# Detecting expectation-based spatio-temporal clusters formed during opportunistic sensing

Matthew Orlinski
Knowledge Discovery
Fraunhofer IAIS
Email: me@shigs.co.uk

Nick Filer
School of Computer Science
University of Manchester
Email: nick@cs.man.ac.uk

***Abstract*—Detecting clusters in the encounter graphs generated from reality mining data is one way of detecting the social and spatial relationships of participants. However, many of the existing clustering algorithms do not factor in the time since encounters, and can only be used to describe a single aggregated snapshot of the data. This paper describes a spatio-temporal clustering technique which has been used to reveal the transient communities within the data.**

## I. Introduction

There exist many different cluster detection algorithms that have been used for a wide variety of different purposes, one application is to analyse the opportunistic encounters between participants of reality mining experiments [1].

Short range, opportunistic, encounters between participants collected during reality mining experiments can be represented using graph theory notation. For example, the participants can be represented as vertices, and encounters between participants can be represented using edges. Using this methodology, a *spatial* cluster can be identified as a group of vertices that have more edges between each other than with other vertices [6]. It is important to note that when talking about encounter graphs the word spatial refers to the topology of the graph (two adjacent nodes are close together and two unconnected nodes are infinitely far apart), whereas the word spatial is commonly used elsewhere in a geographic sense [5].

However, human movement and encounter patterns are dynamic, and we must be careful when analysing human encounters to ensure that this dynamism is preserved. This paper will describe an *expectation-based spatio-temporal clustering* approach which we have used to analysis some popular reality mining datasets. Our new approach differs from existing methods in that it can be used to detect spatio-temporal clusters where edge weights are significantly higher than expected. Thus our method can be used to detect transient meetings between participants that are extraordinary, such as a chance meeting between a group of friends or a flashmob.

### A. Reality mining datasets

Reality mining experiments typically collect human behavioural data using mobile wireless devices belonging or given to participants to use in an unsupervised manner. Devices equipped with Bluetooth and Wi-Fi can be used to scan their surroundings for other nearby devices. The data collected can then be used to analyse which, when, and for how long participants come into close proximity with one another.

| | Infocom5 | Infocom6 | Cambridge | Reality |
|---|---|---|---|---|
| **Environment** | Conference | | Campus | |
| **Duration (days)** | 3 | 3 | 12 | 246 |
| **Number of devices** | 41 | 78 | 36 | 97 |
| **Device type** | iMote | iMote | iMote | Phone |
| **Number of encounters** | 22459 | 128979 | 10641 | 102594 |
| **Daily encounter probability** | 0.78 | 0.73 | 0.24 | 0.01 |
| **Granularity (seconds)** | 120 | 120 | 600 | 300 |
| **Geographic location** | No | No | No | Cell ID |

TABLE I: Comparison of some reality mining datasets. The daily encounter probability is the probability that an encounter with a particular other device will take place on any given day. Granularity is the time between neighbour discovery processes.

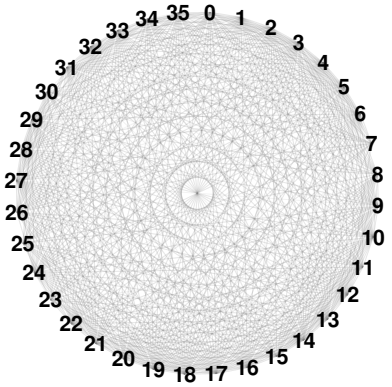
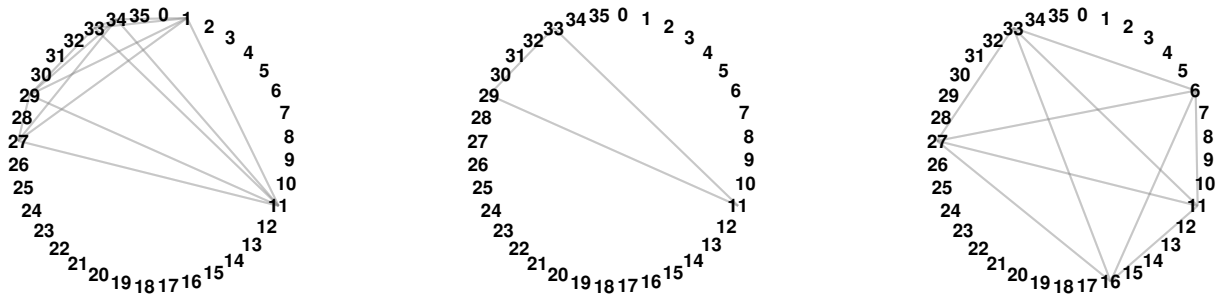



Fig. 1: Aggregated encounters in the Cambridge dataset. Almost all of the temporal information from the experiment is lost when data is presented in this way.

The figures given in Table I only take into account the mobile devices of participants. Any interactions with static devices or devices external to the experiment are omitted from our analysis because our aim is to generate transient clusters formed only by mobile participants.

The Infocom5, Infocom6, and Cambridge datasets were collected as part of the Haggle project from the University of Cambridge [11]. The Reality dataset (Note: Not to be confused with reality mining) was compiled using one hundred mobile phones logging encounters with Bluetooth devices on the MIT campus over a period of nine months [1].



(a) Frame 4 (b) Frame 5 (c) Frame 6

Fig. 2: The encounters present in the $4^{th}$, $5^{th}$, and $6^{th}$ hourly time frames of the Cambridge dataset that form strongly connected subgraphs within each time frame.

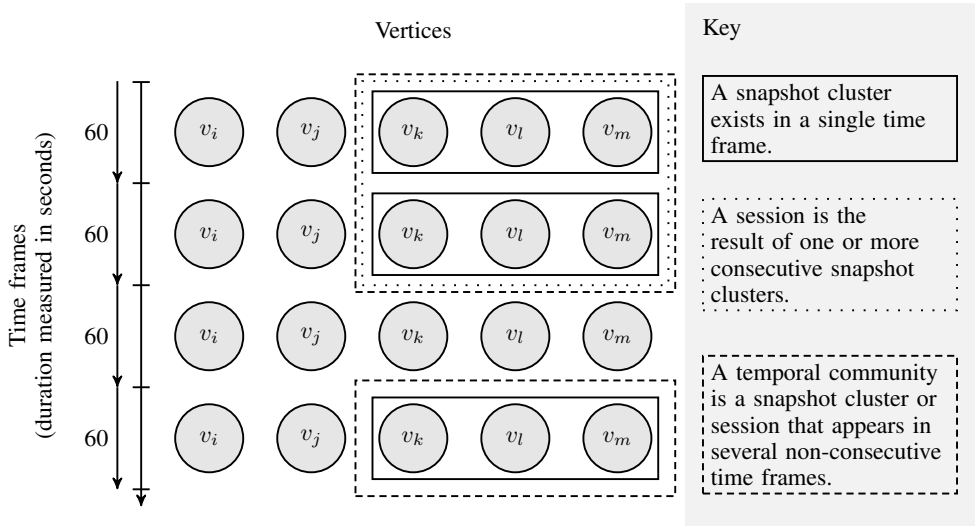



Fig. 3: A series of 60 second time frames (from top to bottom). Each time frame contains the same five vertices. Also shown are examples of Pietilainen's spatio-temporal clusters [10], which are calculated from the encounters between participants that occur within each time frame (encounters are not shown).

### *B. Spatio-temporal clustering*

To better understand the necessity for spatio-temporal clustering when analysing inter-human encounters, let us consider all of the encounters between the 36 participants in the Cambridge experiment [11]. One possible spatial cluster of this data is all of the vertices with edges incident upon them as shown in Figure 1, i.e. all of the people that have been encountered by others. However, this tells us nothing about the start time or duration of clusters.

Now imagine that the Cambridge dataset has been split into a number of discrete, sequential time frames. For example, Figures 2a to 2c show clusters that occur during the $4^{th}$, $5^{th}$, and $6^{th}$ hourly time frames of the dataset. Detecting clusters within a short time frame is one example of spatio-temporal clustering as the clusters describe the encounters which occurred within a particular time frame, and not obsolete encounters which occurred far in the past.

## II. Related work

Pietilainen and Diot detected spatio-temporal clusters within 60 second time frames and called these snapshot clusters. They also found a correlation between snapshot clusters that occur within several time frames, called temporal communities (see Figure 3) and inter-human relationships such as friendship and home city [10]. Pietilainen and Diot also reported that 30-40% of sessions (snapshot clusters that span consecutive time frames, see Figure 3) lasted 10 minutes or more in the campus datasets, and that sessions in the Reality dataset from MIT are long lived. The median being 28 minutes with 25% lasting more than 1 hour [10], possibly due to the campus timetable.

Natarajan et al. took a different approach to spatio-temporal cluster detection and looked for encounters between participants which are longer than a familiar threshold, and which overlap to form "meetings" [4]. Figure 4 shows one example of a meeting from the point of view of one participant/vertex $v_i$. In this example, an encounter with $v_j$ overlaps with the

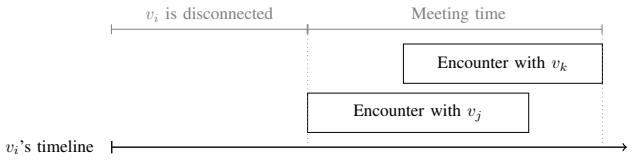



Fig. 4: A single meeting for $v_i$ as described by Natarajan.

encounter with $v_k$. The meeting time is therefore the total time from the start of the first encounter with $v_j$ to the end of the encounter with $v_k$. Using this definition Natarajan et al. found that mean meeting duration is 17 minutes in the Singapore reality mining dataset [4].

It is clear from Figures 3 and 4 that the methods proposed by Pietilainen and Natarajan will detect different patterns of human behaviour, and that the conclusions reached using the two methods are not directly comparable. Sessions can be used to detect clusters that exist over multiple time frames, whilst meetings often describe chains of encounters. The next section will describe a centralised method for detecting spatio-temporal clusters that are formed when edge weights are significantly higher than expected. The proposed approach will allow for individuals moving between social groups during a reality mining experiment, and provide a mechanism to ensure that clusters do not contain obsolete members.

## III. Expectation-based clustering

This section introduces two algorithms for spatio-temporal cluster detection in reality mining datasets. Both approaches are used to analyse the graphs formed within discrete time frames, and both approaches are aimed at detecting spatio-temporal clusters as quickly as possible as encounter data is read chronologically from multiple sources.

Single frame Expectation-Based Spatio-temporal (SEBS) clusters are made up of connected vertices in graphs formed during discrete time frames. The aim of SEBS cluster detection is to be able to detect when clusters of individuals experience a sudden rise in encounter duration with each other (such as at a concert or during a long journey on a crowded train).

The second spatio-temporal clustering algorithm described in this paper is for the detection of Multiple frame Expectation-Based Spatio-temporal (MEBS) clusters. MEBS clusters are formed when edges between a group of vertices form a strongly connected subgraph, and when the strongly connected subgraph is present across multiple consecutive time frames. Thus MEBS clusters are similar to the sessions described by Pietilainen and Diot, but MEBS clustering also provides a mechanism with which to detect smaller significant clusters within larger clusters.

The SEBS and MEBS clustering algorithms belong to the *expectation-based* category of spatio-temporal clustering algorithms [5], [8]. These clustering algorithms detect spatio-temporal clusters using edge weights that are higher compared to the recent past. Critical to the operation of expectation-based spatio-temporal clustering algorithms are the metrics and methods used in baseline calculation, examples of which are described in the following two subsections.

### A. Edges weights and metrics

It is more difficult to meaningfully count new encounters than it is to estimate cumulative encounter duration due to the frequent disruption to encounters experienced during reality mining experiments. Therefore, the weight of edges between vertices in SEBS and MEBS cluster detection represents the cumulative encounter duration within each discrete time frame. For example, the cumulative encounter duration for a vertex $v_i$ with vertex $v_j$ during time frame $t$ is $e^t_{v_i v_j}$. The mean cumulative encounter duration for the vertex $v_i$ for the time frame $t$ is referred to as the metric $m^t_i$, and $m$ for each vertex will be used when calculating baselines in the next section.

### B. Calculating vertex baselines

SEBS and MEBS cluster detection involves automatically calculating the expected values (called *baselines*) for metrics. This removes the need to manually assign thresholds when choosing vertices to cluster together.

Baselines for the current time frame are calculated from previous metric values at the end of each time frame. To make describing this process easier, each time frame of length $l$ is labelled using the time series $t_1, t_2, \ldots, t_{(n-1)}, t_n$, where $t_1$ is the first time frame, $t_{(n-1)}$ is the last complete frame, and $t_n$ is the current frame. This allows the baseline for a vertex $v_i$ for the current frame to be labelled $b^{t_n}_i$.

In SEBS and MEBS cluster detection, the baseline calculation for a vertex $v_i$ for the current time frame $t_n$ is done by calculating the mean of the $m$ values from the past $w$ complete time frames as in Equation 1.

$$b^{t_n}_i = \frac{m^{t_{(n-1)}}_i + m^{t_{(n-2)}}_i + \cdots + m^{t_{(n-w)}}_i}{w} \tag{1}$$

As the number of encounters may vary within the last $w$ time frames, taking the mean of the mean in this way produces an estimated mean. Calculating the estimated mean rather than the true mean may seem a weakness at first, but the baseline is just a guide. This calculation is preferable to setting manual familiar thresholds for each vertex in each experiment, and does not require that a record be kept of all edge weights over a long period of time.

The size of $w$ should be chosen carefully as this will alter the number of time frames from which the baseline is calculated, and thus the baseline itself. Sudden increases and decreases of metric values can be detected by choosing values for $w$ which are greater than 1, but smaller than the total length of the experiment $L$ divided by the time frame length $l$, $1 < w < (L/l)$. For example, in order to detect the bursts of cumulative encounter duration in reality mining datasets [12], the result of $w$ multiplied by $l$ should equal no more than 12 hours.

### C. Strongly connected subgraphs

As encounters between participant's electronic devices are directional [3], the graphs used to represent them can contain directed edges. Directed graphs can also contain *strongly connected subgraphs*, which are sections of a graph where there is a path from each vertex to every other vertex. Figure 2

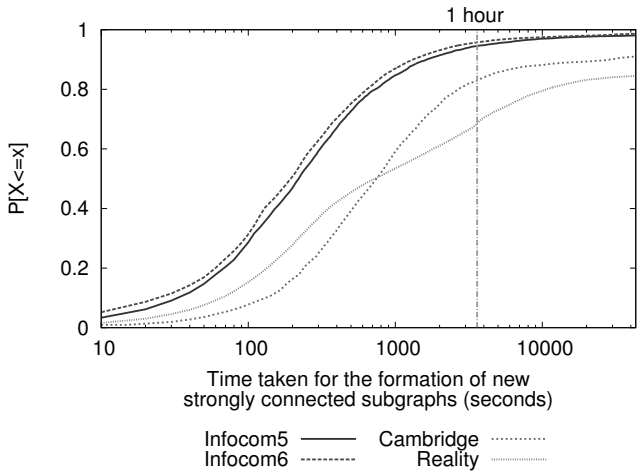



Fig. 5: Cumulative probability distribution of the time taken to form new strongly connected subgraphs.

shows the encounters between participants that form strongly connected subgraphs in the $4^{th}$, $5^{th}$, and $6^{th}$ hourly time frames of the Cambridge dataset.

Strongly connected subgraphs were used as the basis of efficient message forwarding in [7]. However, it is important to note that whilst strongly connected subgraphs were used in that work, strongly connected subgraph detection may be less important for other applications, e.g. flashmob detection.

### D. Length of time frames

In order to form strongly connected subgraphs within discrete time frames, the SEBS and MEBS clustering algorithms use longer time frames than the 60 second frames used by Pietilainen and Diot [10]. This is so that the algorithms can aggregate enough encounters to form large graphs by the end of each time frame. Figure 5 tells us that there is a 0.95 probability that strongly connected subgraphs take less than 1 hour to form in both of the Infocom datasets, whereas in Reality the probability of a strongly connected subgraph forming in less than an hour is just 0.68.

The mechanism used to compile Figure 5 is unusual, and therefore requires some introduction. To generate the data used in Figure 5, encounters within a dataset are aggregated into a non-monotonic encounter graph in chronological order. If a new edge results in a new strongly connected subgraph, then the time the subgraph took to form is calculated as being the time from when the first edge is added to the strongly connected subgraph up to the current time. Furthermore, in order that formation time in Figure 5 relates only to new strongly connected subgraphs, all of the edges in the detected strongly connected subgraph are removed from the encounter graph once the formation time has been calculated and stored.

Figure 5 hints that SEBS and MEBS clustering requires a minimum time frame length of 1 hour. Therefore, time frames of 1, 6 and 12 hours will be used in the analysis in Section IV. In the Reality dataset only 85% of strongly connected subgraphs form within 12 hour time frames, but it is difficult to extend time frame length over 12 hours in SEBS and MEBS cluster detection because of the baseline calculation and time between bursts of inter-human encounters (see Section III-B).

### E. Score function

SEBS and MEBS clustering also involves calculation of the *significance* of strongly connected subgraphs. Significance is determined by comparing the current metrics with baselines of vertices within strongly connected subgraphs.

Furthermore, the SEBS and MEBS clustering algorithms are capable of treating smaller strongly connected subgraphs that are part of larger ones as separate clusters, so long as metric values are higher within the smaller clusters.

The process of assessing the significance of strongly connected subgraphs in SEBS and MEBS clustering involves Neill's score function [5] shown in Equation 2, where $s_x^{t_n}$ is one strongly connected subgraph at time frame $t_n$. Generally speaking, Neill's score function can be used to compare the metrics within a spatial region against its baselines. Here, Neill's score function is being used to assess the significance of strongly connected subgraphs.

$$F_p(s_x^{t_n}) = (\frac{M}{B})^M e^{B-M} \text{ if } M > B, \text{ otherwise } F_p(s_x^{t_n}) = 1 \quad (2)$$

The baseline for a strongly connected subgraph (B) is calculated by summing the baselines from the individual vertices contained within the subgraph. The baseline for the strongly connected subgraph $s_x^{t_n}$ at time frame $t_n$ is then $B = \sum_{v \in s_x^{t_n}} b^{t_n}$, and the metric for $s_x^{t_n}$ is $M = \sum_{v \in s_x^{t_n}} m^{t_n}$. The mathematical constant $e$ is present in the score function following the simplification from the score function's original form as discussed on pages 36 and 37 of [5].

### F. Bringing it all together – SEBS clustering

SEBS cluster detection creates clusters from the graphs produced at the end of each time frame. For example, at the end of time frame $t_n$ the algorithm will attempt to detect strongly connected subgraphs with 3 or more vertices from the newly formed graph. Then the vertices of each strongly connected subgraph are tested to see if their metric values are greater than their baselines, i.e. the significance of all the vertices within a detected strongly connected subgraph $s_x^{t_n}$ are tested using the condition $\forall v_i^{t_n} \in s_x^{t_n} : m_i^{t_n} > b_i^{t_n}$, where $|s_x^{t_n}| \geq 3$.

The score function $F_p(s_x^{t_n})$ is then used to remove less significant strongly connected subgraphs from the analysis with the check, $\forall S^{t_n} \supset s_x^{t_n} : F_p(s_x^{t_n}) \geq F_p(S^{t_n})$. This check ensures that all strongly connected subgraphs detected are not structurally weaker than their parent strongly connected subgraphs ($S^{t_n}$) in terms of the metric used.

Once less significant strongly connected subgraphs have been removed from the analysis using the score function, those remaining are referred to as SEBS clusters. To summarise, in order to classify a strongly connected subgraph $s_x^{t_n}$ which occurred in the time frame $t_n$ as a SEBS cluster, $s_x^{t_n}$ must satisfy the conditions $\forall v_i \in s_x^{t_n} : m_i^{t_n} > b_i^{t_n}$, $|s_x^{t_n}| \geq 3$, and $\forall S^{t_n} \supset s_x^{t_n} : F_p(s_x^{t_n}) \geq F_p(S^{t_n})$. Finally, new baselines are

calculated for each vertex at the end of the SEBS detection process.

### G. Bringing it all together – MEBS clustering

Strongly connected subgraphs may only need to persists over multiple consecutive time frames in order to be labelled a MEBS cluster. A simple way of summarising the MEBS clustering process is that it searches $f$ consecutive time frames where $f > 1$, and looks for strongly connected subgraphs of at least 3 vertices which are present in each time frame.

As well as requiring that strongly connected subgraphs are at least 3 vertices in size, MEBS cluster detection also requires that the strongly connected subgraphs satisfy the following condition in order to discard less significant clusters for the analysis in Section IV-B. In order to be classified as a MEBS cluster, a strongly connected subgraph which spans the interval $t_{(n-(f-1))} \dots t_n$ (called $s_x$), must have a higher score in at least one frame than every strongly connected subgraph which is a super-set of $s_x$ that also spans the interval $t_{(n-(f-1))} \dots t_n$. In other words, for a strongly connected subgraph which spans the interval $t_{(n-(f-1))} \dots t_n$ to be considered a MEBS cluster, there must exist a time frame called $t_{max}$ in the interval $t_{(n-(f-1))} \dots t_n$ where $\forall S^{t_{(n-(f-1))} \dots t_n} \supset s_x^{t_{(n-(f-1))} \dots t_n} : F_p(s_x^{t_{max}}) \geq F_p(S^{t_{max}})$. It should also be added that if there is no super-set of $s_x$ spanning the interval $t_{(n-(f-1))} \dots t_n$, then $s_x$ will also be considered a MEBS cluster.

## IV. Cluster analysis

Pietilainen and Diot showed that temporal communities which exist in reality mining datasets often have less than 10 members [10]. This section will present analysis on the timing and size of SEBS and MEBS clusters in order to offer further insights into how people congregate in reality mining datasets. It is important to mention that the Reality dataset as presented in this analysis is truncated, and only the data between the time-stamps 1094545041 and 1111526856 is used. This is because there is no significant activity before and after these times respectively.

### A. Analysis of SEBS clusters

SEBS cluster analysis has been conducted using time frame lengths $l$ of 1, 6, and 12 hours. The baseline calculation has also been changed between experiments using $w$ values of 2, 4, 12, 18, and 24.

When Infocom6 is excluded from the mean cluster size calculation, the mean SEBS cluster size detected for all datasets is around 5 participants with no consistent change seen when increasing time frame length. However, SEBS cluster size in the Infocom6 experiment can be seen to increase sharply in Figure 6. This is because a small number of very large SEBS clusters are created early on in the Infocom6 experiment when $l = 43200$ seconds, after which time participation in the experiment appears to diminish [9]. This is known as the *premature clustering problem*, and can be caused by a combination of effects including:

1) Early densification, and a low mixing rate in the latter stages of an experiment.
2) High metric values at the start of an experiment, and ever decreasing metric values thereafter.
3) Time frames that encompass one or more complete burst cycles mean that there is little variation in baselines and metric values between frames.

| | Infocom5 | Infocom6 | Cambridge | Reality |
|---|---|---|---|---|
| $\mathbf{l = 3600\ seconds, f = 2}$ | 27 | 212 | 3012 | 15453 |
| $\mathbf{l = 3600\ seconds, f = 3}$ | 10 | 116 | 692 | 2429 |
| $\mathbf{l = 3600\ seconds, f = 4}$ | 2 | 52 | 45 | 290 |

TABLE II: There are different numbers of MEBS clusters in each dataset when using different variable values.

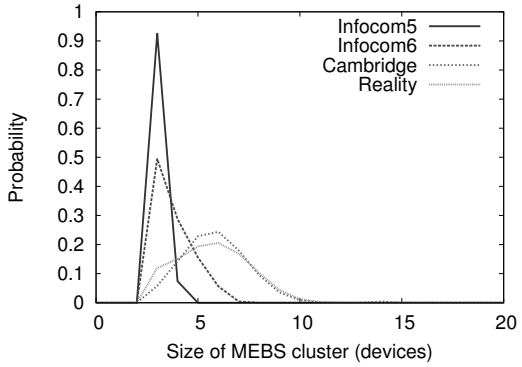



Fig. 7: Probability that a MEBS cluster is a particular size in the tested datasets when $l$=3600 seconds and $f$=2.

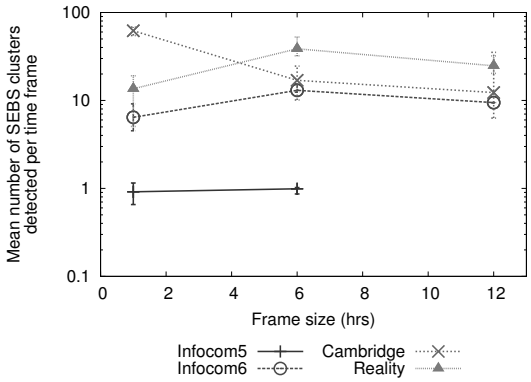
### B. Analysis of MEBS clusters

MEBS cluster detection can be used to tell us about the duration of clusters created from strongly connected subgraphs in reality mining data. The timing, size, and shape of MEBS clusters are very different to that of SEBS because MEBS clustering detects strongly connected subgraphs that exist in sequential time frames.

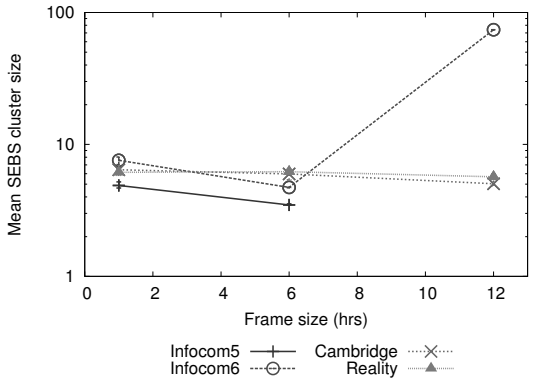
Table II can be used to compare the number of MEBS clusters detected over 2 hourly frames ($l = 3600, f = 2$) with those which span 3 or 4 time frames. It shows that MEBS clusters which last for longer than 3 hours are rare in each of the datasets tested.

1) Only 15.7% of the MEBS clusters which span 2 hourly time frames exist for a third hour in the Reality dataset.
2) Less than 23% of MEBS clusters exist for more than 2 hourly time frames in the Cambridge dataset.
3) MEBS clusters at conferences tend to last longer than in campus experiments. 37% of MEBS clusters exist for longer than 2 hours in the Infocom5 dataset, and 54% of MEBS clusters last for longer than 2 hours in the Infocom6 dataset.

There is also a separation between the data collected at conferences and on campus when looking at the size of MEBS clusters. Figure 7 shows that MEBS clusters tend to be larger in the campus scenarios than they do in reality mining experiments performed at conferences. The MEBS clusters




(a) Number of SEBS clusters




(b) Mean size of SEBS clusters

Fig. 6: Mean size and number of SEBS clusters detected in each of the reality mining dataset tested.

in the Reality and Cambridge datasets are most likely to be around 6 participants in size. Whilst in the Infocom5 dataset very few of the MEBS clusters detected are larger than 3 participants, and only 50% are larger than 3 participants in the Infocom6 dataset. This means that even though the Infocom6 dataset exhibits the premature clustering problem with SEBS cluster detection, recurring strongly connected subgraphs in Infocom6 are between 2-7 participants in size and last no more than 2 to 3 hours.

By considering the duration and size of MEBS clusters, a picture emerges of small yet long lasting MEBS clusters in the conference datasets, with larger MEBS clusters which last for shorter periods in the campus wide experiments. One reason for the observations made about encounters at conferences is that small groups tend to travel and stay together during conferences. This behaviour was also observed in [2] where the groups were given the name of *affiliation communities*. The larger MEBS clusters seen in the campus scenarios are also easily explained; people tend to work individually on a university campus, apart from when they form large short lived clusters in places such as dinning halls and lecture theatres.

## V. Summary

This paper describes the SEBS and MEBS clustering algorithms which detect strongly connected subgraphs within discrete time frames. Our target application for SEBS and MEBS cluster detection was to assess spatio-temporal cluster based data delivery in opportunistic networks [8]. However, similar algorithms might be useful for detecting instances of social stress in call record data. It may also be possible to annotate spatio-temporal clusters with semantic data taken from real time data streams i.e. Twitter.

SEBS and MEBS cluster detection relies on baseline calculation and Neill's score function to test edge weights for significance. Analysis on the resulting SEBS clusters highlighted the premature clustering problem which future expectation-based spatio-temporal clustering algorithms should attempt to avoid.

The time frame lengths of 1, 6, and 12 hours used here can exacerbate the premature cluster problem in SEBS detection by not giving the algorithm enough time to alter baselines to represent a seasonal mean. However, the long time frames are needed for SEBS cluster detection in order to aggregate enough encounters together to form strongly connected subgraphs (see Section III-C). Future iterations of this method may involve separating the baseline calculation and strongly connected subgraph generation into two separate threads which have their own time frames.

## References


[1] N. Eagle and A. (Sandy) Pentland. Reality mining: Sensing complex social systems. *Personal Ubiquitous Computing*, 10(4):255–268, Mar. 2006.

[2] P. Hui and J. Crowcroft. Human mobility models and opportunistic communications system design. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 366(1872), 2008.

[3] A. Madan, M. Cebrian, D. Lazer, and A. Pentland. Social sensing for epidemiological behavior change. In *ACM conference on ubiquitous computing*, pages 291–300, 2010.

[4] A. Natarajan, M. Motani, and V. Srinivasan. Understanding urban interactions from bluetooth phone contact traces. *Passive and Active Network Measurement*, pages 115–124, 2007.

[5] D. Neill. *Detection of spatial and spatio-temporal clusters*. PhD thesis, Carnegie Mellon University, 2006.

[6] M. E. Newman. Analysis of weighted networks. *Physical Review E*, 70(5), 2004.

[7] M. Orlinski. *Neighbour discovery and distributed spatio-temporal cluster detection in pocket switched networks*. PhD thesis, University of Manchester, 2013.

[8] M. Orlinski and N. Filer. Distributed expectation-based spatio-temporal cluster detection for pocket switched networks. In *IFIP Wireless Days*, Dublin, Ireland, Nov. 2012.

[9] M. Orlinski and N. Filer. The rise and fall of spatio-temporal clusters in mobile ad hoc networks. *Ad Hoc Networks*, 11(5):1641–1654, 2013.

[10] A. Pietilainen and C. Diot. Dissemination in opportunistic social networks: the role of temporal communities. In *ACM international symposium on mobile ad hoc networking and computing*, pages 165–174, South Carolina, USA, June 2012.

[11] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau. The haggle reality mining datasets, May 2009.

[12] Y. Wang, B. Krishnamachari, and T. Valente. Findings from an empirical study of fine-grained human social contacts. In *WONS*, pages 153–160, Feb. 2009.