Modeling mobility and communication in a unified way

Jian-Min Jiang, Xiaofei Yu and Zhong Hong
Department of Software Engineering, Fujian Normal University, Fuzhou 350007, China

Abstract—**Traditional formalisms model communication and mobility in a separate way. This may cause complex name management and complex analysis for a communicating and mobile system. In this paper, following the ambient calculus [2], we first propose two types of special events, entering and exiting an ambient, as movement events and discuss the relationship of ambients based on mobility. Then a communication model is introduced based on message movement, which can represent synchronous communication, asynchronous communication and broadcasting communication in a unified way. Finally, we show that such a communication model is contained in a general event-based formal model called a dependency structure [4], [5].**

I. Introduction

Communication and mobility are two essential aspects of complex mobile systems including mobile cyber-physical systems. Most existing formal models and languages have not unified communication and mobility modeling and analysis yet. It is difficult to model and analyze complex mobile systems using formal methods. In the communication aspect, most work assumed a synchronous communication model (e.g., [7]), regarded synchronous communication as special asynchronous communication (e.g., [3]) or independently considered broadcast communication (e.g., [8]). In the mobility aspect, different formal methods have different mechanisms. Mobile Petri nets [1] express process mobility by using variables and colored tokens in an otherwise static net, while dynamic Petri nets [11] extend mobile Petri nets with mechanisms for modifying the structure of a Petri net. The π-calculus [7] is a process algebra where the movement of processes is represented as the movement of channels that refer to processes. One of the most outstanding methods is a calculus of mobile agents called ambient calculus [2]. Ambients are administrative domains and can enter and leave other ambients and perform computations. However, in the ambient calculus, the non-deterministic choice control of processes cannot be expressed like CCS [6]. Moreover, communication needs to use special primitives. More detail refers readers to the literature [5].

Event-based models such as event structures [10] and dependency structures [4], [5] model synchronous communication, asynchronous communication and broadcast communication in a unified way. Movement events are also used to represent mobility [5]. In this paper, we present a unified approach for modeling communication and mobility.

II. Event and movement event

Event is the primitive notion of event-based formal models [9], [5]. Generally, an event refers to an occurrence of an activity or action. It implicitly contains space and time information.

As defined in the ambient calculus [2], an ambient is a closed and bounded place where computation happens. It can be nested in other ambients and can be moved as a whole [2]. To model mobility, we use the two types of special events: *entering* and *exiting* an ambient. The name of an ambient is contained in the two types of events. When a mobile object enters or exits an ambient, the entering or exiting event itself can contain such an ambient. Such consideration can avoid complex name management. For convenience, $\mathbf{M}$ and $\mathbf{A}$ denote the sets of mobile objects (agents) and ambients, respectively.

Definition II.1 Let $\mathcal{M} \in \mathbf{M}$ and $\mathcal{A} \in \mathbf{A}$.

(1) The event of $\mathcal{M}$ for entering $\mathcal{A}$ is called *an entering event*, denoted by $en^{\mathcal{M}}_{\mathcal{A}}$, and the event of $\mathcal{M}$ for exiting $\mathcal{A}$ is called *an exiting event*, denoted by $ex^{\mathcal{M}}_{\mathcal{A}}$.

We define that $\mathcal{M}$ *passes through* $\mathcal{A}$ iff the two events $en^{\mathcal{M}}_{\mathcal{A}}$ and $ex^{\mathcal{M}}_{\mathcal{A}}$ occur in sequence.

(2) An event e is called a *movement event* in $\mathcal{M}$ iff there exists an ambient $\mathcal{A} \in \mathbf{A}$ such that $(e = en^{\mathcal{M}}_{A}) \vee (e = ex^{\mathcal{M}}_{A})$. $E(\mathcal{M})$ denotes the set of all movement events in $\mathcal{M}$.

In the definition, the two events of entering and exiting an ambient are called *movement events*. Note that our framework will not involve other movement events because the two events are sufficiently used to model mobility. According to the definition, movement events in fact contain mobile objects and the ambients involved in the events. For simplicity, *non-movement events* do not consider these information in our framework. Let $\mathbf{E}$ denote the domain (set) of events including movement and non-movement events. Let $e_1, e_2 \in \mathbf{E}$. The notation $e_1 \rightarrow e_2$ is called *a dependency* that denotes that the occurrence of the event e_2 depends on the previous occurrence of the event e_1.

DOI reference number: 10.18293/SEKE 2018-015

III. The relationship of ambients

The ambient calculus [2] of Cardelli and Gordon focuses on the handling of administrative domains where mobile objects may enter a domain or exit from a domain and in this way may change the topology of the network. Since an ambient is a closed and bounded place, any *movement step* is that a mobile object moves from an ambient to one of its adjacent sibling ambients or from a parent ambient to a child ambient, or vice versa. Therefore, there only exists one of the two kinds of relationships between any two ambients in a mobile system: *parent-child* and *adjacency* (see Figure 1). The parent-child relationship is the inclusion relationship while the adjacency relationship is the sibling relationship.

When a mobile object $\mathcal{M}$ moves from an ambient $\mathcal{X}$ to its adjacent sibling ambient $\mathcal{Y}$, it needs to first exit $\mathcal{X}$ and then enter $\mathcal{Y}$. We can use the two events $ex^{\mathcal{M}}_{\mathcal{X}}, en^{\mathcal{M}}_{\mathcal{Y}}$ to model this situation. When a mobile object $\mathcal{M}$ moves from a parent ambient $\mathcal{X}'$ to its child ambient $\mathcal{Y}'$, since $\mathcal{X}'$ contains $\mathcal{Y}'$ and $\mathcal{M}$ is located in $\mathcal{X}'$, $\mathcal{M}$ only need to enter $\mathcal{Y}'$, that is, only one entry movement event $en^{\mathcal{M}}_{\mathcal{Y}'}$ occurs. Similarly, when a mobile object $\mathcal{M}'$ moves from a child ambient $\mathcal{Y}'$ to its parent ambient $\mathcal{X}'$, only one exit movement event $ex^{\mathcal{M}'}_{\mathcal{Y}'}$ occurs. Therefore, we give the following definition.

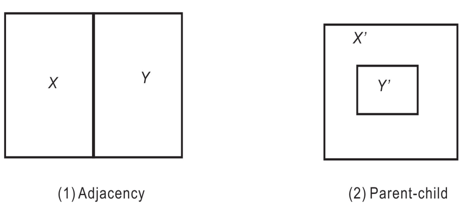

Fig. 1. The relationship of two ambients

Definition III.1 Let $\mathcal{A}_1, \mathcal{A}_2 \in \mathbf{A}$.

(1) (*Parent-child*) $\mathcal{A}_2$ is called a *child ambient* of $\mathcal{A}_1$, denoted by $\mathcal{A}_2 \Subset \mathcal{A}_1$ or $\mathcal{A}_1 \Supset \mathcal{A}_2$, iff for all $\mathcal{M} \in \mathbf{M}$: (i) if $\mathcal{M}$ moves from $\mathcal{A}_1$ to $\mathcal{A}_2$, there exists only one movement event $en^{\mathcal{M}}_{\mathcal{A}_2}$, and (ii) if $\mathcal{M}$ moves from $\mathcal{A}_2$ to $\mathcal{A}_1$, there exists only one movement event $ex^{\mathcal{M}}_{\mathcal{A}_2}$. The notation $\mathcal{A}_2 \not\Subset \mathcal{A}_1$ denotes that $\mathcal{A}_2$ is not a child ambient of $\mathcal{A}_1$.

(2) (*Adjacency*) $\mathcal{A}_1$ is said to be *adjacent to* $\mathcal{A}_2$, denoted by $\mathcal{A}_1 \Rrightarrow \mathcal{A}_2$, iff for all $\mathcal{M} \in \mathbf{M}$, if $\mathcal{M}$ moves from $\mathcal{A}_1$ to $\mathcal{A}_2$, there only exist two movement events $ex^{\mathcal{M}}_{\mathcal{A}_1}$ and $en^{\mathcal{M}}_{\mathcal{A}_2}$ that occur in sequence. The notation $\mathcal{A}_1 \not\Rrightarrow \mathcal{A}_2$ denotes that $\mathcal{A}_1$ is not adjacent to $\mathcal{A}_2$.

(3) (*connectivity*) $\mathcal{A}_1$ is said to be *connected to* $\mathcal{A}_2$, denoted by $\mathcal{A}_1 \gg \mathcal{A}_2$, iff there exists a sequence $\mathcal{B}_1 \cdots \mathcal{B}_n (\mathcal{B}_1, \cdots, \mathcal{B}_n \in \mathbf{A})$ such that $\forall i \in \{1, \cdots, n-1\}, \mathcal{B}_i \Subset \mathcal{B}_{i+1} \vee \mathcal{B}_{i+1} \Subset \mathcal{B}_i \vee \mathcal{B}_i \Rrightarrow \mathcal{B}_{i+1}$. The notation $\mathcal{A}_1 \not\gg \mathcal{A}_2$ denotes that $\mathcal{A}_1$ is not connected to $\mathcal{A}_2$.

The parent-child relationship is bidirectional, that is, mobile objects can move between the parent-child ambients. The adjacency relationship is unidirectional because one ambient is adjacent to another and the reverse adjacency relationship between the two ambients does not necessarily hold. The isolation between ambients means that there does not exist a mobile object that moves between the ambients while the connectivity indicates that any mobile object can move between ambients, but is very possibly unidirectional (because the adjacency relationship as a part of the connectivity is unidirectional).

Proposition III.1 Let $\mathcal{A}, \mathcal{B}, \mathcal{C} \in \mathbf{A}$.
(1) $\mathcal{A} \Subset \mathcal{B} \Longrightarrow \mathcal{A} \gg \mathcal{B} \wedge \mathcal{B} \gg \mathcal{A}$.
(2) $\mathcal{A} \Rrightarrow \mathcal{B} \Longrightarrow \mathcal{A} \gg \mathcal{B}$.
(3) $\mathcal{A} \Subset \mathcal{B} \Subset \mathcal{C} \Longrightarrow \mathcal{A} \gg \mathcal{C} \wedge \mathcal{C} \gg \mathcal{A}$.
(4) $\mathcal{A} \Rrightarrow \mathcal{B} \Rrightarrow \mathcal{C} \Longrightarrow \mathcal{A} \gg \mathcal{C}$.
(5) $\mathcal{A} \gg \mathcal{B} \gg \mathcal{C} \Longrightarrow \mathcal{A} \gg \mathcal{C}$.
Proof This proof is straightforward. □

Proposition III.1 shows that there exists the following properties of the relationship between ambients: (1) if one ambient is a child of another, the two ambients are connected to each other; (2) adjacent ambients have unidirectional connectivity; (3) the transitivity of parent-child relations implies bidirectional connectivity; (4) the transitivity of adjacency relations indicates unidirectional connectivity; (5) the connectivity relation is transitive; and the isolation relation is symmetric.

Theorem III.1 Let $\mathcal{A}, \mathcal{B} \in \mathbf{A}$.
(1) If $\mathcal{A} \Subset \mathcal{B}$, then $\forall \mathcal{X} \Subset \mathcal{A}, \forall \mathcal{Y} \Subset \mathcal{B}, \mathcal{X} \gg \mathcal{Y} \wedge \mathcal{Y} \gg \mathcal{X}$.
(2) If $\mathcal{A} \Rrightarrow \mathcal{B}$, then $\forall \mathcal{X} \Subset \mathcal{A}, \forall \mathcal{Y} \Subset \mathcal{B}, \mathcal{X} \gg \mathcal{Y}$.
(3) If $\mathcal{A} \gg \mathcal{B}$, then $\forall \mathcal{X} \Subset \mathcal{A}, \forall \mathcal{Y} \Subset \mathcal{B}, \mathcal{X} \gg \mathcal{Y}$.
(4) If there exist $\mathcal{X} \Subset \mathcal{A}, \mathcal{Y} \Subset \mathcal{B}$ such that $\mathcal{X} \gg \mathcal{Y}$, then $\mathcal{A} \gg \mathcal{B}$.

Proof

(1) By Proposition III.1(1), $\forall \mathcal{X} \Subset \mathcal{A} \Rightarrow \mathcal{X} \gg \mathcal{A} \wedge \mathcal{A} \gg \mathcal{X}$, $\forall \mathcal{Y} \Subset \mathcal{B} \Rightarrow \mathcal{Y} \gg \mathcal{B} \wedge \mathcal{B} \gg \mathcal{Y}$, and $\mathcal{A} \Subset \mathcal{B} \Rightarrow \mathcal{A} \gg \mathcal{B} \wedge \mathcal{B} \gg \mathcal{A}$. Since $\mathcal{X} \Subset \mathcal{A}$ and $\mathcal{A} \Subset \mathcal{B}$, $\mathcal{X} \Subset \mathcal{A} \Subset \mathcal{B}$. Then, according to Proposition III.1(3), $\mathcal{X} \gg \mathcal{B} \wedge \mathcal{B} \gg \mathcal{X}$. Since $\mathcal{Y} \gg \mathcal{B} \wedge \mathcal{B} \gg \mathcal{Y}$, $\mathcal{X} \gg \mathcal{B} \gg \mathcal{Y}$ and $\mathcal{Y} \gg \mathcal{B} \gg \mathcal{X}$. Therefore, by Proposition III.1(5), $\mathcal{X} \gg \mathcal{Y} \wedge \mathcal{Y} \gg \mathcal{X}$.

(2) By Proposition III.1(1), $\forall \mathcal{X} \Subset \mathcal{A} \Rightarrow \mathcal{X} \gg \mathcal{A}$ and $\forall \mathcal{Y} \Subset \mathcal{B} \Rightarrow \mathcal{B} \gg \mathcal{Y}$. Since $\mathcal{A} \Rrightarrow \mathcal{B}$, by Proposition III.1(2), $\mathcal{A} \gg \mathcal{B}$. Therefore, $\mathcal{X} \gg \mathcal{A} \gg \mathcal{B} \gg \mathcal{Y}$. According to Proposition III.1(5), $\mathcal{X} \gg \mathcal{Y}$.

(3) Since $\mathcal{X} \Subset \mathcal{A}, \mathcal{Y} \Subset \mathcal{B}$, by Definition III.1(3), $\mathcal{X} \gg \mathcal{A} \wedge \mathcal{B} \gg \mathcal{Y}$. Also, since $\mathcal{A} \gg \mathcal{B}$, $\mathcal{X} \gg \mathcal{A} \gg \mathcal{B} \gg \mathcal{Y}$. By Proposition III.1(5), $\mathcal{X} \gg \mathcal{Y}$.

(4) Since $\mathcal{X} \Subset \mathcal{A}, \mathcal{Y} \Subset \mathcal{B}$, by Definition III.1(3), $\mathcal{A} \gg \mathcal{X} \wedge \mathcal{Y} \gg \mathcal{B}$. Also, since $\mathcal{X} \gg \mathcal{Y}$, $\mathcal{A} \gg \mathcal{X} \gg \mathcal{Y} \gg \mathcal{B}$. By Proposition III.1(5), $\mathcal{A} \gg \mathcal{B}$. □

Theorem III.1 states that (1) if one ambient is a child of another, then their children are connected together, (2) if two parent ambients are adjacent to each other, then their children are connected together, (3) if two parent ambients are connected together, their children are all connected together, and (4) if there exist child ambients of two ambients are connected together, then such two ambients are connected together.

IV. Communication model

At a high abstract level, the exchanged information on communication is generally called *messages*. Different communication mechanisms and media form different types of communication such as asynchronous communication, synchronous communication and broadcasting communication. Synchronous communication requires that the sender should wait until the receiver is ready for the message exchange, and then they synchronize by executing the sending and the receiving activity simultaneously. Thus, since a message is directly sent to the receiver by a sender under synchronous communication, we may consider that a message directly moves from the sender into the receiver. By contrast, asynchronous communication requires specific communication media (queues or channels) which store messages. In this setting, a message leaves the sender, and then enters a communication medium. If the receiver needs the message, then the message exits the communication medium and enters the receiver. The broadcasting communication model means that one message can be cloned and transmitted from one sender to multiple receivers.

In a mobile system, mobile objects and ambients may communicate with each other. They are generally composed together by the communication devices (media) and these communication media are also ambients. Messages are mobile objects, and the senders and receivers of messages are mobile objects or ambients in a mobile system. Moreover, there exist many communication media, for example, any communication node on any network. We sometimes need to model all the communication media in order to model and reason about networks and their protocols.

Definition IV.1 A communication model is a tuple $\mathbb{CM} = \langle \mathbf{Mm}, \mathbf{Am}, \mathbf{Em}, \mathbf{Rd} \rangle$ where

•$\mathbf{Mm} \subseteq \mathbf{M}$ is the set of mobile messages,

•$\mathbf{Am} \subseteq \mathbf{A}$ is the set of ambients participating in the movement of messages,

•$\mathbf{Em} \subseteq \{e \mid m \in \mathbf{Mm}, A \in \mathbf{Am}, e = ex_A^m \vee e = en_A^m\}$ is the set of movement events of messages, and

•$\mathbf{Rd}$ is the dependency relation on the set $\mathbf{Em}$ such that for all $m \in \mathbf{Mm}$,
(1) if $\exists A, B \in \mathbf{Am} : A \Subset B$ and m moves between A and B, then $ex_A^m, en_B^m \in \mathbf{Em}$,
(2) if $\exists A, B \in \mathbf{Am} : (A \Rrightarrow B) \vee (\exists C : A \Subset C \wedge B \Subset C)$ and m moves from A to B, then $ex_A^m, en_B^m \in \mathbf{Em} \wedge ex_A^m \to en_B^m \in \mathbf{Rd}$,
(3) if $\exists A, B, C \in \mathbf{Am} : A \Subset B \Subset C$ and m moves between A and C, then $ex_A^m, ex_B^m, en_A^m, en_B^m \in \mathbf{Em} \wedge (ex_A^m \to ex_B^m, en_B^m \to en_A^m \in \mathbf{Rd})$, and
(4) if $\exists A \in \mathbf{Am}$: m passes through A, then $en_A^m \to ex_A^m \in \mathbf{Rd}$.

The communication model considers not only message exchanges between general senders and receivers, but also can handle message migration from a parent ambient to a child, or vice versa.

Proposition IV.1 Let $\mathbb{CM} = \langle \mathbf{Mm}, \mathbf{Am}, \mathbf{Em}, \mathbf{Rd} \rangle$ be a communication model. If $\exists A, B \in \mathbf{Am} : A \gg B$, then there exists a message $m \in \mathbf{Mm}$ such that m moves from A to B.

Next, we discuss the relationship of synchronous communication, asynchronous communication and broadcasting communication.

Definition IV.2 Let $\mathbb{CM} = \langle \mathbf{Mm}, \mathbf{Am}, \mathbf{Em}, \mathbf{Rd} \rangle$ be a communication model.

(1) $\mathbb{CM}$ is said to be *synchronous* if $\forall (e_1, e_2) \in \mathbf{Rd}, \forall e, e' \in \mathbf{Em} : (e, e_1) \notin \mathbf{Rd} \wedge (e_2, e') \notin \mathbf{Rd}$.

(2) $\mathbb{CM}$ is said to be *asynchronous* if $\forall (e_1, e_2) \in \mathbf{Rd}, \exists e, e' \in \mathbf{Em} : (e, e_1) \in \mathbf{Rd} \vee (e_2, e') \in \mathbf{Rd}$.

(3) $\mathbb{CM}$ is said to be *broadcasting* if $\exists e \in \mathbf{Em} : |e^\bullet| > 1$ where $e^\bullet = \{e' \in \mathbf{Em} \mid (e, e') \in \mathbf{Rd}\}$.

In a synchronous communication model, messages only move between two ambients and there does not exist an ambient between sender and receiver ambients. In a asynchronous communication model messages may move from sender ambients to receiver ambients and pass through the ambients between senders and receivers. A broadcasting communication model means that there exist multiple message movement events depending on the same movement events.

Proposition IV.2 If a communication model is broadcasting, it is synchronous or asynchronous.

This propostion shows that broadcast communication is synchronous or asynchronous communication.

V. Dependency structure

An event is an occurrence of an activity or action. If an event occurs, such an event is said to be *available*; otherwise it is *unavailable*. A dependency structure uses an *event set* (a set of events) as a basic element. If all events in an event set are *available*, such an event set is said to be *available*; otherwise it is said to be *unavailable*. For convenience, we first give some notations. Given a set $\mathcal{E}$, $|\mathcal{E}|$ and $P'(\mathcal{E})$ respectively denote the size and the power set of $\mathcal{E}$, and $P(\mathcal{E})$=$P'(\mathcal{E}) \setminus \{\emptyset\}$. **E** denotes the set of events.

Definition V.1 A **dependency structure** ($\mathcal{DS}$) is a tuple $\langle \mathcal{E}, \mathbb{I}, \mathbb{T}, \mathbb{S}, \mathbb{C}, \mathbb{P}, \mathbb{F} \rangle$ with
– $\mathcal{E} \subseteq \mathbf{E}$, a finite set of events,
– $\mathbb{I} \subseteq P'(\mathcal{E})$, the set of initially available event sets,
– $\mathbb{T} \subseteq P(\mathcal{E}) \times P(\mathcal{E})$, the (asymmetric) *transformation* relation,
– $\mathbb{S} \subseteq P(\mathcal{E})$, the *synchronism* relation such that $\forall A \in \mathbb{S} : |A| > 1$,
– $\mathbb{C} \subseteq P(\mathcal{E})$, the *choice* relation such that $\forall A \in \mathbb{C} : |A| > 1$,
–$\mathbb{P} \subseteq P(\mathcal{E}) \times P(\mathcal{E}) \setminus \mathbb{T}$, the (irreflexive and asymmetric) *priority* relation, and
– $\mathbb{F} \subseteq P'(\mathcal{E})$, the set of finally available event sets.

Here, for all $A, B \subseteq \mathcal{E}$, $(A, B) \in \mathbb{T}$ (resp. $\mathbb{P}$) is called a *transformation (resp. priority) dependency*, denoted as $A \rightarrow B$ (resp. $A \multimap B$), all read as *B depending on A*.

When the occurrences of some events completely depend on those of other events, the two groups of events form a (causal) transformation relationship. Transformation is a binary relation between event sets where the intuitive interpretation of a transformation $(A, B)((A, B) \in \mathbb{T})$ is that the availability of all events in B depends on the occurrences of all events in A. A set $A \in \mathbb{S}$ is called a *synchronism set*, and a set $B \in \mathbb{C}$ is called a *choice set*. Definition V.1 requires that any synchronism or choice set should have at least two events. The synchronism relation is not a binary relation. Any set $A \in \mathbb{S}$ means that all events in A synchronize with each other. Only if all events in A have occurred, the events that depend on them will occur. Any set $B \in \mathbb{C}$ means that all events in B are mutually exclusive, that is, if one event occurs, the others cannot occur.

Priorities only control the transformation relation and are not related to synchronism and choice. Actually, the synchronism and choice relations also control the transformation relation. The initially available event set means that the events in such an event set have been available before a system starts to run. The finally available event set means that when the execution of a system makes the events in such an event set available, the system or its subsystems stop running.

Theorem III.1 If $\mathbb{CM} = \langle \mathbf{Mm}, \mathbf{Am}, \mathbf{Em}, \mathbf{Rd} \rangle$ is a communication model of a communicating and mobile system, then there exists a dependency structure $\mathcal{DS} = \langle \mathcal{E}, \mathbb{I}, \mathbb{T}, \mathbb{S}, \mathbb{C}, \mathbb{P}, \mathbb{F} \rangle$ such that $\forall (e, e') \in \mathbf{Rd}, (\{e\}, \{e'\}) \in \mathbb{T}$.

Proof Let *CMS* be a communicating and mobile system. Let $\mathbb{CM} = \langle \mathbf{Mm}, \mathbf{Am}, \mathbf{Em}, \mathbf{Rd} \rangle$, $\mathcal{DS} = \langle \mathcal{E}, \mathbb{I}, \mathbb{T}, \mathbb{S}, \mathbb{C}, \mathbb{P}, \mathbb{F} \rangle$ be the communication model and the behaviour model of *CMS*, respectively. Then, since $\mathbb{CM}$ represent the communication behavior of *CMS*, by Definition IV.1, we can obtain the dependency relation **Rd** of message movement. Obviously, since the communication behavior of *CMS* is part of the whole behavior of *CMS*, and a dependency structure can model the whole behavior of *CMS* (including event dependency, synchronism, choice, priority and loop), the dependency structure can model the communication behavior of *CMS*. According to Definition IV.1, for all $(e, e') \in \mathbf{Rd}$, we can guarantee $(\{e\}, \{e'\}) \in \mathbb{T}$. □

This theorem shows that the dependency structure model contains the communication model of unifying mobility and communication, that is to say, a dependency structure can model not only behavior of a communicating and mobility system, but also represent mobility and communication of such a system in a unified way.

VI. Conclusion

We have discussed the relationship of ambients based on mobility and presented an event-based communication model. We have also shown that a dependency structure can not only unify synchronous, asynchronous and broadcasting communication, but also specify mobility and communication in a unified way. As a general event-based formal model, a dependency structure is easily used to model and reason about mobile cyber-physical systems.

ACKNOWLEDGMENTS

This work is supported by National Natural Science Foundation of China (No. 61772004) and the Natural Science Foundation of Fujian province (No. 2018J01777).

References

[1] Andrea Asperti and Nadia Busi. Mobile petri nets. *Mathematical Structures in Computer Science*, 19(06):1265–1278, 2009.
[2] Luca Cardelli and Andrew D Gordon. Mobile ambients. In *International Conference on Foundations of Software Science and Computation Structure*, pages 140–155. Springer, 1998.
[3] Florent Chevrou, Aurelie Hurault, and Philippe Queinnec. On the diversity of asynchronous communication. *Formal Aspects of Computing*, 28(5):847–879, 2016.
[4] Jian-Min Jiang, Huibiao Zhu, Qin Li, Yongxin Zhao, Lin Zhao, Shi Zhang, Ping Gong, and Zhong Hong. Analyzing event-based scheduling in concurrent reactive systems. *ACM Transactions on Embedded Computing Systems (TECS)*, 14(4):86, 2015.
[5] Jian-Min Jiang, Huibiao Zhu, Qin Li, Yongxin Zhao, Lin Zhao, Shi Zhang, Ping Gong, Zhong Hong, and Donghuo Chen. Event-based mobility modeling and analysis. *ACM Transactions on Cyber-Physical Systems*, 1(2):9:1–9:32, February 2017.
[6] Robin Milner. *A Calculus of Communicating Systems*. Springer-Verlag, New York, NY, 1982.
[7] Robin Milner, Joachim Parrow, and David Walker. A calculus of mobile processes. *Information and Computation*, 100(1):1–40, September 1992.
[8] Sebastian Nanz, Flemming Nielson, and Hanne Riis Nielson. Static analysis of topology-dependent broadcast networks. *Information and Computation*, 208(2):117–139, 2010.
[9] Mogens Nielsen, Gordon Plotkin, and Glynn Winskel. Petri nets, event structures and domains, part i. *Theoretical Computer Science*, 13(1):85–108, 1981.
[10] Glynn Winskel and Mogens Nielsen. *Models for Concurrency*. Handbook of Logic in Computer Science, Semantic Modelling. Oxford Science Publications, Oxford, 1995.
[11] Dianxiang Xu, Jianwen Yin, Yi Deng, and Junhua Ding. A formal architectural model for logical agent mobility. *IEEE Transactions on Software Engineering*, 29(1):31–45, 2003.