The Knowledge Engineering Review, Vol. 31:4, 343–366. © Cambridge University Press, 2016
doi:10.1017/S0269888916000199

Spatial awareness in pervasive ecosystems

SIMON DOBSON[1], MIRKO VIROLI[2], JOSE LUIS FERNANDEZ-MARQUEZ[3], FRANCO ZAMBONELLI[4], GRAEME STEVENSON[1], GIOVANNA DI MARZO SERUGENDO[3], SARA MONTAGNA[2], DANILO PIANINI[2], JUAN YE[1], GABRIELLA CASTELLI[4] and ALBERTO ROSI[4]

[1]*School of Computer Science, University of St Andrews, St Andrews, Fife KY16 9SX, UK;*
e-mail: simon.dobson@st-andrews.ac.uk, graeme.stevenson@st-andrews.ac.uk, juan.ye@st-andrews.ac.uk;
[2]*DISI, University of Bologna, Bologna, Italy;*
e-mail: mirko.viroli@unibo.it, sara.montagna@unibo.it, danilo.pianini@unibo.it;
[3]*Institute of Services Science, University of Geneva, Geneva, Switzerland;*
e-mail: JoseLuis.Fernandez@unige.ch, Giovanna.DiMarzo@unige.ch;
[4]*Department of Engineering Sciences and Methods, University of Modena e Reggio Emilia, Reggio Emilia, Italy;*
e-mail: franco.zambonelli@unimore.it, gabriella.castelli@unimore.it, alberto.rosi@unimore.it

Abstract

Pervasive systems are intended to make use of services and components that they encounter in their environment. Such systems are naturally *spatial* in that they can only be understood in terms of the ways in which components meet and interact in space. Rather than treating spatiality separately from system components, researchers are starting to develop computational models in which the entire structure of a pervasive system is modelled and constructed using an explicit spatial model, supporting multi-level spatial reasoning, and adapting autonomously to spatial interactions. In this paper, we review current and emerging models of spatial computing for pervasive ecosystems, and highlight some of the trends that will guide future research.

1 Introduction

A pervasive ecosystem consists of a number of individuals spread throughout an environment that interoperate opportunistically to achieve their goals in a context-aware manner, but are also globally governed by some infrastructure rules analogous to the 'laws of nature' in natural ecosystems (Viroli *et al*., 2011; Zambonelli *et al*., 2015). Pervasive ecosystems present a vision of the near future where services will be personalised to our needs at a particular time, in a particular location, and for a particular situation; for example in a museum, we can guide the user to an exhibition hall and then route them to the exhibits inside the hall not only based on their artistic preference but also according to the real-time crowdedness or events occurring in each exhibition hall (Stevenson *et al*., 2013a).

These individuals, their smartphones, software services, pervasive displays, sensors and devices, augmented with sensing and computational capabilities, open up an exciting opportunity for harnessing very large volumes of sensed information to support the delivery of context-aware services to human users. The promise of opportunistically networking such devices offers as yet untapped potential in supporting the dynamic provision of services based on *ad hoc*, spontaneous arrangements of devices, services, and content across an infrastructure with few central points of control. Therefore, supporting service delivery on top of this large-scale, unstructured, highly dynamic substrate presents a challenge, with the combination of spatial computing, awareness, and self-organisation techniques offering intriguing possibilities in the search for a solution.

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

Spatial computing provides a flexible framework for distributing, organising, and migrating resources and devices in a space-aware manner, and more importantly, for coordinating the whole system following spatial patterns. Context and situation awareness allows for customised service provision adapting to relevant contextual information from ambient sensing devices. On top of spatiality and awareness, self-organisation techniques provide a means of adapting to changing environmental conditions—the arrival and removal of resources, the mobility of interacting agents, and the ever evolving network topology.

In this paper, we review the state-of-the-art approaches to integrating spatial computing with awareness in pervasive ecosystems. We explore the implications that spatiality has on the design of pervasive ecosystems, including what spatial knowledge a pervasive ecosystem often can acquire or needs to be aware of, how the system design is structured and constrained by spatiality, and how spatiality is leveraged with awareness to achieve advanced self-organising applications. We discuss current approaches and push towards a more systematic integration of pervasive service ecosystem and spatial computing technologies.

Section 2 introduces the background and motivation of a pervasive service ecosystem, from which we extract requirements on system design in terms of spatiality and awareness. In Section 3, we describe an awareness spectrum that characterises functionalities from the basic context representation to more advanced self-organising situation awareness. We review existing approaches to achieving these functionalities and what role space has playing in constraining and structuring these functionalities. The paper concludes in Section 4 with potential directions towards a better integrated pervasive ecosystem design with space and awareness.

2 Background and motivation

The emerging proliferation of devices with sensing and networking technologies offers opportunities for delivering pervasive services through interactions between spatially local resources. Applications are challenged to become aware of their surroundings: to discover, filter, and reason on information relevant to their goals. Although pervasive service ecosystem may utilise centralised services where available at an infrastructure level (e.g. wifi connectivity inside a building), they often do not assume the existence of centralised data management services to which all the data in the network is fed and from which classifications of situations of interest subsequently flow. This is especially the case for genuinely distributed activities, such as large-scale spectator events. Consequently, pervasive ecosystems require mechanisms to partition these responsibilities across the environment's participants, with awareness arising from the interactions between many individuals, each exposing aspects of their own state and perceiving the states of others to further increase their awareness of their surroundings. To explore the key elements of a pervasive ecosystem, we start from a motivating case study that characterises a wide range of pervasive service systems in the near future.

Consider an adaptive pervasive advertising display system (Ferscha *et al.*, 2012) where public displays are deployed in an open environment such as a museum, an airport, a train station, or a shopping mall. When a user is walking by or present in front of a display, the display will present advertisement in a context-aware manner; that is, matching to the current environment context and to the user's preference. Such implicit and spontaneous interaction is fostered by pervasive sensing, communication, and computing technologies. A public display can be installed with various computing devices to gather information about an environment such as weather and about a user such as attention towards the content being shown on the display. These displays can communicate with users' devices like their phones to gather their agenda, mobility pattern, or preference, with nearby service providers like train timetable or a social event, and as well as with each other to spread, coordinate, and schedule contents in a self-organising manner. Not only supporting personalised services, displays can potentially be connected to form a large-scale network so that they can enact global spatial services, such as advertising some content in certain spatial regions, up to steering people along paths from one place to another. This example illustrates the range of potential spatial interactions in a pervasive service ecosystem, from the very local, opportunistic interactions with passers-by, to the longer range, potentially (although not necessarily) supported by existing network infrastructure.

From the above case study, we discuss the three defining characteristics of a pervasive service ecosystem: *awareness*, *adaptation*, and *spatiality*. A display must be aware of the need and preference of a

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

user in order to personalise content to attract user's attention. The awareness is often achieved from two levels: *individual context* and *situation*. At an individual context level, the awareness is on one physical phenomena such as the time of the day, the season of the year, or the current weather like raining or snowing, or on one aspect of user information such as their birthday, marriage anniversary, or a shopping reminder. A situation is defined as an external semantic interpretation of context (Ye *et al*., 2012); that is, a higher-level concept that abstracts and integrates from disparate individual contexts. For example, a situation can be a user being interested in the content by reasoning on their attention towards the display and their personal preference; and another situation can be two friends being co-located by correlating spatial and social information on each user across multiple display spaces. Here a display space can be regarded as a region centring around the front of a display.

Displays must adapt their content to the current context and situation in a *present* and *proactive* manner. The system should not only personalise content based on what is happening right now but also what predictable future situations might be. For example, if the display perceives the delay of the train that a user normally takes to home, then it can push a nearby restaurant or bar advertisement to a screen where the user is present.

As a pervasive ecosystem often assumes an open, large-scale environment, *global* and *local* spatiality becomes very important in terms of computation, based on which awareness and adaptation is linked and bounded. That is, local context and situation awareness spreads across a whole space and other local spaces need to adapt to both their own awareness and the global awareness. For example, each display can locally build display sequences based on relative attraction levels by intersecting preferences of its audience and such established sequences can further be broadcast to other display spaces. The other displays can decide whether they want to merge with this new sequence and adapt their display strategy.

From the above example, we can see awareness and adaptation is tightly coupled: system adaptations—changes in system operation or behaviour—are driven by, and predicated on, observable changes in the environmental state (however simple or complex those state changes may be). More importantly, both are rooted on *spatiality*. A pervasive service ecosystem is characterised by:

- *Spatial embodiment*, in that location is not only the key context that is available in most pervasive ecosystems but it is also the centre that the other contexts are aggregated and organised around. That is, we gather together context relevant to a certain local region, and partition contexts by regions. In addition, awareness often starts from reasoning on locally acquired context to gain an understanding of what is happening in a local region, and then forms a global understanding by connecting the local situations following their spatial patterns. *Locality* is the key towards achieving distributed context and situation awareness.
- *Spatial interaction*, in that location and possible local interactions are the main indicator of current situation in many applications; for example, a user's current location can suggest the user's goal, task, or activities (Ye *et al*., 2014). Furthermore the customisation of service delivery depends on where resources and users are.
- *Multiple spatial scales*, in that communication between devices is constrained by their distance. That is, system design decisions are bounded by space, including how to spread local information to other spaces, how to collect data from sensors widely deployed, and how to distribute reasoners to cover the whole environment.
- *Dynamical spaces*, in that mobility and dynamism are at the centre of a pervasive ecosystem, compared to traditional service-oriented systems. Devices can join or leave the system at any time and they often move around, which presents a major challenge in space-aware ecosystem design. This leads to one of the important questions that we study in this review: *It is not sufficient to design a system to be aware of space, we must ask how we can use spatiality as the spine upon which to hang the system's behaviour.*

3 Spatial awareness spectrum

Pervasive ecosystems envision a point in the near future where devices will opportunistically and spontaneously connect with each other to form dense infrastructures. As mentioned above, *awareness*, *adaptation*, and *spatiality* are primary abstractions in this vision. The path towards realising this vision is

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

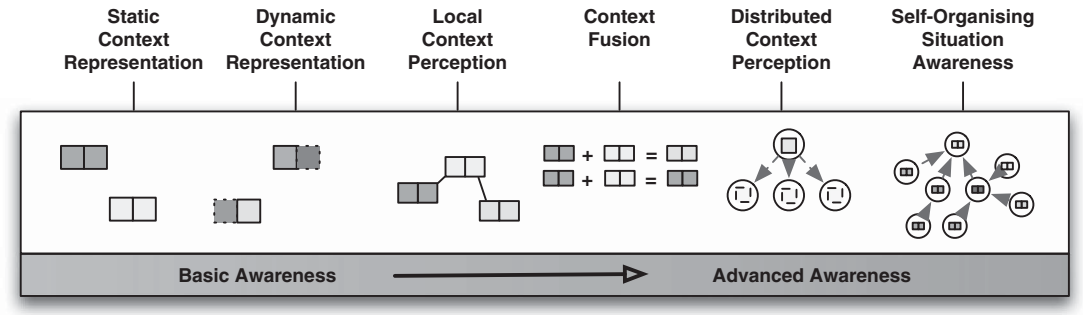


Figure 1 The awareness spectrum

paved with substantial technical challenges to support services in becoming aware of their surroundings: to seek, discover, and filter information relevant to their goals, and to aggregate, interpret, and reason over such information to identify situations—semantically meaningful classifications of states of importance upon which they will adapt their behaviour. To capture forms of awareness with varying complexity, we present a catalogue of spatial awareness attainable as a spectrum, illustrated in Figure 1, with basic data representation at one end and high-level awareness achieved through the collaborations of ecosystem participants at the other (Stevenson *et al.*, 2013b). From left to right we mark six points along this continuum and briefly overview the awareness capabilities available at each. The discussion in the following parts of this section centres around this spectrum, illustrating how information can be acquired, organised, aggregated and fused, and delivered, and how complex spatial data structures can be constructed to support the self-aggregation and self-composition of data (Fernandez-Marquez *et al.*, 2012b) for the purpose of higher-level reasoning.

3.1 Context representation

The representation of static and dynamic states is a basic need of a context-aware system. Static states encompass any information unlikely or less frequent to change, such as a building map or historic information. Such static states are often engineered as *a priori* knowledge in the system. Dynamic states indicate states that often change and are acquired in real time from sensors, which are used to formulate the current context or situation around a user in an environment. Among all different types of context, location is considered as a core element in that it cannot only pinpoint a user's or a resource's location, but also often acts as an active catalyst for binding and matching services.

In this section, we focus the discussion on representing location context, which has two levels of meaning: *modelling data in space*—describing the relation between an entity and a space (e.g. Bob is located in front of DisplayX01), and *modelling space*—describing the relation between spatial entities (e.g. DisplayX01 is next to the hardware shop). The former representation is pervasive in any system or application that can acquire and use location as context. For example, Henricksen and Indulska (2004) propose a three-value logic to represent location contexts, 'Fitzwilliam Darcy is located at Kitchen' and 'Fitzwilliam Darcy is located at DiningTable' from two different sensor reports. Here Kitchen and DiningTable are discrete and unequal values, and without the help of spatial knowledge, these two assertions may evaluate to the third logical value—*unknown*. Space modelling is the key to enable querying and reasoning on the spatial data to derive a higher-level knowledge. In the above example, if a dining table is contained in the kitchen, then we can regard both assertions to be true. Moreover, spatial knowledge can be used to derive that an entity is in a place more meaningful to applications or is co-located with another entity. In the following, we review the work in modelling space and move to modelling data in space. We will see space has very rich semantics which is yet to be fully exploited in the realisation of pervasive systems.

3.1.1 Modelling space

There are two elementary types of spatial representations: geometric and symbolic. A geometric representation characterises a space with its geometric shapes and a set of coordinate points under a certain

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

Cartesian coordinate reference system. A symbolic representation characterises a space with a human-friendly descriptive label. In the following, we focus on LOC8: a location model and extensible framework for programming with location, which was described conceptually in Ye *et al.* (2007) and developed as a programming framework in Stevenson *et al.* (2010). LOC8 supports a comprehensive range of spatial representations and enables rich spatial queries including positioning, range, spatial relation, navigation, and semantic queries. LOC8 shares similar design principle with the other early established space models such as the Location Stack (Hightower *et al.*, 2002), the Aura project (Jiang & Steenkiste, 2002), and the MiddleWhere (Ranganathan *et al.*, 2004b). Here we take it as an example to illustrate the state of the art on the space modelling in pervasive computing.

LOC8 supports representing coordinate points in different coordinate reference systems, which can be a global standard like WGS84 or locally defined to simplify a region's spatial representation. It borrows the simple linear conversion mechanism from Jiang's model (Jiang & Steenkiste, 2002) to support translating coordinate points from one reference system to another by rotating and moving the subject reference system towards the target system. On top of coordinates, LOC8 defines two-dimensional (2D) and 3D geometric shapes, which are further assigned with a symbolic label. Within LOC8, four spatial relationships are explored: containment, adjacency, connectedness, and overlap. The names suggest their meanings, and just to note that connectedness is a particular case of adjacency in which an entity can pass from a space to its adjacent space. It implies both the passage's direction and the notion of an accessible distance between the related locations.

Beyond the above classic spatial representations, LOC8 also supports *relative* spatial representation to enable richer spatial queries: centre and compass. In the centre representation, a target location is a geometric area such as a circle or a cuboid, whose centre is a coordinate and whose edge or radius length is the specified distance. This is particularly useful in formulating an application of finding all the entities within a valid interaction range. The compass representation involves building a coordinate reference system whose origin is a coordinate and whose rotation matrix follows the standard compass direction. The description of a target location contains a distance to its origin; the horizontal angle to the target location, measured clockwise from north, and the angle of elevation from the horizontal plane. The compass representation is beneficial in precisely guiding an entity to navigate towards a target in a 3D space. This can help to address many positioning problems in pervasive computing applications; for example in the Cooperative Object Detection and Ranging system (Kranz & Rockl, 2008) where cars are assumed to be installed with positioning and communication technologies, the compass representation can help to automatically detect the presence of the cars, predict the collision possibility, and send an early warning even when these two cars cannot see each other (Stevenson *et al.*, 2010).

LOC8 adopts a loosely coupled modelling technique that treats location information independently from other forms of context. This allows to treat all locatable objects in the same way, irrespective of their property structure and use by applications. In contrast, Nexus, an early open platform for a common augmented-world model (Hohl *et al.*, 1999), distinguishes representations of the location of static real-world entities from virtual entities with which the real world is augmented.

3.1.2 Modelling data in space

In terms of representing context in space, there are generally two approaches: treating space as *an attribute* to describe data, nothing more than any other attribute; and considering space as a *first-class concept* managed by the infrastructure.

Most existing space models and context models adopt the first approach. For example, Cohen *et al.* (2002) propose a language called *iQL* to specify the composition of pervasive services. Location is included and injected by a service; for example, a badge advertisement service injects a coordinate point of a badge wearer, with which to find all the neighbour wearers, or a local business publishes its advertisement along with its zip code so as to locate all the users present in the designated area. In this way, it is the responsibility of sensors and services to inject and update the location data, while the infrastructure performs the matching in a passive or active manner.

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

The Intentional Naming System (INS) (Adjie-Winoto *et al.*, 1999) supports applications in identifying resources by their properties rather than physical address. In the INS, resources are described using intentional names, a hierarchy of attribute/value pairs (called name specifiers) that describe the properties of the resource, spatial relations being one key descriptor. A name specifier has another name specifier as its parent, if its context is dependent on it (e.g. an identifier for the floor of a building is dependent on the building identifier). Resources advertise themselves to naming resolver components, which maintain a mapping between service description and physical address, while applications use name specifiers to describe the resources that they require. At runtime, a name resolver uses resource descriptions provided by the application to find matching services.

Based on the INS for resource discovery Solar is an infrastructure for collecting, processing, and disseminating context from 'thousands of diverse sensors to hundreds of diverse applications running on thousands of devices' (Chen *et al.*, 2004). Solar decomposes the context-aggregation process into modular and reusable operations, each of which is an object that subscribes to, processes one or more event streams, and publishes another event stream. Using an INS-like scheme, applications can specify the query using a tree of recursively connected operators (starting from sources and sinking in applications) to collect and aggregate context information. The result is a distributed directed acyclic graph (called an operator graph).

The above examples adopt an *extrinsic* approach to representing space, where spaces are named and used primarily as a means of explicitly identifying where an object is located. By contrast, some systems take an *intrinsic* view of space, where digital information is associated with physical objects, and their surrounding physical space and mobility is leveraged to support computing process. The spatial configuration of objects and their movements implicitly control an information system, and there is no explicit space representation beyond the local one implemented by the physical space itself (Banâtre *et al.*, 2010). One way in which Banâtre *et al.* illustrate this concept is the example of a UbiBus system in which a token generated by a pedestrian while in the vicinity of a bus stop requests the next bus to stop as it approaches. It is the spatial configuration of the pedestrian, the bus stop and the bus which controls the token generation and stop requests.

Zambonelli *et al.* (2015) adopt a biochemical approach where spatial computations are structured in terms of chemical-resembling reactions, each occurring in the precise physical point where the space is located, namely, where the device is deployed. They describe a coordination model that is equipped with space–time computing mechanisms, including reification of space–time information and a relocation service for annotations (Viroli & Stevenson, 2012). Within the coordination model, all the entities (including software agents, sensors, or resources) are described in a uniform representation—a *Live Semantic Annotation* (LSA). LSAs continuously represent the state of their associated components and are connected to the domain in which the information can be produced, interpreted, and manipulated.

Here, space is again treated intrinsically. Each LSA carries one property holding one URI value representing the ID of the location. This property is not specified by the agent that injects an LSA, but rather created and maintained by the infrastructure. The value in this property indicates how a local space is shaped around the node that hosts the injecting agent, by locating nearby neighbours in this node's vicinity such as their distance, connectivity, communication bandwidth, and the relative orientation in space. Reifying location into these LSAs enables dynamic behaviours; for example, a lower-level middleware service can be in charge of intercepting such LSAs before they are injected in the space, and relocating them to the proper neighbour, by an asynchronous request (Viroli & Stevenson, 2012). The update and maintenance of reified locations is governed via *eco law* in the infrastructure, while eco law are a set of self-organising rules that manipulate LSA spaces.

3.1.3 Discussion

We have introduced the mainstream representations of space; however, we consider location to have rich semantics that are yet to be fully explored as primary concepts within pervasive system architectures (Dobson, 2005):

- *Format*: it exists in multiple formats such as coordinate points, a geometric shape, a symbolic place, or a relative location.

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

- *Granularity*: it can be abstracted into spatial concepts at varying granularities from raw Global Positioning System (GPS) coordinates, to a street or an area, or up to a city or a country.
- *Relationships*: there are rich relationships between locations, including containment (e.g. one location is contained in another), adjacency (e.g. one location neighbours another), connectedness (e.g. one can pass from one location to another), and overlap (e.g. one location overlaps another).
- *Acquisition channels*: it can be indirectly acquired or inferred from other contexts; for example, a user's scheduled meeting event could suggest that the user would be in a meeting room during a certain time.
- *Interpretation*: it can be interpreted from different contexts; for example, given a coordinate point, we can translate it to a named place like 'Edinburgh Train Station' from a pure spatial context, as 'on the way to work' from a working context by linking the address to the user's work address; or as 'on holiday' from a social context by linking the address to the user's booked trip and calendar.
- *Inference*: it often has a significant impact on inferring a user's current desire or situation and further on adaption; for example, if a user is on holiday, she or he might be more relaxed, and an advertisement on a nearby book store would be appropriate.

How to surface the subtleties of location enabled by pervasive sensing and reasoning technologies to more advanced spatial computing is an interesting research topic. In terms of modelling and managing data in space, using location as a context is widely adopted, but it limits the capability of enabling dynamic behaviours at the infrastructure level such as relocating services. Viroli and Stevenson (2012) regard location as a first-class concept and define a set of meta rules—eco laws (Zambonelli *et al*., 2015)—to allow for infrastructure-level update and management of the spatial aspect of services and thus achieve system dynamics. This is a promising approach and at the current stage it mainly adopts the basic use of space; that is, neighbouring node and distance. It would be a good direction to extend it with more advanced spatial knowledge, implication, and models as presented above to enable richer and more expressive way of spatiality management. Also to note that all the above discussions imply a static (physical) space, not a dynamic one, even if dynamic processes take place into this static setting. Some other example (e.g. MANET or morphogenesis in biology) develop their own space. These examples are out of the scope of the paper.

3.2 Local context perception

Local context perception is about providing entities with the means to inspect the static and dynamic contextual information embedded in the annotations of co-located individuals. To understand this, let us look at the definition of context first. One of the most widely adopted definitions of context is 'any information that can be used to characterise an entity. An entity is a person, place or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves' (Dey, 2001). Here, places are spatial entities and interactions require proximity; that is, an interaction can happen only when two agents are close within a connection range and such interaction is often mediated by some form of local knowledge (Viroli *et al*., 2011). Thus, context has locality implication in that context is information relevant to an entity and accessible within a certain distance. Such locality has a significant impact on organising and distributing context, as well as mediating interactions, which will be discussed in the following.

3.2.1 Acquiring and organising local context

Not only is space a key context but it also serves well as an intuitive metaphor for non-spatial context. That is, context has a strong indication of *locality*, which can be interpreted as

- where the context is *about*; for example, the crowdedness of DisplayX01 is high;
- where the context is *of*; for example, Bob is in front of DisplayX01; and
- where the context is *bound*; for example, Bob's preference is only bound to DisplayX01, which cannot be accessed by or spread to any other displays.

Even though existing in different interpretations, locality is the mechanism to relate different contextual information; that is, if two pieces of context are located in the same local space (in an *absolute* spatial term),

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

or within a short range (in a *relative* spatial term), then they can be related. Hence, locality of context is well suited to organise context in physical locations (Bettini *et al.*, 2010). As we look to design systems with the capacity to handle the ever increasing amount of data generated by myriad physical and Web resources, it becomes less likely that we choose to provide access globally through a centralised service. Although logically centralised services—although implemented on top of distributed systems—are the current architecture of choice (e.g. Google, Amazon, Facebook), there are a number of reasons that suggest they are not well matched for managing data in large-scale pervasive service ecosystems: (1) the amount of information being continuously generated is potentially overwhelming, presenting scalability challenges; (2) requiring more than just a store and query architecture, there is significant computational complexity in reasoning about what is happening in each local space from all the information available; and (3) in the vast majority of cases, information will only be relevant locally and only for a short period of time—it is likely that the majority of information generated will never be used. From the standpoint of efficient resource utilisation, it makes little sense to collect, process, and distribute data from a centralised service under these circumstances. All this said, there will likely remain a class of pervasive applications with data requirements such that a logically centralised service remains a viable implementation choice.

Early context-aware systems like Context Toolkit follow the above centralised solution, where all related context are centralised for decision making and service provision (Heer *et al.*, 2003). Dey *et al.* (2002) later enhance the Context Toolkit as a distributed system that supports distributed feedback based on context acquired in different physical locations. Similarly, Remembrance Agent provides a service that examines users' current location, identifies their nearby individuals, and uses such locality information to retrieve the most relevant context to the users' current situation (Rhodes, 1997). A neural network house learns occupants' patterns of heating and lighting control for various locations in the house and uses this knowledge to predict user needs (Mozer, 1998).

All the above systems demonstrate how to acquire and gather context centring around a certain location, while *Liquid* is one of the very first attempts to deal with context-aware distributed queries towards a more general space-oriented manner (Heer *et al.*, 2003). The system is built on top of Context Fabric (Hong, 2002) that provides the abstraction of information spaces (called *infospaces*), which are logical storage units using the tuple space paradigm as context repositories for individual entities in an environment. Within Liquid, each query is organised into partitions that can be served at individual infospaces. For example, a query retrieval path of 'location.occupant' is designed to retrieve the location of people in the same room as the query issuer.

Liquid supports two operators to service spatial queries: local and external operators. The local operator is responsible for retrieving context data from the local infospace, where a Liquid node resides, and for monitoring the local infospaces for tuple insertions, deletions, and updates. In contrast, the external operators retrieve data from remote infospaces by querying other Liquid nodes. Result items will be streamed back and enqueued into the local operator through a callback from the network layer. These queries can benefit applications by providing augmented awareness within a space. For example, an application can provide resources (such as webpages or e-mails) for all the people currently in the same room as the query issuer. As people enter and exit the room, this result is automatically updated in this room. Also, when the query issuer moves from place to place, the query will be automatically rerouted to the correct infospaces for the new location.

3.2.2 Mediating interactions via locality

Locality is an important concept in coordination models where entities can interact via direct connections (Cabri *et al.*, 2000a). In the early attempts agents need to explicitly name their communication partners (Cabri *et al.*, 2000b). However, in pervasive computing applications, where agents can be dynamically created or removed from an environment, it is difficult to adopt such a spatially coupled model that requires the identification of the communication partners. Also, entities are supposed to be able to move and if they need to communicate directly on a wide scale, then they have to be localised.

The meeting-oriented coordination model attempts to address this spatial coupling problem by virtualising an abstract meeting point where agents can join, communicate, and synchronise with each

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

other (White, 1997). Meeting points are locally constrained in that a meeting takes place in a certain execution environment and only local agents can participate. The major drawback of this model, however, is that it enforces a strict temporal coupling; that is, in order to interact the agents must be at the same meeting point at the same time.

An alternative approach uses blackboard-based coordination, where interactions can occur via shared data spaces (called *blackboards*) (Nii, 1986). Blackboards can be considered as a medium to store and provide access to locally published data and services; agents communicate by leaving messages without the need to know when and by whom the messages will be read. The Linda model (Gelernter, 1985), which provides for a blackboard with associative matching for mediating component interactions through insertion and retrieval of tuple limits, provides the archetypal example. However, the scalability of blackboard approaches to large number of agents remains a concern (Viroli *et al*., 2011; Stevenson *et al*., 2013a). Inspired by nature, TuCSoN (Omicini & Zambonelli, 1999), MARS (Cabri *et al*., 2000a), and chemical-inspired tuple space (Viroli *et al*., 2011) propose to specify policies inherent in the tuple space to control and manage interactions in a self-organising and self-composing way.

3.2.3 Discussion

When it comes to local context perception, most pervasive systems can acquire location context of various kinds, with location one of the contexts key to triggering context-aware service provision. As more and more devices are available, distributed across an open environment, and opportunistically connected, space is no longer just one type of context, but the essence that other context should be organised and structured around. Liquid is one of the first examples of such, which can facilitate distributed computation that will be further elaborated in Section 3.4. However, such a rigid design on space is not suitable for today's highly mobile environment. In terms of mediating local interactions, bio-inspired approaches present a promising research direction; that is, enriching the local space or environment with inherent rules leading to more self-* features of a system.

3.3 Context fusion

An immediate step after perceiving context is *context fusion*—fusing contextual information from multiple sensors so that a notable value can be added. That is, individual sensors are able to capture specific contextual aspects of a situation, typically occurring in the portion of space in which the sensors are situated (Bettini *et al*., 2010). Context fusion indicates the combination of sensory data from disparate sources such that the resulting information is in some sense better than what would be possible when these sources were used individually. The term *better* might stand for more accurate, more complete, or more reliable, or refer to the result of an emerging view, such as a stereoscopic vision, where calculation of depth information is done by combining 2D images from two cameras at slightly different viewpoints.

In the area of context and situation awareness, context fusion can harness the shift from recognising punctual contextual facts to recognising complex, multifaceted, situations (Ye *et al*., 2012). In this regard, we can identify two main ways by which information from multiple sensors can be fused to achieve a higher level of situation awareness:

- *Local fusion of heterogeneous sensors*: this implies exploiting multiple sensors of different types in the same spatial location to acquire a multifaceted perspective of the local situation.
- *Regional fusion of uniform sensors*: this implies aggregating the information by multiple sensors scattered around a spatial region to obtain some global and summarised view of the overall situation in that region.

In the following we will review the state of the art with respect to these two basic approaches, but without forgetting that the approaches can be combined (and they often are); for instance, by first applying local fusion of heterogeneous sensors at different locations in a region, and then by aggregating such information on a spatial basis.

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

3.3.1 Location fusion of heterogeneous sensors

Concerning local fusion of heterogeneous sensors, most of the work in the area focusses on identifying the symbolic and meaningful place in which a user is located by using data coming from different positioning systems. This arises from the need to switch from a perspective in which the geographical *position*, say, from the GPS sensor, is the only raw contextual information available, to a perspective in which the *place* (e.g. office, home, or market) where the user is pinpointed is inferred, which corresponds to the physical and symbolic locations discussed in Section 3.1.1. This switch can be realised in many ways, one example being through fusing GPS information with that coming from co-located microphones (Azizyan & Choudhury, 2009) or other environmental sensors (Jung *et al.*, 2010). The need to considering *place* instead of *position* comes from a wish to enrich positions with semantic data (possibly also related to users' activities), first discussed by Hightower (2003), and further refined and enriched by Castelli *et al.* (2009) and Liao *et al.* (2007) who realise various reasoning engines to semantically identify places visited on the base of temporal patterns, using probabilistic models and Bayesian networks, respectively.

More general approaches focus on sensorial data fusion at different levels, either for acquiring and making accessible all aspects of the local spatial context or for reasoning about the context at a fine-grained perspective in order to get a better understanding of currently happening situations and detecting inconsistencies. Early works consider simple data fusion as aggregation of logically related context data based on simple name transformation (Dey *et al.*, 2001), possibly by adopting a common context specification language (Hong & Landay, 2001). Also, more recent works exploit a similar approach, in which contextual data are represented at a high level with a common structure in order to promote the same management for data coming from diverse sources and to ease the data usage and reasoning (Bortenschlager *et al.*, 2009).

Uncertainty lies at the heart of all descriptions of the sensing and fusion process. Probabilistic and knowledge models (or the combination of both) provide a powerful means of describing uncertainty and lead naturally into ideas of information fusion. Such models are often used to achieve context fusion at a finer-grained level. Ganti *et al.* (2010) use lightweight Bayesian learning algorithms based on hidden Markov model when integrating data from the microphone and inertial sensors on the mobile phones in order to infer high-level human activities. Tran and Phung (2006) exploit the combination of multiple probabilistic techniques to reduce the number of model parameters to be taken into account when a large number of sensors are used in a smart home environment. Liu and Singh (2004) propose a promising approach of applying commonsense knowledge to better correlate information coming from different sensors (Mamei, 2010; Bicocchi *et al.*, 2012a). The idea is to use a database of commonsense semantic information in order to associate a comprehensive meaning (i.e. in the form of a word or a sentence) to multiple information (in the form of semantic tags) coming from a set of multiple heterogeneous sensors. Dempster–Shafer theory (Dempster, 1968; Shafer, 1976), a mathematical theory of evidence that provides the ability to combine evidence from different sources to arrive at a degree of belief in a proposition while preserving ignorance, has been applied to a wide range of sensor fusion tasks, spatial data included (Wu *et al.*, 2002; Zhang *et al.*, 2009; Mckeever *et al.*, 2010). Its application relies on expert knowledge to define an evidential reasoning hierarchy and quantify the accuracy of sensors. Consequently, its application is limited to domains in which experts can reasonably be expected to provide this knowledge.

3.3.2 Regional fusion of uniform sensors

Many works in the area of wireless sensor networks for mobile users are concerned with regional fusion of uniform sensors (Yick *et al.*, 2008). Most of the approaches on sensor networks assume the presence of base stations to which sensed data should flow. In such situations, the basic solution is to have sensors build a tree rooted at the sink and supporting the routing of sensed data towards it (Polastre *et al.*, 2004), possibly performing limited forms of data fusion (e.g. averaging or max/min determination) as data from sensors climb the tree (Gehrke & Madden, 2004), with the goal of reducing communications between sensors and thus, the energy costs. However, when mobile users wishing access surrounding sensorial information from their current location are involved, tree-based approaches can hardly apply as a general solution. In fact, the costs of building a tree on demand for many possible mobile users at various locations

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

would be unbearable, and none of the many optimisations proposed (Kabaday & Julien, 2007; Yang *et al*., 2008) fully eradicates the identified flaws of tree-based approaches.

To accommodate the need of mobile users to achieve an understanding of the situation around, as sensed by different sensors dispersed around, one has to aggregate data in the sensor network (or in confined spatial regions of the network) such that queries by multiple and mobile users can be answered (Lotfinezhad *et al*., 2008; Bicocchi *et al*., 2012b). This basically corresponds to fusing information about a specific aspect within a region so as to gather a more global understanding of a specific phenomena. For instance, in Sarkar *et al*. (2007), each sensor in the environment can compute, at predefined communication and energy costs, aggregated information about sensorial information in its neighbourhood, and the aggregation functions can be concurrently computed for neighbourhoods of different sizes. However, the definition of the regions in which to produce aggregated data is mostly network driven, and does not take into account specific characteristics of the environment being monitored or of the phenomena being sensed. To overcome this limitation, Bicocchi *et al*. (2012b) propose the virtual macro sensors approach to make a sensor network self-organise into regions characterised by similar sensing patterns, which helps to promote aggregation of data on a per-region basis, as if each region were monitored by a single macro sensor. Such an approach can be very effective in facilitating data collection in dense large-scale sensor networks, and as well as in identifying the regions in which specific situations take place.

Several research works in the area of middleware and programming languages for distributed sensor networks start recognising the need to support various forms of spatial aggregation of data (Mottola & Picco, 2011). They define suitable general-purpose primitives and language constructs (together with the supporting middleware infrastructures) to enable users to flexibly query the network and obtain information about individual sensor data and aggregated data related to specific spatial regions. Approaches such as logical neighbourhood (Mottola & Picco, 2006) and TeenyLIME (Costa *et al*., 2007) can be considered as general tools to program and enforce regional fusion of contextual information.

As far as distributed data aggregation is concerned, although not always connected with spatial awareness, diffusive algorithms (Corradi *et al*., 1999) and gossip-based aggregation algorithms (Jelasity *et al*., 2005) are worth mentioning. Such approaches have been proposed as simple yet very effective approaches to compute and make available at each node of a network that aggregates values related to some global property of a network. When applied to nodes in spatial proximity relations (Dimakis *et al*., 2006), such approaches can be indeed exploited as a means to achieve spatial fusion of contextual information. Interestingly, it has been recently shown that gossip-based aggregation algorithms can also be exploited to perform distributed learning of contextual information (Ormándi *et al*., 2013), by preventing the distribution of raw contextual information in respect of those privacy concerns.

3.3.3 Discussion

Context fusion is about aggregating and abstracting information to attain higher-level insights. We have discussed two aspects of fusion: fusing of heterogeneous sensors and of uniform sensors. The former mainly involves reasoning; that is, how to abstract raw context from different phenomena in different formats and data types to attain higher-level situations. Different probabilistic models and knowledge-driven techniques have been applied and a hybrid approach of the two presents a promising future direction (Ye *et al*., 2014). In terms of fusing uniform sensor data, the research mainly concerns about how to collect sensor data in a distributed fashion, where spatiality plays an important role. Gossip-based aggregation algorithms have started gaining wide recognition when performing distributed context awareness while preserving privacy (Ormándi *et al*., 2013).

3.4 Distributed context perception

Moving further towards the right end of the spectrum, we reach *distributed context perception* where more complex interactions and communication strategies emerge. It aims to seek or project the presence of resources across a region of space, informing interested parties of state, advertising services, and serving as the basis for further interactions. Information spreading is one way to enable ecosystem participants to perceive context in their non-immediate proximity. It works by replicating context across neighbouring

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

devices, which is often bounded by a metric such as hops or physical distances. Propagating context information from context sources to remote nodes is one way of letting context be known at a remote location. Inversely, nodes can also proactively look for and gather context. In the following, we will review the works on how distributed contextual information can be either shared or gathered in a completely decentralised way in order to achieve high-level contextual reasoning. Additionally, since situation recognition is the immediate next step after raw context perception, we will review as well distributed situation recognition. Therefore, the discussion will centre around *proactively and reactively sharing contextual information beyond local perception* and *achieving distributed situation recognition*.

3.4.1 Sharing contextual information proactively

Generally, the most intuitive way of sharing contextual information in a pervasive system is to proactively propagate among nodes. A naive process potentially involves a high bandwidth usage and a high volume of information stored at each node. Propagation can then take different forms, from pure flooding to more subtle forms of broadcasting aiming at reducing bandwidth. Additionally, information injected and propagated in the pervasive system can be outdated or no longer valid because of the environment dynamism. To overcome each of these disadvantages, three self-organising mechanisms have been proposed: *spreading*, *aggregation*, and *evaporation*.

Spreading mechanism. Contextual information often needs to be propagated so that remote nodes can perceive it beyond their local perception and thus increase their knowledge about the global state of the system. Spreading has been proposed as a primitive mechanism for a large number of self-organising complex mechanisms (Fernandez-Marquez *et al*., 2012a), which is also alternatively called as information diffusion (Khelil *et al*., 2002), information or data dissemination (Sabbineni & Chakrabarty, 2005), flooding (Yi & Gerla, 2003), broadcast (Tseng *et al*., 2002), or epidemic spreading (Khelil *et al*., 2002). In Beal ,and Viroli (2014), a general purpose, reusable and self-stabilising building block 'G' for spreading is studied, used to propagate information outward a source and along a gradient-like data structure, with possibility of on-the-fly computation of information.

The propagation of information using broadcasting may result in serious redundancy, contention, and collisions (Tseng *et al*., 2002). Many optimised spreading algorithms have been proposed in the literature, mainly focussed on reducing bandwidth usage. These algorithms range from pure flooding (Obraczka *et al*., 2001), to probabilistic propagation (Eugster *et al*., 2003; Sasson *et al*., 2003), to dynamic probabilistic propagation (Zhang & Agrawal, 2005), to location-based propagation (Tseng *et al*., 2002), and to hybrid propagation where algorithms are switched depending on contextual information (Fernandez-Marquez *et al*., 2014). Location-based propagation is a family of algorithms that take into account the position of neighbouring nodes in order to decide which nodes should re-send the information. The main goal is to reduce the number of nodes that re-send the information. Location-based approaches present very good results compared to pure and probabilistic propagation schemes, however, the computation carried on at each node is more complex, and relies on nodes providing their positions voluntarily and accurately. Location-based implementations can be extended by taking into account neighbouring nodes of two-hop distance, further optimising the reduction in the number of messages sent, but increasing dramatically the computation at each node.

Aggregation mechanism. Even though these optimised techniques mentioned above can help to reduce the bandwidth usage during spreading, we still need a mechanism to reduce the volume of information (e.g. information coming from a sensor sampling every minute, or information coming from several sources). The *Aggregation* mechanism (Gardelli *et al*., 2007; Fernandez-Marquez *et al*., 2012a) as a way to reduce the amount of information in the system by synthesising meaningful information. Aggregation, also known as fusion (Niu & Varshney, 2005), locally applies a fusion operator to process the information and synthesises macro information. This fusion operator can take many forms, such as filtering, merging, aggregating, or transforming (Chen & Kotz, ,2002). Different aggregation strategies have been discussed in Section 3.3.2. Similarly to the case of G, in Beal and Viroli (2014) a building block 'C' for collecting values into sources is proposed, in which information moves back to a source with progressive aggregation.

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

Evaporation mechanism. Even using aggregation, outdated, or no longer valid information can remain in the system, which could lead to a wrong contextual perception and thus misguiding the system decisions. The *Evaporation* mechanism (Fernandez-Marquez *et al*., 2012a), also known as decay (Huebel *et al*., 2008), temporal degradation function (Ye *et al*., 2008), or freshness (Ranganathan *et al*., 2004a), has been proposed to deal with dynamic environments. Basically, the Evaporation progressively reduces the relevance of information. That is, recent information becomes more relevant than information processed some time ago. Additionally, evaporation can be directly used to remove no longer valid or updated information from the system.

3.4.2 Gathering distributed contextual information on request

The above three mechanisms such as Spreading, Aggregation, and Evaporation form the primitives for actively propagating distributed contextual information among a pervasive ecosystem. However, usually we do not need to actively propagate in the system. For example, when information is not relevant for a large number of users, when information is only useful at the time a user provides it, or when information is shared just with a subset of people interested. Following the example in Section 2, information such as the availability of advertisement and incoming events of a certain area would be specially interesting to users who are in the same area or nearby. Such a contextual information should be only spread to this subset of users.

Compared to the above proactively sharing approaches, gathering information on request presents advantages in dramatically reducing resource consumption such as memory and bandwidth. Gathering distributed contextual information requires two phases: (1) spreading the request in order to find the information at remote locations, (2) routing back the so-found information to the query node. Similarly to the above sub-section, we highlight the self-organising mechanisms useful for gathering distributed contextual information, either for spreading the request or for bringing back the results, which are *Gradient*, *Dynamic Gradient*, *Chemotaxis*, *progressive in-network aggregation*, and *remote query and retrieval*.

Gradient mechanism. The *Gradient* mechanism (Fernandez-Marquez *et al*., 2012a) is to diffuse certain information together with additional metadata related to the distance of the gradient source (i.e. a type of hop count). Diffusing a gradient actually involves spreading information, incrementing a hop counter, and aggregating information when different routes reach an identical node. Once diffused in a geographic area, a gradient then provides additional useful information, such as the distance from the current node to the source of the gradient, direction towards or away from the gradient source, equi-potential lines of distances from the source, and the shortest path towards the gradient source. Gradients take inspiration from nature, where they correspond to concentrations of substance. Gradients are a crucial mechanism in spatial awareness, and serve to form spatial structures, geographically dispersed but with a precise meaning. Indeed, by combining spreading and aggregation, gradients provide spatial information such as the shortest paths, distance, and directions to gradient's source, which can also serve to route back information towards the source. Gradients are particularly useful for a query node when retrieving distributed contextual information. The query node becomes a gradient source and spreads a gradient whose main information is the requested query. Nodes answering the query then simply send the answer back along the paths built by the gradient towards the query node.

Dynamic Gradient mechanism. The *Dynamic Gradient* mechanism (Fernandez-Marquez *et al*., 2012a), also called Active Gradient (Clement & Nagpal, 2003), is a variant of the Gradient mechanism, where in addition to spreading and aggregation, evaporation also comes into play. In a mobile and dynamically changing environment, where nodes are constantly moving, appearing, and disappearing, a static gradient becomes quickly updated and cannot be used for routing information back to the gradient source. When the gradient information evaporates at the source node, the latter diffuses a new gradient structure. Dynamic Gradients are then particularly useful for spreading queries over mobile or dynamically changing infrastructure.

Chemotaxis mechanism. The *Chemotaxis* mechanism (Fernandez-Marquez *et al*., 2012a) is about following up-hill, down-hill, or along equipotential lines a gradient previously established either by a

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

given environment or engineered on purpose. Chemotaxis is a way to route back the information to the source of the gradient following the shortest path up-hill, thus answering the query.

Progressive In-Network Aggregation. In addition to Gradient and Chemotaxis described above, many different routing algorithms have been proposed in the literature, such as Ad-Hoc On-Demand Distance Vector (Perkins *et al*., 2003), Dynamic Source Routing (Johnson & Maltz, 1996), or Fisheye State Routing (Pei *et al*., 2000). Routing protocols are able to route information to a destination node. However, gathering distributed contextual information may involve thousands or even millions of pieces of data distributed among the nodes (e.g. calculating the average speed of cars driving in a city or specific road, or calculating the number of people attending a demonstration). Thus, it is not sufficient to route all data to the node interested in the information and let that node compute it. In order to gather a large amount of distributed data in a more efficient way in terms of energy, computation, and storage, in-network aggregation techniques have recently been proposed for the domain of wireless sensor networks: instead of sending all data to a destination node and letting this node compute (i.e. aggregate) the data, in-network aggregation provides the desired aggregation as a result of the distributed computation at intermediary nodes involved in the routing task (Fasolo *et al*., 2007).

Even though in-network aggregation has been mainly analysed in the wireless sensor network domain, it will play a key role for distributed context perception in future pervasive ecosystems. They share constraints including low energy (most of the devices will be sensors, laptops, mobile phones, tablets, or smart watches), low computation power, and low storage capacity. Node mobility involves different concentration on nodes, segmentation, and frequent disconnections. In a pervasive ecosystem each node executes a different operating system, and has different hardware and different capabilities. This calls for an additional requirement related to a common middleware that allows transparent interactions between different nodes. A pervasive ecosystem is not designed for a specific goal, instead the system contains many applications with different purposes interacting among themselves. A major difference between the concept of aggregation in wireless sensor network (WSN) and pervasive ecosystems is that the aggregations in WSN are usually arithmetical or statistical aggregations such as minimum, maximum, average, or count. Aggregation in a pervasive ecosystem often involves reasoning, aggregation of functionalities and predictions, as discussed in Section 3.3.

Fernandez-Marquez *et al*. (2012b) extend the general in-network aggregation and apply it to pervasive ecosystems, where a gradient is used both to propagate a query and to create a routing structure to bring the information back to the source node. Analogously to in-network aggregation, information is progressively aggregated at intermediary nodes, while it is being routed to the query node. Space plays a key role in this approach since data is gathered from nodes geographically dispersed across which gradients spread spatial information.

Remote Query and Retrieval. The combination of Gradient and Chemotaxis to query and retrieve information constitutes a mechanism on its own, termed Remote Query and Retrieval (Di Marzo Serugendo & Fernandez-Marquez, 2013). It employs progressive in-network aggregation when routing back information. It has been used for progressive self-aggregation of context information (Fernandez-Marquez *et al*., 2012b), progressive self-aggregation of situation information (Stevenson *et al*., 2012), and semantic resource discovery (Stevenson *et al*., 2013a).

Additionally, the mechanisms described above are also provided individually under the form of services, in an effort to separate concerns and to ease the engineering and development effort of self-aware spatial applications.

3.4.3 Distributed/decentralised situation recognition

The above two sub-sections discuss the case of sharing or gathering raw context information, possibly aggregating or evaporating it along the way to retain pertinent or up-to-date information. On the basis of context information, it becomes possible to reason and identify actual *situations*, that is from Bluetooth proximity signals identifying that a group of people are crowded in front of a display. Section 3.5 will discuss this issue in more detail, however, in relation to distribution and context perception and the above described mechanisms, we highlight here the case of distributed and decentralised situation recognition (Stevenson *et al*., 2012).

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

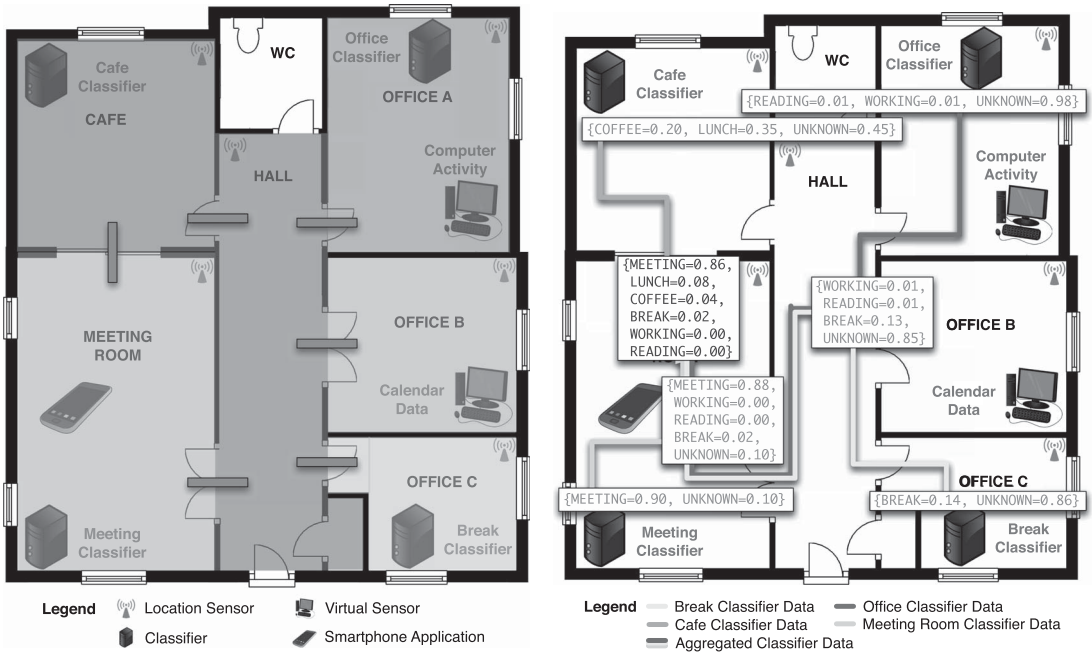


Figure 2 (Left) The architectural layout of networked spaces in a building. (Right) An illustration of the in-network aggregation of situation classifications *en route* to a common destination

As for context perception, a request, originating from an application, spreads throughout the network establishing a gradient. Typically, a sensor or a reasoner agent receiving the request evaluates whether its capability matches the request. If it matches, the sensor or reasoner agent either retrieves the matched data or starts reasoning and then encapsulates the result in a reply message that is sent back to the source according to the *Chemotaxis* mechanism. But if it does not match, the agent passes the request to the neighbouring node. To reduce communication cost, if there are more than one result matching the request, they are aggregated along the path back to the requester using the *Aggregation* mechanism. Aged information will be progressively discarded following the *Evaporation* mechanism. Inferred results are progressively aggregated while being sent back from sensors and reasoners to the application. Applications or reasoners need not know which reasoners or sensors are available in advance. The routes between an application and a reasoner or sensor are constructed based on the matching mechanism and the *Gradient* mechanism. Where the state of intermediate nodes along the transmission path changes due to failure or mobility, a dynamic gradient allows routing to automatically adapt when assuming a path remains open.

As an example of the above distributed situation recognition that combines both local and regional fusion of sensor data in a way that is decentralised and self-organising, consider a smart building containing a number of sensors and localised classifiers that determine whether or not a situation is occurring in the immediate proximity based only on spatially local sensor data. The left-hand side of Figure 2 illustrates this arrangement using one floor of a building. The spatial extent of each computational node is denoted by areas of different colours, with connections between nodes represented by grey bars. Distributed across this infrastructure are nodes containing location, calendar, and activity sensors, and four local situation classifiers that interpret this information.

We consider the case where an application located on a mobile phone in the meeting room may request that this spatially local classification information be routed towards the node hosting the application. Adopting the biological communication metaphor, computational *gradient* structures extending from the nodes hosting the classifiers and the mobile phone application broadcast requests for sensor and classification information across the network, respectively, while *Chemotaxis* draws matched information towards the query origins. Using Dempster–Shafer theory as a fusion function, *Aggregation* is applied to integrate the classifications and their uncertainties *en route* to their destination

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

(Fernandez-Marquez *et al*., 2012b). From this process, a global picture of the situation occurring emerges from the in-network generation of a consensus, calculated in a fully distributed fashion as data flows from its origins to its consumer. This is shown on the right-hand side of Figure 2.

Although this simple example considers a single floor of a building, the same concepts, interactions, and mechanisms can be employed at larger scales where the biological communication metaphor is applied.

3.4.4 Discussion

We have discussed distributed context perception and situation recognition, in particular, how to either share or retrieve context or situation in a completely decentralised manner using only local interactions and exploiting the spatial organisation of nodes.

Beside the smart building example presented in the previous section, the application domains of the presented mechanisms range from pervasive systems, to swarm robotics, to sensor networks, or to distributed traffic management. That is, all systems composed of a set of spatially distributed nodes where (1) nodes have computation and communication capabilities, (2) a node has a local perception of the system, and (3) there is no central entity controlling the system (i.e. the desired behaviour emerges from the local interactions between spatially distributed nodes).

Future directions of research suggest (1) providing those mechanisms as services (Di Marzo Serugendo & Fernandez-Marquez, 2013); (2) building spatial services on-demand relying on those mechanisms (Di Marzo Serugendo *et al*., 2014), thus easing the engineering of spatial-aware applications; (3) investigating the links between functional and non-functional aspects, and how they can be ensured for a given mechanism. Examples of such non-functional aspects include quality of service requests such as minimum bandwidth, highest accuracy, highest coverage of propagation, quickest gathering of context, etc. (Fernandez-Marquez *et al*., 2014).

3.5 Self-organising situation awareness

At the complete right side of the spectrum we have a full intertwining of situation awareness and self-organisation. Namely, situation awareness becomes a truly global, distributed, and spatially situated process, inherently robust and adaptive. Self-organisation is defined as the process of obtaining global and coherent spatio-temporal structures out of local interaction of components. Accordingly, self-organising situation awareness is achieved solely by single-device computations (environment perceptions and local reasoning) and local interactions between the involved devices (a device asynchronously interacting with its spatial neighbourhood), which ultimately result in the formation of global spatio-temporal patterns of behaviour, precisely reifying the system-level situation awareness. Such patterns are typically multi-level: the basic mechanisms of Spreading, Aggregation, and Evaporation occurring locally lie in the bottom, basic Gradient and Chemotaxis patterns stay above, and finally on top we have a whole multi-layering of further self-organisation patterns, combining and manipulating the various levels of situation awareness. In this section, we review the basic approach and existing proposals for one such view of self-organising situation awareness.

3.5.1 Spatial computations

Spatial computing (Zambonelli & Mamei, 2005; Beal *et al*., 2013) amounts to conceive computations as occurring on a spatial substrate (a physical environment as the main example), and following the constraints that this space enacts typically in terms of topology of interactions between computing devices, as well as the timing aspects including speed/bandwidth of communications, rate of computations in each device, and so on. Situation awareness enters the picture of spatial computing precisely when the situation of interest becomes tightly dependent on (mutual) position of computations/data/events present in the network, and when actions to accordingly take are again expressed in terms of spatial computations. Various approaches surveyed by Beal *et al*. (2013) mostly depend on the language used to describe the spatial computation: functional approaches like Proto (Beal & Bachrach, 2006), rule-based like MGS (Giavitto & Spicher, 2008) and Self-Aware Pervasive Service Ecosystems (Zambonelli *et al*., 2015),

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

Java-based like Tota (Mamei & Zambonelli, 2008), and others featuring logic programming and process algebraic ones. A uniform way to view at how they work is to consider the paradigmatic case of the gradient data structure described in the previous section, generalise over it, and consider the notion of computational field as a first-class one. A computational field is simply a map from the network of devices to values of some data structure (Booleans, numbers, tuples, and directional vectors): other than a gradient, examples can include the field of values perceived by a homogeneously distributed set of sensors, the Boolean field of obstacles in a traffic area, and the field of directional vectors towards a POI—point of interest. Accordingly, it is key to conceive spatial computations as computational processes with the goal of creating certain computational fields (static or changing over time), possibly as a combination of existing ones by means of the constructs that the specific model/language supports. This is extremely useful in the context considered in this paper, since such computational fields can be considered as the global situation that emerges out of local interactions, out of which decisions concerning the robust and self-* provision of some service have to be taken.

3.5.2 Layering spatial computations

It is by the process of combination of basic computational fields that more structured ones can arise, providing useful services in the context of pervasive computing, or display ecosystems in particular. A series of work (Zambonelli & Mamei, 2005; Beal & Bachrach, 2006; Beal, 2010; Viroli & Damiani, 2014) investigate the problem of identifying a minimal set of constructs for combining fields, typically including functional point-wise computation, repetition over time of a computation, extraction of values from neighbours, and restriction of computation in certain regions. As an example, a gradient structure can be achieved out of a source field, by iteratively applying repetition and extraction of neighbour values, and by suitably applying an additive point-wise function to make the field values increase with distance. Higher-level patterns can be devised accordingly as follows.

- By suitably preventing a gradient value from increasing over a certain level, at which point propagation is prevented, one can create a spatially bounded version of the gradient. This is useful when one wants to advertise a service in a limited portion of the network, or when a circular region around a centre has to be identified (e.g. to aggregate all its values).
- By creating a gradient out of multiple sources, and taking care that in a node only the value coming from the nearer source is retained, one can partition the network into as many regions as sources (by a so-called Voronoi partition): each region is identified by one source node (the one closest to its points), and its shape fully adapts to the position of others. One such pattern can be used to spatially divide a process or data throughout the network (Viroli & Stevenson, 2012).
- By summing the two gradient structures created out of two nodes a to b, we obtain a field that has the minimum values precisely in the points in the shortest routes between them; by properly filtering and spatial expansion (again by a bounded gradient), one can devise the so-called channel pattern (Viroli & Damiani, 2014), which is a region of the network connecting the two nodes, above which long-range communications can be spread. Alternatively, as of the display ecosystem scenario, such an area can be the one used to activate a steering service to guide people from a to b.
- An orthogonal problem is how we can control the actual area where the above fields spread: in fact, we might be in need of specifying that certain areas are forbidden, and the various gradients should try to circumvent it. This is achieved by simply blocking propagation in those areas (in a way actually similar to the bounded gradient) defined above. In crowd steering by display ecosystems, this can be useful to avoid certain areas when steering, for example due to some road construction.

3.5.3 Knowledge, space, and time with situation awareness

An advanced way of providing powerful situated awareness functions is based on the idea of constructing computational fields that reify or exploit not only spatial information, but also knowledge available in the networked environment (and how two knowledge items can match against each other), and timing of

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

relevant events (either happened in past, happening right now, or expected to happen in the future). All such information can diffuse and re-aggregate so as to make the overall knowledge base and time information being available globally for intercepting the situation of interest.

The first example of this approach is developed by Montagna *et al.* (2013), aiming to guide people inside a museum towards their preferences, following the shortest and possibly quickest path using the pervasive ambient displays deployed in the museum. In this example, the gradient data structure is deviated so as to penalise areas where a certain situation arises. Accordingly, the overall gradient data structure become 'aware' of those situations, such that Chemotaxis can automatically take this into account when used to move data/agents towards the source. In particular, this is achieved by extending the notion of distance reified by the gradient in those areas; for example, by an additive amount of space. A paradigmatic example in the case of pervasive displays used to steer people is when some sensors on a display detect the present of a crowded area, one which we want to circumvent while suggesting a route towards the POI. Interestingly, in an open computational environment, judging what areas have to be a source of deviation, and whether a given user might be affected by it (and how), should be based on a semantic match between user preferences/profile and data available in the environment, for example, as produced by sensors. As an example, depending on a person profile, a steering service can or cannot take into consideration how crowded is a given area.

Another advanced example is to mix self-organisation and situation awareness in the context of computational fields (Stevenson *et al.*, 2013a), where discovering of resources in spatial settings exploit semantic information about such resources to properly aggregate them *en route* (Fernandez-Marquez *et al.*, 2012b), so as to promote cooperation/competition. The basic idea is that many resources located in proximity of each other should cooperate making the whole area more appealing for the user seeking for a suitable target, while resources located far away from each other must compete, and the best one should be chosen. Accordingly, the user spreads a gradient data structure, such that information about the seek resources—in the form of a reply—can be gathered by Chemotaxis. However, as soon as two such replies reside in the same node, some degree of semantic match can activate a combination service that either drops one (competition) or aggregate them (collaboration). This can be shown to automatically select a route towards the most suitable area for the discovering at hand, computed in a fully distributed and context-aware way.

Self-organisation is also exploited by Montagna *et al.* (2012) to make timing aspects enter the realm of situation awareness—relying on so-called 'future awareness'. There, the gradient pattern is equipped with *anticipative adaptation*; namely, it can take advantage of the local availability of information about future events, once this has been provided by some component in charge of situation recognition (Ye *et al.*, 2012). Such a knowledge is used in advance to stretch and deviate certain parts of a gradient structure so as to promote alternative paths that could be longer, but guarantee to eventually circumvent any forbidden area in the precise moment in time when they would not be available for transit. This pattern finds applications in the context of steering services (e.g. traffic control or crowd steering), which can rely on local predictions about future events to anticipate alternative routes that circumvent potential sources of jams. The anticipative gradient can be seen as a time-dependent computational field, obtained by functional combination of lower-level fields along with the field of 'time passing' that is trivially assumed to be available throughout the network. One such lower-level field is called *wave*, and is used to advertise the future events in just those points from which an agent moving at a given velocity could cross the future event area when it will be active—such field is time-dependent, as it shrinks down to the future event area as time passes. Another one is called *shadow*: it spreads from the POI and tags all the positions from which an agent would cross an obstacle area while moving towards POI. A special combination of these two fields create the anticipative gradient; it is a gradient that is identical to a standard one outside the intersection of shadow and wave, otherwise distance and orientation values in each node reflect the optimal path among the two following alternative: the one crossing the future event (and waiting behind it that it is over) and the one circumventing it.

3.5.4 Discussion

Recent works pushed forward the idea of combining traditional situation awareness techniques with self-organisation, so as to design distributed and situated processes with the goal of intercepting relevant

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

data/events (which are itself intrinsically distributed) and reify them as spatio-temporal structures. When proper design techniques are exploited, such structures are often usefully seen as computational fields, and are able to self-adapt to changes in the environment and to the unpredictability of situations to occur, mostly in an emergent way. Although a relevant set of mechanisms and specific solutions have already been studied, as reviewed in this section, several issues remain open that concern the identification of a rigorous engineering methodology. As we discussed, advanced forms of self-organising situation awareness can be achieved via a combination of low-level mechanisms. On the one hand, it is key to find a suitable set of such mechanisms, including useful forms of aggregation, diffusion, and evaporations, but also of analysis techniques useful at the local level as discussed in previous sections. On the other hand, it is also important to identify the useful ways of combining those low-level mechanisms; for example, the work by Viroli and Damiani (2014) is the first attempt in the direction of finding a minimal set of constructors for computational fields.

Once a method for defining self-organising situation awareness techniques and combinators is identified, formal techniques will become useful to evaluate which behavioural properties the resulting processes have. For example, Viroli (2013) defines the problem of identifying *self-stabilising* computational fields, and sketches guidelines on how one such property can be enforced. Studying such property is key to suitably control that the possibly complex spreading processes produce a meaningful global result. Other interesting aspects include the identification of a universal set of constructs, and studying the relationship of the expected global result and the actual shape of the environment and the devices therein; for example, following the approach developed by Beal and Viroli (2014); and finally, addressing openness by deploying and managing new code in a self-organised way as advocated in Damiani *et al*. (2015).

The specific case of integrating the three key dimensions we consider, namely, space, time, and knowledge, is yet to be unveiled: we argue there is a high potential in studying patterns and applications using all of them for the identification of complex situation recognition. Finally, an appropriate tooling is key, primarily in the form of simulation techniques and suitable libraries and application programming interfaces for smooth exploitation in practice.

4 Concluding remarks

Pervasive systems are inscribed in a physical and geographical environment. In such a framework, spatial computing is a needed feature to engineer spatial concerns. Adequate mechanisms provided as services, for exploring space and retrieving information from that space, help separate concerns and help engineer pervasive systems in a dependable way. This paper reviews existing attempts to apply and integrate spatial computing technologies within pervasive service ecosystems: from modelling context, especially location context and knowledge, to organising and distributing other types of context on top of space, to perceiving and fusing context information locally in a certain space, and to achieve context and situation awareness in a distributed manner.

This review has shown that spatial computing has gained wider recognition in pervasive computing and their integration is becoming more evident in recent pervasive ecosystems. One representative example is the bio-inspired computational model, which takes biochemical metaphors to design a set of laws of nature and design patterns to govern spontaneous and self-organising interactions between entities and across the whole computation space. That is, it manipulates and reifies spatial and temporal annotations like reagents in a chemical reaction to regulate the interactions between annotations, relate them dynamically based on content, synthesise new information from old, and manage the diffusion of annotations across the environment. Entities only interact with the ecology through manipulating and observing the structure and content of their annotations. Such closer integration results in autonomous, dynamic, and self-organising system behaviours, which are desired features exhibited in pervasive ecosystems.

Zambonelli *et al*. (2015) have demonstrated a promising starting point towards systematically integrating spatial computing in designing pervasive service ecosystems, which can be further improved with rich semantics and advanced awareness that are empowered by pervasive sensing and reasoning technologies. The notation of space is no longer just bound to neighbours and distances, which can be expressed in many different semantic forms such as meaningful places or relative locations, and acquired

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

from various channels beyond traditional positioning sensors. Such semantics will play a significant role in discovering and matching services and resources, and furthermore in allowing spreading and organising information and as well as distributing reasoning capabilities in a more subtle and flexible manner. However, these advantages come at a price; that is, enabling rich spatial expressions and reasoning might require a re-thinking or a re-design of existing spatial computing technologies and their corresponding tools. We note that there are many associated challenges to spatial models of computation that we have not reviewed in this article, including, but not limited to, robustness, stability, security, and optimality of pervasive ecosystems.

Another profound implication of integrating spatial awareness with pervasive ecosystem design is *privacy*. On the one hand, spatial awareness suggests to make (absolute or relative) positions of entities explicit so as to be useful. On the other hand, space can serve as a shield to protect the privacy or control the accessibility of information. For example, Jiang and Landay (2002) bound information within a certain spatial region; that is, only when the user enters this region, can the information be accessed; and when the user moves out of the region, the access right will be revoked. Also, local context perception and situation awareness protects privacy of information further in that rather than broadcasting all the information globally, a spatial-aware pervasive system (Ormándi *et al.*, 2013) allows to locally aggregate and abstract information to higher-level concepts, which prevents finer-grained specific information from being shared across the whole network. In summary, the system designers could leverage such locality of spatial awareness in preserving privacy.

Acknowledgement

This work is supported by the FP7 FET proactive project SAPERE—Self-Aware Pervasive Service Ecosystems, under grant no. 25873.

References

Adjie-Winoto, W., Schwartz, E., Balakrishnan, H. & Lilley, J. 1999. The design and implementation of an Intentional Naming System. *ACM SIGOPS Operating Systems Review* **33**(5), 186–201.

Azizyan, M. & Choudhury, R. 2009. Surroundsense: mobile phone localization using ambient sound and light. *ACM SIGMOBILE Mobile Computing and Communications Review* **13**, 69–72.

Banâtre, M., Allard, F. & Couderc, P. 2010. A spatial computing approach for integrity checking of objects groups. In *2010 Fourth IEEE International Conference on Self-Adaptive and Self-Organizing Systems Workshop (SASOW)*, 80–84, September.

Beal, J. 2010. A basis set of operators for space-time computations. In *SASO Workshops*, 91–97. IEEE Computer Society.

Beal, J. & Bachrach, J. 2006. Infrastructure for engineered emergence on sensor/actuator networks. *IEEE Intelligent Systems* **21**, 10–19.

Beal, J., Dulman, S., Usbeck, K., Viroli, M. & Correll, N. 2013. Organizing the aggregate: languages for spatial computing, chapter 16. In *Formal and Practical Aspects of Domain-Specific Languages: Recent Developments*, Mernik, M. (ed.). IGI Global, 436–501.

Beal, J. & Viroli, M. 2014. Building blocks for aggregate programming of self-organising applications. In *Eighth IEEE International Conference on Self-Adaptive and Self-Organizing Systems Workshops, SASOW 2014*, 8–13, 8–12 September.

Bettini, C., Brdiczka, O., Henricksen, K., Indulska, J., Nicklas, D., Ranganathan, A. & Riboni, D. 2010. A survey of context modelling and reasoning techniques. *Pervasive and Mobile Computing* **6**(2), 161–180.

Bicocchi, N., Lasagni, M. & Zambonelli, F. 2012a. Bridging vision and commonsense for multimodal situation recognition in pervasive systems. In *2012 IEEE International Conference on Pervasive Computing and Communications*, 48–56, 19–23 March.

Bicocchi, N., Mamei, M. & Zambonelli, F. 2012b. Self-organizing virtual macro sensors. *ACM Transactions on Autonomous and Adaptive Systems* **7**(1), 2.

Bortenschlager, M., Castelli, G., Rosi, A. & Zambonelli, F. 2009. A context-sensitive infrastructure for coordinating agents in ubiquitous environments. *Multiagent and Grid Systems* **5**(1), 1–18.

Cabri, G., Leonardi, L. & Zambonelli, F. 2000a. MARS: a programmable coordination architecture for mobile agents. *IEEE Internet Computing* **4**(4), 26–35.

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

Cabri, G., Leonardi, L. & Zambonelli, F. 2000b. Mobile-agent coordination models for internet applications. *Computer* **33**(2), 82–89.

Castelli, G., Mamei, M., Rosi, A. & Zambonelli, F. 2009. Extracting high-level information from location data: the W4 diary example. *Mobile Networks and Applications* **14**(1), 107–119.

Chen, G. & Kotz, D. 2002. Context aggregation and dissemination in ubiquitous computing systems. In *Proceedings of the Fourth IEEE Workshop on Mobile Computing Systems and Applications, WMCSA '02*, 105–. IEEE Computer Society.

Chen, G., Li, M. & Kotz, D. 2004. Design and implementation of a large-scale context fusion network. In *The First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, 2004. MOBIQUITOUS 2004*, 246–255, August.

Clement, L. & Nagpal, R. 2003. Self-assembly and self-repairing topologies. In *Workshop on Adaptability in Multi-Agent Systems, First RoboCup Australian Open (AORC 2003)*.

Cohen, N. H., Lei, H., Castro, P., Davis, J. S. II & Purakayastha, A. 2002. Composing pervasive data using iQL. In *Proceedings of the Fourth IEEE Workshop on Mobile Computing Systems and Applications, WMCSA '02*, 94–. IEEE Computer Society.

Corradi, A., Leonardi, L. & Zambonelli, F. 1999. Diffusive load balancing policies for dynamic applications. *IEEE Concurrency* **7**(11), 22–31.

Costa, P., Mottola, L., Murphy, A. & Picco, P. 2007. Programming wireless sensor networks with the TeenyLIME middleware. In *ACM Middleware Conference*.

Damiani, F., Viroli, M., Pianini, D. & Beal, J. 2015. Code mobility meets self-organisation: a higher-order calculus of computational fields. In *Formal Techniques for Distributed Objects, Components, and Systems*, Graf, S. & Viswanathan, M. (ed.), Lecture Notes in Computer Science **9039**, 113–128. Springer International Publishing.

Dempster, A. P. 1968. A generalization of Bayesian inference. *The Royal Statistical Society, Series B* **30**, 205–247.

Dey, A. K. 2001. Understanding and using context. *Personal and Ubiquitous Computing* **5**(1), 4–7.

Dey, A. K., Abowd, G. D. & Salber, D. 2001. A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. *Human-Computer Interaction* **16**, 97–166.

Dey, A. K., Mankoff, J., Abowd, G. D. & Carter, S. A. 2002. Distributed mediation of ambiguous context in aware environments. In *Proceedings of the Fifteenth Annual Symposium on User Interface Software and Technology (UIST 2002)*, 121–130.

Di Marzo Serugendo, G. & Fernandez-Marquez, J. L. 2013. Self-organising services. In *IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO2013)*. IEEE Computer Society.

Di Marzo Serugendo, G., Fernandez-Marquez, J. L. & De Angelis, F. L. 2014. Engineering spatial services: concepts, architecture, and execution models. chapter 6. In *Handbook of Research on Architectural Trends in Service-Driven Computing*, Raja, R. & Raja, K. (eds). IGI Global, 136–159.

Dimakis, A., Sarwate, A. & Wainwright, M. 2006. Geographic gossip: efficient aggregation for sensor networks. In *International Conference on Information Processing in Sensor Networks*.

Dobson, S. 2005. Leveraging the subtleties of location. In *sOc-EUSAI'05: Proceedings of the 2005 Joint Conference on Smart Objects and Ambient Intelligence*, 175–179. ACM Press.

Eugster, P. Th., Guerraoui, R., Handurukande, S. B., Kermarrec, A.-M. & Kouznetsov, P. 2003. Lightweight probabilistic broadcast. *ACM Transaction on Computer Systems* **21**, 341–374.

Fasolo, E., Rossi, M., Widmer, J. & Zorzi, M. 2007. In-network aggregation techniques for wireless sensor networks: a survey. *Wireless Communications, IEEE [see also IEEE Personal Communications]* **14**(2), 70–87.

Fernandez-Marquez, J. L., Di Marzo Serugendo, G., Montagna, S., Viroli, M. & Arcos, J. L. 2012a. Description and composition of bio-inspired design patterns: a complete overview. *Natural Computing* **12**, 1–25.

Fernandez-Marquez, J. L., Di Marzo Serugendo, G., Stevenson, G., Ye, J., Dobson, S. & Zambonelli, F. 2014. Self-managing and self-organising mobile computing applications: a separation of concerns approach. In *Proceedings of the 29th Annual ACM Symposium on Applied Computing, SAC '14*, 458–465. ACM.

Fernandez-Marquez, J. L., Stevenson, G., Tchao, A. E., Ye, J., Di Marzo Serugendo, G. & Dobson, S. 2012b. Analysis of new gradient based aggregation algorithms for data-propagation in distributed networks. In *1st International Workshop on Adaptive Service Ecosystems: Natural and Socially Inspired Solutions (ASENSIS 2012)*, Fernandez-Marquez, J. L., Montagna, S., Omicini, A. & Zambonelli, F. (eds). SASO, September 2012.

Ferscha, A., Zia, K. & Gollan, B. 2012. Collective attention through public displays. In *2012 IEEE Sixth International Conference on Self-Adaptive and Self-Organizing Systems (SASO)*, 211–216, September.

Ganti, R. K., Srinivasan, S. & Gacic, A. 2010. Multisensor fusion in smartphones for lifestyle monitoring. In *Proceedings of the 2010 International Conference on Body Sensor Networks, BSN '10*, 36–43. IEEE Computer Society.

Gardelli, L., Viroli, M. & Omicini, A. 2007. Design patterns for self-organizing multiagent systems. In *Proceedings of EEDAS*.

Gehrke, J. & Madden, S. 2004. Query processing in sensor networks. *IEEE Pervasive Computer* **3**(1), 46–65.

Gelernter, D. 1985. Generative communication in Linda. *ACM Transactions on Programming Languages and Systems* **7**(1), 80–112.

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

Giavitto, J.-L. & Spicher, A. 2008. Topological rewriting and the geometrization of programming. *Physica D* **237**(9), 1302–1314.

Heer, J., Newberger, A., Beckmann, C. & Hong, J. 2003. Liquid: context-aware distributed queries. In *UbiComp 2003: Ubiquitous Computing*, Dey, A. K., Schmidt, A. & McCarthy, J. F. (eds), Lecture Notes in Computer Science **2864**, 140–148. Springer Berlin Heidelberg.

Henricksen, K. & Indulska, J. 2004. A software engineering framework for context-aware pervasive computing. In *Proceedings of the Second IEEE International Conference on Pervasive Computing and Communications (PerCom'04)*, 77–. IEEE Computer Society.

Hightower, J. 2003. From position to place. *In Proceedings of the 2003 Workshop on Location-Aware Computing*, 10–12.

Hightower, J., Brumitt, B. & Borriello, G. 2002. The Location Stack: a layered model for location in ubiquitous computing. In *Proceedings Fourth IEEE Workshop on Mobile Computing Systems and Applications*, 22–28.

Hohl, F., Kubach, U., Leonhardi, A., Rothermel, K. & Schwehm, M. 1999. Next century challenges: Nexus— an open global infrastructure for spatial-aware applications. In *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking, MobiCom '99*, 249–255. ACM.

Hong, J. I. 2002. The Context Fabric: an infrastructure for context-aware computing. In *CHI '02 Extended Abstracts on Human Factors in Computing Systems, CHI EA '02*, 554–555. ACM.

Hong, J. I. & Landay, J. A. 2001. An infrastructure approach to context-aware computing. *Human-Computer Interaction* **16**, 287–303.

Huebel, N., Hirche, S., Gusrialdi, A., Hatanaka, T., Fujita, M. & Sawodny, O. 2008. Coverage control with information decay in dynamic environments. In *Proceedings of the 17th IF AC World Congress*, 4180–4185.

Jelasity, M., Montresor, A. & Babaoglu, O. 2005. Gossip-based aggregation in large dynamic networks. *ACM Transactions on Computer Systems* **23**(3), 219–252.

Jiang, C. & Steenkiste, P. 2002. A hybrid location model with a computable location identifier for ubiquitous computing. In *Proceedings of the 4th International Conference on Ubiquitous Computing, UbiComp '02*, 246–263. Springer-Verlag.

Jiang, X. & Landay, J. A. 2002. Modeling privacy control in context-aware systems. *Pervasive Computing, IEEE* **1**(3), 59–63.

Johnson, D. B. & Maltz, D. A. 1996. Dynamic Source Routing in ad hoc wireless networks. In *Mobile Computing*, 153–181. Kluwer Academic Publishers.

Jung, D., Teixeira, T. & Savvides, A. 2010. Towards cooperative localization of wearable sensors using accelerometers and cameras. In *Proceedings of the 29th Conference on Information Communications, INFOCOM'10*, 2330–2338. IEEE Press.

Kabaday, S. & Julien, C. 2007. Scenes: abstracting interaction in immersive sensor networks. *Journal on Pervasive and Mobile Computing* **3**(6), 635–658.

Khelil, A., Becker, C., Tian, J. & Rothermel, K. 2002. An epidemic model for information diffusion in MANETs. In *MSWiM '02: Proceedings of the 5th ACM International Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems*, 54–60. ACM.

Liao, L., Fox, D. & Kautz, H. 2007. Extracting places and activities from GPS traces using hierarchical conditional random fields. *International Journal of Pattern Recognition and Artificial Intelligence* **26**, 119–134.

Liu, H. & Singh, P. 2004. Conceptnet, a practical commonsense reasoning tool-kit. *BT Technology Journal* **22**, 211–226.

Lotfinezhad, M., Liang, B. & Sousa, E. 2008. Adaptive cluster-based data collection in sensor networks with direct sink access. *IEEE Transactions on Mobile Computing* **7**, 884–897.

Kranz, M., Franz, A., Strang, T. & Rockl, M 2008. Codar viewer—a v2v communication awareness display. In *Pervasive 2008 Late Breaking Results*, 79–82.

Mamei, M. 2010. Applying commonsense reasoning to place identification. *IJHCR* **1**(2), 36–53.

Mamei, M. & Zambonelli, F. 2008. Programming pervasive and mobile computing applications: the TOTA approach. *ACM Transactions on Software Engineering and Methodology* **18**, 4.

Mckeever, S., Ye, J., Coyle, L., Bleakley, C. & Dobson, S. 2010. Activity recognition using temporal evidence theory. *Journal of Ambient Intelligence and Smart Environments* **2**(3), 253–269.

Montagna, S., Pianini, D. & Viroli, M. 2012. Gradient-based self-organisation patterns of anticipative adaptation. In *Proceedings of 6th IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO 2012)*, 169–174, September. IEEE.

Montagna, S., Viroli, M., Fernandez-Marquez, J. L., Di Marzo Serugendo, G. & Zambonelli, F. 2013. Injecting self-organisation into pervasive service ecosystems. *Mobile Networks and Applications* **18**(3), 398–412.

Mottola, G. & Picco, G. P. 2006. Logical neighborhoods: a programming abstraction for wireless sensor networks. In *IEEE International Conference on Distributed Computing in Sensor Systems*.

Mottola, L. & Picco, G. P. 2011. Programming wireless sensor networks: fundamental concepts and state of the art. *ACM Computing Surveys* **43**(3), 19.

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

Mozer, M. 1998. The neural network house: an environment that adapts to its inhabitants. In *Proceedings of the American Association for Artificial Intelligence*, 110–114.

Nii, H. P. 1986. The blackboard model of problem solving and the evolution of blackboard architectures. *AI Magazine* **7**(2), 38.

Niu, R. & Varshney, P. K. 2005. Distributed detection and fusion in a large wireless sensor network of random size. *EURASIP Journal on Wireless Communications and Networking* **2005**, 462–472.

Obraczka, K., Viswanath, K. & Tsudik, G. 2001. Flooding for reliable multicast in multi-hop ad hoc networks. *Wireless Networks* **7**(6), 627–634.

Omicini, A. & Zambonelli, F. 1999. Tuple centres for the coordination of internet agents. In *Proceedings of the 1999 ACM Symposium on Applied Computing, SAC '99*, 183–190. ACM.

Ormándi, R., Hegedüs, I. & Jelasity, M. 2013. Gossip learning with linear models on fully distributed data. *Concurrency and Computation: Practice and Experience* **25**(4), 556–571.

Pei, G., Gerla, M. & Chen, T.-W. 2000. Fisheye State Routing in mobile ad hoc networks. In *ICDCS Workshop on Wireless Networks and Mobile Computing*, 71–78.

Perkins, C., Belding-Royer, E. & Das, S. 2003. *Ad-Hoc On-Demand Distance Vector (AODV) Routing*. RFC 3561, July 2003.

Polastre, J., Szewcyk, R., Mainwaring, A., Culler, D. & Anderson, J. 2004. Analysis of wireless sensor networks for habitat monitoring. *Wireless Sensor Networks* 399–423.

Ranganathan, A., Al-Muhtadi, J., Chetan, S., Campbell, R. & Mickunas, M. D. 2004a. Middlewhere: a middleware for location awareness in ubiquitous computing applications. In *Proceedings of Middleware '04*, 397–416.

Ranganathan, A., Al-Muhtadi, J., Chetan, S., Campbell, R. & Mickunas, M. D. 2004b. Middlewhere: a middleware for location awareness in ubiquitous computing applications. In *Proceedings of the 5th ACM/IFIP/USENIX International Conference on Middleware, Middleware '04*, 397–416. Springer-Verlag New York, Inc.

Rhodes, B. J. 1997. The wearable Remembrance Agent: a system for augmented memory. In *First International Symposium on Wearable Computers, 1997. Digest of Papers*, 123–128.

Sabbineni, H. & Chakrabarty, K. 2005. Location-aided flooding: an energy-efficient data dissemination protocol for wireless sensor networks. *IEEE Transactions on Computers* **54**, 36–46.

Sarkar, R., Zhu, X. & Gao, J. 2007. Hierarchical spatial gossip for multi-resolution representations in sensor networks. In *International Conference on Information Processing in Sensor Networks*.

Sasson, Y., Cavin, D. & Schiper, A. 2003. Probabilistic broadcast for flooding in wireless mobile ad hoc networks. In *Wireless Communications and Networking, 2003. WCNC 2003. 2003 IEEE*, **2**, 1124–1130.

Shafer, G. 1976. *A Mathematical Theory of Evidence*. Princeton University Press.

Stevenson, G., Ye, J., Dobson, S. & Nixon, P. 2010. LOC8: a location model and extensible framework for programming with location. *IEEE Pervasive Computing* **9**, 28–37.

Stevenson, G., Fernandez-Marquez, J. L., Montagna, S., Rosi, A., Ye, J., Di Marzo Serugendo, G., Viroli, M., Dobson, S. & Tchao, A.-E. 2012. Situated awareness in urban networks: a bio-inspired approach. In *First International Workshop on Adaptive Service Ecosystems: Nature and Socially Inspired Solutions (ASENSIS) at Sixth IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO12)*. IEEE Computer Society.

Stevenson, G., Pianini, D., Montagna, S., Viroli, M., Ye, J. & Dobson, S. 2013a. Combining self-organisation, context-awareness and semantic reasoning: the case of resource discovery in opportunistic networks. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing*.

Stevenson, G., Ye, J., Dobson, S., Castelli, G., Rosi, A. & Zambonelli, F. 2013b. A bio-chemical approach to awareness in pervasive systems. In *Proceedings of First International Workshop on Sensing and Big Data Mining, SENSEMINE'13*, 7:1–7:6. ACM.

Tran, D. T. & Phung, D. Q. 2006. A probabilistic model with parsinomious representation for sensor fusion in recognizing activity in pervasive environment. In *Proceedings of the 18th International Conference on Pattern Recognition—Volume 03, ICPR '06*, 168–172. IEEE Computer Society.

Tseng, Y.-C., Ni, S.-Y., Chen, Y.-S. & Sheu, J.-P. 2002. The broadcast storm problem in a mobile ad hoc network. *Wireless Networks* **8**(2/3), 153–167.

Viroli, M. 2013. Engineering confluent computational fields: from functions to rewrite rules. In *Spatial Computing Workshop (SCW 2013), AAMAS 2013*.

Viroli, M., Casadei, M., Montagna, S. & Zambonelli, F. 2011. Spatial coordination of pervasive services through chemical-inspired tuple spaces. *ACM Transactions on Autonomous and Adaptive Systems* **6**(2), 14:1–14:24.

Viroli, M. & Damiani, F. 2014. A calculus of self-stabilising computational fields. In *Coordination Languages and Models*, Eva, K. & Rosario, P. (eds), LNCS **8459**, 163–178. Springer-Verlag, June. Proceedings of the 16th Conference on Coordination Models and Languages (Coordination 2014), 3–5 June. Best Paper of Discotec 2014 Federated Conference.

Viroli, M. & Stevenson, G. 2012. On the space-time situation of pervasive service ecosystems. In *Workshop on Spatial Computing*.

White, J. 1997. Mobile agents. In *Software Agents*, Bradshaw, J. (ed.), AAAI Press.

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press

Wu, H., Siegel, M., Stiefelhagen, R. & Yang, J. 2002. Sensor fusion using Dempster-Shafer theory. In *Proceedings of the IEEE Instrumentation and Measurement Technology Conference*, **1**, 7–12, May.

Yang, H., Ye, F. & Sikdar, B. 2008. A swarm intelligence based protocol for data acquisition in networks with mobile sinks. *IEEE Transactions on Mobile Computing* **7**(8), 931–945.

Ye, J., Coyle, L., Dobson, S. & Nixon, P. 2007. A unified semantics space model. In *Location- and Context-Awareness*, Jeffrey, H., Bernt, S. & Thomas, S. (eds), LNCS **4718**, 103–120. Springer.

Ye, J., Dobson, S. & McKeever, S. 2012. Situation identification techniques in pervasive computing: a review. *Pervasive and Mobile Computing* **8**, 36–66.

Ye, J., McKeever, S., Coyle, L., Neely, S. & Dobson, S. 2008. Resolving uncertainty in context integration and abstraction. In *ICPS '08: Proceedings of the International Conference on Pervasive Services*, 131–140. ACM, July.

Ye, J., Stevenson, G. & Dobson, S. 2014. KCAR: a knowledge-driven approach for concurrent activity recognition. *Pervasive and Mobile Computing* **19**, 47–70.

Yi, Y. & Gerla, M. 2003. Efficient flooding in ad hoc networks: a comparative performance study. In *Proceedings of the IEEE International Conference on Communications (ICC)*, 1059–1063.

Yick, J., Mukherjee, B. & Ghosal, D. 2008. Wireless sensor network survey. *Computer Networks* **52**(12), 2292–2330.

Zambonelli, F. & Mamei, M. 2005. Spatial computing: an emerging paradigm for autonomic computing and communication. In *1st International Workshop on Autonomic Communication*, Lecture Notes in Computer Science **3457**, 44–57. Springer.

Zambonelli, F., Omicini, A., Anzengruber, B., Castelli, G., De Angelis, F. L., Di Marzo Serugendo, G., Dobson, S., Fernandez-Marquez, J. L., Ferscha, A., Mamei, M., Mariani, S., Molesini, A., Montagna, S., Nieminen, J., Pianini, D., Risoldi, M., Rosi, A., Stevenson, G., Viroli, M. & Ye, J. 2015. Developing pervasive multi-agent systems with nature-inspired coordination. *Pervasive and Mobile Computing* **17** (Part B), 236–252.

Zhang, D., Cao, J., Zhou, J. & Guo, M. 2009. Extended Dempster-Shafer theory in context reasoning for ubiquitous computing environments. In *CSE '09: Proceedings of the 2009 International Conference on Computational Science and Engineering*, 205–212. IEEE Computer Society.

Zhang, Q. & Agrawal, D. P. 2005. Dynamic probabilistic broadcasting in {MANETs}. *Journal of Parallel and Distributed Computing* **65**(2), 220–233.

https://doi.org/10.1017/S0269888916000199 Published online by Cambridge University Press