_i \) do
3. if \( x_i \notin [\text{median}(X) \pm \lambda \text{MAD}] \) then
4. \( X_{\text{MAD,o}} = \text{append}(x_i, X_{\text{MAD,o}}) \)
5. else
6. \( X_{\text{MAD}} = \text{append}(x_i, X_{\text{MAD}}) \)
7. end if
8. end for
9. return \( \bar{x}_{\text{MAD}} = \text{mean}(X_{\text{MAD}}) \)

Notice that the value of \( \lambda \) provides a degree of freedom for system administrators to decide the range of the measurements and the consideration of outliers. \( \lambda \) is a parameter that controls the degree of outlier tolerance. For absolute outlier intolerance, a \( \lambda = 2 \) is sufficient as the resulting \( \delta < 3 \) is within the 3-\( \sigma \) range. On the other hand, for outlier tolerance, a choice of \( \lambda \geq 4 \) is recommended as \( \delta > 3 \) is beyond the 3-\( \sigma \) range. Since \( \lambda \) has no upper bound, it needs to be carefully selected by the administrator to avoid including abnormal outliers. However, the breakdown point of the MAD-mean, \( b_{\bar{x}_{\text{MAD}}} \), is impacted by the choice of \( \lambda \); in particular, \( b_{\bar{x}_{\text{MAD}}} < 0.5 \) for \( \lambda > 4 \), as the MAD-mean is based on the median. The MAD-based outlier detection is a technique that is especially useful for
small sample sizes, which makes it useful for the small-sample scenarios present in MCS systems’ spatiotemporal cells [85]. The MAD estimator is a very good option that is further characterized by low complexity. Its parameter $\lambda$ provides an interesting trade-off between accuracy and robustness, that is very important for commercial systems. Later simulations show deeper considerations of this trade-off.

3.2 The Non-Parametric Bootstrap

The non-parametric bootstrap [86] is a population agnostic method, that allows the construction of sample distributions without prior assumptions about the population’s distribution. Its numerical nature allows it to achieve its target by means of sampling with replacement a large number of times, $B$. The non-parametric bootstrap resamples the original sample $X = \{x_1, x_2, \ldots, x_N\}$, generating $B$ resamples, $X_b = \{x_{b1}, x_{b2}, \ldots, x_{bN}\}$ where $x_{bi}$ can appear more than once in $X_b$, and $x_{bi}$ is a random variable that samples uniformly from $X$. From each bootstrapped sample set $X_b$, a bootstrap statistic $\theta_b$ can be acquired. To construct the sample distribution of a statistic $\theta$, each $X_b$ is employed to compute $B \theta_b$ bootstrap statistics, collected in a vector of bootstrap statistics $\theta^* = \{\theta_1, \theta_2, \ldots, \theta_B\}$, whose histogram represents the sample distribution.

Despite its numerical complexity for a large $B$, the bootstrap is useful for crowd sensing applications where the scenario is sparse [1]. It is of particular usefulness for small-sample cases due to the fact that its fair resampling has a low probability of selecting a homogeneous sample (i.e., all samples in $X_b$ being exactly the same); which is obtained as:

$$P(X_b = \{x_i, x_i, \ldots, x_i\}) = [1 - (1 - 1/N)^N]^N \tag{6}$$

which is the probability of a binomial case. Inspecting the Equation (1), out of $B$ resamples, only 3.45% will be extremely biased for a sample of size $N = 8$.

The usefulness of the bootstrap in obtaining the quality of small-sample scenarios comes
from the presence of outliers. Since resampling is uniform, the outliers’ presence is promoted
to have a probability of $1/N$, like any other sample $x_i$ in $X$. Further analysis of the binomial
probability shows that the probability of an element’s inclusion is:

$$P(x_i \in X_b) = 1 - \left(1 - \frac{1}{N}\right)^N$$

which converges for a large $N$ to 67%, i.e., each sample is present in 67% of the $B X_b$ sample
sets.

Furthermore, since $B$ is a sufficiently large number, the properties of the Central Limit
Theorem (CLT) are also applicable to the $B \theta_b$ statistics obtained. This property is of par-
ticular usefulness in the discussion of central measures of tendency (mean, median, mode,...
etc), as it causes the bootstrap for the mean to follow the normal distribution. However, due
to the robustness of the MAD-mean, the resulting distribution is inherently multi-modal,
even if it is seemingly normal, as the employment of the median in the MAD-mean’s trim-
ing process generates multi-modalities. It limits the set of medians to be selected to a
discrete number of medians (for an odd $N$ there are $N$ possible medians, for an even $N$
there are $N^2 - \sum_{i=0}^{N-1} i$ possible medians).

3.3 The Bootlier

The Bootlier, a graphical tool developed in [87], exploits the outlier promotion flaw in the
bootstrap to detect outliers by computing the difference statistic between the mean $\bar{x}$ and the
trimmed mean $\bar{x}_k$. The trimmed mean is a robust estimate of centrality with a breakdown
point $b_{\bar{x}_k} = k\%$. In the Bootlier, the difference between the mean and the trimmed mean,
$\bar{x} - \bar{x}_k$ is bootstrapped to construct a sample distribution. The histogram of the resulting
sample distribution is called a bootlier plot, which shows the impact of outliers present in
a sample. The authors in [87] investigated the bootlier’s multimodality and smoothness,
and developed “the bootlier index” as a measure to quantify its smoothness. However, their
work heavily depends on human intuition for assessing the quality of a sample, and for the
detection of outliers, in addition to the lack of a straight metric for the evaluation of a sample
set’s quality.

We analyzed the bootstrap plot, by considering an infinite sample size \((N \to \infty)\), and found
that its ideal reference is in fact the distribution \(N(0, 0)\), which is best described as the dirac
delta impulse distribution [88]. We also found that the bootstrap, from a signal processing
perspective, is a superposition of leaking impulses, which are very distinct when an outlier is
present as it introduces another impulse at \(x_o - \bar{x}_k\), where \(x_o\) is an outlier sample and \(\bar{x}_k\) is
the trimmed mean of the sample. The source of leakage is the variation within the original
sample, which causes the resulting sample distribution, i.e., the bootstrap, to be smooth. The
bootstrap statistic hides the resulting multimodalities which combine to form a seemingly
normal distribution around 0, that should ideally - for an absolutely perfect sample - be an
impulse. Chapter 4 introduces the Mean-MAD Mean-Trimmed Mean (MMTM) statistic and
discusses the modalities in its distribution based on the trimmed mean and the MAD-mean.

3.4 MCS Spatiotemporal Model

In order to achieve efficient characterization of an area of interest under an MCS system,
it needs to be appropriately divided into geofences or cells to which MCS participants are
assigned. In addition to such a spatial division, a temporal division is also required for proper
real-time sensing. These divisions in space and time can be represented as a spatiotemporal
diagram as shown in Figure 4, based on that conceived in [7]. An MCS administrator
could divide the spatiotemporal space in a manner that satisfies the objective of the system
while being consistent with the spatial and temporal versions of the Sampling Theorem [89].
MCS Participants are assigned tasks according to their availability, for example, if the MCS
administrator requires temperature values to be sensed by MCS participants, the \(m^{th}\) cell
3.4 MCS Spatiotemporal Model

will have a set of readings, $X_m$:

$$X_m = \{x_{m,1}, x_{m,2}, \ldots, x_{m,N_m}\}$$  \hspace{1cm} (8)

where the reading obtained by the $i$th participant is modeled as a random variable $x_{m,i}$, and $N_m$ is the number of participants who executed the task in the $m$th spatiotemporal cell. Each cell corresponds to a 3-tuple $(a, b, c)$, where $a$ maps to the $x$-location, $b$ maps to the $y$-location, and $c$ maps to the $c$th sensing cycle (temporal location).

![Figure 4: Spatiotemporal Diagram](image)

The true value of the sensed quantity in the $m$th cell, $\mu_m$, is estimated from the sample obtained in Equation (8) by computing the mean, $\bar{x}_m$:

$$\hat{\mu}_m = \bar{x}_m = \text{mean}(X) = \frac{1}{N_m} \sum_{i=1}^{N_m} x_{m,i}$$  \hspace{1cm} (9)
The standard deviation for the $m^{th}$ cell can be estimated by means of the sample error:

$$\hat{\sigma}_m = \sqrt{\frac{\sum_{i=1}^{N_m} (x_{m,i} - \hat{\mu}_m)}{N_m - 1}}$$  \hspace{1cm} (10)$$

However, this is only valid when the sample is large enough that $N_m \geq 30$ samples are suitable to represent the population. At such a size, the CLT is viable as the combination of all $x_{m,i}$ random variables would ultimately conjure a normal distribution [90]. While this assumption is very useful, it is not the case when $N \ll 30$, when there are barely enough measurements within a spatiotemporal cell. The problem, then, is of the SD scale. An MCS system should be capable of infering as much as possible under the stringent conditions of SD. Eqs. (9, 10) become less accurate as the presence of $N_{m,o}$ outliers, $x_{m,o}$, would throw off the estimation.

Outlier values, $x_{m,o}$, can be viewed as an estimation that is offset from $\mu_m$:

$$x_{m,o} = \mu_m + \Theta_{m,o}\sigma_m$$  \hspace{1cm} (11)$$

where $\Theta_{m,o} \in \mathbb{R}$ is the Outlier Deviation Factor (ODF), which we define as the multiple of standard deviations, $\sigma_m$, the $o^{th}$ outlier value, $x_{m,o}$, is far from the mean, $\mu_m$. This allows the expression of Equation (11) as:

$$\hat{\mu}_m = \mu_m + \bar{\Theta}_m\sigma_m$$  \hspace{1cm} (12)$$

where $\bar{\Theta}_m = \sum_{o=1}^{N_o} \Theta_{m,o}/N_o$ is the Average Outlier Deviation Factor (AODF).

The mean, $\bar{x}_m$, can then be rewritten as:

$$\bar{x}_m = \frac{1}{N_m} \sum_{i=1}^{N_m} x_{m,i} = \frac{1}{N_m} \left[ \sum_{i=1}^{N_m-N_{m,o}} x_{m,i} + \sum_{j=N_{m,o}}^{N_m} x_{m,j} \right]$$

$$= \frac{N_m - N_{m,o}}{N_m} \hat{\mu}_m + \frac{N_{m,o}}{N_m} (\mu_m + \bar{\Theta}_{\bar{x}_m}\sigma_m)$$
\[
\hat{x}_m = \hat{\mu}_x + \frac{N_{m,o}}{N_m} (\mu_m - \hat{\mu}_x) + \frac{N_{m,o}}{N_m} \bar{\Theta}_m \sigma_m
\]

(13)

where \(\hat{\mu}_x\) corresponds to the mean’s estimate of the true value, \(\mu\), and \(\bar{\Theta}_x\) corresponds to the average outlier deviation factor of the mean.

The second and third terms of Equation (13) correspond to the errors induced by the presence of the outliers. The second is a sample size error due to the presence of outliers that inflates the denominator \(N_m\), while the third is a deviation error due to distance between the outliers and \(\mu\). The third term contributes the majority of the error in the non-robust mean.

Thus, the error of the mean, denoted \(\xi_{\bar{x}_m}\), can be defined as the distance between the mean, \(\bar{x}_m\), and the true value, \(\mu_m\), where:

\[
\xi_{\bar{x}_m} = \bar{x}_m - \mu_m = (\hat{\mu}_x - \mu) + \frac{N_{m,o}}{N_m} \left[ (\mu_m - \hat{\mu}_x) + \bar{\Theta}_x \sigma \right]
\]

(14)

The errors of the trimmed mean \(\bar{x}_k\) and the MAD-mean \(\bar{x}_{\text{MAD}}\) can be derived in a manner similar to that employed in Eqs. (13, 14).

Equation (14) can be rewritten as:

\[
\xi_{\bar{x}_m} - (\hat{\mu}_x - \mu) = \frac{N_{m,o}}{N_m} \left[ (\mu_m - \hat{\mu}_x) + \bar{\Theta}_x \sigma \right]
\]

(15)

indicating that for a more accurate estimation, the errors on the right hand side need to be minimized. This can be achieved by two options: (a) reducing the proportion of outliers, \(N_{m,o}/N_m \to 0\), which translates to outlier removal, or (b) increasing the sample size by taking \(N_m \to \infty\). The latter case is when the data is large-scale and CLT is applicable, while the former case is the problem faced under SD. In SD, the proportion of outliers cannot easily become zero since \(N_{m,o} \geq 1\) would render the proportion \(N_{m,o}/N_m\) significant. Due to the
nature that the MCS data comes from participants whose sensors are different, prone to error, or even maliciousness, its susceptibility to outlier errors is more significant when SD is involved.
4.1 MMTM Statistic, and the MAD-Q Quality Metric

Chapter 4

Cell-Specific Quality Metrics

In the previous chapter, we saw how the MAD-mean is a robust centrality estimate for the true value, $\mu$. We have also covered the basics of the statistical bootstrap. Now, we employ both tools to develop a cell-specific quality metric that accounts for both range and accuracy. For range, a suitable solution is the difference between the mean and the MAD-mean, $\bar{x} - \bar{x}_{\text{MAD}}$, as the administrator’s choice of $\lambda$ allows the extension of the MAD’s trimming range, either to include or exclude normal outliers. For accuracy, the difference between the mean and the trimmed mean, $\bar{x} - \bar{x}_k$, provides a degree of outlier intolerance, as the trimmed mean suppresses all outliers, including normal outliers, because it trims at least a single outlier as long as $k \neq 0$. However, in case the case that $kN < N_o$, some outliers might be present in the trimmed mean term. However, for the scale of SD, this is unlikely to happen as the proportion of $k$ is larger for SD ($k = 10\%$ trims at most 1 to 3 samples for the range $8:30$, this implies that outliers will be filtered for $k > N_o/N$).

In this section we introduce the hybrid Mean-MAD mean-Trimmed Mean (MMTM) statistic, and use it as the basis upon which we formulate two quality metrics for SD in MCS. One is computationally simple for samples as small as 11 by measuring the closeness of the MMTM statistic’s magnitude to zero, while the other is slightly computationally demanding yet it extends the characterization of MMTM quality for sample sets as small as $N = 8$, by measuring the similarity between the bootstrap distribution of the MMTM statistic and the dirac delta distribution [88]. The developed quality metrics allow the MCS system administrators to control the trade-off between accuracy, range, and the number of MCS participants per cell, by tweaking an outlier sensitivity variable. This flexibility is significant in the design of commercial systems as it establishes a relation between cost, i.e., the number of participants, and quality.
4.1 MMTM Statistic, and the MAD-Q Quality Metric

The selection of a suitable point in the range-accuracy trade-off depends on the scenario, the application, as well as economic concerns. In this subsection, we describe the MAD quality metric (MAD-Q) that allows the MCS administrator to include or exclude normal/abnormal outliers by means of a sensitivity parameter, denoted $\beta$. The MMTM statistic, $\theta_{\text{MMTM}}$ is defined as:

$$
\theta_{\text{MMTM}} = \beta(\bar{x} - \bar{x}_k) + (1 - \beta)(\bar{x} - \bar{x}_\text{MAD}) = \bar{x} - \beta \bar{x}_k - (1 - \beta)\bar{x}_\text{MAD}
$$

(16)

where $\beta \in [0, 1]$, where $\bar{x}_k$ is the $k$-trimmed mean. The $k$-trimmed mean, discussed in Section 4.3, removes the rounded $k\%$ highest and/or lowest samples of $X$ prior the computation of the mean.

The $\theta_{\text{MMTM}}$ statistic is the difference between the mean and a weighted average of the more robust centrality estimates: the trimmed mean $\bar{x}_k$ and the MAD-Mean $\bar{x}_\text{MAD}$. $\theta_{\text{MMTM}}$ can be rewritten in terms of the errors of the mean, $\xi_{\bar{x}}$, the trimmed mean, $\xi_{\bar{x}_k}$, and the MAD-mean, $\xi_{\bar{x}_\text{MAD}}$; all errors can be derived in a manner similar to that for Equation (14). $\theta_{\text{MMTM}}$ becomes:

$$
\theta_{\text{MMTM}} = \mu + \xi_{\bar{x}} - \beta(\mu + \xi_{\bar{x}_k}) - (1 - \beta)(\mu + \xi_{\bar{x}_\text{MAD}}) = \xi_{\bar{x}} - \beta \xi_{\bar{x}_k} - (1 - \beta)\xi_{\bar{x}_\text{MAD}}
$$

(17)

where $\hat{\mu}_k$, $\hat{\mu}_\text{MAD}$ correspond to the estimation of the trimmed mean and the MAD-mean;
4.2 The MADBS-Q Bootstrap-based Quality Metric

$N_{k,o}, N_{MAD,o}$ correspond to the outliers present in the trimmed mean set $X_k$ and the MAD-mean $X_{MAD}$; $N_k, N_{MAD}$ correspond to $|X_k|$ and $|X_{MAD}|$; and $\Theta_k, \Theta_{MAD}$ correspond to the average deviation due to the presence of post-trimming outliers in $X_k$ and $X_{MAD}$.

Equation (17) comprises the sum of three errors. The first being between the outlier-free estimation errors of the mean, the trimmed mean, and the MAD-mean; the second being the sample size error due to the outlier samples present in $X$; and the third is the error due to the presence of outliers in the mean $\bar{x}$ and the trimmed mean $\bar{x}_k$, and the $\bar{x}$ and the MAD-mean $\bar{x}_{MAD}$. Furthermore, if $\lambda$, the parameter that defines the MAD-mean’s outlier removal range, is less than 2, the MAD-mean becomes very aggressive in selecting which samples are non-outliers, permitting it to be very robust with $b_{\bar{x}_{MAD}} \rightarrow 0$. Ideally, for a perfect sample, the $\theta_{MMTM}$ should be zero. However, this is impossible to happen empirically. Thus, we define quality as the closeness of this value to zero, and we formulate the MAD-Q quality, $Q_{MAD}$, based on the $\theta_{MMTM}$ statistic as:

$$Q_{MAD} = \log_{\gamma_{MAD}} \left( \frac{1}{\theta_{MMTM}} \right) = \log_{\gamma_{MAD}} \left( \frac{1}{\bar{x} - \beta \bar{x}_k - (1 - \beta)\bar{x}_{MAD}} \right)$$

where $\gamma_{MAD}$ is a scaling factor. Algorithm 2 shows the steps to compute the quality of an MCS sample. $\gamma_{MAD}$ can be changed as needed to control the scale, however the saturation value in line 6 in Algorithm 2 needs to be adjusted accordingly.

Algorithm 2 Algorithm for MAD-Q Quality Assessment

**Input:** Readings from $N$ sensors: $X = \{x_1, x_2, ..., x_n\}$

**Output:** Quality of Source: $Q_{MAD}$

1. $\bar{x}_{MAD} = \text{mean}(X_{MAD})$
2. $\bar{x} = \text{mean}(X)$
3. $\bar{x}_k = \text{trim\_mean}(X)$
4. $Q_{MAD} = \log_{\gamma_{MAD}} \left( (\bar{x} - \beta \bar{x}_k - (1 - \beta)\bar{x}_{MAD})^{-1} \right)$
5. if $Q_{MAD} \rightarrow \infty$ then
6. $Q_{MAD} = 6$
7. end if
8. return $Q_{MAD}$
4.2 The MADBS-Q Bootstrap-based Quality Metric

In the previous section we described how the non-parametric bootstrap is useful for acquiring the sample distribution of a statistic from small sample sizes. We combine the bootstrap with the $\theta_{\text{MMTM}}$ statistic to develop the MAD Bootstrap quality metric (MADBS-Q), $Q_{\text{BS}}$ that performs in a manner similar to $Q_{\text{MAD}}$, but capable of assessing to even smaller sample sizes [91]. Rather than formulating MADBS-Q based on the $\theta_{\text{MMTM}}$ statistic, it is formulated based on the sample distribution of $\theta_{\text{MMTM}}$. The bootlier, discussed in Section 3, is a tool that employs human intuition in its assessment on top of being too sensitive to disregard normal outliers as abnormal. This “over-sensitivity” to outliers impacts the perception of quality, which is an important factor in reducing the MCS system’s costs. To benefit from the bootstrap, we define the quality metric to relate to the closeness of the $\theta_{\text{MMTM}}$ sample distribution to the ideal impulse (dirac delta) best captured in terms of location and spread. We define the bootstrap-based quality, $Q_{\text{BS}}$ as:

$$Q_{\text{BS}} = \frac{1}{2} \left[ \log_{\gamma_{\text{BS}}} (\mu_{\text{MMTM}}^{-1}) + \log_{\gamma_{\text{BS}}} (\sigma_{\text{MMTM}}^{-2}) \right]$$

(19)

where $\mu_{\text{MMTM}}$ and $\sigma_{\text{MMTM}}^2$ are the mean and variance of the $\theta_{\text{MMTM}}$ sample distribution, respectively. And $\gamma_{\text{BS}}$ is a scaling factor. The mean of the resulting $\theta_{\text{MMTM}}$ distribution indicates its location, and the variance indicates its spread. However, it is best compared with the dirac-delta impulse.

The bootstrap-based quality, $Q_{\text{BS}}$, is thus defined as the average of the logs of the $\theta_{\text{MMTM}}$’s mean and variance. The closer they are to zero, the higher the quality. This renders $Q_{\text{BS}}$ as an absolute quality metric as it is free from any reference distributions (other than the dirac delta) or thresholds. Algorithm 3 shows the steps required to obtain the bootstrapped sample quality, $Q_{\text{BS}}$. In Algorithm 3, $\theta_{\text{MMTM}}^*$ refers to the vector containing the bootstrapped resamples of $\theta_{\text{MMTM}}$.

While both metrics, MAD-Q and MADBS-Q, are interchangeable, MADBS-Q is more
4.3 The Impact of Control Parameters

**Algorithm 3 Algorithm for MADBS-Q Quality Assessment**

**Input:** Readings from \( N \) sensors: \( X = \{x_1, x_2, ..., x_n\} \)

**Output:** Quality of Source: \( Q_{BS} \)

1. \( \theta_{MMTM}^* = \text{bootstrap}(X, B, \theta_{MMTM}) \)
2. \( \mu_{MMTM} = \text{mean}(\theta_{MMTM}^*) \)
3. \( \sigma_{MMTM} = \text{variance}(\theta_{MMTM}^*) \)
4. \( \text{return} \ Q_{BS} = \frac{1}{2} \left[ \log_{\gamma_{BS}} (\mu_{MMTM}) + \log_{\gamma_{BS}} (\sigma_{MMTM}^{-2}) \right] \)

reliable when data sets are very small, but it comes with a computational cost, mainly from the computation of the \( \theta_{MMTM} \) statistic \( B \) times. For the sample mean, the complexity of the bootstrap is \#P-hard, which can be solved in polynomial time [91]. MAD-Q, on the other hand, is cheaper to compute and serves data sets which are not as small as those encountered by MADBS-Q. Ultimately, MADBS-Q is derived from, and extends, MAD-Q.

### 4.3 The Impact of Control Parameters

The MMTM quality metrics, \( Q_{MAD} \) and \( Q_{BS} \), are based on the \( \theta_{MMTM} \), which is in turn based on the parameters \( \beta \) and \( \lambda \). \( \beta \), referred to as sensitivity from Equation (16), allows the control of whether the trimmed mean, \( \bar{x}_k \), or the MAD-mean, \( \bar{x}_{MAD} \), is more significant in \( \theta_{MMTM} \). \( \lambda \), on the other hand, is only specific to the MAD-mean, as it controls its filtering of outliers. In \( \theta_{MMTM} \), the presence of the term \( \bar{x}_k \) causes \( \theta_{MMTM} \) to maintain a narrow distribution, as outliers are always filtered for \( k > N_o/N \), while the presence of \( \bar{x}_{MAD} \) allows the control of the range based on the choice of \( \lambda \). \( \lambda \), as a parameter, allows the administrator to completely remove outliers by selecting \( \lambda = 3 \), or tolerate outliers where \( \lambda = 4 \) tolerates normal outliers and \( \lambda > 4 \) would tolerate abnormal outliers depending on their outlier deviation factor, \( \Theta \).

As a result, \( \theta_{MMTM} \), can be viewed as a sum of two distributions:

\[
\theta_{MMTM} = \frac{\theta_N}{\bar{x}} - \beta \bar{x}_k + (1 - \beta) \bar{x}_{MAD}
\]  

(20)
where $\theta_A$ is the sample distribution of the mean, and $\theta_B$ represents a mixture of the sample distributions of the trimmed mean and the MAD-mean, with $\beta$ as a mixing parameter.

The distribution of $\theta_{MMTM}$, as a result, is a multi-modal distribution with $\theta_A$ contributing a mode, and $\theta_B$ contributing either a mode or two based on the mixing parameter of the mixture between $\bar{x}_k$ and $\bar{x}_{MAD}$. However, it is hard for $\theta_B$ to be bimodal as both estimates are robust measures of centrality, except for the case in which $\lambda > 4$ which is no longer tolerant. For $\lambda < 4$, it is safe to assume that both, the trimmed mean and the MAD-mean, follow a distribution that represents the distribution of a sample mean: the normal distribution. For a mixture of two equally weighted normal distributions with similar variability to be bimodal, the difference between their means has to be greater than the sum of both their standard errors (or deviations, if not sample distributions) [92]. For $\theta_B$, the conditions are necessary (a):

$$|\mu_{\bar{x}_k} - \mu_{\bar{x}_{MAD}}| > \bar{\sigma}_{\bar{x}_k} + \bar{\sigma}_{\bar{x}_{MAD}},$$

(21)

and (b) $\beta = 0.5$ for them to be equally weighted. Since both are robust measures of centrality, then it is always the case that $|\mu_{\bar{x}_k} - \mu_{\bar{x}_{MAD}}| < \bar{\sigma}_{\bar{x}_k} + \bar{\sigma}_{\bar{x}_{MAD}}$, where $\bar{\sigma}_{\bar{x}_k}$ and $\bar{\sigma}_{\bar{x}_{MAD}}$ refer to the sample distribution of the trimmed mean and the MAD-mean’s standard errors, respectively. Thus bimodality is unlikely. However, for unimodality to occur in a mixed distribution, the ratio of the difference between the centers to the double of the product of the standard deviation has to be less than or equal to 1 [93], i.e.,

$$\frac{|\mu_{\bar{x}_k} - \mu_{\bar{x}_{MAD}}|}{2\sqrt{\bar{\sigma}_{\bar{x}_k} \bar{\sigma}_{\bar{x}_{MAD}}}} \leq 1$$

(22)

However, this implies that there is a region between unimodality and bimodality where:

$$\bar{\sigma}_{\bar{x}_k} + \bar{\sigma}_{\bar{x}_{MAD}} > |\mu_{\bar{x}_k} - \mu_{\bar{x}_{MAD}}| > 2\sqrt{\bar{\sigma}_{\bar{x}_k} \bar{\sigma}_{\bar{x}_{MAD}}}$$

(23)

In this region, the distribution of $\theta_B$ is neither unimodal nor bimodal, but trimodal [94].
4.3 The Impact of Control Parameters

However, this trimodality is unlikely to happen due to the fact that both sample distributions, \( \bar{x}_k \) and \( \bar{x}_{MAD} \) are robust estimates of the true mean \( \mu \), where the difference, \( |\mu - \bar{x}_k| \) is not in the interval \( (2\sqrt{\bar{x}_k \cdot \sigma_{x_k}} - \sigma_{x_k} + \sigma_{x_{MAD}}) \).

Moreover, the standard deviation of the sample distribution of the trimmed mean, \( \sigma_{\bar{x}_k} \) (not to be confused with the standard error of the trimmed mean, \( \bar{\sigma}_{x_k} \)) can be defined as a distribution [95]:

\[
\sigma_{\bar{x}_k} = \sqrt{\frac{E[(X - \mu)^2I(\Phi^{-1}[k/2],\Phi^{-1}[1-k/2])]}{1-k}} \tag{24}
\]

where \( E \) is the expectation operator, and \( I(a,b) \) is the identity function over the interval \([a,b]\). Product with the identity function effectively trims the tails of the distribution of \( X \).

Similarly, the standard deviation of the sample distribution of the MAD-mean, \( \sigma_{x_{MAD}} \), (not to be confused with the standard error of the MAD-mean, \( \bar{\sigma}_{x_{MAD}} \)) can also be defined as a distribution:

\[
\sigma_{x_{MAD}} = \sqrt{\frac{E[(X - \mu)^2I(\Phi^{-1}[2-\lambda],\Phi^{-1}[2+\lambda])]}{(1-\Phi^{-1}[2-\lambda]) - (1-\Phi^{-1}[2+\lambda])}} \tag{25}
\]

And since the standard errors, \( \bar{\sigma}_{x_k} \) and \( \bar{\sigma}_{x_{MAD}} \) are estimates of \( \sigma_{x_k} \) and \( \sigma_{x_{MAD}} \), respectively, it can be seen that the multimodality of \( \theta_B \) is mainly controlled by the administrator’s choice of \( \lambda \) rather than the choice of \( k \), as a choice of \( k \neq 0 \) will always trim at least a single outlier.

Consequently, \( \theta_{MMTM} \) has either a bimodal or a trimodal distribution (with three modes) as \( \theta_A \) contributes an extra mode due to the non-robust mean, while \( \theta_B \) could be unimodal or trimodal, depending on the choice of \( \lambda \). There are five possible scenarios for the sample distribution of \( \theta_{MMTM} \) based on the ratio of outliers present, \( N_o/N \):

(a) That no outliers are present \( (N_o/N = 0) \) and all measures of centrality are close to each other and \( \theta_{MMTM} \) is unimodal;

(b) The outlier ratio is less than the breakdown point of the MAD-mean term \( (b_{x_{MAD}} > N_o/N > 0) \) and \( \lambda <= 2 \), which will cause \( \theta_B \) to be unimodal, thus resulting in a bimodal \( \theta_{MMTM} \) due to the non-robustness of \( \theta_A \);
4.3 The Impact of Control Parameters

(c) The outlier ratio is less than the breakdown point of the MAD-mean term and $4 > \lambda > 2$, which will cause $\theta_B$ to be bimodal due to the presence of normal outlier, however that bimodality will not be apparent (i.e., it will not cause a local minima between the two modes), and therefore $\theta_{\text{MMTM}}$ will be trimodal. However, this trimodality will be a superposition of leaking impulses as previously mentioned in Section 3.3, where the leakage covers the mode due to the trimmed mean term.

(d) The outlier ratio is less than the breakdown point of the MAD-mean term and $\lambda > 4$, which will cause $\theta_B$ to be bimodal due to the presence of abnormal outliers, and thus $\theta_{\text{MMTM}}$ will be trimodal.

(e) The outlier ratio is more than the breakdown point. In this case, the outliers are no longer outliers and the modality of $\theta_{\text{MMTM}}$ is unknown as the variance would be large due to the spread of the values.
Chapter 5

MCS-Specific Coverage Metric

While MCS systems can function without full coverage, it is important for MCS administrators to be able to enhance coverage by identifying which spatiotemporal cells are below a desired coverage threshold. In this section, we provide a simple scheme that builds upon the quality metrics, discussed in the chapter 4, to evaluate the quality of the area as a whole during a specific sensing cycle. This quality evaluation comes after the cell-specific quality has been evaluated. The method described in this section provides the MCS administrators with two quantities, MAD-Q and MADBS-Q, that allow the characterization of MCS coverage quality. The first definition is an indicator of how the average quality is over the whole space, while the second definition is a detailed map of which cells have a higher quality relative to their neighbouring cells. By using the second definition, the MCS system can recruit participants from cells with over-satisfied quality and ask them to move to cells with lower quality and perform a required MCS task there. By having a metric that indicates where participants need to be moved, the MCS system becomes capable of achieving uniform MCS coverage as a part of participatory sensing, unlocking a scheme in which participants could be asked to (or incentivized to) voluntarily move from a place to another. Coverage, in that sense, extends the definition of the MAD-Q and MADBS-Q quality metrics, or any other cell-specific metric, over the area of interest, while uniformity refers to having good quality overall. While MCS coverage is independent of both MAD-Q and MADBS-Q, it is capable of augmenting both techniques for more control over participatory sensing. Figure 5 provides a description of the developed coverage metric.

5.1 Overall Coverage Quality

For algorithmic convenience, the cells in the area of interest are assumed to be equally spaced resulting in a $C \times C$ square grid. For each cell throughout the grid, the algorithms covered in Section 4 return the quality of each cell. This allows the definition of an overall
5.2 Relative Coverage Quality

Figure 5: Description of MCS-Specific Coverage Metric over a $C \times C$ square grid

The quality metric, $Q_{tot}$, is defined as

$$Q_{tot} = \text{mean}(Q_{1,1}, Q_{1,2}, \ldots, Q_{1,C})$$

where $Q_{map}$ is the resulting matrix of all quality evaluations for all cells, and $Q_{i,j}$ corresponds to the cell’s coordinates. $Q_{tot}$ can be seen in Figure 5 where it corresponds to a plane elevated to the average value of all cells’ MMTM quality.

It is important that the MCS administrator sets a minimum threshold for quality, $Q_{min}$, which can be useful in two ways: (a) to evaluate whether the overall quality is satisfied i.e., is $Q_{tot} \geq Q_{min}$, or (b) to identify specific cells which are below the threshold. Combining these together allows the characterization of overall quality, however it does not indicate how the quality can be adjusted to achieve uniformity.
5.2 Relative Coverage Quality

In order to characterize how quality can be improved over a space, we developed a method that measures the angle between each two adjacent quality points and constructs a $C^2 \times C^2$ matrix that maps this relation, denoted $Q_{rel}$ whose elements are defined as per the Pythagoras’s theorem as:

$$Q_{rel}(Q_{i,j}, Q_{a,b}) = \sin^{-1}\left[\frac{Q_{i,j} - Q_{a,b}}{\sqrt{|Q_{i,j} - Q_{a,b}|^2 + 1}}\right]$$  \hspace{1cm} (27)

where $(i, j)$ are the coordinates of the current cell, and $(a, b)$ are the coordinates of the adjacent cell, and $\sin^{-1}$ comes from trigonometric ratios of a right-angled triangle on the midpoints of quality values $Q_{i,j}$ and $Q_{a,b}$. Figure 5 provides an illustration of Equation (27) with $Q_i \equiv Q_{i,j}$ and $Q_j \equiv Q_{a,b}$.

In order to implement this in an automated algorithm, the 2-tuple map of a $C \times C$ square grid, $Q_{map}$ maps to a singleton map, $\hat{Q}_{map}$ such that:

$$\begin{array}{c|c|c|c|c|c|c|c}
Q_{1,1} & Q_{1,2} & \ldots & Q_{1,C} \\
Q_{2,1} & Q_{2,2} & \ldots & Q_{2,C} \\
\vdots & \vdots & \ddots & \vdots \\
Q_{C,1} & Q_{C,2} & \ldots & Q_{C,C} \\
\hline
Q_{map} & \rightarrow & Q_1 & Q_2 & \ldots & Q_C \\
Q_{C+1} & Q_{C+2} & \ldots & Q_{2C} \\
\vdots & \vdots & \ddots & \vdots \\
Q_{\Delta+1} & Q_{\Delta+2} & \ldots & Q_{C^2} \\
\hline
\hat{Q}_{map} & \rightarrow & Q_1 & Q_2 & \ldots & Q_C \\
Q_{C+1} & Q_{C+2} & \ldots & Q_{2C} \\
\vdots & \vdots & \ddots & \vdots \\
Q_{\Delta+1} & Q_{\Delta+2} & \ldots & Q_{C^2} \\
\end{array}$$  \hspace{1cm} (28)

where $\Delta = C(C - 1)$.

Within the singleton map, cells adjacent to the $k^{th}$ cell are corresponding to the cardinal directions: north ($k - C$), west ($k - 1$), east ($k + 1$), and south ($k + C$), for any cell that is not an edge or a corner. As a result, the $Q_{rel}$ can then be described by substituting Equation (27) for the corresponding elements.

The Relative Coverage Quality, $Q_{rel}$ looks at the quality, described by the $Q_{map}$, as a surface and measures the angles between adjacent points on it. In that sense, the matrix
5.2 Relative Coverage Quality

$Q_{rel}$ is defined as an anti-symmetric matrix in which transposing elements are a pair of alternate interior angles, and thus the sign change. As a result, $Q_{rel}$ is a sparse matrix, which can be visualized using color-coded tables or spy plots, as illustrated in Figure 13 in Section 6. If the corresponding value for $(k \rightarrow l)$ is positive, then the uniformity of coverage can be improved by moving participants from cell $k$ to cell $l$, while negative implies that participants need to be moved from $l$ to $k$ and/or incentivate new participants in $k$. 
Chapter 6

Simulation and Results

In order to test the proposed algorithms, we consider a temperature evaluation scenario in MATLAB, where temperature measurements are acquired from $N(24, 0.5)$, and two outliers are obtained from an abnormal outlier distribution located $N(24 + \bar{\Theta}, 0.3)$. $\gamma = 3.1$ is chosen for the MAD-Q, while $\gamma = 10$ is selected for MADBS-Q to ensure comparable scales. $k = 10\%$ is decided for the trimmed mean. The cell specific quality metrics are evaluated over an ensemble of 30 randomly generated samples for each data point. For the MADBS-Q, $B = 250$ is employed for the purpose of this simulation.

6.1 Cell-Specific Quality

We tackle the case of variable $N_{tot}$, $\bar{\Theta}$, $\lambda$, and $\beta$, in order to illustrate the impact of these variables on the quality.

Figure 6 illustrates $Q_{MAD}$ and $Q_{BS}$ for different sample sizes. Both $Q_{MAD}$ and $Q_{BS}$ show that with larger sample size, the quality increases. Moreover, this figure illustrates the sample size required to achieve a certain quality threshold, which is useful in selecting a suitable number of participants for recruitment in a specific MCS application. This can be further seen in Figure 9, for which the sample size is evaluated for $Q_{MAD}$ at different AODFs, $\bar{\Theta}$. Figure 7 represents $Q_{MAD}$ and $Q_{BS}$ for different $\bar{\Theta}$. It is observable that the curves are seemingly symmetric as they tend to follow the shape of a bell curve. This is due to the fact that the closer the outliers are to the center of the true value’s population, the less of an outlier they are. At $\bar{\Theta} = 0$, the outliers constitute the sample itself, i.e., no outlier are present. At the extremes, $\bar{\Theta} = \pm 10$, the quality worsens significantly. Figures 6 and 7 are slices of Figure 9 at $\bar{\Theta} = 6$ and $N_{tot} = 20$, respectively.

In Figure 8, $Q_{MAD}$ and $Q_{BS}$ for different $\lambda$ are shown. Figure 10 provide a view of how $Q(\lambda)$ varies with $\bar{\Theta}$, raising the general quality for high $\lambda$. For low $\lambda$, $\lambda < 2$, the quality
6.1 Cell-Specific Quality

Figure 6: $Q_{MAD}(N)$ and $Q_{BS}(N)$

Figure 7: $Q_{MAD}(\Theta)$ and $Q_{BS}(\Theta)$
6.1 Cell-Specific Quality

Figure 8: $Q_{MAD}(\lambda)$ and $Q_{BS}(\lambda)$

Figure 9: $Q_{MAD}(N, \Theta)$
6.1 Cell-Specific Quality

Figure 10: $Q_{MAD}(\tilde{\Theta}, \lambda)$ for $\lambda \in [0, 10]$, $N_{tot} = 20$, and $\beta = 0.8$

Figure 11: $Q_{MAD}(\lambda)$ for different $\beta$ for $\beta \in [0, 1]$, $N_{tot} = 20$, and $\Theta = -7$
6.1 Coverage Quality

Figure 12: $Q_{\text{MAD}}(\beta)$ for different $\lambda$

is consistent as no outliers are tolerated. However for larger values of $\lambda > 4$, the metric becomes more tolerant towards outliers, and thus raises the quality perceived as $\lambda$ increases.

Figures 11 and 12 provide a view of $Q_{\text{MAD}}$ for different $\lambda$ and $\beta$. Since $\beta$ controls the presence of the MAD-mean term in the $\theta_{\text{MMTM}}$ statistic, at $\beta = 0$ the trimmed mean term is entirely absent, thus causing a larger variation in quality. For $\lambda < 2$, the quality is not very high nor very low. This is due to the fact that $Q_{\text{MAD,BS}}$ measures the closeness of a sample set’s values. For $\lambda = 4$, the quality can be seen to have started increasing as outliers gradually become tolerable. For $\lambda > 4$, the quality increases as $\lambda$ increases, as it allows more outliers to be tolerated. However, for $\beta = 1$, when the trimmed mean is fully present, $\lambda$ has no effect at all. Similarly for $\beta$, the quality increases from $\beta = 0$ to $\beta = 1$ except when $\lambda \geq 4$.

The difference between both methods lies in the fact that the MADBS-Q can quantify, consistently, samples as small as $N = 8$ at a computational cost without falling into degeneracies, unlike the MAD-Q which often falls in the log function’s negative domain for
6.3 Discussion

$N < 11$, being sometimes undefined (returning a NaN for a negative $\theta_{\text{MMTM}}$). However, the behavior for $Q_{\text{BS}}$, as a surface, is similar to that illustrated in figures 9, 10, 11, and 12.

6.2 Coverage Quality

![Spy plot of the adjacency matrix, $Q_{\text{rel}}$](image)

To test our coverage quality metric, we generate a $3 \times 3$ map with values obtained from a distribution $N(2.2, 0.5)$, to simulate diverse values obtained from the metrics discussed in the previous subsection, where $Q_{\text{map}}$ is:

$$Q_{\text{map}} = \begin{bmatrix}
3.09 & 1.04 & 2.23 \\
2.81 & 2.65 & 2.22 \\
1.56 & 1.28 & 3.32
\end{bmatrix} \quad (29)$$

The overall quality metric, $Q_{\text{tot}}$ is found to be 2.24, and $Q_{\text{rel}}$, illustrated in Figure 13, is obtained. We use a color code to identify quality, where green denotes that the corresponding cells are near each other, i.e., $Q_{\text{rel}}$ is closer to zero at these two cells, and red denotes that there is a significant contrast between the values, i.e., $Q_{\text{rel}}$ far from zero. The resultant matrix, illustrated as a sparse adjacency matrix in Figure 13, is anti-symmetric as the relations between cells are bidirectional. This result is very important as it shows the status-quo to the administrator and sheds the light on where more participants are needed.
in order to balance the distribution among the cells.

6.3 Discussion

The proposed algorithm has two main parameters, $\lambda$ and $\beta$, both which impact the $\theta_{\text{MMTM}}$ parameter, as previously discussed in Section 4.3. Another parameter that is also considered is $k$, the amount of trimming done by the trimmed mean term $\bar{x}_k$. Increasing $k$ increases the intolerance of the trimmed mean to outliers, however in most cases 10% is more than enough in SD scenarios where $N \in [8, 30]$, as increasing it would be of little benefit as removing 20%, for example, of 20 samples reduces it to 16 samples; which is unlikely for abnormal outliers.

The choice of $\beta$ and $\lambda$ relies on the application on question. If the application requires being more tolerance towards outliers, a small value of $\beta$ would increase the significance of the $\bar{x}_{\text{MAD}}$ term and thus $\lambda$ would be the only parameter impacting the filtration of outliers. However, the application requires intolerance to outliers, then choosing a high $\beta$ increases the significance of the trimmed mean term, $\bar{x}_k$ thus filters outliers aggressively. However, changing $k$, as previously mentioned, is of little benefit.


We have presented a new direction for looking at scenarios in which MCS participants present are not enough or the data reported by them is not entirely reliable due to the presence of abnormalities, and thus errors. Sometimes, the available data may not be enough to fully characterize the MCS area of interest. To enhance the MCS system’s robustness, we introduced SD quality techniques to allow MCS systems to evaluate a group of participants’ reports in a specific spatiotemporal cell based on the data they reported in scenarios where the scale of data is insufficient for proper inference. This evaluation was performed without knowledge of the true value, ascertaining as much as possible from the least amount of data.

The proposed quality metrics, MAD-Q and MADBS-Q, allow the characterization of a sample’s quality - typically an MCS sample being a set of physical readings whose readings come from a symmetric distribution - when the number of contributors is minimal. The MAD-Q technique allows the MCS system to evaluate the quality of samples as small as 11, while MADBS-Q performs the evaluation of samples as small as eight with a little computational cost by employing the non-parametric bootstrap. Both metrics are based on the MMTM statistic, which is designed to allow the formulation of a flexible quality metric, that takes into consideration tolerating specific outliers, by tweaking the sensitivity parameter $\beta$ that controls the weights of the trimmed-mean ($\beta = 1$) and the MAD-mean ($\beta = 0$), and a range parameter $\lambda$ that allows the control of the MAD-mean’s outlier tolerance. This control, while allowing the tolerance of certain values, allows MCS administrators to define, subjectively, what a usable MCS data set is in a domain-agnostic manner. Furthermore, the importance of the quality metrics lies in their relationship to the sample-size, as illustrated in the MAD-Q technique, allowing the administrators control in the quality-cost trade-off.

We also proposed an MCS-specific coverage metric to employ the notion of a cell-specific
quality, no matter how it is defined, in the definition of an MCS system’s coverage. The coverage metric aims to achieve uniformity by describing quality as a whole, and also in a relative manner between the MCS system’s spatiotemporal cells. This gives the MCS system a way of measuring which cells are overstaffed or understaffed by MCS participants as well as a metric that indicates how to manipulate the current participant topology (i.e., the participants’ distribution over the grid) to achieve coverage uniformity, either by recruiting more participants or by requesting their movement around the area of interest. The developed quality metrics, MAD-Q and MADBS-Q, along the MCS-specific coverage metric, provide a clear relationship for the MCS system between sensing quality and sensing cost.

In conclusion, the MAD-Q and MADBS-Q are controllable quality metrics for scarce scenarios in MCS systems that measures quality of a data set similar to how entropy measures uncertainty in a random variable. The MMTM statistic, upon which both metrics are developed, provides a degree of control for MCS administrators to control the influence of anomalous values, namely outliers, on the perception of quality. Both metrics provide a relationship between the number of MCS participants and the perceived quality, for scarce MCS scenarios. In addition, both metrics can be extended to form a coverage metric that is suitable for the unorthodox nature of MCS tasks.

7.2 Recommendations and Further Work

For the implementation of the proposed quality metrics, we recommend that MCS administrators identify the thresholds of each quality metric theoretically based on a sample of smart devices that MCS participants could use. The data sheets of such sensors provide information on the error margins of the sensors’ readings. Quality thresholds would allow the MCS administrator to make decisions regarding MCS quality control - i.e., to select more or less participants - and thus mitigate the cost.

For further work, academic research based on these quality metrics, a form of feedback could be implemented to involve historical information. While this approach has been im-
implemented in numerous works, the MCS environment is a dynamic environment in which phenomena does not often remain static - otherwise, it would not need to be sensed. Thus, feedback for MAD-Q and MADBS-Q ought to be limited by means of a window and independent of the participants to avoid biasing readings towards regular participants.

Another potential further work is to consider non-symmetric distributions, as there are various sensed phenomena that come from non-symmetric distributions (e.g., GPS or virtual sensors), however it is a more challenging direction as less generalizations can be made.
References


References


Appendix: Bootstrap Probability Analysis

The purpose of this appendix is to introduce the reader to the MMTM statistic defined in Equation (15) under the statistical bootstrap. The reader will find a derivation of the sampling distribution of the sample median within the bootstrap. This result would be helpful for further derivations under the bootstrap.

Within the bootstrapped sample space, the sample distributions of the centrality estimates (mean, trimmed mean, MAD mean) are different, but they all share an overlap. Equation (16) is a function of three random variables, this can be illustrated as a sample space denoted $\mathcal{U}$, containing subsets corresponding to each of the random variables as shown in Figure 14.

![Venn Diagram of the Bootstrapped Sample Space and its subsets](image)

Figure 14: Venn Diagram of the Bootstrapped Sample Space and its subsets

Inferences can be made, at this point, regarding each of the random variables. The sample distribution $\bar{x}$ and the $\bar{x}_k$ both follow normal distributions, thanks to the central limit theorem. The distribution of the MAD-mean, $\bar{x}_{\text{MAD}}$, however is different as it is a result of the distribution of the median. The median, however, thanks to its discrete nature
Appendix: Bootstrap Probability Analysis

has various cases. If the sample size was odd, the probability that a specific bootstrap sample, \( x_n \) is selected as the median, \( \tilde{x}_b \), can be expressed by Equation (30), where \( r_U, r_L \) correspond to the number of values greater than the \( n^{th} \) and values lower than the \( n^{th} \), respectively, and \( N \) is the sample size. The numerator is the probability of a specific event, whereas the denominator is the total, which is an extension of the multinomial distribution that treats the possibilities of values greater or smaller than the median coexisting with it, in a bootstrap resample such that the bootstrap median would reflect the \( n^{th} \) sample.

For an even sample size, the expression becomes more tedious since the possibilities of the bootstrap median increase, as the averages among samples become median candidates in addition to the samples themselves. Equation (31) shows the expression we have derived for that, where \( r_{M1} \) and \( r_{M2} \) represent the two upper and lower values which would contribute to the median, while \( r_U \) and \( r_L \) represent the rest of the values within the sample that are remote to the center:

\[
P(\tilde{x}_b = x_n) = \frac{\sum_{r_U=0}^{N/2} \sum_{r_L=0}^{N/2} \frac{N!}{r_U!(N-r_U-r_L)!r_L!} \left( \frac{n-1}{N} \right)^{r_L} \left( \frac{1}{N} \right)^{N-r_U-r_L} \left( \frac{N-n}{N} \right)^{r_U}}{\sum_{n=1}^{N/2} \sum_{r_U=0}^{N/2} \sum_{r_L=0}^{N/2} \frac{N!}{r_U!(N-r_U-r_L)!r_L!} \left( \frac{n-1}{N} \right)^{r_L} \left( \frac{1}{N} \right)^{N-r_U-r_L} \left( \frac{N-n}{N} \right)^{r_U}}
\]  

\[
P(\tilde{x}_b = \tilde{x}_j) = \frac{\sum_{j=1}^{N/2} \sum_{j=1}^{N/2} \sum_{r_{M1}=1}^{N/2} \sum_{r_{M2}=1}^{N/2} \frac{N!}{r_{M1}!r_{M2}!r_U!} \left( \frac{N-f_{Uj}}{N} \right)^{r_{M1}} \left( \frac{N-f_{Uj}}{N} \right)^{r_{M2}} \left( \frac{1}{N} \right) \left( \frac{1}{N} \right)^{r_{M1}+r_{M2}}}{\sum_{j=1}^{N/2} \sum_{j=1}^{N/2} \sum_{r_{M1}=1}^{N/2} \sum_{r_{M2}=1}^{N/2} \sum_{r_U=0}^{N/2} \frac{N!}{r_U!r_{M1}!r_{M2}!r_U!} \left( \frac{N-f_{Uj}}{N} \right)^{r_{M1}} \left( \frac{N-f_{Uj}}{N} \right)^{r_{M2}} \left( \frac{1}{N} \right) \left( \frac{1}{N} \right)^{r_{M1}+r_{M2}}}
\]

It is possible to see from both equations that the probability distribution tends to have a central limit. However, it is not a smooth distribution since the median is, by nature, discrete. Thus the distribution of the medians attempts to approach that of the normal distribution - due to the central limit - however the shape is not always normal. As a result, the MAD-mean’s distribution is the discrete impulses representing that of the median convolved with
the normal distribution. While it does have a clear central tendency to be normal, it is not necessarily normal if the medians are distant from each other, which would result in a bumpy distribution. Ultimately, to obtain the closed-form approximation of the bootstrap, the compound probabilities formed from the simple ones described in the Venn Diagram in Figure 14, and have to be computed, as well as their expectations.

Finally, \( f_{jl} \) and \( f_{ju} \) are the number of occurrences (frequency) necessary for the \( j^{th} \) median out of the range of all possible medians, for the even sample size. The expression for the median probability for an even sample can be obtained using Algorithm 4.

**Algorithm 4** Algorithm for computing median probability for even samples.

**Input:** Sample \( X = \{x_1, x_2, ..., x_n\} \)

**Output:** \( \tilde{x}_j \)

1: for \( a = 1 \) to \( N \) do
2: for \( b = a \) to \( N \) do
3: store \( (x_a + x_b)/2 \) in \( \tilde{x}_j[i, 1] \)
4: store \( x_a \) in \( \tilde{x}_j[i, 2] \)
5: store \( x_b \) in \( \tilde{x}_j[i, 3] \)
6: end for
7: end for
8: [\~, idx] = sort(\( \tilde{x}_j[:, 1] \))
9: \( \tilde{x}_j = \tilde{x}_j[idx, :] \)
10: return \( \tilde{x}_j \)