Networkmetrics: Multivariate Big Data Analysis in the Context of the Internet

May 25, 2016

José Camacho, Roberto Magán-Carrión, Pedro García-Teodoro, James J. Treinen

Abstract

Multivariate problems are found in all areas of knowledge. In chemistry and related disciplines, the chemometric community was developed in a joint effort to understand and solve problems mainly from a multivariate and exploratory perspective. This perspective is, indeed, of broader applicability, even in areas of knowledge far from chemistry. In this paper, we focus on the Internet: the net of devices that allow an interconnected world where all types of data can be shared and unprecedented communication services can be provided. Problems in the Internet, or in general in networking, are not very different from chemometric problems. Building on this parallelism, we review four classes of problems in networking: estimation, anomaly detection, optimization and classification. We present an illustrative set of problems and show how a multivariate perspective may lead to significant improvements from state-of-the-art techniques. In absence of a better name we call the approach of treating these problems from that multivariate perspective networkmetrics. Networkmetric problems have their own specificities, mainly their typical Big Data nature and the presence of unstructured data. We argue that multivariate analysis is, indeed, useful to tackle these specificities.

Keywords: Multivariate analysis; Networking; Networkmetrics; Big Data

1 Introduction

The Internet has been recognized as one of the main technological advances in human history. In developed parts of the world, we have changed the way we interact and the way we seek information by relaying more and more on computer networks. This causes a fast and continuous alteration of human habits, since advances in communication technologies are giving way to the development of new Internet services we can access everywhere trough mobiles, wearable devices and vehicular communications, among others.

The goal of this paper is to illustrate how Internet communications and services can benefit from advances in multivariate analysis, making a parallelism with the achievements accomplished in analytical chemistry or process analysis, among others, thanks to chemometrics. This parallelism is, indeed, the reason why we refer to the study of multivariate data problems in the Internet as networkmetrics[1]. Same types of problem arise in both chemometrics and networkmetrics: estimation, optimization, anomaly detection and classification problems, among others. Still, two main differences are found: (i) data-based problems in networkmetrics are commonly Big Data problems by definition, and (ii) networkmetric data are mostly unstructured, which complicates but also increases the flexibility and possibilities of the analysis.

The rest of paper is organized as follows. In Section 2, the Internet networking structure and functioning is briefly introduced to contextualize the remainder of the paper. Section 3 discusses similarities and differences between chemometrics and networkmetrics from a broad perspective. The following four sections present some of the main data-driven problems in the context of networking, organized according to their goal in Sections 4 to 7: estimation, anomaly detection, optimization and classification, respectively. The list of problems provided here is not intended to be complete, but illustrative of the potential benefit of an adequate application of multivariate analysis to networking. In each of these sections, a brief literature review is performed, discussing

[1]It should be noted that this is not a widely extended term: researchers working on networking with multivariate analysis methods do not have a sense of belonging to a research community, nor there are journals or conferences focused on this topic.

whether main references apply or not multivariate analysis and, where appropriate, suggesting how we can make the most of advanced chemometrics methods to improve state-of-the-art approaches in networking. Also, the specificities of the data used in each of the problems are described and illustrative examples are included at the end of each section or subsection. Examples have clear resemblance to problems dealt with in the chemometric literature, so the goal here is to motivate the application of approaches that worked well in chemometrics to some of the problems below or similar ones. Finally, Section 8 presents the conclusions of the work.

2 The Internet and the Networked World

The widespread use of the Internet is the result of the combination of very smart design decisions [1]. Some of these are described in this section.

Computer communications were preceded by the invention of telephone communications, and the Public Switched Telephone Network (PSTN) was well understood when the seed of the Internet, the ARPAnet project, was initiated. In the traditional PSTN, each telephone call between a pair of phones reserves dedicated resources in the network for that communication. Even if the users of the call remain silent, the resources cannot be used by thirds. To avoid this waste of resources, the Internet embraced the *packet switching* approach. Every message we send through the Internet is cut into a number of *packets*, also named *datagrams* (Fig. 1). Each datagram contains a part of the message plus a set of control variables added at the beginning, forming what is called the *header* of the packet. One principal control information in the header is the datagram destination. With this information, each datagram travels the Internet on its own, so that different datagrams of a same message may follow different paths to their destination. This approach optimizes the use of resources and makes communications more robust to malfunctioning. This was principal for the adoption of the Internet, since reasonably low investments were required to offer very useful services, like web navigation or the e-mail.

Another main design choice in the Internet is its layered architecture. For a proper commu-

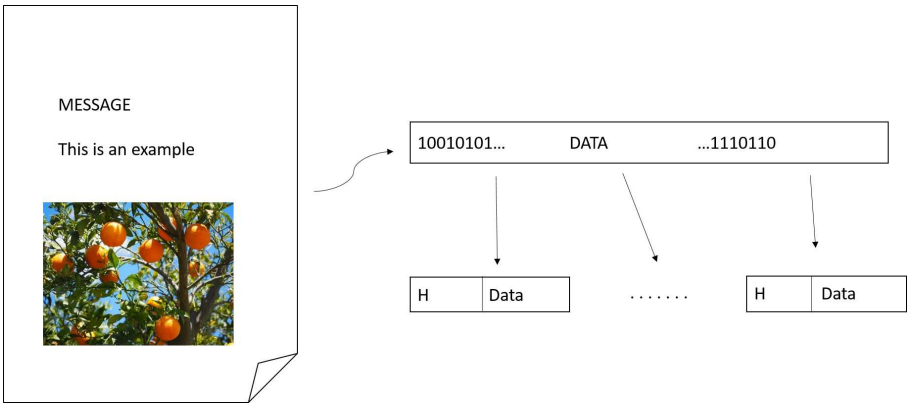


Figure 1: Illustration of how a message is sent through a packet-switching network: the information, which is stored in binary form in a computer, is split into a number of packets and combined with control information (headers H) prior to be sent.

nication between two devices, there are many problems that need to be solved. The fathers of the Internet realized that it was too difficult for a single computer program to incorporate all the functionalities needed to solve all of these problems. Instead, they decided to separate functionalities into layers. In each layer, a handful of functionalities are implemented, and it is assumed that the remaining functionalities are handled somewhere else. The functionalities in each layer are accomplished by software entities (programs). Thus, the communication between two devices in a network is supported by the communication between several paired entities in the devices, each pair belonging to a different layer and talking in a specific language named protocol, see Fig. 2. To make this possible, the datagram headers are divided in as many parts as protocols are used in the communication, and then headers and data in the datagram are sent through the communication channel.

The definition of layers is a very powerful idea in computing environments. A main advantage is that a layered architecture can hide the specificities of the communication hardware. Thus, networks with different technologies, from legacy to the most modern technologies, can communicate through a world wide interconnection of heterogeneous networks: the Internet. The layer that most contributed to the development of the Internet has been the so-called Internet Protocol (IP)

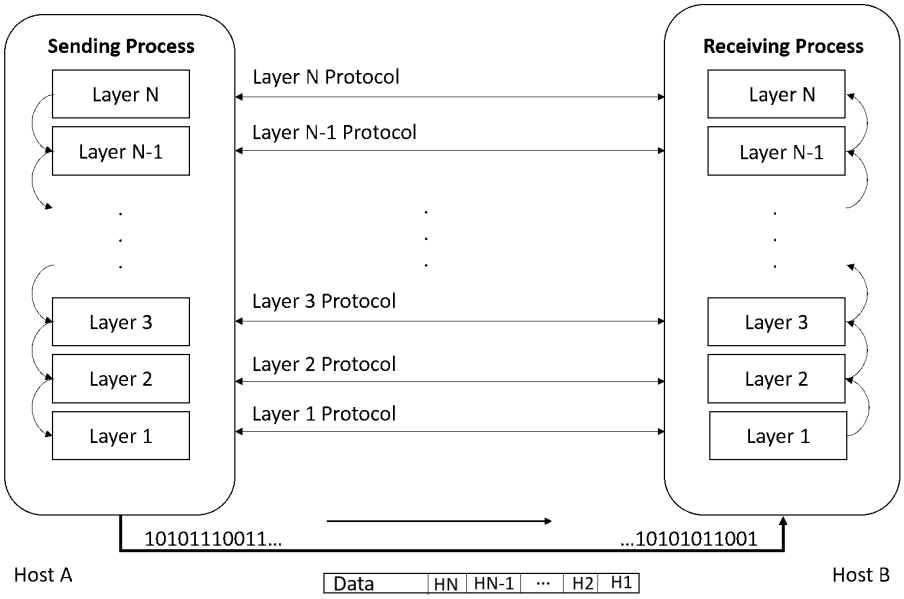


Figure 2: Layered network architecture: horizontal (virtual) communication is performed among entities in the same layer, though real (physical) communication is done over the links in binary codification. The virtual communication in layer n is carried out thanks to the information in the corresponding header (Hn) of the packet, on the bottom of the figure.

layer. Software entities in this layer are responsible for identifying the devices in the whole Internet with the *IP addresses*[2] and routing the datagrams from the origin to the destination through the mesh of interconnected networks. To reach the destination, the datagram goes through a number of intermediate devices, referred to as *routers*, each of them is interconnected to the rest of the Internet through (wired or wireless) links, see Fig. 3. Routers are responsible for maintaining updated information for datagrams routing towards their destination.

As a summary, the proper functioning of the Internet is a mixture of complex functionalities and decisions made on a distributed basis, in which a large number of entities at different parts of the Internet contribute. In the end, the Internet is kind of a *swarm intelligence*, made up of simple computing programs whose interaction create a complex behavior. Understanding such a behavior is very much like understanding macroeconomics or the human body. This paper discusses the

[2]An IP address is the identifier of a device in a computer network. For instance 150.214.191.5

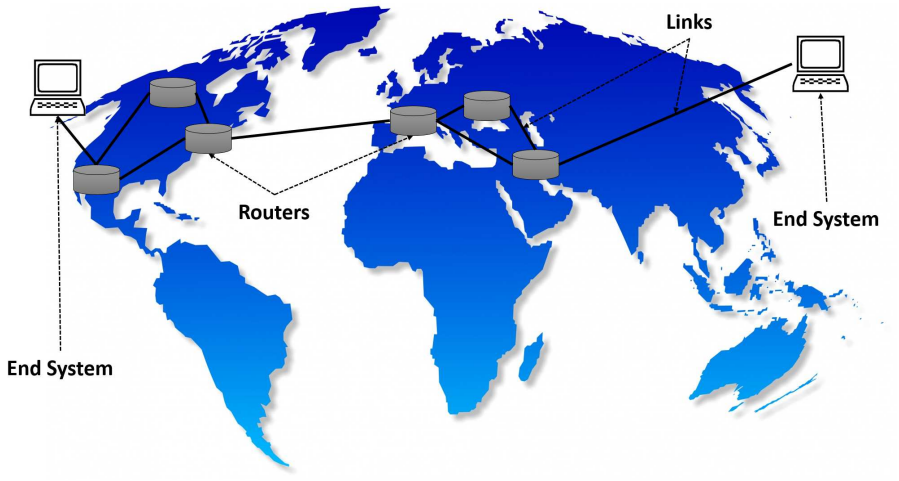


Figure 3: Illustration of the Internet topology: end systems (e.g. PCs) communicate through a network of routers connected by links.

potential contribution of multivariate analysis to this end.

3 Networkmetrics vs Chemometrics

It has already been discussed, and will be treated in detail in the following sections, that many data-driven applications in networking are very close to other applications in the chemometric literature. As a motivating example, the detection of cybersecurity issues and cyberwarfare activity can be approached following the Multivariate Statistical Process Control (MSPC) philosophy [2]. However, the main differences relate to the data that need to be handled in each area.

A large number of networkmetric applications are naturally Big Data applications, while most chemometric applications are not (yet). This may not be a relevant difference, since many chemometric applications where sampling is applied, like in industrial processing, could be thought of as Big Data. However, networkmetric applications are among the first applications where tons of data needed to be handled. They share a number of features with other typical Big Data problems, the so-called 4 Vs [3]:

- Variety: Data are varied in nature. Different sources, including unstructured and structured

Table 1: Memory storage units.

Name (Symbol)	Standard SI
Kilobyte (KB)	10^3
Megabyte (MB)	10^6
Gigabyte (GB)	10^9
Terabyte (TB)	10^{12}
Petabyte (PB)	10^{15}
Exabyte (EB)	10^{18}
Zettabyte (ZB)	10^{21}
Yottabyte (YB)	10^{24}

information, need to be properly combined for problem solving.

- Veracity: The search for valuable information in large data sets is very much like the problem of finding the needle in a haystack. Big Data present low signal to noise ratio, and data analysis techniques are needed to find patterns or trends, which are more reliable than punctual measures.

- Volume: The amount of data that needs to be handled simultaneously makes hardware parallelism a must in many cases. Exabytes, Zettabytes, and even higher amounts of data are described in Big Data applications (see a list of storage units in Table 1).

- Velocity: In networkmetric problems, a high rate of incoming information is common. This further complicates the analysis and makes parallelism even more necessary. Quoting Robert J. Moore, CEO and co-founder of RJMetrics, "*There was 5 exabytes of information created between the dawn of civilization through 2003, but that much information is now created every 2 days, and the pace is increasing*".

Another challenge in networkmetrics is to handle unstructured data, while typical chemometric data are structured. Structured data refer to data with a fixed format (top of Fig. 4), e.g. the data you can store in several fields of a spreadsheet. This includes three-way or multi-block data.

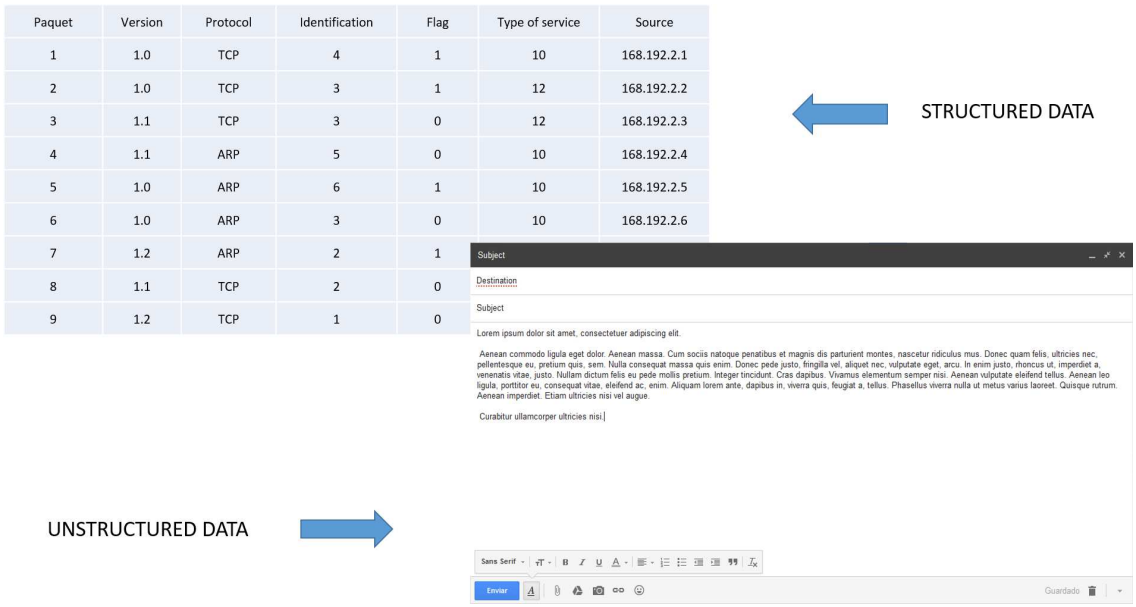


Figure 4: Examples of structured and unstructured data.

For instance, process data is structured, since for every sampling time we obtain a set of process variables we can directly input to our multivariate model. Instead, unstructured data need to be translated into a set of variables to make possible the multivariate analysis. Take the example of the e-mail in Fig. 4 and think of an automatic procedure to distinguish useful e-mails from *spam*. The management of unstructured data is a main problem in anomaly detection, and will be discussed in more detail in Section 5.

The networking literature is full of different approaches on how to deal with unstructured and Big data, ranging from parallel processing to visualization methods. However, the analysis of networking data can also benefit from identifying similarities with chemometric data and using chemometric algorithms. Thus, the ability of chemometric methods to handle highly multivariate data is of utmost importance to analyze unstructured data, since the amount of variables that can be artificially defined from these is unlimited. For instance, following with the example of the e-mail anti-spam application, we may select a parametrization that includes different statistics of the apparition of specific letters, words and/or combinations of words. This is likely to be highly multivariate. Most of the time, these types of approaches are avoided in the networking community,

where the expertise on multivariate analysis is not common. This also opens interesting questions on how to design the parametrization of unstructured data to make the most of a given multivariate model. Another typical problem in networkmetrics is that of data fusion, where several sources of information need to be combined to fulfill a specific goal. Multi-block methodologies can be useful for that. Finally, temporal and spatial patterns in networkmetric problems may be adequately handled following the N-way methodology, opening interesting research opportunities. This will be further discussed in following sections.

Most networkmetric problems are Big Data problems. This means that the application of multivariate algorithms in Big Data need to be worked out. There are at least two ways to do so. A first approach is to program chemometric algorithms in parallel processing platforms like Apache Hadoop (https://hadoop.apache.org) or Spark (http://spark.apache.org). This is the case, for instance, of the Singular Value Decomposition (SVD) implementation of the Mahout machine learning library (https://mahout.apache.org). However, implementations programmed so far following this or similar philosophies [4, 5, 6] are more aimed at the automatic use of algorithms advocated in the machine learning area. Multivariate models in chemometrics are more often used in an interactive basis, an approach similar to that of the so-called visual analytics research area [7].

A second approach to perform multivariate analysis in Big Data is to reduce/compress data and then apply methods in a similar way how they are applied to short data. This is the goal of the MEDA Toolbox [8], where iterative calibration methods based on kernel computations are combined with clustering-based visualization tools, like the Compressed Score Plots (CSP) [9]. An example of this approach is included below in one of the examples of the paper.

4 Estimation

Estimation is applied when there are important variables that cannot be measured, are missing, or their measurement is complex and/or expensive. A common estimation problem in networking

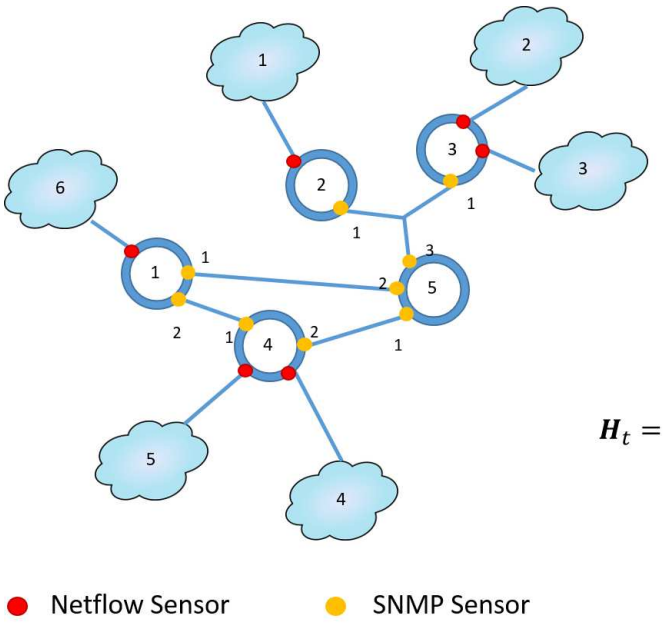
is that of the traffic load in a network, that is, the amount of traffic in terms of packets or bytes of data transmitted per time unit. When the load is too high for the technology of the network, the delay of the communication grows, effecting the quality of the network services. This is equivalent to a traffic jam in a road network. Following this parallelism, a computer network (a road network) gets congested when the amount of packets (cars) is too large for the capacity of the network links (roads). Estimating the traffic load in a network is just as useful as in a road network. This estimation is necessary for the design of the network, including the choice of the number of links and routers to be deployed (Fig. 3) and their traffic capacity, the number of alternative paths to a set of destinations, etc. [10]. This estimation is also useful for security, since some of the most harmful attacks in communication networks, like *Denial of Service* (DoS) attacks, can be detected from the amount of traffic [11].

Another typical estimation problem is that of missing data imputation. Missing data imputation is especially relevant in a class of networks, the so-called *sensor networks*, which send sensed information to a Central Unit (CU) for analysis. An industrial network is a sensor network that is also used to communicate controllers and actuators in an industrial process. The problem of missing data estimation in sensor or industrial networks is very similar to that in the process industry. However, missing data imputation in networking explores the ability of missing data imputation algorithms when the information was lost due to communication problems, including malfunction in communication devices and/or malicious attacks [12, 13].

4.1 Traffic Matrix Estimation

A network-wide view of the traffic load is obtained with the traffic matrix (TM) [14, 15]. The TM contains the amount of traffic from each pair of origin-destination (OD) in the network. Once estimated, the TM can be used in a large set of applications, mainly to optimize the performance of the network (see Section 6).

There are different types of TMs depending on the specific goal (network design, network monitoring, etc.). A main design choice for a TM is that of spatial aggregation [10]. Thus,



$$\boldsymbol{H}_t = \begin{bmatrix} 0 & 102 & 23 & 54 & 102 & 804 \\ 100 & 0 & 44 & 46 & 22 & 55 \\ 12 & 34 & 0 & 130 & 12 & 12 \\ 60 & 32 & 204 & 0 & 32 & 45 \\ 120 & 28 & 103 & 5 & 0 & 82 \\ 1005 & 34 & 54 & 114 & 73 & 0 \end{bmatrix}$$

Figure 5: Illustrative network and corresponding traffic matrix where the OD pairs are the subnets (represented by clouds) and counts are in Mbps. The typical location of Netflow and SNMP sensors is also shown.

depending on the size of the network and the goal of the TM, we may use a different definition for the OD nodes [15]: from considering every single node of the network, and thus generating a potentially huge TM, to measuring the traffic among large network areas. Other choices for a TM are the units in which the traffic is measured (e.g. bits per second (bps) or packets per second (pps)) and the measurement frequency. Fig. 5 illustrates a network and potential TM design, where the traffic in each element corresponds to each pair of OD sub-networks (represented by clouds) and the units could be Mbps[3]. In this case, the TM is not symmetric, which tells us that the amount of traffic in a OD pair is not the same as in the other way round. Also, the diagonal is zero because traffic within one subnet does not go through the transport network.

Let us call $\mathbf{x}_t$ the column vectorization of the TM at a given sampling time t. The value of $\mathbf{x}_t$ can be directly measured using a traffic flow registering protocol like *Netflow* [16]. The measurement is distributed across the network. Netflow sensors, measuring one or several of the

[3]Mbps reads Mega-bit-per-second and equates to 10^6 bps.

1970-01-02 01:13:53,1970-01-02
01:14:59,66.822,33.4.1.0,0.0.0.0,0,0,,.....,0,0,25224,1320528,0,0,0,0,0,0,0,0,0,0,0.0.0.0,0.0.0.0,0,0,00:00:00:00:00:00:00,00:00
:00:00:00:00,00:00:00:00:00:00,00:00:00:00:00:00:00,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0.0.0.0,0/0

1970-01-02 01:14:55,1970-01-02
01:16:14,79.112,33.4.1.0,0.0.0.0,0,0,,.....,0,0,28614,1487928,0,0,0,0,0,0,0,0,0,0,0.0.0.0,0.0.0.0,0,0,00:00:00:00:00:00:00,00:00
:00:00:00:00,00:00:00:00:00:00,00:00:00:00:00:00:00,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0.0.0.0,0/0

1970-01-02 01:15:56,1970-01-02
01:17:15,79.162,33.4.1.0,0.0.0.0,0,0,,.....,0,0,26681,1387412,0,0,0,0,0,0,0,0,0,0,0.0.0.0,0.0.0.0,0,0,00:00:00:00:00:00:00,00:00
:00:00:00:00,00:00:00:00:00:00:00,00:00:00:00:00:00:00,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0.0.0.0,0/0

1970-01-02 01:16:57,1970-01-02
01:18:16,79.072,33.4.1.0,0.0.0.0,0,0,,.....,0,0,25757,1344764,0,0,0,0,0,0,0,0,0,0,0.0.0.0,0.0.0.0,0,0,00:00:00:00:00:00:00,00:00
:00:00:00:00,00:00:00:00:00:00:00,00:00:00:00:00:00:00,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0.0.0.0,0/0

1970-01-02 01:17:58,1970-01-02
01:19:13,74.800,33.4.1.0,0.0.0.0,0,0,,.....,0,0,15572,809744,0,0,0,0,0,0,0,0,0,0,0.0.0.0,0.0.0.0,0,0,00:00:00:00:00:00:00,00:00:
00:00:00:00,00:00:00:00:00:00:00,00:00:00:00:00:00:00,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0-0-0,0.0.0.0,0/0

Figure 6: Netflow records.

elements of $\mathbf{x}_t$, are located at different parts of the network where the incoming traffic is registered. This information is regularly sent to a central device, where the complete $\mathbf{x}_t$ is put together. In Fig. 6, an example of Netflow file of registers (a.k.a. Netflow log) is shown. The log retains useful information about the flows, mainly, the origin and destination IP addresses, the origin and destination communication ports and the amount of packets and bytes in the flow. IP addresses identify the devices and the ports identify the computer programs (processes) in the devices that are communicating through the network. Thus, the origin and destination IP addresses and ports provide of a unique identifier for a traffic flow, *i.e.* two flows cannot share same pairs of IPs and ports.

Netflow and similar tools are resource demanding in terms of traffic generated and processing time and storage capacity in the routers. For this reason, measuring the $\mathbf{x}_t$ vector each sampling interval can be cumbersome for the network hardware. The network is designed to provide certain services to users while minimizing the cost of deployment (capital expenditures or CAPEX) among other costs. Thus, network managers are reluctant to devote a large part of the network resources to

192.168.4.11	IF-MIB::ifInOctets.1	IF-MIB::ifInOctets.2
[12:09 7/23]	814768.00	31750774.00
[12:10 7/23]	909022.00	36295730.00
[12:11 7/23]	917352.00	36802806.00
[12:12 7/23]	884206.00	34970580.00
[12:13 7/23]	893056.00	35885934.00
[12:14 7/23]	881923.00	33974831.00
[12:15 7/23]	835326.00	32906544.00
[12:16 7/23]	864102.00	34287672.00
[12:17 7/23]	939600.00	37733404.00

Figure 7: Example of SNMP log with two link counters and measurement time stamp.

obtain the TM. An alternative is to estimate it from link counts, a method referred to as network tomography [17] in the literature. Link counts refer to the number of incoming or outgoing packets/bytes in a link of the network. In these counts, the information about the origin and destination of the traffic is lost. Measuring link counts, for instance with the Simple Network Management Protocol (SNMP), is much less resource consuming than using Netflow. An example of SNMP record is shown in Fig 7. The typical location of both Netflow and SNMP sensors to measure the TM is also shown in Fig. 5.

The relationship between link counts and the TM at a given time interval t can be specified as follows:

$$\mathbf{y}_t = \mathbf{x}_t \mathbf{R}_t \tag{1}$$

where $\mathbf{y}_t$ are the link counts and $\mathbf{R}_t$ is a binary matrix with routing information, where elements $\mathbf{r}_t^{ij}$ equal to 1 when the traffic of the i-th OD is routed through the j-th link in the network and 0 otherwise. Both $\mathbf{y}_t$ and $\mathbf{R}_t$ can be easily measured in the network. The routing information is typically fixed in time or slowly varying, so that $\mathbf{R}_t$ may be replaced by $\mathbf{R}$ in eq. (1).

Estimating $\mathbf{x}_t$ from $\mathbf{y}_t$ is an ill-posed problem similar to those we encounter in chemometrics, since typically the number of OD pairs is much higher than the number of links. This is also the reason why measuring flows is more resource demanding than measuring link counts. Solutions

in the literature are based on including some additional information to the problem. Nucci and Papagiannaki [14] identify three generations of solutions:

- A first generation, e.g. [17], where a univariate distributional model for traffic flows is assumed and combined with eq. (1). The estimation performance of these methods is reduced [14].

- A second generation where additional link counts measurements are performed to improve the estimation performance. Examples of this generation are the route change [18] and the tomogravity [19] methods.

- A third generation, in which the inversion of eq. (1) is carried out with partial Netflow measurements. Using measurements of both $\mathbf{y}_t$ and $\mathbf{x}_t$, prediction models can be calibrated to establish the relationship between link counts and the TM. Examples are the fanout method in [20] and a PCA-based method proposed in [11]. For their resemblance with chemometric approaches, these methods will be treated in detail in the following. The third generation improves the estimation performance of the other two only considering a reduced amount of Netflow measurements [14].

The fanout method is based on the cyclostationary nature of network traffic. In Fig. 8 we illustrate this nature with real measurements. Typical traffic profiles of 24 hours are repetitive, passing from almost inactivity (nightly periods) to peak use during working periods. Not all the networks exhibit the same profile, since this depends on the specific activity carried out over the network and the number of connected remote sites at potentially different hour zones, but the cyclostationarity is a common feature. The fanout method makes of most of this feature to predict future traffic with measured traffic at the same time of the day.

The PCA method in [11] is based on a number of consecutive $\mathbf{x}_t$ and $\mathbf{y}_t$ measurements conforming matrices $\mathbf{X}$ and $\mathbf{Y}$. First, matrix $\mathbf{X}$ is approximated by its main PCs following the SVD of $\mathbf{X}$:

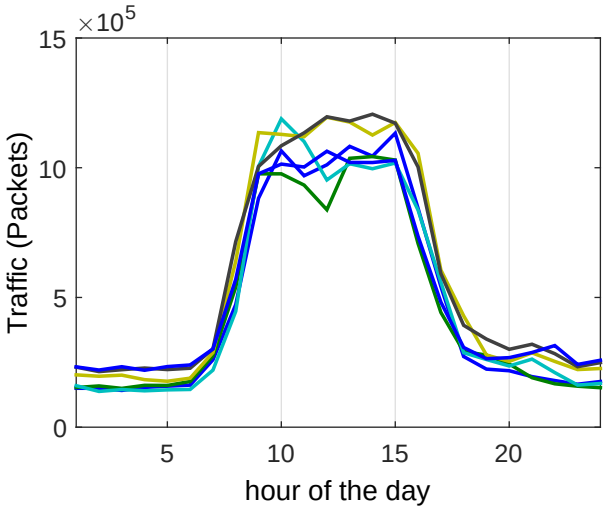


Figure 8: Cyclostationarity: amount of traffic in different days.

$$\mathbf{X} \approx \mathbf{U}_A \mathbf{S}_A \mathbf{V}'_A \tag{2}$$

Combining eqs. (1) and (2) for matrices $\mathbf{X}$ and $\mathbf{Y}$, and fixed $\mathbf{R}$, follows:

$$\mathbf{Y} \approx \mathbf{U}_A \mathbf{S}_A \mathbf{V}'_A \mathbf{R} \tag{3}$$

The relationship between $\mathbf{Y}$ and $\mathbf{U}_A$ can be established with the Moore–Penrose pseudoinverse of matrix $\mathbf{Q} = \mathbf{S}_A \mathbf{V}'_A \mathbf{R}$. The complete prediction model is the following:

$$\hat{\mathbf{X}} = \mathbf{Y}\mathbf{Q}'(\mathbf{Q}\mathbf{Q}')^{-1}\mathbf{S}_A \mathbf{V}'_A \tag{4}$$

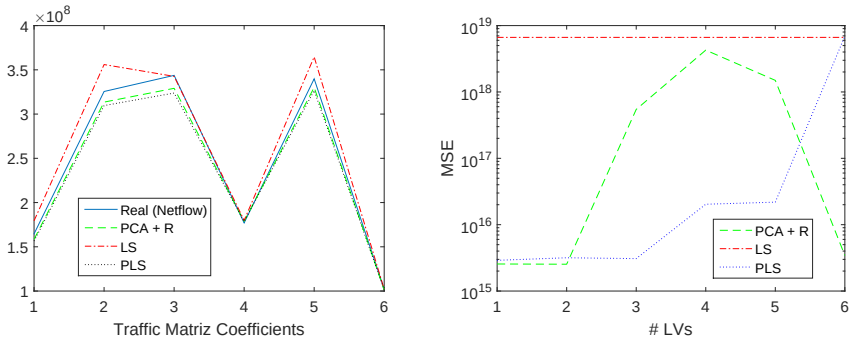
Another interesting classification of the aforementioned methods is based on the type of information contained in the models [14]:

- Spatial models: those that incorporate the relationship among OD pairs.
- Temporal models: those that incorporate univariate dynamics per OD pair.
- Spatio-Temporal models: those that incorporate dynamics and multivariate information at the same time.

The estimation of the TM can be regarded as a Big Data application, although of course this depends on the amount of data handled, whether sampling is used or not, etc. Data employed (Netflow, SNMP, routing information) are structured data, which simplifies the problem.

To exemplify the TM estimation problem, we collected Netflow and SNMP data in a teaching lab in Communications Engineering. The network interconnected three local networks that shared information. Traffic data from and to the local network was simulated using a network probe called *iperf*. Considering that the amount of traffic in one direction may be different from the other, the TM has 6 no-zero coefficients (considering zero diagonals). The amount of traffic was measured during an hour, with measurements every 1 minute interval. From the total set of 60 measurements, 40 were used for calibration and 20 for testing. Results are shown in Figure 9. Figure 9(a) compares the actual values of the TM coefficients with the estimation with the PCA method in [11] and the use of Least Squares (LS) and PLS between $\mathbf{Y}$ and $\mathbf{X}$. For PCA and PLS, the optimum number of Latent Variables (LVs) were considered, 2 and 3 respectively. In Figure 9(b), we compare the Mean Squared Error (MSE) for different LVs of the PCA and PLS estimators. LS is also included in the comparison. The results in this toy example show that the use of contextual information, in this case the routing matrix in the PCA approach, can be beneficial in the estimation. It should be noted that retrieving this information from the network may not be straightforward depending on the network design and size.

Different chemometric approaches can be applied to this problem. As shown in the example, biased regression techniques can be used to find the relationship between link counts and the TM. The combination of regression with routing information, using techniques like grey modeling [21], can also be interesting. Finally, the cyclostationarity of the traffic, which makes suitable the arrangement of the data in a 3-way matrix [15], can be modeled with batch processing chemometric approaches like [22]. Combinations of these three ideas may produce interesting approaches to calibrate spatio-temporal models for network tomography. Noticeably, Acar et *al.* [23] have already approached the TM estimation with partial Netflow measurements using chemometric tools (PARAFAC).



(a) (b)

Figure 9: Comparison between estimation methods: (a) TM estimation, and (b) Mean Squared Error of estimation.

4.2 Missing Data Imputation

Missing data imputation is another relevant estimation problem. In the context of networking, missing data imputation is of especial interest in sensor networks. A sensor networks is a group of sensor devices intended to monitor a given area by measuring several physical variables [24]. Every certain time interval, each sensor introduces its measurements in a packet that is sent to a Central Unit (CU), where the data corresponding to all the sensors are combined for analysis.

A sub-class of sensor networks with a great measurement flexibility are the Wireless Sensor Networks (WSNs). A main advantage of using wireless technologies is that sensors can be easily deployed, without the necessity of cabling installations. Sensor hardware has been greatly miniaturized, leading to the tiny sensors called *motes* (see Fig. 10). Hundreds of even thousands of these motes can be randomly distributed in an area for monitoring [24]. These sensors send information to the CU until the battery expires and need to be replaced for re-charging.

In a WSN, there are different strategies to route the packets with measurement information towards the CU. Sensors may use GPRS, a legacy mobile communications technology. With GPRS, the information can be sent to a mobile telephone antenna, and in turn to the CU through the Internet. This approach, however, is too consuming in terms of energy, reducing the sensors autonomy. Alternatively, the WSN can use the so-called *ad-hoc communication* approach, where

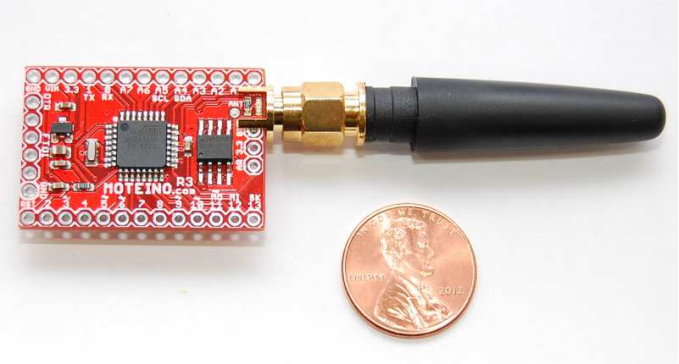


Figure 10: Illustration of a mote.

sensors send their information to sensors nearby that forward the information until it reaches the CU. Thus, only a subset of the sensors, if any, need large range communication means.

Sensor networks are used for monitoring in military, medical and/or industrial applications [25], among others. Like in any industrial set-up, there is the risk of sensor malfunction. Sensors can also be the target of hackers [26][27], with unpredictable consequences. An example of attack is a *packet dropping attack* [28], where a hacked sensor selects a subset of packets to discard (delete) at random, so that their information do not reach the CU.

When the CU cannot gather information from a set of sensors, missing data imputation can be used to estimate their values. An interesting feature of the missing data problem in a WSN is that the lost information is dependent on the routing approach that is used to send the information towards the CU. For instance, Fig. 11 illustrates a WSN where the sensors are regularly distributed. Three routing strategies, and the consequence of data loss, are shown: i) direct communication (e.g. GPRS) between each sensor and the CU (top-left corner); ii) linear routing from left to right; and iii) cluster head routing where all sensors in a square send their information towards the center, which in turn uses GPRS. Thus, in the first case, the missed data only correspond to that measured by hacked/malfunctioning sensors. However, if ad-hoc communications are used instead, a hacked/malfunctioning sensor may have a broader (and more harmful) effect, affecting data from other sensors. See the affected sensors for each routing in black color in Fig. 11.

Several missing data imputation techniques for WSNs have been recently proposed. A neural

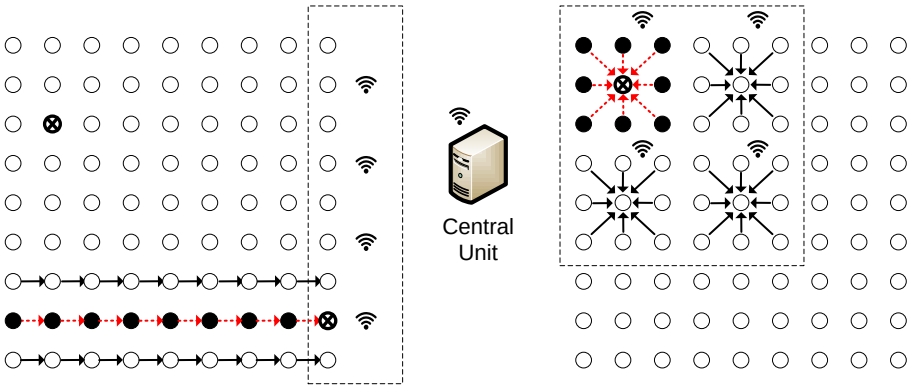


Figure 11: Illustration of a WSN with regularly distributed sensors. Hacked/malfunctioning sensors and affected sensors are highlighted for different routing strategies.

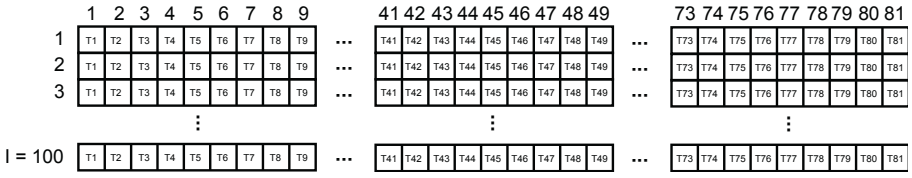


Figure 12: Global modeling approach according to [12]. Each sensor is treated as a variables.

network-based anomaly detection scheme and missing data imputation algorithm were developed in [29]. The authors in [30] introduce a data mining methodology based on exploiting spatio-temporal relationships among sensors for imputation. A missing data recovery proposal using sparsity-spatial interpolation is addressed in [31, 32]. So far, exception made on [12], the application of multivariate methodologies to WSNs is limited and mainly restricted to monitoring applications [33, 34].

To impute missing data using methods in the chemometric literature, like those in [35, 36] and references therein, data have to be arranged in a proper way [12]. A direct way to do so is combining the variables collected by each sensor in a unique, highly-multivariate, matrix. See Fig. 12 for an illustration of that data arrangement. This approach, however, may not be optimum from the recovery performance. An alternative proposed in [12] is to use what the authors call *local models*. Fig. 13 illustrates the local modeling approach. The measurement of one sensor are combined with those of its N nearest-neighbors to conform the matrix of data. Then, the

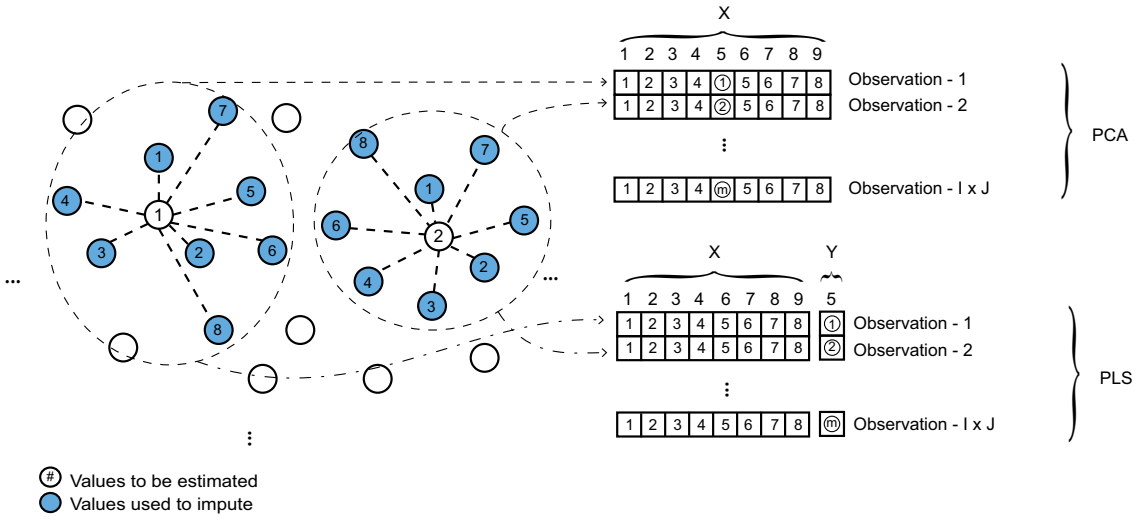


Figure 13: Local modeling approach according to [12] using the nearest-neighbors idea for PCA and PLS.

information of a given sensor is recovered using PCA modeling plus Trimmed Score Regression (TSR) [35] or PLS.

To illustrate the missing data problem in WSN, we developed a simulator [12] based on Matlab 2009b. We assumed a 1000 m×1000 m square forestry area where 81 (9×9) temperature sensors are regularly distributed, *i.e.*, each sensor is located ∼100 m away from its neighbors. Every sensor gathers the ambient temperature each sampling time and sends the measurements to the CU. The proposed deployment for the measurements is inspired in a real system provided by the Libelium company[4]. In Table 2, estimation error values in terms of the Mean Squared Error (MSE) are provided for different modeling and routing strategies. Two conclusions can be drawn from the results: i) there is a strong interplay between routing and imputation performance, and ii) local models outperform global models in several orders of magnitude.

The study of alternative data arrangements [37], as well as alternative missing data imputation methods [38, 35, 36], can be an interesting research opportunity in order to improve data imputation performance in this problem.

[4]http://www.libelium.com/wireless_sensor_networks_to_detec_forest_fires/

	Global model	Local model	
Attack scenarios	**MSE (TSR-PCA)**	**MSE (TSR-PCA)**	**MSE (TSR-PLS)**
Direct communication	1900.8	2.7	3.4
Linear routing	2472.0	67.8	71.1
Cluster head	4391.6	149.6	149.6

Table 2: MSE comparison between global and local models for the routing approaches illustrated in Fig. 11.

5 Anomaly Detection

The outstanding capability of multivariate analysis to detect anomalies has been recognized also in networking [39, 40, 41, 42, 43]. The pioneering work by Lakhina et *al.* [39] introduced the use of PCA for network anomaly detection. Their approach received a lot of attention from the networking community one decade ago, and most of the existent works on PCA-based anomaly detection in networking are developed taking their work as a base.

The approach of Lakhina et *al.* was inspired by the chemical engineering literature. However, there are main differences between this approach and the state-of-the-art in MSPC with PCA:

- Lakhina et *al.* use PCA to divide data in two subspaces for normal and anomalous behavior. Anomaly detection is performed only in the latter. In MSPC, PCA is used to split data in a structured subspace and a residual subspace. Detection is performed in both subspaces using different statistics [44, 45, 46, 47].

- Lakhina et *al.* use data for the calibration of the anomaly detector that may incorporate anomalies. In MSPC, a two phases approach is performed so as to avoid this problem [22, 48, 49].

- Lakhina et *al.* select the number of Principal Components to capture a specific amount of variance. Subsequently, in [50] they suggest the use of the Scree plot. These approaches are well known to be impractical in most MSPC set-ups [51, 52].

- Lakhina et *al.* use a supervised fault diagnosis system referred to as Reconstruction-Based Contributions (RBC) [53], which were defined as an alternative to contribution plots. In the original definition, RBC plots require an a priori set of common faults. Subsequently, Lakhina et *al.* extended in [50] their approach for unsupervised diagnosis, yet assuming a predefined structure in the fault. In MSPC, contribution plots are commonly used for faults diagnosis without the need of any set of previously defined faults [44, 54, 55].

Ringberg et *al.* [43] claimed that the approach of Lakhina et *al.* is sensitive to calibration settings. However, this is the consequence of the flaws in the approach, as the sensitivity problem was solved by applying state-of-the-art MSPC [56]. We call Multivariate Statistical Network Monitoring (MSNM) the approach that follows the MSPC theory for anomaly detection in communication networks. MSNM has specificities not found in MSPC.

A main challenge in MSNM, and also a main difference between MSPC and MSNM, is the data preparation[5]. Let us illustrate this in the most simple setting, that of two-way analysis– e.g. MSPC of continuous processes. In MSPC, monitored variables, like temperatures, pressures, concentrations, etc., are directly measured from the process. Thus, none or very little preparation is needed. Observations are typically ordered in time, for regular or variable sampling rates, and again with little or none pre-processing.

In MSNM we have the opposite situation. In a network, most of the information comes in the form of logs or packets of data, unstructured information that cannot be directly used in a MSNM set-up. Rather, logs and packets need to be translated into quantitative variables, and there is a bunch of possibilities to do so. This is typically referred to as *data parsing* or *feature engineering* [57]. The parsing needs to be programmed for very different sources of data, from structured information similar to that in the process industry, to completely unstructured/textual information like the e-mails. Thus, the parsing is in charge of converting textual information into quantitative variables of value for anomaly detection. Clearly, there is no systematic way to do

[5]Data preparation is tipically referred to as data preprocessing in the networking community, though the meaning of preprocessing is different from that in chemometrics.

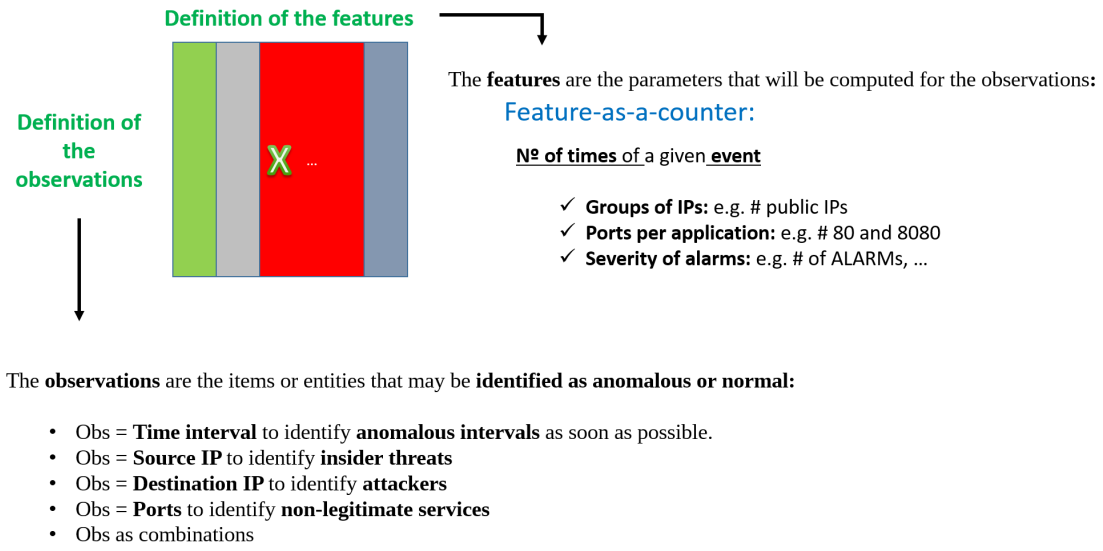
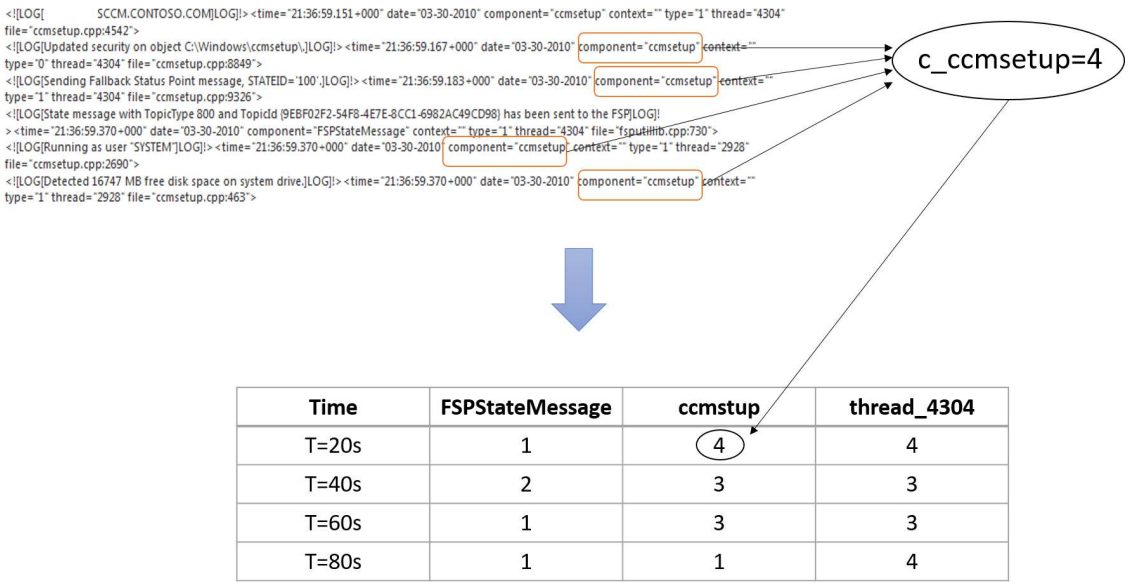


Figure 14: Diagram illustrating the flexibility of MSNM.

this, which opens very interesting research directions.

On the other hand, the definition of the observations in MSNM is not straightforward. Although observations are typically ordered in time, it may be interesting to define the observations in terms of relevant entities in a network, such as source or destination devices or protocols being used in the network. Combining this with temporal information generates a three-way matrix of data. However, the number of devices and protocols involved in communications through the network is varying in time, which makes the multivariate analysis of these entities specially challenging. The flexibility in the definition of both variables and observations in MSNM, see Fig. 14, makes it more challenging but also more powerful than traditional MSPC.

Lakhina et *al.* [39] proposed the definition of counters as quantitative variables. The counters were restricted to counts of packets (datagrams) and bytes arranged by origin-destination. Camacho et *al.* [2] generalized this definition to consider several sources of data. They proposed the *feature-as-a-counter* approach, so that variables are basically counters for the number of associated events (see Figure 15). Another class of variables are those representing a sample distribution. These are commonly more suitable than counters to summarize the information in



Time	FSPStateMessage	ccmstup	thread_4304
T=20s	1	4	4
T=40s	2	3	3
T=60s	1	3	3
T=80s	1	1	4

Figure 15: Illustration of the feature-as-a-counter approach.

traffic data, and less sensitive to packet sampling according to [50]. Histograms are useful artifacts to transform a distribution into one or more quantitative variables. Alternatively, the histograms can be summarized, for which measures of central tendency (*e.g.*, averages) and dispersion (*e.g.*, standard deviations) may be adequate. Also, Lakhina et *al.* proposed in [50] the use of entropy to summarize the information in very large histograms, such as those obtained in terms of source devices. Callegari et *al.* [58] also proposed the use of the Kullback-Leibler (K-L) divergence to capture dynamical information.

Let us briefly discuss an example of MSNM in Big Data. This was an study hired by Protectwise Inc. (http://www.protectwise.com), network security company based in Colorado, USA. The goal was to investigate the potentialities of MSPC in network security. Protectwise provides of security services to other companies. For that, they monitor several sources of information from the network of their clients. This generates tons of data, and most anomaly detectors generate a great amount of false positives. We applied standard MSPC techniques coupled with the Big Data extensions in the MEDA Toolbox [8]. In Fig. 16 we show an example of monitoring charts for one of the clients of Protectwise. The excursions highlighted anomalous events, which where diagnosed using

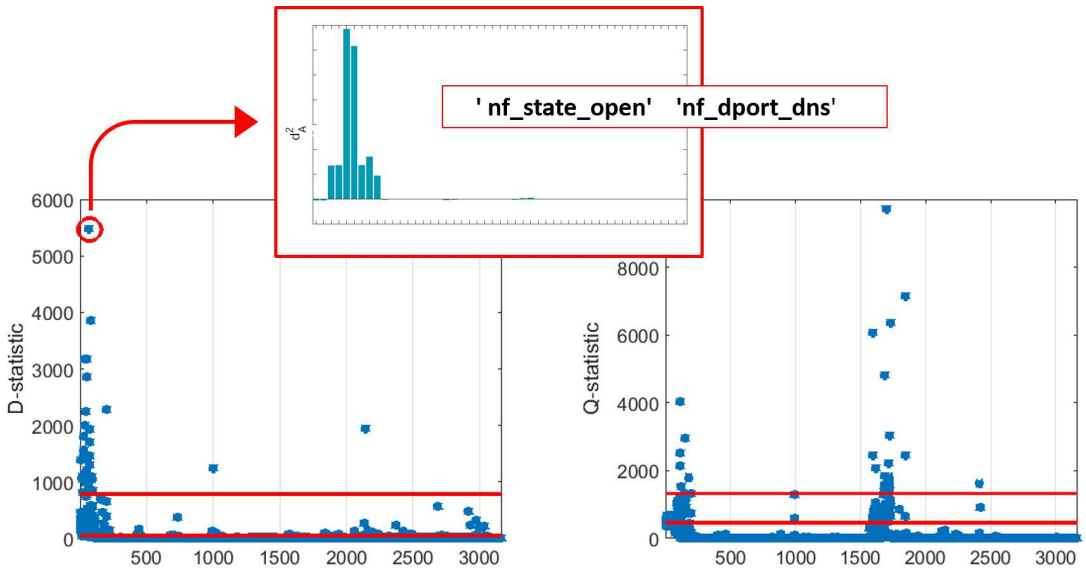


Figure 16: Monitoring charts (D-statistic or Hotelling T^2 [60] for the model space and Q-statistic for the residual space [61]) and oMEDA diagnosis plot on the top showing the index d_A^2 [59]. From the features highlighted at the top, the anomaly is related to a larger than usual number of DNS connections that remain open.

oMEDA plots [59] with the index d_A^2 [6]. Also, each dot in the charts was connected with a Big Data Latent Model from where Compressed Score Plots (CSP) [9] for different objects definition could be issued (see Fig. 17 for an CSP in the IPs and another CSP in the ports). A CSP is a plot of clusters of scores in a given subspace. In Fig. 17, each single element in the D-statistic represents a time interval where many network items (IPs or ports) are active. The CSP shows detail on this, pointing to the main active network elements. Thus, combining MSPC plots with CSP plots we can detect anomalies and locate them in the network. Apart from these visualizations, a report was issued with a list of anomalies and related variables, devices and services in the network.

Clearly, there are tons of MSPC contributions in chemometrics that can be seamlessly applied to MSNM. However, the flexibility on the definition of variables and observations is a new challenge to deal with.

[6] This type of plot is an alternative to contribution plots. Please refer to the cited reference for more information.

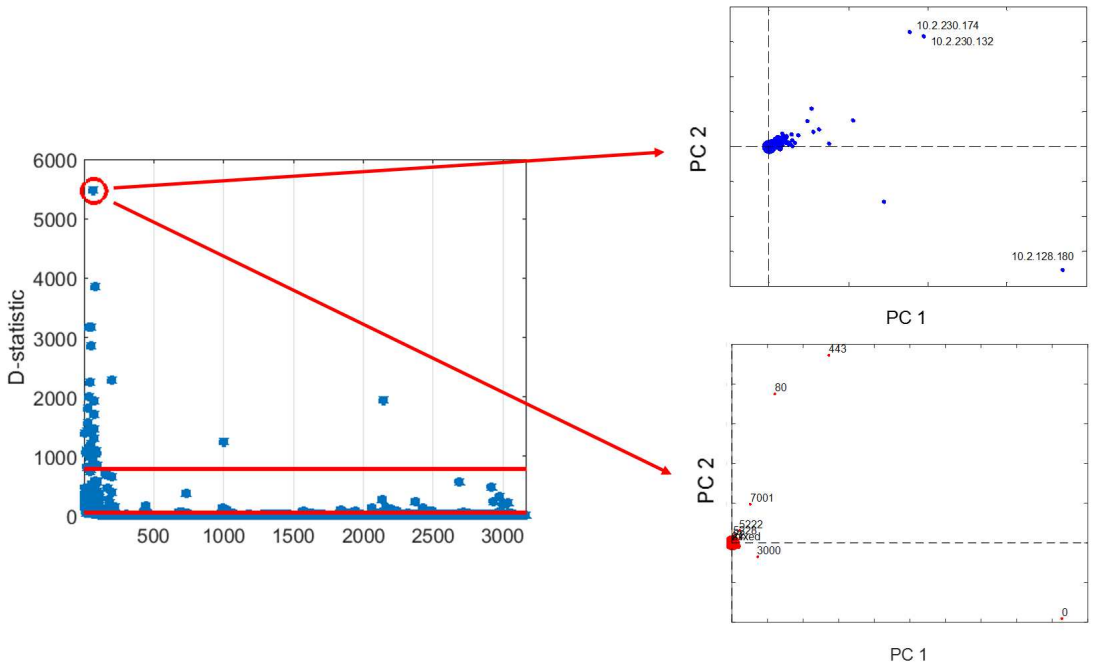


Figure 17: Compressed Score Plots for several thousands of observations, corresponding to one of the anomalous data points in the D-statistic. For the same point, we show the CSP for IP addresses (top-right) and ports (bottom-right), providing a more complete visual information.

6 Optimization

Like in many other fields of knowledge, optimization problems are central in chemometrics. The formulation of a problem as a mathematical optimization implies the definition of an objective function $f(\mathbf{x})$ to minimize or maximize within a search space S, where the best solution $\mathbf{x}^*$ is to be found. This space is often constrained by equality and/or inequality constraints, $g_i(\mathbf{x})$ and $h_j(\mathbf{x})$. The optimization can be generally formulated as follows, where the goal is to find the values in $\mathbf{x}$ that minimize $f(\mathbf{x})$:

$$\mathbf{x}^* := \underset{\mathbf{x} \in S}{\arg\min} \{ f(\mathbf{x}) \mid g_i(\mathbf{x}) \leq 0, \ for\ i = 1...I\ and\ h_j(\mathbf{x}) = 0, \ for\ j = 1...J \} \tag{5}$$

To formulate the optimization problem we need to define $\mathbf{x}$, the search space S and the functions $f(\mathbf{x})$, $g_i(\mathbf{x})$ and $h_j(\mathbf{x})$. Once formulated, the solution is obtained using an adequate algorithm,

sometimes referred to as a *solver*. Many times the optimization problem is too cumbersome, or the search space too wide, to obtain the global optimal solution, and we rely on approximate algorithms to find a sub-optimal solution. For instance, heuristic bio-inspired optimization algorithms, like Particle Swarm Optimization (PSO) [62] or genetic algorithms [63], have been used for chemometric problems.

In some optimization problems, parametric analytical solutions can be obtained from algebra without the need of a solver. In those cases, the optimum is obtained from a single expression or an iterative succession of expressions. This is for instance the case of many chemometric calibration algorithms like NIPALS.

Finally, there are optimization problems where we do not know the objective function $f(\mathbf{x})$ explicitly. Instead, we have pairs of measurements of $\mathbf{x}$ and $\mathbf{y} = f(\mathbf{x})$. Variables in $\mathbf{x}$ are often referred to as independent variables, while variables in $\mathbf{y}$ are the responses or dependent variables. A form of pseudo-optimization in this class, which seeks to minimize the number of measurement pairs in $\mathbf{x}$ and $\mathbf{y}$, is the design of experiments [64], central in chemometrics. Another typical example of data-driven optimization is that used for process optimization, for instance in run-to-run optimization (see [65] and references therein).

Optimization is also principal in networking. Tasks like network design, capacity planning[7], traffic engineering[8], etc., can be defined as purely optimization problems [66, 67]. An interesting reference for that is the book by Pióro and Medhi [68], where explicit formulations are provided for different optimization problems in networking. An example of formulation is that of capacity problems, where the traffic capacity of the links of a network are optimized taking as input the TM previously estimated according to Section 4.

However, as already discussed, there are cases in which we cannot define an explicit formulation of the problem, because of its complexity. For instance, take the case of wireless networks, or more specifically of ad-hoc wireless networks, where all network devices are allowed to make direct

[7]Capacity planning refers to the choice of the capacity of the links in the network according to the traffic demand.

[8]Traffic engineering refers to a set of mechanisms to manage traffic according to performance goals.

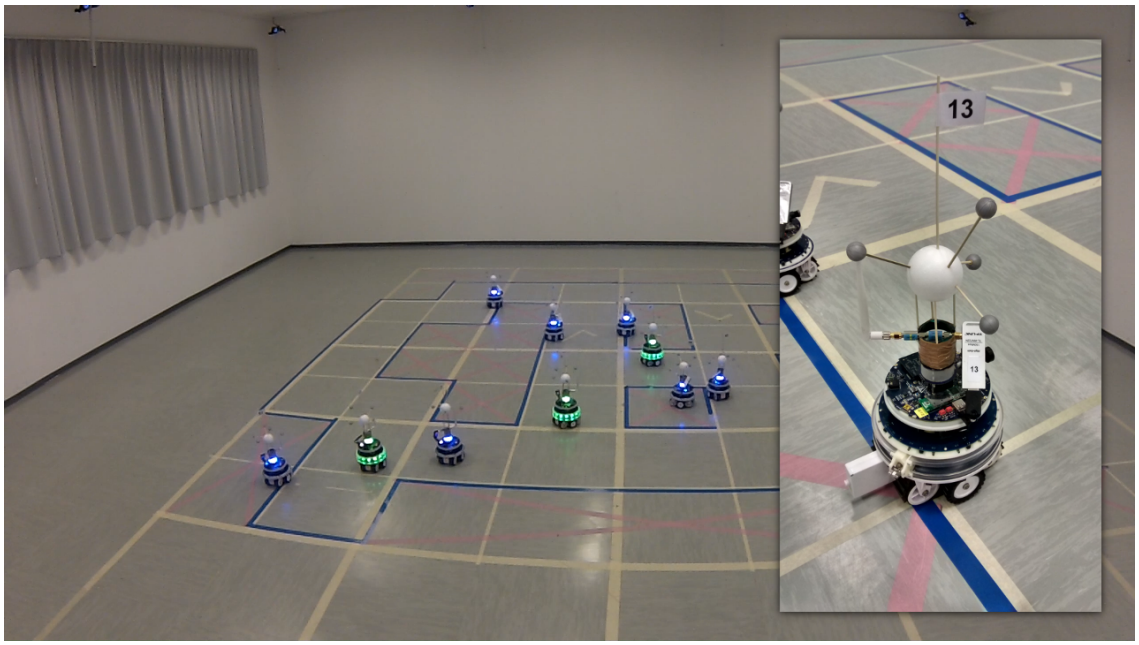


Figure 18: Robotic ad-hoc network at IDSIA lab (http://www.idsia.ch): communicating nodes have blue light while relay nodes have green light. Rely nodes choose their location to allow the communication among all the nodes .

communications. In this type of networks, existent communications links depend on the distance of the communicating devices. Thus, if two devices are close enough, they can communicate through a direct wireless link. Otherwise, they need to send their communication through relaying devices. A very interesting problem [69] is how to place the relaying devices in the network to maximize the communications. Even more challenging is that problem when communicating devices are allowed to move (see Fig. 18 for an illustration with a robotic network). The relay placement problem will give way to networks based on autonomous relaying devices like mobile or flying robots (drones). Robots act like communication antennas in the ad-hoc network, but with the additional movement capability. Thus, they can relocate to improve communications as the other devices, e.g. mobile phones, other robots, etc., also move. This is especially interesting for certain applications, like military actions or in crisis management and disaster recovery, where the communication infrastructure of an area needs to be recovered to facilitate rescue missions.

If the placement problem is explicitly formulated, the potential number of variables in $\mathbf{x}$ in eq. 5 would be too demanding for finding a solution with common solvers. Furthermore, because of the mobile and wireless nature of the network, the optimal solution is continuously changing, which introduces an upper limit for the time to obtain a solution. Although Big Data platforms could help in that, a common alternative is to approach this problem with heuristic optimization methods like PSO [69, 70].

The relay placement problem in ad-hoc networks can be tackled with heuristic optimizations. In particular, Fig. 19 shows a comparison between PSO and the PLS optimization algorithm proposed in [65]. This comparison was performed in a simulator based on Matlab [69]. We chose a deployment rectangular area of $6.6m \times 5.4m$ with a node coverage range of $1m$ assuring network disconnections. The connectivity is measured as a percentage of interconnected nodes, measured from 0 (no nodes interconnected) to 1 (all nodes interconnected). Results show that the optimization algorithms provide effective relay placements, since the connectivity grows with the number of relay nodes. Furthermore, we can see that both PSO and the PLS optimizer attain similar results.

The previous example illustrates a way to use data-driven chemometric methods in network optimization. In a similar way, DoE can be applied in optimization problems for fixed topology networks, especially when deciding the number and location of network elements. On the other hand, considering the cyclostationrity of most networks, the batch processing parallelism also holds in optimization, and techniques applied to optimize batch processes [65] can be applied to network optimization. In particular, mid-course corrections [71, 72] are suitable for the daily optimization operations typical in traffic engineering.

7 Classification

A number of classification problems arise in networking, two of the most popular ones being those of malware and traffic classification. They are further described in the next.

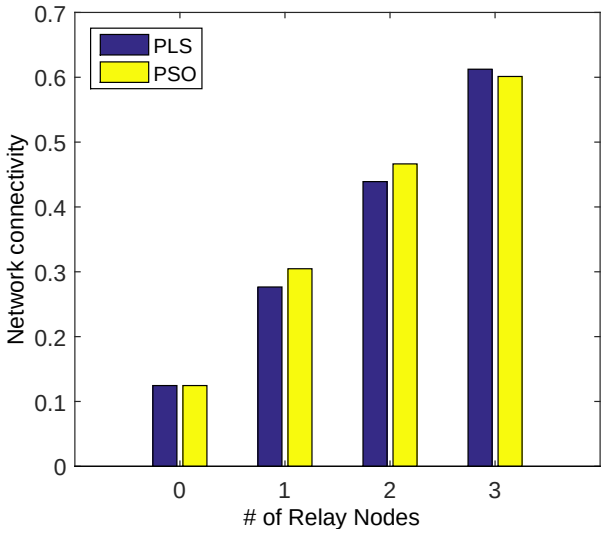


Figure 19: Comparison in terms of connectivity of two optimization algorithms in the problem of relay node placement in a mobile ad-hoc network. The algorithms are the PSO and the PLS optimization algorithm of [65].

The term 'malware' means malicious software, that is, software mainly intended to carry out some harmful actions. There is a continuous race between antivirus companies and malware developers. Antivirus detect malware by looking for patterns in the malware files. However, recent tools for automatic malware generation have made possible to generate malware without technical knowledge, which has given way to an increase of the amount of malware generated. In this context, automatic classification [73] can be of use to simplify malware detection.

The type of data analysis techniques used in detection and classification of malware are diverse. From bayesian-based [74] to SVM-based [75], as well as heuristic-based [76]. Others, like [77], rely on graph related techniques to classify malware. Authors in [78] introduce an immune-based system for malware detection in smartphones, while Zolotukhin *et al.* combine in [79] SVM, genetic and game-based techniques for detection and classification.

In malware classification almost all current proposals make use of a variety of feature types over which the detection/classification procedure is carried out. Most of the data used represent behavioral information of the device: file system, network activity, OS commands, etc. Again,

most of this information is unstructured. So far, very little attempts to use multivariate analysis can be found in the literature [80].

Another main classification problem in networking is traffic classification, where the protocol or type of protocol of a traffic flow is identified. The goal, thus, is to determine the type of information a datagram contains, so as to infer the use of services in a network. This is a matter of great importance for different applications. In the security ambit, traffic identification is a main information source for anomaly detection. Since each type of service has its own traffic particular features, from the knowledge of the used services we can estimate the behavior of the traffic. Therefore, the traffic identification is also principal for network optimization and maintenance [81] and to establish different priority levels for the network traffic [82].

There are three main network traffic classification strategies [83]: one based on the communication ports [84], another based on the packet content (payload) inspection [85], and a third one based on the application of machine learning techniques over traffic statistics [86]. The first two approaches present severe limitations [87] and the research community has moved towards the use of classifiers over traffic statistics [83].

There are three choices to make in the design of a classifier over traffic statistics. First, traffic is a mixture of structured and non-structured information. The proper selection of the features that characterize the network traffic is essential for an adequate classification, and a wide variety of proposals exists [88, 89, 90, 91]. Once the traffic has been parameterized, the classification can be performed at different aggregation levels [92, 93]. The last choice is the classification method, and again there is a bunch of methods in the literature for that [94, 86], including Support Vector Machines (SVMs), Hidden Markov Models (HMM), k-Nearest Neighbors (kNN), etc.

Most recent publications in the literature claim results of very high classification accuracy, very close to a 100%. However, these results are at the expense of high dependence on the considered scenario and the specific traffic conditions. This lack of generalization questions the practical applicability of the methods, since a network is a dynamic entity in ever changing conditions. In recognition of this problem, Li et *al.* [95] compared the stability of a number of classification

approaches. Also, Camacho et *al.* [87] showed the limitations of high accuracy classifiers. This reference shows that achieving generalization ability in a classifier from traffic statistics is very challenging, but doing it from dynamical relationships is simple. The authors propose a dynamic model with simple rules to relate traffic flows with past flows associated to the same service, so that only a minimum percentage of flows need to be actually classified.

Potential contributions of chemometrics to traffic classification algorithms are similar than in previous applications. As already discussed, chemometric tools are suited to deal with the high number of potential features for parameterizing unstructured information. Variable selection procedures [96] can be helpful to that end. Also, the use of three-way models to account for the cyclostationarity of the traffic may lead to more generalizable methods. To the best of our knowledge, the notion of cyclostationarity has not been applied to traffic classification. Finally, the use of dynamical models in the mode of the days to update the classifiers, e.g [97], may also be convenient.

Surely, the multivariate nature of some of the classification methods like PLS-Discriminant Analysis (PLS-DA) [98] can be helpful in the networking domain. However, in agreement with the thesis in [99], the most interesting potential contribution may be the exploratory capability of the methods. Let us take the following example using a data-set [100] available at the CRAWDAD repository (http://crawdad.cs.dartmouth.edu/) and published in [101]. It consists of an outdoor experiment for the comparison of four different communication protocols in a mobile ad-hoc network formed by 33 laptops in movement. The evaluated protocols are: Any Path Routing without Loops (APRL), Ad hoc On-demand Distance Vector (AODV), On-Demand Multicast Routing Protocol (ODMRP) and System- and Traffic-dependent Adaptive Routing Algorithm (STARA). Rather than a traffic classification problem, this is a traffic understanding problem, but the goal is to understand differences among protocols, which is quite close to classification.

The main results provided in [101] are shown in Table 3. For an optimum performance, a communication protocol should present the highest delivery ratio [9] using the lowest amount of

[9]The message delivery ratio is the percentage of datagrams reaching the destination.

Table 3: Results provided in [101].

Routing algorithm	Message delivery ratio	Packets per message	Average number of hops
AODV	0.50	7.50	1.61
APRL	0.20	33.30	2.11
ODMRP	0.77	45.59	2.47
STARA	0.08	150.67	1.18

traffic generation, that is, minimum number of packets and hops[10]. According to the table, ODMRP is the protocol that attains the highest message delivery ratio and STARA the one attaining the lowest. On the other hand, AODV is the protocol that generates the lowest amount of packets per message. Finally, ODMRP and APRL show a high hop ratio in their routes. These results lead to the conclusion that AODV is the best trade-off between message delivery and traffic generation.

In this example, an exploratory analysis with PLS-DA is used to unveil more details of the experiment. For this purpose, a set of new statistics are computed from the original data at regular intervals of time. The first 10 variables are related to the distribution and location of the devices (laptops) in the field, while the remaining 8 variables are related to the network traffic.

The score plot, in Fig. 20, shows that the observations related to each algorithm are easily distinguished with the designed variables. In Fig. 21, the differences between AODV and APRL are studied with oMEDA [59]. Surprisingly, the main differences found are related to the laptop spatial distribution and not to the routing performance. During the experiment, when APRL was used, the laptop distribution presented a higher dispersion. Clearly, under these circumstances, the communication with AODV is less challenging than with APRL just because of the location of the laptops, not the routing algorithm. This feature can be also observed when comparing APRL with the rest of the routing algorithms (not shown). In this situation, where the dispersion of the stations are significantly higher for APRL than for the others, it is not possible to carry out

[10]The hop is the number of intermediate nodes between a source and a destination, which depends on the routing algorithm that decides the paths of the datagrams.

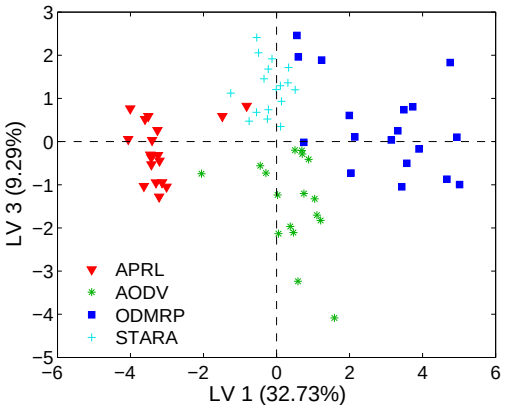


Figure 20: PLS-DA score plot for the MANET experiment data.

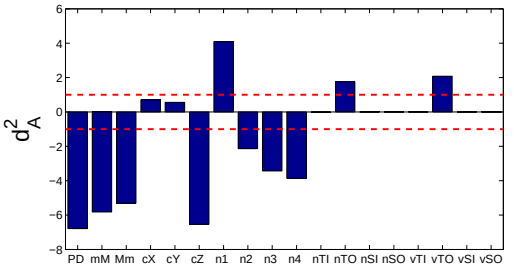


Figure 21: oMEDA plot showing the index d_A^2 [59] for comparison between the routing algorithms AODV vs APRL.

a reliable comparison with the rest of the algorithms. Stating otherwise, the comparison is not fair because APRL is working on a more complicate scenario than its opponents. Therefore, the results provided in Table 3 for APRL must not be taken into account.

The clear benefit in the application of the multivariate exploratory methodology in this example is the better understanding of the multivariate nature in the data. When analyzing data sets using traditional methods, the analyst needs to summarize the variables in a reduced set of statistics. This may obscure the truth underlaying the data, as it was the case in the example. More details on this example can be found on the Networkmetrics technical report in the MEDA toolbox [8].

8 Conclusion

In this paper, we have built upon the concept we call networkmetrics: using the multivariate analysis perspective in networking problems. The paper illustrates, through a number of examples, the potentiality on the use of multivariate analysis methods developed mainly in the chemometric area. A recurrent parallelism is done with three-way time series modeling, like in batch processes modeling, but other tools like biased regression, missing-value imputation, multi-block methods, variable selection, grey modeling, among others, are valuable in different networkmetric problems.

The extension of chemometric methods to networkmetrics is not straightforward, since the special features of networkmetrics, mainly its Big Data nature and the need to handle unstructured data, complicate the application of multivariate methods. These are challenging problems that open many research directions, where the contribution of experts on multivariate analysis is encouraged. On the other hand, understanding the problems and solutions applied by the networking community may provide new strategies for tackling chemometric problems, especially in a world where more and more data are becoming available.

Acknowledgements

This work is partly supported by the Spanish Ministry of Economy and Competitiveness and FEDER funds through project TIN2014-60346-R. We would like to thank Julio Piñar-Figueroa for the design of several of the illustrative figures of this paper, and Juan Manuel Martín-Doñas and Jose Antonio Rodríguez-Fernández for obtaining the Netflow and SNMP data in the lab.

References

[1] Stallings William. *Data and Computer Communications (5th Ed.)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc. 1997.

[2] Camacho J, Maciá-Fernández G, Díaz-Verdejo J, García-Teodoro P. Tackling the big data 4 Vs for anomaly detection. *Proceedings of the IEEE Conference on Computer Communica-*

tions Workshops (INFOCOM). 2014:500–505.

[3] Schroeck M, Shockley R, Smart J, Romero-Morales D, Tufano P. Analytics: The Real-World Use of Big Data. ibm institute for business value - executive reportIBM Institute for Business Value 2012.

[4] Rabani E, Toledo S. Out-of-Core SVD and QR Decompositions. in *Proceedings of the 10th SIAM Conference on Parallel Processing for Scientific Computing* 2001.

[5] Halko N, Martinsson Per-G, Shkolnisky Y, Tygert M. An Algorithm for the Principal Component Analysis of Large Data Sets *SIAM J. Scientific Computing.* 2011;33:2580–2594.

[6] Vogt F, Tacke M. Fast principal component analysis of large data sets. *Chemometrics and Intelligent Laboratory Systems.* 2001;59:1–18.

[7] Ellis G, Dix A. A taxonomy of clutter reduction for information visualization. *IEEE Transactions on Visualization and Computer Graphics.* 2007;13:1216–1223.

[8] Camacho J, Pérez-Villegas A, Rodríguez-Gómez R.A, Jiménez-Mañas E. Multivariate Exploratory Data Analysis (MEDA) Toolbox for Matlab. *Chemometrics and Intelligent Laboratory Systems.* 2015;143:49–57.

[9] Camacho J. Visualizing Big data with Compressed Score Plots: Approach and research challenges. *Chemometrics and Intelligent Laboratory Systems.* 2014;135:110–125.

[10] Medina A, Fraleigh C, Taft N, Bhattacharyya S, Diot C. Taxonomy of IP traffic matrices 2002.

[11] Lakhina A, Papagiannaki K, Crovella M., Diot C, Kolaczyk E.D, Taft N. Structural Analysis of Network Traffic Flows *SIGMETRICS Perform. Eval. Rev.*. 2004;32:61–72.

[12] Magán-Carrión R, Camacho J, García-Teodoro P. Multivariate Statistical Approach for Anomaly Detection and Lost Data Recovery in Wireless Sensor Networks. *International Journal of Distributed Sensor Networks.* 2015;2015:1-20.

[13] Gollmann D, Gurikov P, Isakov A, Krotofil M, Larsen J, Winnicki A. Cyber-Physical Systems Security: Experimental Analysis of a Vinyl Acetate Monomer Plant. in *Proceedings of the 1st ACM Workshop on Cyber-Physical System Security (CPSS '15)*:1–12 2015.

[14] Nucci A, Papagiannaki K. *Design, Measurement and Management of Large-Scale IP Networks - Bridging the Gap between Theory and Practice.* Cambridge University Press 2008.

[15] Tune P, Roughan M. Internet Traffic Matrices: A Primer. in *Proccedings of Recent Advances in Networking - ACM SIGCOMM*;1 2013.

[16] Introduction to Cisco IOS NetFlow - A Technical Overview. http://www.cisco.com/c/en/us/products/collateral/ios-nx-os-software/ios-netflow/prod_white_paper0900aecd80406232.html

[17] Vardi Y. Network Tomography: Estimating Source-Destination Traffic Intensities from Link Data. *Journal of the American Statistical Association.* 1996;91:365–377.

[18] Soule A, Nucci A, Cruz R, Leonardi E, Taft N. How to Identify and Estimate the Largest Traffic Matrix Elements in a Dynamic Environment. *SIGMETRICS Perform. Eval. Rev..* 2004;32:73–84.

[19] Zhang Y, Roughan M, Duffield N, Greenberg A. Fast Accurate Computation of Large-scale IP Traffic Matrices from Link Loads. *SIGMETRICS Perform. Eval. Rev.* 2003;31:206–217.

[20] Papagiannaki K, Taft N, Lakhina A. A Distributed Approach to Measure IP Traffic Matrices in *Proceedings of the 4th ACM SIGCOMM Conference on Internet Measurement (IMC '04)*:161–174 2004.

[21] Gurden S.P, Westerhuis J.A, Bijlsma S, Smilde A.K. Modelling of spectroscopic batch process data using grey models to incorporate external information. *Journal of Chemometrics.* 2001;15:101–121.

[22] Nomikos P, MacGregor J.F. Monitoring batch processes using multiway principal component analysis. *AIChE Journal.* 1994;40:1361–1375.

[23] Acar E, Dunlavy D.M, Kolda T.G, Mørup M. Scalable tensor factorizations for incomplete data. *Chemometrics and Intelligent Laboratory Systems.* 2011;106:41–56.

[24] Al-Karaki J.N, Kamal A.E. Routing techniques in wireless sensor networks: a survey. *IEEE Wireless Communications.* 2004;11:6–28.

[25] Yick J, Mukherjee B, Ghosal D. Wireless sensor network survey. *Computer Networks.* 2008;52:2292–2330.

[26] Xiangqian Chen K, Makki K.Y, N Pissinou. Sensor network security: a survey. *IEEE Communications Surveys & Tutorials.* 2009;11:52–73.

[27] Wood A.D, Stankovic J.A. Denial of service in sensor networks. *IEEE Computer.* 2002;35:54–62.

[28] García-Teodoro P, Sánchez-Casado L, Maciá-Fernández G. Taxonomy and Holistic Detection of Security Attacks in MANETs. in *Security for Multihop Wireless Networks* (Khan Shafiullah, Lloret Mauri Jaime. , eds.):1–12. CRC Press 2014.

[29] Li Y, Parker L.E. A spatial-temporal imputation technique for classification with missing data in a wireless sensor network. in *Proccedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*:3272–3279 2008.

[30] Gruenwald L, Sadik M.S, Shukla R, Yang H. DEMS: A Data Mining Based Technique to Handle Missing Data in Mobile Sensor Network Applications. in *Proceedings of the Seventh International Workshop on Data Management for Sensor Networks (DMSN '10)*:26–32 2010.

[31] D Guo, X Qu, L Huang, Y Yao. Sparsity-Based Spatial Interpolation in Wireless Sensor Networks. *Sensors Journal.* 2011;11:2385–2407.

[32] D Guo, X Qu, L Huang, Y Yao, M.T Sun. Sparsity-Based Online Missing Data Recovery Using Overcomplete Dictionary. *Sensors Journal.* 2012;12:2485–2495.

[33] Livani M.A, Abadi M. A PCA-based distributed approach for intrusion detection in wireless sensor networks. in *Proceedings of the 2011 International Symposium on Computer Networks and Distributed Systems (CNDS)*:55–60 2011.

[34] Chitradevi N, Baskaran K, Palanisamy V, Aswini D. Designing an Efficient PCA Based Data Model for Wireless Sensor Networks. in *Proceedings of the 1st International Conference on Wireless Technologies for Humanitarian Relief (ACWR '11)*:147–154 2011.

[35] Arteaga F, Ferrer A. Dealing with missing data in MSPC: several methods, different interpretations, some examples. *Journal of Chemometrics.* 2002;16:408–418.

[36] Arteaga F, Ferrer A. Framework for regression-based missing data imputation methods in on-line MSPC. *Journal of Chemometrics.* 2005;19:439–447.

[37] Camacho J, Picó J, Ferrer A. Bilinear modelling of batch processes. Part I: theoretical discussion. *Journal of Chemometrics.* 2008;22:299–308.

[38] Nelson P.R.C., Taylor P.A., MacGregor J.F.. Missing data methods in PCA and PLS: score calculations with incomplete observations *Chemometrics and Intelligent Laboratory Systems.* 1996;35:45–65.

[39] Lakhina A, Crovella M, Diot C. Diagnosing network-wide traffic anomalies. *ACM SIGCOMM Computer Communication Review.* 2004;34:219.

[40] Chatzigiannakis V, Papavassiliou S, Androulidakis G. Improving network anomaly detection effectiveness via an integrated multi-metric-multi-link (M3L) PCA-based approach. *Security and Communications Networks.* 2009;2:289–304.

[41] Münz G. *Traffic Anomaly Detection and Cause Identification Using Flow-Level Measurements.* PhD thesis 2010.

[42] Brauckhoff D, Salamatian K, May M. Applying PCA for Traffic Anomaly Detection: Problems and Solutions. in *Proceedings of IEEE INFOCOM*:2866–2870 2009.

[43] Ringberg H, Soule A, Rexford J, Diot C. Sensitivity of PCA for traffic anomaly detection. *ACM SIGMETRICS Performance Evaluation Review.* 2007;35:109.

[44] MacGregor J.F, Kourti T. Statistical process control of multivariate processes. *Control Engineering Practice.* 1995;3:403–414.

[45] Jackson J.E. *A user's guide to principal components.* Wiley series in probability and mathematical statistics. Probability and mathematical statistics.Wiley-Interscience 2003.

[46] Kresta J.V, Macgregor J.F, Marlin T.E. Multivariate statistical monitoring of process operating performance. *The Canadian Journal of Chemical Engineering.* 1991;69:35–47.

[47] Wise B.M, Ricker N.L, Veltkamp D.F, Kowalski B.R. Theoretical basis for the use of principal component models for monitoring multivariate processes. *Process Control and Quality.* 1990;1:41–51.

[48] Tracy N.D, Young J.C, Mason R.L. Multivariate Control Charts for Individual Observations. *Journal of Quality Technology.* 1992;24:88–95.

[49] Ferrer A. Latent Structures-Based Multivariate Statistical Process Control: A Paradigm Shift. *Quality Engineering.* 2014;26:72–91.

[50] Lakhina A, Crovella M, Diot C. Mining anomalies using traffic feature distributions. *ACM SIGCOMM Computer Communication Review.* 2005;35:217.

[51] Camacho J. *New Methods Based on the Projection to Latent Structures for Monitoring, Prediction and Optimization of Batch Processes.* PhD thesis 2007.

[52] P Nomikos, J.F MacGregor. Multivariate SPC Charts for Monitoring Batch Processes. *Technometrics.* 1995;37:41–59.

[53] Dunia R, Joe Qin S. Subspace approach to multidimensional fault identification and reconstruction. *AIChE Journal.* 1998;44:1813–1831.

[54] Kourti T, Nomikos P, MacGregor J.F. Analysis, monitoring and fault diagnosis of batch processes using multiblock and multiway PLS. *Journal of Process Control.* 1995;5:277–284.

[55] Westerhuis J.A, Gurden S.P, Smilde A.K. Generalized contribution plots in multivariate statistical process monitoring. *Chemometrics and Intelligent Laboratory Systems.* 2000;51:95–114.

[56] Camacho J, Pérez-Villegas A, García-Teodoro P, Maciá-Fernández G. PCA-based Multivariate Statistical Network Monitoring for Anomaly Detection. *Submitted to Computers & Security.* 2015.

[57] Marty R. *Applied Security Visualization.* USA: Pearson Education 2008.

[58] Callegari C., Gazzarini L., Giordano S., Pagano M., Pepe T.. A novel PCA-based network anomaly detection. in *Proccedings of IEEE International Conference on Communications (ICC'11)*:1–5 2011.

[59] Camacho J. Observation-based missing data methods for exploratory data analysis to unveil the connection between observations and variables in latent subspace models. *Journal of Chemometrics.* 2011;25:592–600.

[60] Hotelling H. *Multivariate Quality Control. Techniques of Statistical Analysis.* New York: MacGraw-Hill 1947.

[61] Jackson J E, Mudholkar G S. Control procedures for residuals associated with Principal Component Analysis. *Technometrics.* 1979;21:331–349.

[62] Marini F, Walczak B. Particle swarm optimization (PSO). A tutorial. *Chemometrics and Intelligent Laboratory Systems.* 2015;149:153–165.

[63] Broadhursta D, Rowlandb J.J, Kelp D.B. Genetic algorithms as a method for variable selection in multiple linear regression and partial least squares regression, with applications to pyrolysis mass spectrometry. *Analytica Chimica Acta.* 1997;348:71–86.

[64] Lundstedt T, Seifert E, Abramo L, et al. Experimental design and optimization. *Chemometrics and Intelligent Laboratory Systems.* 1998;42:3–40.

[65] Camacho J, Lauri D, Lennox B, Escabias M, Valderrama M. Evaluation of smoothing techniques in the run to run optimization of fed-batch processes with u-PLS. *Journal of Chemometrics.* 2015;29:338–348.

[66] Schupke D, Gonzalez de Dios O, Tipper D. Advances in network planning - part I: fixed networks and clouds [Guest Editorial]. *Communications Magazine, IEEE.* 2014;52:24–25.

[67] Schupke D, Gonzalez de Dios O, Tipper D. Advances in network planning - Part II: wireless networks and traffic uncertainty [Guest Editorial]. *Communications Magazine, IEEE.* 2014;52:146–146.

[68] Pióro M, Medhi D. *Routing, Flow, and Capacity Design in Communication and Computer Networks.* Morgan Kaufmann Publishers Inc. 2004.

[69] Dengiz O, Konak A, Smith A.E. Connectivity management in mobile ad hoc networks using particle swarm optimization. *Ad Hoc Networks.* 2011;9:1312–1326.

[70] Magán-Carrión R, Camacho J, García-Teodoro P. Optimal Relay Placement in Multi-Hop Wireless Networks. *Submitted to Ad Hoc Networks being under moderate revision.* 2015:1–34.

[71] Yabuki Y, MacGregor J.F. Product quality control in semibatch reactors using midcourse correction policies. *Industrial & Engineering Chemistry Research.* 1997;36:1268–1275.

[72] Flores-Cerrillo J, MacGregor J.F. Control of Particle Size Distributions in Emulsion Semibatch Polymerization Using Mid-Course Correction Policies. *Industrial and Engineering Chemical Research.* 2002;41:1805–1814.

[73] Mehra V, Jain V, Uppal D. DaCoMM: Detection and Classification of Metamorphic Malware. in *Proceedings of the Fifth International Conference on Communication Systems and Network Technologies (CSNT)*:668–673 2015.

[74] Yerima S.Y, Sezer S, McWilliams G. Analysis of Bayesian classification-based approaches for Android malware detection. *IET Information Security.* 2014;8:25–36.

[75] tugsSanjaa B, Chuluun E. Malware detection using linear SVM. in *Proceedings of the 8th International Forum on Strategic Technology (IFOST)*;2:136–138 2013.

[76] Khodamoradi P, Fazlali M, Mardukhi F, Nosrati M. Heuristic metamorphic malware detection based on statistics of assembly instructions using classification algorithms. in *Proceedings of the 18th CSI International Symposium on Computer Architecture and Digital Systems (CADS)*:1–6 2015.

[77] Cesare S, Xiang Y. Malware Variant Detection Using Similarity Search over Sets of Control Flow Graphs. in *Proceedigns of the IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*:181–189 2011.

[78] Bin Wu, Tianliang Lu, Kangfeng Zheng, Dongmei Zhang, Xing Lin. Smartphone malware detection model based on artificial immune system *Communications, China.* 2014;11:86-92.

[79] Zolotukhin M, Hamalainen T. Support vector machine integrated with game-theoretic approach and genetic algorithm for the detection and classification of malware. in *Proccedings of the IEEE Globecom Workshops (GC Wkshps)*:211–216 2013.

[80] Ki-Hyeon K, Mi-Jung C. Android malware detection using multivariate time-series technique. in *Proccedings of 17th Network Operations and Management Symposium (APNOMS)*:198–202 2015.

[81] De Turck F, Vanhastel S, Vandermeulen F., Demeester P. Design and implementation of a generic connection management and service level agreement monitoring platform supporting

the virtual private network service. in *Proceedings of the IEEE/IFIP International Symposium on Integrated Network Management*:153–166 2001.

[82] Stewart L, Armitage G, Branch P, Zander S. An architecture for automated network control of QoS over consumer broadband links. in *Proceedings of the IEEE International Region 10 Conference (TENCON 05)*:1–6 2005.

[83] Dainotti A, Pescape A, Claffy K.C. Issues and future directions in traffic classification. *Network, IEEE.* 2012;26:35–40.

[84] Internet Assigned Numbers Authority (IANA). (Accessed January 29, 2016) http://www.iana.org/assignments/port-numbers.

[85] Mochalski K, Schulze H. Deep Packet Inspection: Technology, Applications and Net Neutrality. *Forum American Bar Association.* 2009.

[86] Nguyen T.T.T, Armitage G. A survey of techniques for internet traffic classification using machine learning. *Communications Surveys Tutorials, IEEE.* 2008;10:56–76.

[87] Camacho J, Padilla P, García-Teodoro P, Díaz-Verdejo J. A generalizable dynamic flow pairing method for traffic classification. *Computer Networks.* 2013;57:2718 - 2732.

[88] Sen S, Wang J. Analyzing peer-to-peer traffic across large networks. *IEEE/ACM Trans. Netw.* 2004;12:219–232.

[89] Madhukar A, Williamson C. A Longitudinal Study of P2P Traffic Classification. in *Proceedings of the Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS). 14th IEEE International Symposium on*:179–188 2006.

[90] Yeon-sup L, Hyun-chul K, Jiwoong J, Chong-kwon J, Ted Taekyoung K, Yanghee C. Internet traffic classification demystified: on the sources of the discriminative power. in *Proceedings of the 6th International COnference (Co-NEXT '10)*:9:1–9:12 2010.

[91] Karagiannis T, Broido A, Faloutsos M, Claffy K. Transport layer identification of P2P traffic. in *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement (IMC '04)*:121–134 2004.

[92] Callado A, Kamienski C, Szabo G, et al. A Survey on Internet Traffic Identification *Communications Surveys Tutorials, IEEE.* 2009;11:37–52.

[93] Keralapura R, Nucci A, Chuah C. A novel self-learning architecture for P2P traffic classification in high speed networks. *Computer Networks.* 2010;54:1055–1068.

[94] Sousal M, Schmidt E.G. Machine learning algorithms for accurate flow-based network traffic classification: Evaluation and comparison. *Performance Evaluation.* 2010;67:451–467.

[95] Efficient application identification and the temporal and spatial stability of classification schema. *Computer Networks.* 2009;53:790–809.

[96] Mehmood T, Hovde Liland K, Snipen L, Saebo S. A review of variable selection methods in Partial Least Squares Regression. *Chemometrics and Intelligent Laboratory Systems.* 2012;118:62–69.

[97] Dayal B.S, Macgregor J.F. Recursive exponentially weighted PLS and its applications to adaptive control and prediction. *Journal of Process Control.* :169–179.

[98] Barker M, Rayens W. Partial least squares for discrimination. *Journal of Chemometrics.* 2003;17:166–173.

[99] Brereton R.G, Lloyd G.R. Partial least squares discriminant analysis: taking the magic away. *Journal of Chemometrics.* 2014;28:213–225.

[100] Gray R.S, David K, Newport C, et al. CRAWDAD data set dartmouth/outdoor (v. 2006-11-06) 2006. (Accessed January 29, 2016) `http://crawdad.cs.dartmouth.edu/dartmouth/outdoor`.

[101] Gray R.S, David K, Newport C, et al. Outdoor experimental comparison of four ad hoc routing algorithms. in *Proceedings of the 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems (MSWiM '04)* 2004.