MOCAVI: An Efficient Causal Protocol for Cellular Networks

Eduardo Lopez Dominguez, Saul E. Pomares Hernandez and Gustavo Rodriguez Gomez

Computer Science Department
National Institute of Astrophysics, Optics and Electronics (INAOE),
Luis Enrique Erro #1, 72840, Tonantzintla, Puebla, Mexico

Summary
A cellular network consists of two main components: base stations (BS) and mobile hosts (MH). It is through base stations that mobile hosts can communicate. In order to reduce the causal overhead and the computational cost over mobile hosts, most of the existing protocols for cellular networks ensure causal order at BSs. However, these protocols introduce unnecessary inhibition of delivery of messages since the causal ordering is carried out according to the causal view of the BSs and not in absolute accordance with the causal view of the MHs. In this paper, we present *MOCAVI*, an efficient protocol that ensures causal ordering according to the causal view of mobile hosts, through which we avoid the unnecessary inhibition of message delivery while maintaining a low overhead and computational cost.

Key words:
Causal ordering, mobile causal view, cellular networks, immediate dependency relation.

1. Introduction

In recent years, cellular networks have experienced several exciting innovations and will continue to represent a rapidly growing sector in the near future. The evolution of cellular networks establishes a trend to use portable computing devices, such as smart phones and personal digital assistants (PDAs). In conjunction with wireless communication technologies, cellular networks enable users to access the internet at anytime and anywhere in the world. The goal is to provide users with access to desktop applications, applications specially suited for mobile users, and multimedia applications. However, cellular networks involve new characteristics and constraints, such as changeable physical network connections, limited processing and storage capabilities in mobile devices, as well as limited bandwidth on wireless communication channels.

Many protocols have been proposed for cellular networks in different research areas such as causally ordered message delivery [2-9], mutual exclusion [13], and checkpoint protocols [14]. In this paper, we consider the problem of causal order message delivery among mobile hosts in the context of group communication.

Some protocols [2-9] have been proposed to implement causal message ordering over cellular networks. In order to reduce computational costs and communication loads on mobile hosts, most of these protocols store relevant data structures in the base stations (BS), and they are executed by the BSs on behalf of the mobile hosts (MH). These methods give rise to two main problems. First, the causal order seen by the BSs differ from the causal order seen by the MHs. Secondly, they introduces unnecessary inhibition of message delivery. This unnecessary inhibition is due to the serialization of messages at the BSs level. The serialization of messages appears since a base station is unable to detect mutual concurrency between messages occurring at different MHs in its cell.

In this paper, we propose a new protocol called *MOCAVI*, which ensures the causal ordering according to the causal view that the mobile hosts perceive during the system execution, avoiding unnecessary inhibition of message delivery while maintaining a low overhead and computational cost. To achieve this, we differentiate two communication levels according to the connection type (wired and wireless): intra-base communication level and inter-base communication level. At the intra-base communication level (wireless connection) we only send as causal overhead, between a BS and the MHs attached to it, a vector of bits Φ of size n ($\Phi(n)$, where n is the number of participants in the group). At the inter-base communication level (wired connection) we only send as causal overhead, between BSs, information about messages that are related through immediate dependence [11]. As we will show, the vector $\Phi(n)$ used at the intra-base communication level is sufficient to ensure causal message ordering as seen by the MHs of the system.

The rest of this paper is organized as follows. Section 2 presents the system model, background, and definitions. A description of the proposed protocol in this work is provided in Section 3. Next, in Section 4 we compare our protocol with other works in two aspects: message overhead and unnecessary inhibition in message delivery. Finally, conclusions are presented in Section 5.

Manuscript received January 5, 2007
Manuscript revised January 20, 2007

2. Preliminaries

2.1 The System Model

A cellular network consists of two kinds of entities: mobile support stations or base stations and mobile hosts. A base station (BS) has the necessary infrastructure to support and communicate with mobile hosts. The BS communicates with mobile hosts through wireless communication channels. The geographic area covered by a base station is called *cell,* figure 1. A mobile host (MH) is a host that can move while retaining its network connection. At any given time, a MH is assumed to be within a cell of at most one BS, which is called its local BS. A MH can communicate with other MHs and BSs only through its local BS.

We assume in this paper that the wireless communication channels between MHs and BSs are FIFO and reliable. The base stations are connected among themselves using wired channels. The BSs and the wired channels constitute the static network. We assume that the wired channels are reliable and take an arbitrary but finite amount of time to deliver messages. Due to system asynchrony and unpredictable communication delays, the sent messages on cellular network can arrive in a different order as they were sent.

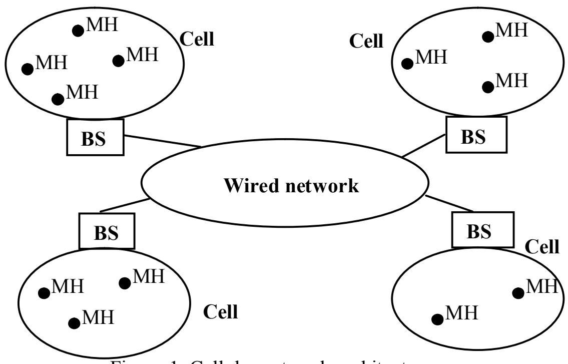

Figure 1. Cellular network architecture

In a cellular network, a mobile host can move from one BS to another. In this case, a hand-procedure (not presented in this paper) is performed to transfer the communication responsibilities of MH to the new BS.

With respect to the logical specification in our work, the application under consideration is composed of a set of mobile hosts $P = \{i, j, \ldots n\}$ organized into a group that communicates by reliable broadcast asynchronous message passing. We consider a finite set of messages M, where each message $m \in M$ is identified by a tuple $m = (p,t)$, where $p \in P$ is the sender of m, denoted by $Src(m)$, and t is the sequential ordered logical clock for messages of p when m is broadcasted. The set of destinations of a message m is always P.

2.2 Background and Definitions

Causal ordering delivery is based on the causal precedence relation defined by Lamport [10]. The happened-before relation establishes, over a set of events, possible precedence dependencies without using physical clocks. It is a partial order defined as follows:

Definition 1. The causal relation "→" is the least partial order relation on a set of events satisfying the following properties:

- If a and b are events belonging to the same process and a was originated before b, then $a \rightarrow b$.
- If a is the send message of a process and b is the reception of the same message in another process, then $a \rightarrow b$.
- If $a \rightarrow b$ and $b \rightarrow c$, then $a \rightarrow c$.

By using Definition 1 we can cay that a pair of events are *concurrent* related "a || b" only if $\neg(a \rightarrow b \vee b \rightarrow a)$.

The precedence relation on messages denoted by $m \rightarrow m'$ is induced by the precedence relation on events, and is defined by:

$$m \rightarrow m' \Leftrightarrow send(m) \rightarrow send(m')$$

The Immediate Dependency Relation. The Immediate Dependency Relation (IDR) formalized in [11] is the propagation threshold of the control information regarding the messages sent in the causal past that must be transmitted to ensure a causal delivery. We denote it by ↓, and its formal definition is the following:

Definition 2. Immediate Dependency Relation "↓" (IDR):

$$m \downarrow m' \Leftrightarrow [\,(m \rightarrow m') \wedge \forall\, m'' \in M, \neg(m \rightarrow m'' \rightarrow m')]$$

Thus, a message m directly precedes a message m', iff no other message m'' belonging to M exists (M is the set of messages of the system), such that m'' belongs at the same time to the causal future of m and to the causal past of m'.

This relationship is important because if the delivery of messages respects the order of the diffusion for all pairs of messages in an IDR, then the delivery respects the causal order for all messages. This property is formally defined for the broadcast case as follows:

Property 1:

$$\text{If } \forall\, m,m' \in M, m\downarrow m' \Rightarrow \forall\, p \in P: delivery(p,m) \rightarrow deliver(p,m') \text{ then}$$
$$m \rightarrow m' \Rightarrow \forall p \in P : delivery(p,,m) \rightarrow delivery(p,m')$$

Causal information that includes the messages immediately preceding a given message is sufficient to ensure a causal delivery of such message.

Transmission time of message *m* ($\delta(m)$)

We analyze the necessary time for transmitting a message by a mobile host p_i to another mobile host p_j. Both mobile hosts within the cell are covered by the base station BS_r. First, the mobile host p_i sends a message m to its local base station BS_r over a wireless communication channel. The local base station BS_r receives the message m at a time t. Before the transmission of message m to mobile host p_j, the base station BS_r attached control information to message m in a time period called *processing_timeBS*$_r(m)$. Finally, BS_r sends the message m at time t' to mobile host p_j through a wireless communication channel, figure 2.

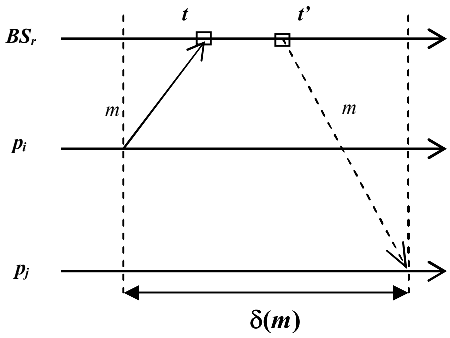

Figure 2. Transmission time of a message into its local cell.

Thus, we can divide the necessary time to transmit a message m into three parts.

uplink_time(m): transmission time in wireless communication channels for a message m from the mobile hosts to its base station.

processing_time(m): period of time necessary used by a base station to attach control information to the received message.

downlink_time(m): transmission time in wireless communication channels for a message m from the base station to a mobile host.

Therefore, the total transmission time of a message m is equal to:

$$\delta(m)= uplink\_time(m) + processing\_time(m) + downlink\_time(m) + \varepsilon$$

In our case, we consider a possible error in the variation time represented by the variable ε.

3. The Causal View Protocol

Protocol overview

From the point of view of the physical architecture, we consider two communication levels in a mobile system: Intra-base and Inter-base. The Intra-base communication level is formed by a wireless network integrated by a base station and mobile hosts. In this level, the base station provides communication services to mobile hosts in its cell. The inter-base communication level is formed by a wired network constituted by several base stations.

On the other hand, from a logical point of view, we consider only one communication level. This communication level is formed by the view that mobile hosts have during the system execution. In our work, we propose a protocol that carries out a causal ordering according to the causal view that the mobile hosts perceives during the system execution. In our case, the base stations are in charge of carrying out the causal delivery of messages according to the order in which messages were observed by the mobile hosts, thus avoiding the serialization of the messages implicitly established by the causal view of the base station.

The mobile host uses a bit vector Φ_i, to establish the immediate dependency relation among messages. The number of bit vectors at a mobile host is equal to the number of base stations involved in system. In our protocol, each bit vector logically represents the mobile host of a cell. The size of Φ_i is equal to the number of mobile hosts that are within the cell of the base station i. A bit of Φ_i is equal to 1 if a message sent by a mobile host of the base station i has an immediate dependency relation (definition 2) with the next message to send.

The bit vectors are the only control information attached to messages sent in the wireless communication channel which determine the immediate dependency relation between messages.

The base stations keep control information of messages causally sent in the cellular network. Through this control information and the bit vectors attached to messages by the MHs, the base stations can determine precedence immediate dependencies between messages sent by MHs on different BSs.

3.1 Data Structures

Each base station has the following data structures:

- $VT(p)$ is the vector time. For each mobile host p there is an element $VT(p)[j]$, where j is a mobile host identifier. The size of VT is equal to the number of mobile hosts in the group. $VT(p)$ contains the local view that a mobile host p has of the elements of the system. In particular, element $VT(p)[j]$ represents the greatest element number of the identifier j and 'seen' in causal order by p. It is through the $VT(p)$ structure that we are able to guarantee the causal delivery of elements.

- A structure to keep control information $CI(p)$ of messages sent. The structure $CI(p)$ is a set of entries (k, t, d, Δ, ip). Each entry in $CI(p)$ denotes a message that is not ensured by participant p of being delivered in a causal order. The entry (k, t, d, Δ, ip) represents a diffusion by participant k at a logical local timeclock $t = VT(p)[k]$, where d is equal to the number of messages sent by the base station, Δ is a variable of type *time*, and ip is a boolean variable, which is used to indicate if the message represented by the entry (k, t, d, Δ, ip) immediately precede a received message.

- An integer counter *sent_messages* that is incremented each time a message is sent by the base station.

- The structure of a message m'' sent in the wired communication channels is a quintuplet $m'' \equiv (i, t, BS_k, data, H(m))$, where i is the mobile host identifier, t is the message identifier, BS_k is the base station identifier, *data* is the information in question, and $H(m)$ is composed of a set of elements $h_r(m)$, $H(m)=\{h_r(m), h_s(m), \ldots h_w(m)\}$, where w is equal to the number of base stations in the group. Each element $h_r(m)$ of $H(m)$ is formed by a set of entries (k, t), which represent messages that that have an IDR with m''. Structure $H(m)$ is created at the moment of diffusion of a message by a base station.

Data structures maintained at the mobile host are:

- An integer counters *received_messages,* which is incremented each time that a message is received by the mobile host p.

- A bit vector Φ_r, for each base station r in the communication group. The size of Φ_r is equal to the number of mobile hosts that are within the cell of the base station r. The bits put to 1 in Φ_r indicate the immediate dependency relation that the transmitted message has with the messages sent by mobile hosts in the BS_r. For example, the bit of $\Phi_r(p)[j] = 1$, indicates that a sent message by mobile host j in the cell covered by the base station r immediately precedes the next message to send by mobile host p.

- The structure of a message m sent in the wireless communication channels has the following form: $m \equiv (i, t, data, received\_messages, \{\Phi_r(p), \Phi_s(p) \ldots \Phi_w(p)\}$, where i is the mobile host identifier, t is the message identifier, *received_messages* is a counter that is incremented each time a message is received by the mobile host i, and $\Phi_r(p), \Phi_s(p) .. \Phi_w(p)$ are bit vectors that logically represent to the mobile host of the cells covered by the base stations r, s, and w, respectively.

In our work, a message transmitted by a mobile host is denoted by m_i. When a message m_i is sent by the base station to local mobile hosts, the message is denoted by m_i', and a message m_i sent by the base station to another base station is denoted by m_i''.

3.2 Specification of the protocol

Table 1. Initially

/* Data structures maintained at base station p */
$VT(p)[j] = 0\ \forall\, j{:}1\ldots n.$
$CI(p)[j] \leftarrow \emptyset\ \ \forall\, j{:}1\ldots w$
IDR_bits(p) /* vector of bits of size n */
IDR_bits(p)[j] = 0 $\forall\, j$: $1\ldots n$.
/* Data structures maintained at mobile host */
$\Phi_r(p)[j] = 0;\ j = 1\ldots n\ ;\ r = 1\ldots w$
received_messages = 0 /* account of received messages */
let *update_CI*($H(m)$) { // update the control information at BS
For each $h_r(m) \in H(m)$
$\forall\,(x,y) \in h_r(m)$ if ($x \neq i$) then
if $\exists\,(k,t,d,\Delta,ip) \in CI(p) \mid k = x$ and $t \leq y$ and $\Delta = 0$ then

14.	$(k, t, d, \Delta, ip) \leftarrow (k, t, d, \Delta = current\_time, ip)$
15.	Endif
16.	Endfor
17.	For each $(k,t,d,\Delta,ip) \in CI(p)$ /* Delete some entries of CI */
18.	if ($\Delta + \delta(m) \leq current\_time$) then
19.	$CI(p) \leftarrow CI(p) / (k,t,d,\Delta,ip)$
20.	Endfor
21.	}

Table 2. Diffusion of message *m* by mobile host *p*

1.	/* building of message *m* */
2.	$id\_message = id\_message + 1$
3.	/* $m \equiv (i = id\_host,\ t = id\_message,\ received\_messages,\ data,\ \{\Phi_r(p), \Phi_s(p) \ldots \Phi_w(p)\})$ */
4.	**Difussion:** send(m) /*sent of message m to local base station p*/

Table 3 Reception of message m at base station $BS_k = \{p_i, p_j \ldots p_n\}$

1.	/* $m \equiv (i, t, received\_messages, data, \{\Phi_r(p), \Phi_s(p) \ldots \Phi_w(p)\})$ */
2.	if $i \in BS_k$ then
3.	if not ($t = VT(p)[i] + 1$) then
4.	wait
5.	Else
6.	delevery(m)
7.	$VT(p)[i] = VT(p)[i] + 1$
8.	/* bit vectors to atach the sent message m to local mobile hosts*/
9.	For each $\Phi_r(p) \in m \mid r = s,t,\ldots\ldots w$
10.	For all $\Phi_r(p)[j] == 1;\ j = 1,2..n$
11.	if $\exists\ (k,t,d,\Delta,predecesor) \in CI(p) \mid k = j$ and $d \leq received\_messages$ and $predecesor = false$ then
12.	$IDR\_Bits_r(p)[j] = \Phi_r(p)[j]$
13.	$(k,t,d,\Delta, predecesor) \leftarrow (k,t,d,\Delta,predecesor=true)$
14.	else
15.	$IDR\_Bits_r(p)[j] = 0$
16.	Endif
17.	Endfor
18.	Endfor
19.	$m' \equiv (i, t' = sent\_messages+1, data, \{IDR\_Bits_r(p), IDR\_Bits_s(p) \ldots IDR\_Bits_w(p)\})$
20.	**Difussion :** send(m') /* sent of message m' to local mobile hosts */
21.	For each $\Phi_r(p) \in m \mid r = s,t,\ldots\ldots w$ // forming $H(m)$
22.	For all $\Phi_r(p)[j] == 1;\ j = 1,2..n$
23.	$\forall\ (k,t,d,\Delta,predecesor) \in CI(p)$
24.	if (($k = j$) and ($d \leq received\_messages$))then
25.	$h_r(m) \leftarrow h_r(m) \cup (k, t)$
26.	endif
27.	Endfor
28.	Endfor
29.	$m'' \equiv (i, t, BS_k, datos, H(m))$
30.	**Difussion :** send(m'') /* sent of message m'' to other Mobile hosts*/
31.	Endif /* End of if, line 3 */
32.	Else /* $i \notin BS_k$ */
33.	/* $m'' \equiv (i, t, BS_k, datos, H(m))$ */
34.	if not ($t = VT(p)[i] + 1$ and $\forall\ (s,x) \in H(m)$: $x \leq VT(p)[s]$) then
35.	Wait
36.	Else
37.	delivery(m)
38.	$VT(p)[i] = VT(p)[i] + 1$
39.	/* message m to sent by BS to local mobile hosts */
40.	For each $h_r(m) \in H(m)$
41.	$\forall\ (x,y) \in h_r(m)$
42.	if $\exists (k,t,d, \Delta, predecesor) \in CI(p) \mid x = k$ and $y = t$ and $predecessor = false$ then
43.	$IDR\_Bits_l(p)[k] = 1$
44.	$(k,t,d,\Delta, predecesor) \leftarrow (k,t,d,\Delta, predecesor=true)$
45.	Else
46.	$IDR\_Bits_l(p)[k] = 0$
47.	endif
48.	Endfor
49.	$m' \equiv (i, t' = sent\_messages+1, BS, datos, \{IDR\_Bits_r(p), IDR\_Bits_s(p), \ldots IDR\_Bits_w(p)\})$
50.	**Difussion :** send(m) /* sent of message m to local mobile hosts */
51.	Endif
52.	Endif
53.	$sent\_messages = sent\_messages + 1$
54.	$update\_CI(H(m))$
55.	$CI(p) \leftarrow CI(p) \cup \{ (i,t, sent\_messages, \Delta = 0, predecesor = fase)\}$

Table 4. Reception of message m by mobile host p.

1.	/* $m \equiv (i, t', BS_k, datos, \{IDR\_Bits_l(p), IDR\_Bits_{l+1}(p), \ldots\ldots IDR\_Bits_m(p)\})$ */
2.	If not ($t' = received\_messages + 1$) then
3.	Wait
4.	Else
5.	Delivery(m')
6.	$received\_messages = received\_messages + 1$
7.	For each $IDR\_Bits_l(p) \in m \mid l = s,u,\ldots\ldots w$
8.	For all $IDR\_Bits_l(p)[j] == 1;\ j = 1,2\ldots n : j \neq i$
9.	if ($\Phi_l(p)[j] == 1$) then
10.	$\Phi_l(p)[j] = 0$
11.	Endif
12.	Endfor
13.	Endfor
14.	$\Phi_k(p)[i] = 1$
15.	Endif

3.3 Scenario Example

Consider the group of mobile hosts g = $\{h_1, h_2, h_3, h_4\}$ and the diffusion of message m_3 to other mobile hosts, where h_1 and h_2 are present in the cell served by BS_1, and h_3, and h_4 are present in the cell served by BS_2, see figure 3. Before the delivery of m_3 to BS_1, $CI(BS_1)=(h_1,1,2,0,false)$, $CI(BS_2)=(h_1,1,2,0,false)$, $VT(BS_1)=(1,0,1,0)$ and $VT(BS_2)=(1,0,1,0)$. These values are deduced from our *MOCAVI* protocol shown in tables 1-4. In figure 3, we show transmitted messages during the execution of our *MOCAVI* protocol.

Diffusion of message m_3 by h_2 at BS_1

Lines 2-4, **table 2**. The value of *id_message* is incremented by one. The variable *id_message* identifies the message sent by h_2. Diffusion of message $m_3=(h_2,1,2,data,10,00)$, send(m_3).

Delivery of message m_3 to its local BS_1

Lines 2-31, table 3. Each time that a *BS* receives a message from a local mobile host. The *BS* only verifies that the received message satisfies the FIFO delivery condition. In this case, message $m_3=(h_2,1,2,data,\Phi_1=10, \Phi_2=00)$ satisfies the delivery condition (**Line 3)** because $t=1$ and $VT(BS_1)[h_2] = 0$. Thus, the condition $t=VT(BS_1)[h_2]+1$ is satisfied. Now, message m_3 is delivered and the vector is increased by one in $VT(BS_1)[h_2]$, resulting in $VT(BS_1)=(1,1,1,0)$, **lines 6-7**. The bit put to 1 in Φ_1 indicates that a message sent by h_1 served by BS_1 immediately precedes m_3. Later on, the BS_1 must send the message m_3 to its local mobile hosts and to the others mobile hosts in the base station BS_2, see figure 3.

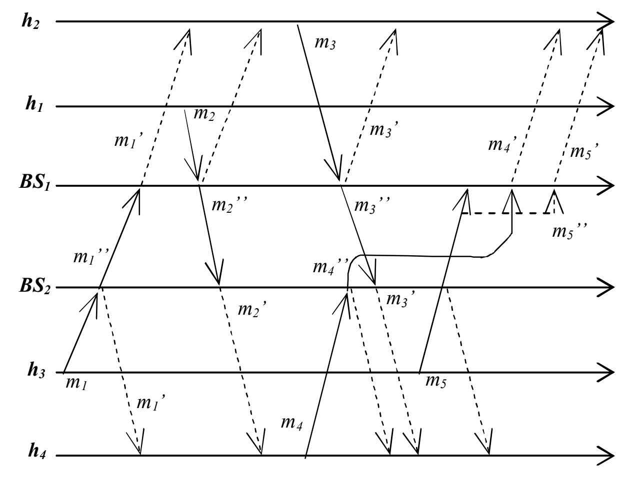

Figure 3. A scenario of group communication formed by four mobile hosts.

Lines 9-20. Sending by BS_1 of message m_3 to the local mobile hosts. BS_1 builds the bit vectors attached to m_3'. In this case, the message to send is $m_3' \equiv (i=h_2, t'=3, data, IDR\_Bits_1=10, IDR\_Bits_2=00)$.

Diffusion of message m_3'' by BS_1 to base station BS_2

Lines 21-30. Sending by BS_1 of message m_3 to base station BS_2. BS_1 identifies the messages that have an immediate dependency relation with m_3 through of the bit vectors Φ_r received attached to it and the control information stored in the structure $CI(BS_1)$. In Φ_1 the bit equals to 1 indicates that h_1 has sent a message that immediately precedes to m_3. In order to identify the message that immediately precedes to m_3, BS_1 verifies if an entry (k,t,d) of mobile host h_1 already exists in $CI(BS_1)$ with a d minor or equal to the variable *received_messages* of m_3. In this case, there is an entry storage at $CI(BS_1)$ about h_1 with $d=2$ and the condition $d \leq received\_messages$ is satisfied. This entry is the control information of message m_2. Thus, the only control information attached to m_3 in order to ensure a causal order relates to m_2, which is the only message in an immediate dependency relationship with m_3, see figure 3. Therefore, the message to send by BS_1 to BS_2 is $m_3''=(h_2,1, BS_1,data, (h_1,1))$, **Line 29**.

Diffusion of message m_4 by h_4 at BS_2

Lines 2-4, **table 2**. The value of *id_message* is incremented by one. The variable *id_message* identifies the message sent by h_4. Diffusion of message $m_4=(h_4,1,2,data,10,00)$, send(m_4).

Delivery of message m_4 to its local BS_2

Lines 2-31, table 3. The BS_2 verifies that only the received message satisfies the FIFO delivery condition. In this case, message $m_4=(h_4,1,2,data,10,00)$ satisfies the delivery condition (**Line 3).** Now message m_4 is delivered and the vector is increased by one in $VT(BS_2)[h_4]$, resulting in $VT(BS_2)=(1,1,1,1)$, **Line 6-7**. Later on, the BS_2 must send the message m_4 to its local mobile hosts and to base station BS_1.

Lines 9-20. Sending by BS_2 of message m_4 to the local mobile hosts. In this case, the message to send is $m_4'=(h_4,3,data, IDR\_Bits_1=10, IDR\_Bits_2=00)$.

Diffusion of message m_4'' by BS_2 to base station BS_1

Lines 21-30. In Φ_1 the bit equal to 1 indicates that h_1 has sent a message that immediately precedes to m_4. In order to identify the message that immediately precedes to m_4, BS_1 verifies if an entry (k,t,d) of mobile host h_1 already exists in $CI(BS_1)$ with a d minor or equal to the variable *received_messages* of m_4. In this case, there is an entry storage at $CI(BS_2)$ about h_1 with d =2, and $d \leq$ *received_messages* is satisfied. Thus, the only control information attached to m_4 in order to ensure a causal order relates to m_2, which is the only message that has an immediate dependency relationship with m_3, see figure 3. **Line 29,** Therefore, the message to send by BS_2 to BS_1 is $m_4''=(h_4,1,\ BS_2,data,\ (h_1,1))$. **Line 30,** Diffusion of m_4'', $send(m_4'')$.

Delivery of message m_3'' to BS_2

Lines 33-55, table 3. Each time a *BS* receives a message from a mobile host within cell different to cover by it. The *BS* verifies that the received message satisfies the FIFO and causal delivery condition. In this case, message $m_3''=(h_2,1,\ BS_1,data,\ (h_1,1))$ satisfies both conditions (**Line 34)** because t=1 and $VT(BS_2)[\ h_2]$ = 0, and the conditions $t=VT(BS_1)[h_2]+1$ and $\forall\ (s,x)\in H(m):x\leq VT(p)[s]$ are satisfied. Now message m_3'' is delivered and the vector is increased by one in $VT(BS_2)[h_2]$, resulting in $VT(BS_2)$=(1,1,1,1). Later on (lines 40-50), the BS_2 must send message m_3'' to its local mobile hosts. The message to send by BS_2 to local mobile hosts is $m_3'=(h_2,4,\ BS_1,data,$ 00,00), **Line 49**.

Delivery of message m_3' to mobile host h_3

Lines 1-10, table 4. Each time that a mobile host receives a message *m,* the *MH* only verifies that the received message satisfies the FIFO delivery condition. In this case, message $m_3'=(h_2,4,\ BS_1,data,$ 00,00) satisfies the delivery condition (**Line 3)** because t'=4 and t'= 3 + 1. Later on, h_3 updates its bit vectors, **Line 2-8**. After updating the data structures at h_3, the bit vectors are Φ_1=01, and Φ_2=01. In the bit vectors Φ_1 and Φ_2, the bits equals to 1 indicate that last messages received by h_3 were sent by h_2 and h_4, respectively.

Diffusion of message m_5 by h_3 at BS_2

Lines 2-4, **table 2**. The value of *id_message* is incremented by one, *id_message*=2. **Line 4,** Diffusion of message $m_5=(h_3,2,4,data,$01,01), send(m_5).

Delivery of message m_5 to its local BS_2

Lines 2-31, table 3. In this case, message $m_5=(h_3,2,4,data,$ Φ_1=01,Φ_2=01) satisfies the FIFO delivery condition (**Line 3).** Now message m_5 is delivered, and the vector is increased by one in $VT(BS_2)[h_3]$, resulting in $VT(BS_2)$=(1,1,2,1). Later on, the BS_2 must send message m_5 to its local mobile hosts and to base station BS_1. **Lines 9-20.** Sending by BS_2 of message m_5 to the local mobile hosts. The message to send is $m_5'=(h_3,5,\ BS_2,data,$01,01). **Line 20**, Diffusion of message m_5', send(m_5').

Diffusion of message m_5'' by BS_2 to base station BS_1

Lines 21-30. In Φ_1 and Φ_2 the bits set to 1 indicate that the messages sent m_3 and m_4 by h_2 and h_4, respectively, immediately precede to m_5. Thus, the only control information attached to m_5 in order to ensure a causal order relates to m_3 and m_4, which are the only messages that have an immediate dependency relation with m_5, see figure 3. **Line 29,** Therefore, the message to send by BS_2 to BS_1 is $m_5''=(h_3,2,BS_2,data,\{(h_2,1),(h_4,1)\})$. **Line 30,** Diffusion of m_5'', $send(m_5'')$.

Delivery of message m_5'' to BS_1

Lines 33-55, table 3. The BS_1 verifies that the received message satisfies the FIFO and causal delivery condition, **Line 34**. In this case, message $m_5''=(h_3,2,BS_2,data,\ \{(h_2,1),(h_4,1)\})$ satisfies only the FIFO delivery condition (**Line 34)** because t=2 and $VT(BS_1)[h_3]$=1 and the condition $t=VT(BS_1)[h_3]+1$ is satisfied. Because the message m_4 hasn't been received by mobile hosts within the cell covered by BS_1, the causal delivery condition (**Line 34**, $1\leq VT(BS_1)[h_4]=0$) is not satisfied; therefore, message m_5'' cannot be delivered causally and it is delayed (**Line 35**).

Delivery of message m_4'' to BS_1

Lines 33-55, table 3. The BS_1 verifies that the received message satisfies the FIFO and causal delivery condition. In this case, message $m_4''=(h_4,1,\ BS_2,data,\ (h_1,1))$ satisfies both conditions (**Line 34)**. Message m_4'' is delivered, and the vector is increased by one in $VT(BS_1)[h_4]$, resulting in $VT(BS_1)$=(1,1,1,1). Later on, the BS_1 must send message m_4'' to its local mobile hosts. The message to send by BS_1 to local mobile hosts is $m_4'=(h_4,4,BS_2,data,$00,00), **Line 49**.

Delivery of message m_4' to mobile host h_1

Lines 1-10, table 4. The mobile host h_1 updates its bit vectors with the reception of message $m_4'=(h_4,1,BS_2,data,$ 00,00), **Lines 2-8**. The bit vectors after updating the data structures at h_1 are Φ_1=01, and Φ_2=01. **Line 10**, the variable *received_messages* is incremented by one, *received_messages*= 4.

Delivery of message m_4' to mobile host h_2

Lines 1-10, table 4. The bit vectors after updating the data structures at h_2 are Φ_1=01 and Φ_2=01. **Line 10**, The account of received messages is incremented by one, *received_messages*= 4.

Delivery of message m_5'' in causal ordering at BS_1

Lines 33-55, table 3. Finally, the BS_1 after the causal delivery of message m_4'' verifies if message m_5'' satisfies the causal delivery condition, **Line 34**. Now, message $m_5'' = (h_3, 2, BS_2, data, \{(h_2,1), (h_4,1)\})$ satisfies the causal delivery condition (**Line 34)** because message m_4 has been received by mobile hosts within the cell covered by BS_1. The causal delivery condition (**Line 34**, $1 \leq VT(BS_1)[h_4]=1$) is satisfied, and therefore message m_5'' can be delivered causally (**Line 37**). Later on, the BS_2 must send message m_5'' to its local mobile hosts. The message to send by BS_1 to local mobile hosts is $m_5' = (h_3, 5, data, 01, 01)$, **Line 49**.

4. Comparisons

We compare our algorithm versus the related work in two aspects: message overhead and unnecessary inhibition in the message delivery (see Table 5).

4.1 Message overhead

In order to ensure the causal delivery of messages, all algorithms need to add control information to the header of each message. This control information is overhead that will increase the bandwidth used. However, the amount of overhead added to the messages by each one of these algorithms is considerably different.

Table 5. Comparison between causal algorithms for cellular networks (where n = number of MHs and m=number of BSs)

Protocol	Overhead (wireless network)	Overhead (wired network)	Communication type	Unnecessary delivery inhibition
AV-1	0	$O(n^2)$	Unicast	No
AV-2	0	$O(m^2)$	Unicast	Yes
AV-3	0	$O(m^2 * k^2)$	Unicast	Yes
YHH	0	$O(n * m)$	Unicast	Yes
Mobi Causal	$O(\sum_{i=1}^{m} li)$	$O(\sum_{i=1}^{m} li)$	Unicast	NO
LH	0	$O(m)$	Multicast	Yes
PRS	$O(n)$	$O(n)$	Group communication	No
KHC	0	$O(m)$	Multicast	Yes
MOCAVI	$\Phi(n)$	$O(s)$: $1 \leq s \leq n$	Group communication	No

Hence, the control information size attached to messages send over the wired network of the algorithms AV-2, AV-3 [2], YHH [3], and Mobi_Causal [9] is $O(m^2)$, $O(m^2 k^2)$, $O(n * m)$, and $O(\sum_{i=1}^{m} li)$ respectively, where k is a predetermined integer parameter, and l_i represents the number of messages sent by base station i. In these algorithms, each message is intended for a single destination site (mobile host), whereas algorithms LH [4], PRS [8], KHC [5], and our proposed protocol MOCAVI allows a message to be destined for n sites, see table 5.

The lowest message overhead, $O(m)$, is proposed in LH, and KHC [5]. However, these algorithms can unnecessarily delay the delivery of a message since they preserve the causal ordering at a base station level. In our proposal, the size of control information differs from the intra-base communication level and inter-base level. For the intra-base level, we send a constant overhead of n bits per message. And for inter-base level (communication among BSs) is given by the cardinality of $H(m)$, which can fluctuate in our case between 1 and n ($0 \leq |H(m)| \leq n$), where n is equal to the number of mobile hosts in the group. This is because $H(m)$ only has information about the most recent messages that immediately precede a message m. In the best case, dealing with the serial case, we note that the message overhead is $|H(m)|=1$, and in the case of concurrent messages, the worst case is $|H(m)|=n$. We notice that in our protocol, as for the minimal causal algorithm in [11], the likelihood that the worst case will occur approaches zero as the number of participants in the group grows. Compared with other works that are exclusively based on vectors clocks [2-9], our worst case denotes for them the constant overhead that must always be attached per message.

4.2 Unnecessary delivery inhibition

Before we compare the probability of unnecessary delivery inhibition among all protocols, we illustrate the phenomenon of unnecessary inhibition with an example. Consider the example shown in figure 4. In this scenario, the mobile host h_2 sends message m_1 to its local base station BS_2. After delivering m_1, the base station BS_2 sends message m_1 to base station BS_1. Another message m_2 is sent by h_3 to BS_2. After delivering m_2, the base station BS_2 sends message m_2 to BS_1. As we can see, m_2 has been sent before the delivery of m_1 at h_3, and therefore, messages m_1 and m_2 are concurrent.

In this case, message m_2 is received before m_1 at BS_1. Some protocols that carry out a causal ordering at a base station level can inhibit the delivery of m_2 until m_1 has been delivered to BS_1. This is because the messages are delivered according to the causal view of base station BS_2. For BS_2 message m_1 happened before m_2 ($m_1 \rightarrow m_2$). See figure 4. On the other hand, the protocols that perform a causal ordering according to the causal view of the mobile host avoid the phenomenon of unnecessary inhibition.

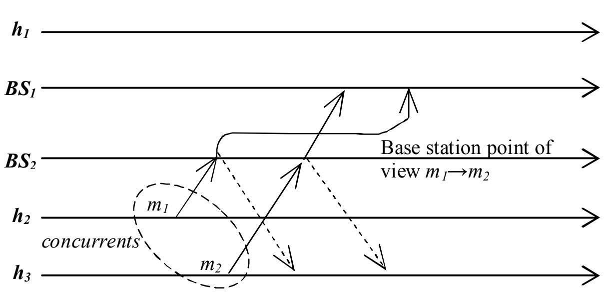

Figure 4. An example of unnecessary inhibition.

Algorithm AV-2 has the highest probability of unnecessary inhibition since it only maintains the causal ordering among base stations [3], see table 5. AV-3 reduces the probability of unnecessary inhibition by dividing the physical cells in logical cells k and maintaining the causal order among logical cells. In contrast, algorithms PRS and Mobi_Causal ensure causal ordering explicitly according to MHs. Hence, inhibition never occurs. However, these protocols have a message overhead O(n) and O($\sum_{i=1}^{m} li$) over the wireless and wired communication channels, respectively, table 5. Another drawback of Mobi_causal is the non bounded growth of information control storage on each mobile host in order to achieve the causal ordering.

In our approach, causal ordering is carried out by base stations in accordance with the causal view that the mobile hosts have during the system execution, avoiding the unnecessary delivery inhibition. Our algorithm has a message overhead of n bits, one bit by each mobile host in the communication group.

5. Conclusions

An efficient causal ordering protocol, *MOCAVI*, for cellular networks has been presented. *MOCAVI* performs the causal ordering according to the causal view that the mobile host perceives during the system execution, eliminating with this the inhibition effect in the message delivery. *MOCAVI* maintains a low overhead attached to the transmitted messages in the wireless communication channels. Moreover, our protocol is efficient since the amount of computation performed by mobile hosts is low and the control information (overhead) attached to the sent messages in the wired communication channels is dynamically adapted to the behavior of the system.

References

[1] ETSI, European Telecommunications Standards institute GSM recommendations.
[2] S. Alagar and S. Venkatesan, “Causally Ordered Message Delivery in Mobile Systems”, In Proc.Workshop on Mobile Computing Systems and Applications, 169-174, 1994.
[3] L. H. Yen, T.L. Huang, and S.Y. Huang, “A protocol for causally ordered message delivery in mobile computing systems”, ACM/Baltzer Mobile Networks and Applications, vol. 2, no.4, 365-372, 1997.
[4] C.P. Li and T.L. Huang, “A mobile support station based causal multicast algorithm in mobile computing environment”, Proc. 11th Int’l Conf. info. Networking, vol. 2, 9C-1.1-9C-1.10, 1997.
[5] K. H. Chi, L.H. Yen, C.C. Tseng, and T.L. Huang, “A Causal Multicast Protocol for Cellular networks”, IEICE TRANS. INF. & SYST., vol. E83-D. no. 12, 2065-2073,2000.
[6] P. Chandra, and A. Kshemkalyani, “Causal Multicast in Mobile Networks”, Proc. The IEEE Computer Societyapos 12th Annual International Symposium on Volume , Issue , 213 – 220, 4-8 Oct. 2004.
[7] R. Praskash and Mukesh Singhal, “Dependency sequences and hierarchical clocks: efficient alternatives to vector clocks for mobile computing systems”, Wireless Networks, (3):349--360, 1997.
[8] R. Praskash, M. Raynal.and Mukesh Singhal, “An Efficient Causal ordering Algorithm for Mobile Computing Environments”, Proceedings of the 16th International Conference on Distributed Computing Systems, p. 744, 1996.
[9] C. Benzaid and N. Badache, “Mobi_Causal: A protocol for Causal Message Ordering in Mobile Computing Systems”, Mobile computing and communications Review, vol. 9, 19-28,2005.
[10] L. Lamport, “Time Clocks and the Ordering of Messages in Distributed Systems”, Communications ACM 21(7), pp. 558-565, 1978.
[11] S. Pomares Hernández, J. Fanchon, K. Drira,”The Inmediate Dependency Relation: An Optimal Way to Ensure Causal Group Communication”, annual Review of Scalable Computing , Vol. 6, Series on Scalable Computing. Ed. Y.C. Kwong, World Scientific, pp. 61-79, 2004.
[12] C. Skawratananond, N. Mittal, and V. K. Garg, “A Lightweight Algorithm for Causal Message Ordering in Mobile Computing Systems”, In Proc. of 12th ISCA Intl. Conf. on Parallel and Distributed Computing Systems (PDCS), 245-250, Florida, USA, 1999.
[13] B.R. Badrinath, A. Acharya and T. Imielinsky, “Structuring Distributed Algorithms for Mobile Hosts”, Conf. on Distributed Computing Systems, June 1994.
[14] C.Yi Lin, S. Chi Wang and S. Yen Kuo, “An Efficient Time-Based Checkpointing Protocol for Mobile Computing Systems over Mobile IP”, Mobile Networks and Application, 687-697, 2003. h