(Im)possibility of a Coordinated Attack

Faustine van der Grijn
advised by Dick de Jongh

University of Amsterdam, June 2004

Abstract

Making use of epistemic logic it is provable that whenever the smallest uncertainty of message delivery is present, common knowledge via communication is impossible. This implies that a coordinated attack, which demands common knowledge, really can never take place. But there might be possibilities for a coordinated attack when you drop the demand of common knowledge and have a look at the probabilities of proper message delivery. A tentative start is made to develop a theory to explore this.

Contents

1 Introduction

On two mountain tops at both sides of a valley two generals have settled with an army. In the valley lives their common enemy. When the two generals attack from both sides at the same moment, they definitely will win the battle, but when either attacks on his own, he will lose. If one decides to start the battle at a certain moment in the future, he can send a messenger to alert the other general. But the messenger could on his way be captured by the enemy, so the other general in his turn has to send another messenger back to inform the first general that the message has been delivered. Only then the first general knows that the other general knows at what time he has to attack from the opposite side. But a new problem arises: now the second general needs to know whether his message has been delivered, to make sure that he is not the only general that will attack, the first general still waiting for his message to be delivered. But then again the first general needs a new message of receiving and so on...This endless sending of messages will never bring enough knowledge to both generals and so a coordinated attack cannot take place. This is called *the coordinated attack problem.*

For understanding the coordinated attack problem and the proof of its impossibility when you demand common knowledge, an introduction in epistemic logic is needed. Making use of the language of epistemic logic we can reason about knowledge. A short introduction can be found in chapter one. In chapter two and three we extend this logic for describing a multi-agent system.

Then in chapter four we apply this language to prove that common knowledge via communication with uncertain message delivery is impossible, whatever the probability of right message delivery might be. So a coordinated attack which demands common knowledge can never take place.

In these first four chapters I tried to follow as much as possible the notations of *'Reasoning about Knowledge'*[1].

Although common knowledge via communication is impossible, I evolved some theory to estimate probabilities of sufficient shared knowledge for a succesfull coordinated attack. Depending on the chosen strategy it seems sometimes important to ask for a message of receipt and sometimes this isn't important at all.

2 Method

The first thing to do, for me and my teacher D. de Jongh, was to get into epistemic logic. Therefore we read the first two chapters of *'Epistemic Logic for AI and computerscience'*[2], guided by the course notes of R. Verbrugge[3]. After this we stepped over to the book *'Reasoning about Knowledge'*[1] and read chapter 4 and 5 for understanding chapter 6.1, which treats the coordinated attack problem. I obtained a clear picture of the proof of the impossibility of a coordinated attack based on common knowledge. I here give a presentation of

the theory on which it is based and of the proof itself. I added some pictures and a little extra explanation where I thought this might be helpful. After this I brought some stochastics into the subject and made a tentative start to develop a theory to explore the possibilities for a succesfull attack, when you drop the demand for common knowledge.

3 The Language of epistemic logic

Definition 3.1 *When P is a set of primitive propositions, $P = \{p_m, m \in \mathbb{N}\}$ or $P = \{p_0, ..., p_m\}$, and G is a set of m agents $G = \{1, ..., n\}$, then the* ***set of epistemic formulas*** *$\mathcal{L}_K^n(P)$, is the smallest set closed under:*

- $p \in P \Rightarrow p \in \mathcal{L}_K^n(P)$,
- $\varphi, \psi \in \mathcal{L}_K^n(P) \Rightarrow (\varphi \wedge \psi)$, $\neg\varphi \in \mathcal{L}_K^n(P)$,
- $\varphi \in \mathcal{L}_K^n(P) \Rightarrow \forall i \in G : K_i\varphi \in \mathcal{L}_K^n(P)$.

The other connectives $\{\vee, \rightarrow, \leftrightarrow\}$ are defined with $\{\neg, \wedge\}$ in the usual manner. You can read for $K_i\varphi$, 'agent i knows that φ'.

Definition 3.2 *With the language $\mathcal{L}_K^n(P)$ we can construct an* ***axiomatic system*** *$K_{(n)}$, by:*

- *Axioms*

  A1 All propositional tautologies

  A2 $(K_i\varphi \wedge K_i(\varphi \rightarrow \psi)) \rightarrow K_i\psi \quad i = 1, ..., n$

- *Derivation Rules*

  R1 $\frac{\varphi, \varphi \rightarrow \psi}{\psi}$

  R2 $\frac{\varphi}{K_i\varphi} \quad i = 1, ...n$

This axiomatic system corresponds with a class of models $\mathcal{K}_{(n)}$, in the sense that if you can derive a formula in $K_{(n)}$, it is true in each model in this class, and when a formula is true in all models in $\mathcal{K}_{(n)}$, there exists a derivation of it in $K_{(n)}$. In other words, $K_{(n)}$ is sound and complete with respect to $\mathcal{K}_{(n)}$ (Full proofs can be found in [2], Chapter 1). Let us have a look for which class of models this is the case.

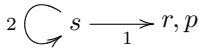
Definition 3.3 *The* ***class of Kripke Models*** *$\mathcal{K}_{(n)}$ is the set of all models $\mathbb{M} =< S, \pi, R_1, ..., R_n >$ with:*

- *S is a non-empty set of states*
- *π: $S \times P \longrightarrow \{true, false\}$*
- *$R_i \subseteq S \times S$ for $i = 1, ..., n$*

Now $(\mathbb{M}, s) \models K_i\varphi$, means that $\forall r((s, r) \in R_i) : (\mathbb{M}, r) \models \varphi$. When $(s, r) \in R_i$, in $(\mathbb{M}, s)$ agent i considers $(\mathbb{M}, r)$ as a possible state. You can read for $(\mathbb{M}, s) \models K_i\varphi$, 'when agent i is in state s, he knows that φ'.

Example 1

fig.1



In *fig.1*:

1. $(\mathbb{M}, s) \vDash K_1 p$
2. $(\mathbb{M}, s) \nvDash K_2 p$
3. $(\mathbb{M}, r) \nvDash K_1 p$

Sometimes we want to look at the reduced class of all Kripke models, with R_i an equivalence relation for $i = 1, ..., n$. This class is denoted by $\mathcal{S}5_{(n)}$. We get a sound and complete axiomatization $S5_{(n)}$ of this, when we add three other axioms to $K_{(n)}$:

Definition 3.4 *The* ***axiomatic system*** $S5_{(n)}$ *is* $K_{(n)}$*, expanded with*

A3 $K_i\varphi \to \varphi \quad i = 1, ..., n$
(Known facts are true)

A4 $K_i\varphi \to K_i K_i \varphi \quad i = 1, ..., n$
(Positive introspection: an agent knows that he knows something)

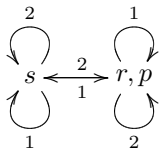
A5 $\neg K_i\varphi \to K_i \neg K_i \varphi$
(Negative introspection: an agent knows that he doesn't know something)

(again proofs can be found in [2], chapter 1).

Example 2

If we want *fig.1* to be a model in $\mathcal{S}5_{(n)}$, then since R_1 has to be an equivalence relation, also $(s, s) \in R_1$ (reflexivity), so the picture has to be extended to this:

fig.2



Now we have $(\mathbb{M}, s) \nvDash K_1 p$, although in *fig.1* we had the opposite. This is sound with respect to the axiomatization, since if we had $(\mathbb{M}, s) \vDash K_1 p$, then with axiom A3 in $S5_{(n)}$ we also would have $(\mathbb{M}, s) \vDash p$, but this is not the case.

We can also extend the language with two operators other than K_i: C and E. We call this expanded language $\mathcal{L}^m_{KEC}(P)$. To define the meaning of these operators in a model, we first need to introduce some notation:

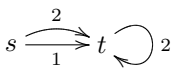
Notation 1

- $s \to t \Longleftrightarrow (s,t) \in R_1 \cup ... \cup R_n$,
- $s \to_i t \Longleftrightarrow (s,t) \in R_i$
- $\forall k \geq 0 : s \to^k t \Longleftrightarrow \exists$ *a sequence* $s = s_0 \to s_1 \to ... \to s_k = t$,
- $\forall k \geq 0 : s \to^k_G t \Longleftrightarrow \exists$ *a sequence* $s = s_0 \to_{i_1} s_1 \to_{i_2} ... \to_{i_k} s_k = t$ *with* $i_1, i_2, ..., i_k \in G$,
- $s \twoheadrightarrow t \Longleftrightarrow \exists k \geq 0 : s \to^k t$,
- $s \twoheadrightarrow_G t \Longleftrightarrow \exists k \geq 0 : s \to^k_G t$.

You can see that $\twoheadrightarrow$ is the reflexive, transitive closure of $\to$. Furthermore if $s \twoheadrightarrow t$, we say t ***is reachable from*** s.

Example 3

fig.3



In *fig.3*:

1. $\neg(s \to s)$
2. $t \to t$
3. $\forall k \in \mathbb{N}_{>0} : \quad s \to^k t$
4. $s \twoheadrightarrow s$

Definition 3.5 *: $KEC_{(m)}$ is the class of models $\mathbb{M} =< S, \pi, R_1, ..., R_m, E, C >$, with $E = R_1 \cup ... \cup R_m$ and $C = E^*$, i.e. C is the reflexive, transitive closure of E. We have:*

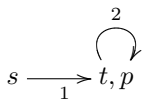
- $(\mathbb{M}, s) \models E\varphi \Leftrightarrow \forall t(s \to t) : \quad (\mathbb{M}, t) \models \varphi$
- $(\mathbb{M}, s) \models C\varphi \Leftrightarrow \forall t(s \twoheadrightarrow t) : \quad (\mathbb{M}, t) \models \varphi$

When we have a subset G of two of more agents among all the agents in the system, then we can also look at E_G and C_G, being defined as if the agents in G are the only agents in the system:

- $(\mathbb{M}, s) \models E_G\varphi \Leftrightarrow \forall t(\exists i \in G(s \rightarrow_i t)) : \quad (\mathbb{M}, t) \models \varphi$
- $(\mathbb{M}, s) \models C_G\varphi \Leftrightarrow \forall t(s \twoheadrightarrow_G t) : \quad (\mathbb{M}, t) \models \varphi$

Example 4

fig.4



In fig.4:

1. $(\mathbb{M}, s) \vDash Ep$
2. $(\mathbb{M}, s) \nvDash Cp$
3. $(\mathbb{M}, t) \vDash Cp$

Lemma 1 *Given a symmetric model in $KEC_{(m)}$, if t reachable from s, then $(\mathbb{M}, s) \vDash C\varphi \Leftrightarrow (\mathbb{M}, t) \vDash C\varphi$.*

Proof:

$\Rightarrow$): Given is $s \twoheadrightarrow t$, so $\exists l \in \mathbb{N}$ with $s \rightarrow^l t$. If $(\mathbb{M}, s) \vDash C\varphi$, then by definition of $C\varphi$, $(\mathbb{M}, s) \vDash E^k\varphi$ for $k = 0, 1, 2, \ldots$. So we know $(\mathbb{M}, t) \vDash E^k\varphi$ for $k = 0, 1, 2, ...$, since $(\mathbb{M}, s) \vDash E^k\varphi$ for $k = l, l+1, l+2, \ldots$. By definition of $C\varphi$ it now follows that $(\mathbb{M}, t) \vDash C\varphi$.

$\Leftarrow$): When $(\mathbb{M}, t) \vDash C\varphi$, by definition of $C\varphi$, $(\mathbb{M}, t) \vDash E^k\varphi$ for $k = 0, 1, 2, \ldots$ Furthermore $s \twoheadrightarrow t \Leftrightarrow t \twoheadrightarrow s$, since we are in a symmetric model, so R_i is symmetric for i=1, ..., n. This means that $\exists l \in \mathbb{N}$ with $t \rightarrow^l s$. Now we know that $(\mathbb{M}, s) \vDash E^k\varphi$ for $k = 0, 1, 2, ...$, since $(\mathbb{M}, t) \vDash E^k\varphi$ for $k = l, l+1, l+2, \ldots$ So by definition $(\mathbb{M}, s) \vDash C\varphi$.

4 A model of a multi-agent system

Definition 4.1 *We call any collection of interacting agents a **multi-agent system**.*

You can think for example of players in a game or processes in a computer network.

Definition 4.2 *In a multi-agent system with agents $i = 1, ..., n$, at a certain time, every agent i is in a certain individual state, the **local state**, denoted by s_i.*

The local state contains all the information which the agent at that particular moment has access to. This can be (too) much information, so most of the times we try to look only at the information that is relevant for our goal.

It is not only important what the local states of the agents are. When we consider communicating agents, we also need to know which messages are on the way and if the communication line is working. We therefore define also a 'state of the environment'.

Definition 4.3 *The **environment's state**, s_e, contains all the information that, besides the local states of the agents, is relevant for our analysis.*

Definition 4.4 *Looking at a system with agents $i = 1, ..., n$ the **global state** of the system, is the (n+1)-tuple $(s_e, s_1, ..., s_n)$. It describes the system at a given time.*

Notation 2

- L_e: *the set of possible states of the environment,*
- L_i: *the set of possible local states s_i of agent i,*
- $\mathcal{G} := L_e \times L_1 \times ... \times L_n$: *the set of all possible global states.*

Definition 4.5 *A **run over** $\mathcal{G}$ is a function $r : \mathbb{N} \longrightarrow \mathcal{G}$, a sequence of global states in $\mathcal{G}$. For (r, m), with r a run and $m \in \mathbb{N}$, $r(m) := (s_e, s_1, ..., s_n)$, with $r_e(m) = s_e$, $r_i(m) = s_i$. **Round** m takes place between time $m - 1$ and m. Agents perform **actions** during a round.*

In the above definition, we think of $\mathbb{N}$ as a set of discrete times. When we look at interacting computers, it is definitely allowed to consider discrete time steps. But also when we look at interacting people, the relevant actions are often taken at discrete times. You can think of rounds in a game. The reason we look at an infinite time domain is that we often do not know in advance how much time is needed to achieve a certain end position. When you have estimated how much time is needed and it is only a finite set of k time steps, then you can simply say that the global state remains the same each step after the k^{th}.

The agents do not have to know what time it is. We have an external clock. An agent has access to the time only in so far as information about it is contained in its local state.

Although we already used the word, we couldn't define a system before we knew about runs:

Definition 4.6 *A **system** R **over** $\mathcal{G}$ is a non-empty set of runs over $\mathcal{G}$. We say (r, m) is a **point** in a system R, if $r \in R$, $m \in \mathbb{N}$.*

What do we know about the individual knowledge of the agents? The knowledge of agent i is determined by s_i, the local state. This means that agent i cannot distinguish between two points (r, m), (r', m') for which $r_i(m) = r'_i(m')$.

Definition 4.7 *In an **interpreted system** $\mathcal{I} = (R, \pi)$, R is a system over G and π is an interpretation of it: π is a function $\pi : \mathcal{G} \times P \longrightarrow \{true, false\}$, i.e. $\forall p \in P$, $\forall s \in \mathcal{G}, \pi(s)(p) \in \{true, false\}$. Again point $(r, m) \in \mathcal{I}$, if $r \in R$.*

Definition 4.8 *A further expansion of the system, gives us a **model** $\mathbb{M}_{\mathcal{I}} = (S, \pi, R_1, ..., R_n)$, with S the set of points in $\mathcal{I}$, and R_i for $i = 1, ..., n$ binary relations on S, with $(r, m)R_i(r', m')$, also denoted by $(r, m) \sim_i (r', m')$, iff $r_i(m) = r'_i(m')$.*

Of course in the above definition, the relations R_i need to be reflexive, symmetric and transitive and so we are in an $\mathcal{S}5_{(m)}$-model, which was defined in chapter 1.
R_e is not included, since we are not interested in what the environment knows, but only in what the agents know.

Notation 3 *For $p \in \Phi$:*

- $(I, r, m) \vDash p \Longleftrightarrow \pi(r(m))(p) = true$,
- $(I, r, m) \vDash K_i\varphi \Longleftrightarrow \forall(r', m')((r, m) \sim_i (r', m')) : \quad (I, r', m') \vDash \varphi$.

So, agent i in state $r_i(m)$ does not know φ, iff $\exists(r', m')$ with $r_i(m) = r'_i(m')$ and $r'_i(m') \vDash \neg\varphi$.

The truth of some propositions, like ψ ='eventually φ', depends on more than the global state and therefore π can't give the proposition a truth value. It could be that $r(m) = r'(m')$, so for any $p \in P(\pi(r(m))(p) = \pi(r'(m'))(p))$, but $(I, r, m) \vDash \psi$ and $(I, r', m') \nvDash \psi$. But this does not contradict the validity of our definitions, since in our frame with language $\mathcal{L}^m_{KEC}(P)$ temporal propositions cannot be formulated.

5 Actions within a multi-agent system

Definition 5.1 *A **protocol** P_i for agent i is a function $P_i : L_i \longrightarrow (\mathcal{P}(ACT_i) - \{\emptyset\})$, in which L_i is the set of possible local states for agent i and ACT_i the set of actions that can be performed by agent i. A **deterministic protocol for agent** i is a function $P_i : L_i \longrightarrow ACT_i$.*

If the protocol maps all local states to singletons, it means that the agent has no choice, the protocol is deterministic. In case of a deterministic protocol it is more natural to think of the protocol as a function from states to actions in stead of a function to sets of actions.

When you look at a situation in which messages are sent, it could be that the message will get lost. You can use the environment protocol P_e to capture this possibility. When the agents' protocols are deterministic, the possibility of different runs is caused by the environment's protocol only.

If you want to know how the global state of the system changes when an agent is acting, you need to know also what the other agents are doing. When one agent is opening a door and an another agent pulls at the opposite side, the outcome is not simply a function of the outcome of individual actions.

Definition 5.2 *A **joint protocol** P is a tuple $(P_1, ..., P_n)$ consisting of protocols P_i for each of the agents $i = 1, ..., n$. A **deterministic protocol** is a joint protocol in which each of the agents' protocols is deterministic.*

You may be surprised that P_e isn't in the tuple. This is because of P_e's special role. You often study cases in which the protocols of the agents are deterministic and in each state the environment can take different actions, like causing a message to get lost.

Definition 5.3 *A **joint action** is a tuple $(a_e, a_1, ..., a_n)$, where a_e is an action of the environment and a_i is an action of agent i.*

Definition 5.4 *A **global state transformer** $\mathcal{T}$ is a function $\mathcal{T} : \mathcal{G} \longrightarrow \mathcal{G}$. A joint action changes a global state s via the associated global state transformer to $\mathcal{T}(s)$.*

How do you know which global state transformer belongs to which action? To answer this question we also need to define a transition function.

Definition 5.5 *A **transition function** is a function $\tau : (a_e, a_1, ..., a_n) \mapsto \tau(a_e, a_1, ..., a_n)$, with $\tau(a_e, a_1, ..., a_n)$ a global state transformer.*

With these definitions we know that the global state $(s_e, s_1, ..., s_n)$ is changed to $(\tau(a_e, a_1, ..., a_n))(s_e, s_1, ..., s_n)$ when the joint action $(a_e, a_1, ..., a_n)$ is performed.

To describe the behavior of the whole system the protocols together don't suffice yet. We need to define some context, in which the protocols exist.

Definition 5.6 *A **context** γ is a tuple $(P_e, \mathcal{G}_0, \tau, \Psi)$, where P_e is the protocol of the environment, $\mathcal{G}_0$ is a nonempty subset of $\mathcal{G}$, τ is a transition function, and Ψ is a condition on runs. Given a context γ and a joint protocol P we can derive a **system representing P in context** γ, denoted by $R^{rep}(P, \gamma)$. This system consists of all **runs consistent with P in context** γ, i.e. all runs r with:*

- $r(0) \in \mathcal{G}_0$,
- $\forall m \geq 0 : \quad (r(m) = (s_e, s_1, ..., s_n) \Rightarrow \exists (a_e, a_1, ..., a_n) \in P_e(s_e) \times P_1(s_1) \times ... \times P_n(s_n)$ *such that* $r(m+1) = \tau(a_e, a_1, ..., a_n)(r(m))$,

- $r \in \Psi$.

We call a ***system*** $\mathcal{R}$ ***consistent with*** P ***in context*** γ *iff* $\mathcal{R} \subseteq R^{rep}(P, \gamma)$

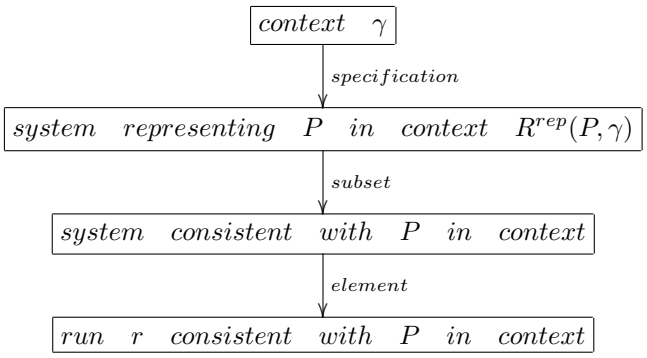
We think of $\mathcal{G}_0$ as the set of possible initial states.

With τ in the tuple γ, implicitly also $L_e, L_1, ..., L_n$ and $ACT_e, ACT_1, ..., ACT_n$ are included in the context, since
$\tau(ACT_e \times ACT_1 \times ... \times ACT_n) : L_e \times L_1 \times ... \times L_n \rightarrow L_e \times L_1 \times ... \times L_n$.

Ψ is a set of runs such that $r \in \Psi$ if r satisfies the condition Ψ. If Ψ is $True$, Rel or $Fair$, this means that we respectively have the condition consisting of all runs, $\{r|$all messages sent in r are eventually delivered$\}$ and
$\{r|$all messages that are repeatedly sent in r are eventually delivered$\}$.

As with systems you can also speak of an **interpreted context** (γ, π), with π an interpretation as defined before.

fig.5



Definition 5.7 *Let* $\sum_i$ *be a set of possible initial states for process* i, INT_i *a set of internal actions for* i *and* MSG *a set of messages.* $\mathcal{G}_0 = L_e \times \sum_1 \times ... \times \sum_n$ *and* $ACT_i = INT_i \cup \{send(\mu, j, i), receive(\mu, j, i) \mid \mu \in MSG, j \in G\}$. *A* ***history*** *for process* i *is then a sequence whose first element is in* $\sum_i$ *and whose later elements are nonempty sets with elements of the form* ACT_i.

The history of an agent describes the run through his or her eyes until a given moment. You can think of the histories as the local states of the agents.

Definition 5.8 *A* ***message-passing system*** *is a system so that for each* (r, m) *and* $i = 1, ..., n$*:*

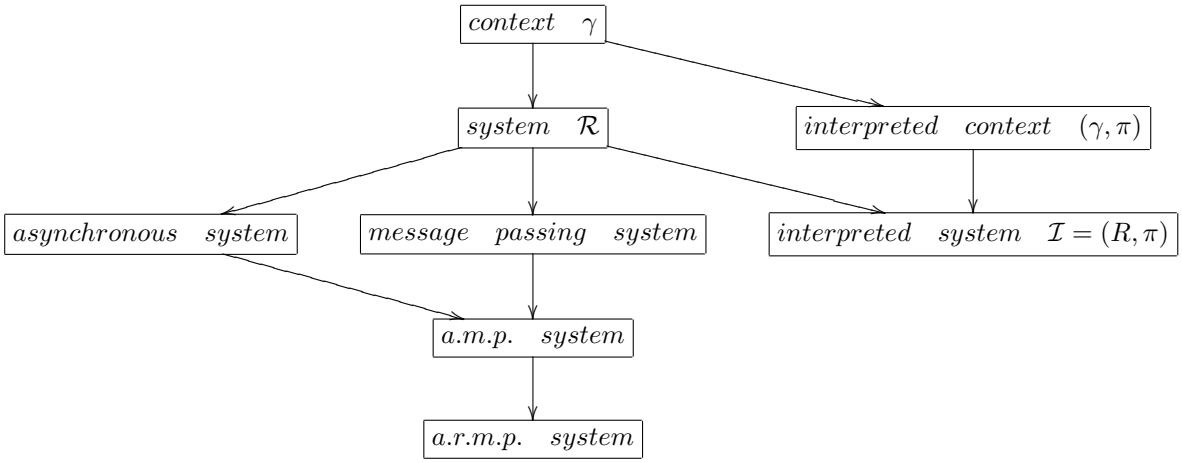
- $r_i(m)$ *is a history over* $\sum_i$, INT_i, *and* MSG,
- *for every event* $receive(\mu, i, j)$ *in* $r_i(m)$ *there exists a corresponding event* $send(\mu, j, i)$ *in* $r_j(m)$,

- $r_i(0)$ *is a sequence of length one (consisting of the initial state of* i*) and* $r_i(m+1)$ *is equal to* $r_i(m)$ *or the result of attaching to* $r_i(m)$ *a set of actions performed by* i *in round* $m+1$*.*

Definition 5.9 *A* ***synchronous system*** *is a system in which every processor knows exactly what time is is. A* ***prefix-closed set*** V ***of histories****, is a set for which* $h \in V$ *implies that also every initial sequence of* h *(prefix) that is not the empty sequence is also in* V*. An* ***asynchronous message-passing system*** *(a.m.p) is a message passing system that is not synchronous, consisting of all runs such that all local states* s_i *of runs in the system are in a prefix-closed set* V_i *of histories. This system is reliable (a.r.m.p) if it satisfies*

- *for all processes* i, j*, and all points* (r, m) *if* $send(\mu, i, j) \in r_j(m)$*, then there exists an* $n \geq m$ *such that* $receive(\mu, j, i) \in r_i(n)$

fig.6



6 The Coordinated Attack Problem

The coordinated attack problem has already been described in the introduction. We saw that the endless sending of messages will never bring the common knowledge we defined in chapter one, so there never can be a coordinated attack in this situation.

We will describe this situation purely mathematically, to give a full proof of the impossibility of a coordinated attack. Afterwards we will describe an attempt to create a somewhat different situation in which a coordinated attack may judged to be possible.

Definition 6.1 *A **message delivery context** is an interpreted context (γ, π), for which:*

- *The environment and the agents have actions resulting in messages being delivered to agents,*
- *The context γ is a **recording context**, i.e. the environment's state includes the sequence of joint actions that have been performed,*
- *The language contains the proposition '**delivered**' (in a model we say this proposition is true, if at least one message has been delivered).*

*A **message delivery system** is a set of runs $(R^{rep}(P, \gamma), \pi)$, with (γ, π) a message delivery context.*

With the coordinated attack problem, we are in a system that displays unbounded message delivery (umd). This means that when agent i receives a message at (r, l) in $\mathcal{R}$ and no other agent receives a message from i in run r between l and m, then at time m all the other agents think it is possible that i has not (yet) received the message.

Notation 4
$d(r, m) = k \Leftrightarrow$ *exactly k messages are delivered in the first m rounds*

Definition 6.2 *If $(\mathcal{R}, \pi)$ is a message delivery system, then $\mathcal{R}$ **displays umd** if $\forall (r, m) \in \mathcal{R}$ with $d(r, m) > 0$ (so $m \neq 0$), $\exists i, \exists r'$ such that*

- $\forall j \neq i, \forall m' \leq m : r'_j(m') = r_j(m')$,
- $d(r', m) < d(r, m)$.

An example of a message delivery system is an a.m.p. system as defined in chapter 3, since such a system contains *every* run r in which for all $m \in \mathbb{N}$ and for all $i = 1, ..., n$ $(r_i(m)) = s_i$, with s_i contained in a set V_i. So when we have a run $r = (r(0), r(1), ..., r(m), r(m+1), ...)$ in an a.m.p. system $\mathcal{R}$ with $r(m) = (s_e, s_1, ..., s_n)$ and $r(m+1) = (t_e, t_1, ..., t_n)$, this means that $s_i, t_i \in V_i$, so $\mathcal{R}$ also contains a run $r' = (r'(0), r'(1), ..., r'(m), r'(m+1), ...)$ with $r'(j) = r(j)$ for $0 \leq j \leq m$, $r'(m+1) = (t_e, t_1, ..., t_{k-1}, s_k, t_k + 1, ..., t_n)$ and for $j > m+1$, $r'(j)$ a tuple of arbitrary extensions of the sequences in the tuple $r'(j-1)$ in $L_e \times V_1 \times ... \times V_n$. An a.r.m.p system is an example of a system displaying umd as well, since a message eventually has to be delivered in such a system, but nothing is demanded about when. So you can complete the sequence to a run r', starting with $r'(0), r'(1), ..., r'(m+1)$ as above, with $r'(j)$ again tuples in $L_e \times V_1 \times ... \times V_n$ for $j > m+1$, almost arbitrarily as long as the tuple r'(j) contains extensions of the sequences in the tuple r'(j-1) and there is one $m' > m+1$ with $r'_k(m')$ a state in which the message (which was delivered to k in $r(m+1)$) is delivered.

Definition 6.3 *A **context** γ **displays umd** iff all systems $R^{rep}(P,\gamma)$ display umd for every protocol P that can be run in the given context.*

If we look at the coordinated attack problem, we need a system that displays umd, since it can always be the case that one message has not (yet) been delivered, no matter how the protocols of the generals may be. So we look at a context displaying umd, to find within it a system with a protocol such that it displays umd.

Theorem 1 *If $\mathcal{I} = (\mathcal{R}, \pi)$ is a message delivery system that displays umd, then $\mathcal{I} \vDash \neg C(delivered)$.*

Proof: by induction on $d(r,m)$.

Base: Look at an arbitrary (r,m) in $\mathcal{R}$. If $d(r,m) = 0$, then $(\mathcal{I}, r(m)) \vDash \neg C(delivered)$, since there are no delivered messages at all.

Induction-step: Suppose $\forall (r',m') \in \mathcal{R} : d(r',m') \leq k \Rightarrow (\mathcal{I}, r'(m')) \vDash \neg C(delivered)$ (IH). Now look at an arbitrary $r \in \mathcal{R}$, $m \in \mathbb{N}$ with $d(r,m) = k{+}1$. Since the system displays umd, $\exists r', i'$ with $\forall l \leq m, j \neq i : r'_j(l) = r_j(l)$ and $d(r',m) < d(r,m)$, so by (IH) $(\mathcal{I}, r'(m)) \vDash \neg C(delivered)$. Since $r'_j(m) = r_j(m)$, (r',m) is reachable from (r,m), so by Lemma 1, also $(\mathcal{I}, r(m)) \vDash \neg C(delivered)$.

Conclusion: $\forall (r,m) \in \mathcal{R} : (\mathcal{I}, r(m)) \vDash \neg C_G(delivered)$, what means $\mathcal{I} \vDash \neg C_G(delivered)$

This theorem tells us that the generals cannot obtain common knowledge by communication only.

Definition 6.4 *An interpreted context (γ, π) is a **ca-compatible context** if:*

- *(γ, π) is a message delivery context,*
- *there are at least two agents, named general A and general B,*
- *$\forall i \in \{A, B\}$: $attack_i \in ACT_i$,*
- *$\forall i \in \{A, B\}$ the language contains the proposition $attacked_i$, with $[(\mathcal{I}, (r(m)) \vDash attacked_i] \Longleftrightarrow [attack_i \in h]$, in which h is the sequence of joint actions (since a message delivery by definition is a recording context, $L_e =< ..., h, ... >$).*

For reasoning about the possibility of a coordinated attack, we need some notation for describing which actions are going to be taken in the following round:

Notation 5

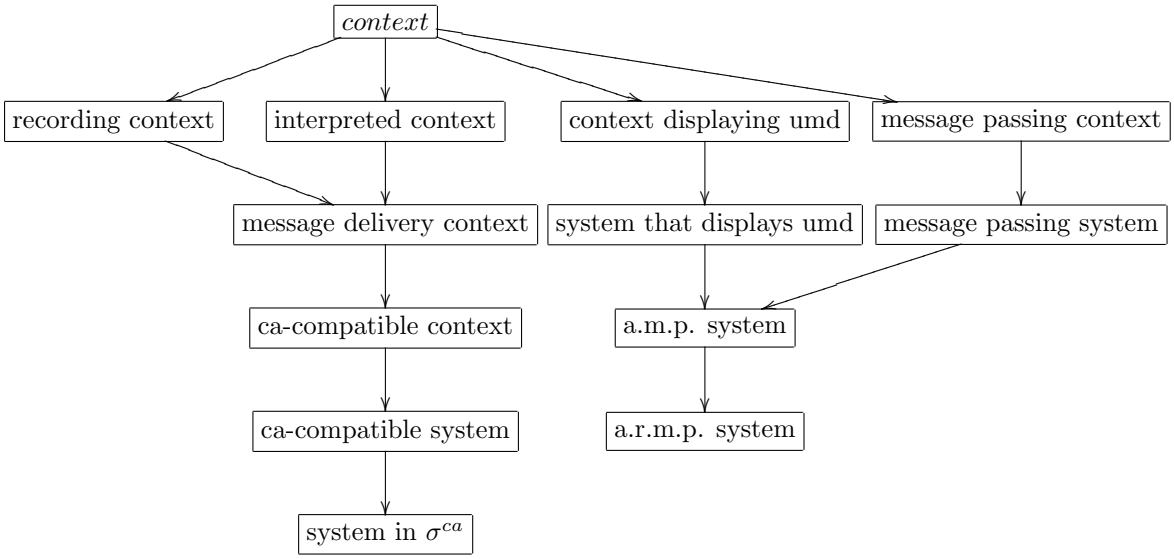
- $(\mathcal{I}, (r(m)) \vDash \bigcirc\varphi \Longleftrightarrow (\mathcal{I}, r(m+1)) \vDash \varphi$,

- $(\mathcal{I}, r(m)) \models attacking_i \iff (\mathcal{I}, r(m)) \models \neg attacked_i \wedge \bigcirc attacked_i$,
- $(\mathcal{I}, r(m)) \models attack \iff (\mathcal{I}, r(m)) \models attacking_A \wedge attacking_B$.

Definition 6.5 *By σ^{ca} we denote the set of all ca-compatible interpreted systems $\mathcal{I} = (\mathcal{R}, \pi)$ for which:*

- $\mathcal{I} \models attacking_A \Leftrightarrow attacking_B$,
- $\mathcal{I} \models \neg delivered \Rightarrow \neg attack$,
- $\exists (r, m) \in \mathcal{R} : (\mathcal{I}, r, m) \models attack$.

fig.7



When we look at the coordinated attack problem, we see that we are in a ca-compatible context, and the situation in which a succesfull attack can take place, is precisely described by the requirements in σ^{ca}. This is why we use this definition:

Definition 6.6 *P **is a protocol for coordinated attack in a ca-compatible context** $(\gamma, \pi) \Leftrightarrow P$ satisfies σ^{ca} in (γ, π).*

Lemma 2 *Let (γ, π) be a ca-compatible context and $\mathcal{I} = (R^{rep}(P, \gamma), \pi)$. If $\mathcal{I} \models attack \Rightarrow E_{\{A,B\}}(attack)$, then $\mathcal{I} \models attack \Rightarrow C_{\{A,B\}}(attack)$.*

Proof:
Let (r, m) be an arbitrary point in $\mathcal{I}$. Suppose that $(\mathcal{I}, r(m)) \models attack \Rightarrow$

$E_{\{A,B\}}(attack)$. We will show that $\forall k \in \mathbb{N} : (\mathcal{I}, r(m)) \vDash attack \Rightarrow E^k_{\{A,B\}}(attack)$, with induction on k.

Base: for $k = 1$, this is just our assumption.

Induction Step: Let the (IH) be that $\forall l \leq k : (\mathcal{I}, r(m)) \vDash attack \Rightarrow E^l_{\{A,B\}}(attack)$. The (IH) gives us that if $(\mathcal{I}, r(m)) \vDash attack, \quad \forall r'(m') \in \mathcal{I}$ which are k-reachable from $r(m)$ via R_A, R_B : $(\mathcal{I}, r'(m')) \vDash attack$. Furthermore since $\mathcal{I} \vDash attack \Rightarrow E_{\{A,B\}}(attack)$ (our assumption), $(\mathcal{I}, r'(m')) \vDash attack \Rightarrow (\mathcal{I}, r'(m')) \vDash E_{\{A,B\}}(attack)$. So when $(\mathcal{I}, r(m)) \vDash attack, \quad \forall (r", m") \in \mathcal{I}$ which are $k+1$-reachable from (r, m) with R_A and R_B, $(\mathcal{I}, r"(m")) \vDash attack$. This gives us what we wanted: $(\mathcal{I}, r(m)) \vDash attack \Rightarrow E^{k+1}_{\{A,B\}}(attack)$.

Conclusion: Given $(\mathcal{I}, r(m)) \vDash attack \Rightarrow E_{\{A,B\}}(attack)$,
$\forall k \in \mathbb{N} : \quad (\mathcal{I}, r(m)) \vDash attack \Rightarrow E^k_{\{A,B\}}(attack) \Longleftrightarrow_{def} \mathcal{I} \vDash attack \Rightarrow C_{\{A,B\}}(attack)$.

Theorem 2 *When (γ, π) is a ca-compatible context, P is a deterministic protocol and $\mathcal{I} = (R^{rep}(P, \gamma), \pi)$ satisfies σ^{ca}, then $\mathcal{I} \vDash attack \Rightarrow C_{\{A,B\}}(attack)$.*

Proof:
Let (r, m) be an arbitrary point in $\mathcal{I}$. Suppose $(\mathcal{I}, r(m)) \vDash attack$. By definition of *attack*, this is equivalent to $(\mathcal{I}, r(m)) \vDash attacking_A \wedge attacking_B$. This means $P_A(r_A(m)) = attack_A$ and $P_B(r_B(m)) = attack_B$: A and B both attack in round $m+1$. Since P_A and P_B are both protocols within the deterministic joint protocol P:

- If $(r', m') \in \mathcal{I}$ with $r_A(m) = r'_A(m')$, then $P_A(r_A(m)) = P_A(r'_A(m'))$. So $(\mathcal{I}, r'(m')) \vDash attacking_A$ and since $\mathcal{I}$ satisfies σ^{ca}, also $(\mathcal{I}, r'(m')) \vDash attacking_B$. Now we have $(\mathcal{I}, r'(m')) \vDash attacking_A \wedge attacking_B$ which means that $(\mathcal{I}, r'(m')) \vDash attack$. So for all points (r', m') which A considers possible in (r, m) counts $(\mathcal{I}, r'(m')) \vDash attack$ and so $(\mathcal{I}, r(m)) \vDash K_A(attack)$.

- If $(r", m") \in \mathcal{I}$ with $r_B(m) = r"_B(m")$, then $P_B(r_B(m)) = P_B(r"_B(m")$. So $(\mathcal{I}, r"(m")) \vDash attacking_B$ and since $\mathcal{I}$ satisfies σ^{ca}, also $(\mathcal{I}, r"(m")) \vDash attacking_A$. Now we have $(\mathcal{I}, r"(m")) \vDash attacking_A \wedge attacking_B$ what means that $(\mathcal{I}, r"(m")) \vDash attack$. So for all points $(r", m")$ which B considers possible in (r, m) counts $(\mathcal{I}, r"(m")) \vDash attack$ and so $(\mathcal{I}, r(m)) \vDash K_B(attack)$.

Now we have $(\mathcal{I}, r(m)) \vDash K_A(attack) \wedge K_B(attack) \Leftrightarrow (\mathcal{I}, r(m)) \vDash E_{\{A,B\}}(attack)$, and since (r, m) was arbitrary in $\mathcal{I}$ with $(\mathcal{I}, r(m)) \vDash attack$, even $\mathcal{I} \vDash attack \Rightarrow E_{\{A,B\}}(attack)$. So, making use of lemma 2 we can say $\mathcal{I} \vDash attack \Rightarrow C_{\{A,B\}}(attack)$.

Theorem 3 *If (γ, π) is a ca-compatible context, P is a deterministic protocol and $\mathcal{I} = (R^{rep}(P, \gamma), \pi)$ satisfies σ^{ca}, then $\mathcal{I} \vDash attack \Rightarrow C_{\{A,B\}}(delivered)$.*

Proof: Suppose for an arbitrary (r', m') that $(\mathcal{I}, r'(m')) \vDash E_{\{A,B\}}(attack)$. Then for all (r, m) with $(r, m)R_i(r', m')$, $i \in \{A, B\}$, $(\mathcal{I}, r(m)) \vDash attack$. By definition of σ^{ca}, $\mathcal{I} \vDash \neg delivered \Rightarrow \neg attack$. So $\forall (r, m) \in \mathcal{I} : \quad (\mathcal{I}, r(m)) \vDash \neg delivered \Rightarrow \neg attack$. In particular $\forall (r, m) \in \mathcal{I}$ with $(r, m)R_i(r', m')$, $\quad i \in \{A, B\}, \quad (\mathcal{I}, r(m)) \vDash \neg delivered \Rightarrow \neg attack$. So $\forall (r, m) \in \mathcal{I}$ with $(r, m)R_i(r', m')$: $(\mathcal{I}, r(m)) \vDash delivered$. This gives us $(I, r'(m')) \vDash E_{\{A,B\}}(delivered)$. So $(\mathcal{I}, r'(m')) \vDash E_{\{A,B\}}(attack) \Rightarrow E_{\{A,B\}}(delivered)$. Since (r', m') was arbitrary in $\mathcal{I}$, $\mathcal{I} \vDash E_{\{A,B\}}(attack) \Rightarrow E_{\{A,B\}}(delivered)$. Using induction on k we can show similarly that for all $k \in \mathbb{N}, \quad \mathcal{I} \vDash E^k_{\{A,B\}}(attack) \Rightarrow E^k_{\{A,B\}}(delivered)$. By definition of $C_{\{A,B\}}$, we have then that $\mathcal{I} \vDash C_{\{A,B\}}(attack) \Rightarrow C_{\{A,B\}}(delivered)$. Combining this result with theorem 2 gives us $\mathcal{I} \vDash attack \Rightarrow C_{\{A,B\}}(delivered)$.

Theorem 4 *If (γ, π) is a ca-compatible context and γ displays umd, then there doesn't exist a deterministic protocol P that satisfies σ^{ca} in (γ, π).*

Proof:
Theorem 1 tells us $\mathcal{I} \vDash \neg C_{\{A,B\}}(delivered)$. Theorem 3 says $\mathcal{I} \vDash attack \Rightarrow C_{\{A,B\}}(delivered)$. Combining these results shows that $\mathcal{I} \vDash \neg attack$.

Even if the generals are in a situation that every message is properly delivered in the same round that it has been sent, there cannot be common knowledge because of the uncertainty of the delivery. Hard as it is to accept, there never can be any coordinated attack.

7 Probabilities for message delivery

In the previous chapter we wanted to be one hundred percent sure that both generals knew about each other that they were to attack. But perhaps sometimes it is possible to estimate the probability that both generals know about the attack and when it's only 98 percent sure that general B is also to attack at the same moment, isn't that enough for general A to risk a battle?

We still have to look at ca-compatible contexts, since these describe the possible actions and select the language to formulate propositions about the results of these actions. Furthermore the context still displays umd: it still may take an arbitrarily long time for the message to be delivered. But the difference in the situation lies in the requirements of the system. We can delete this item from the specification σ^{ca}:

$I \vDash \neg delivered \Rightarrow \neg attack$

and replace it by:

$I \vDash P(\neg delivered) > q \Rightarrow \neg attack$

Suppose that the probability of a message sent in a round being delivered to the right person in the same round is p_1, by the second round after sending p_2, by the third round after sending p_3, etcetera. It is obvious then that $p_1 \leq p_2 \leq p_3 ...$

Now we will look for reasonable values of p_i for $i = 1, ..., k$. Let us assume for a moment that the messenger cannot be kept by the enemy or die on his way. So eventually he will deliver the message to the receiver, but it can take arbitrarily long, he may be delayed (unbounded message delivery). Let the probability of the messenger delivering the message (if it's not already received earlier) from general A to general B in each round be $\frac{r}{s}$, $r, s \in \mathbb{Z}_+$. Then the probability p_k of the message having been delivered from general A to general B within k rounds is:

$p_k =$

$= P(\text{message delivered within k rounds}) =$

$= \sum_{i=1}^{k} P(\text{message delivered in } i^{th} \text{ round}) =$

$= \sum_{i=1}^{k} \frac{r}{s}(1 - \frac{r}{s})^{i-1} =$

$$= \frac{\frac{r}{s} - \frac{r}{s}(1 - \frac{r}{s})^k}{1 - (1 - \frac{r}{s})} =$$

$$= \frac{\frac{r}{s} - \frac{r}{s}(1 - \frac{r}{s})^k}{\frac{r}{s}} =$$

$= 1 - (1 - \frac{r}{s})^k.$

This seems reasonable, since this gives us $lim_{k\to\infty} p_k = 1$ and we assumed that the message eventually will be delivered.

But we have to add the possibility that the enemy captures the messenger or the messenger dies during the trip. When this happens, the message won't be delivered at all. On the one side you could think that the probability of the messenger on his way being captured by the enemy or dying of a disease grows when the messenger is getting more delayed: the duration of the trip is longer. But a delay could also mean a decrease of the probability since it could be caused by higher caution of the messenger. I therefore just assume that the probability of the message never being delivered to general B is independent of the time the trip takes, let's say it is $1 - c$, with $c \in (0, 1)$. Then we have:

$p_k =$

$= P(\text{message delivered within } k \text{ rounds}) =$

$= \sum_{i=1}^{k} P(\text{message delivered in } k^{th} \text{ round}) =$

$= \sum_{i=1}^{k} c\frac{r}{s}(1 - \frac{r}{s})^{i-1} =$

$$= c\frac{\frac{r}{s} - \frac{r}{s}(1 - \frac{r}{s})^k}{1 - (1 - \frac{r}{s})} =$$

$$= c\frac{\frac{r}{s} - \frac{r}{s}(1 - \frac{r}{s})^k}{\frac{r}{s}} =$$

$$= c - c(1 - \tfrac{r}{s})^k.$$

Now p_k converges to c for k to infinity, the probability of a message being eventually delivered.

It could be that A wants a confirmation of his or her message being delivered to B first, before he or she really attacks in the planned k^{th} round. The message that general A sends to B not only contains information about the planned attack then, but also asks for sending a confirmation of receipt. It might be that the probability of a message sent by B to A within a given number of rounds is higher than in the opposite direction, since it could be for example that the top of the hill where general B is settled is much higher than the one where general A is, so the messenger has to climb a lot more for delivering the message from A to B. Let us suppose the probability for a message being sent from B to A in each round until it really is delivered is $\frac{t}{u}, \quad t, u \in \mathbb{Z}_+$. We then can calculate the probability q_k of a message being sent from A to B and also a confirmation being delivered in return within the k rounds before the attack is planned. We again first ignore the probability of the message being not delivered at all and assume the messenger can only be delayed.

$q_k =$

$= P(\text{messages delivered there and back within } k \text{ rounds}) =$

$= \sum_{i=1}^{k-1} P(\text{message delivered to B in } i^{th} \text{ round, confirmation to A within } k-i \text{ rounds}) =$

$= \sum_{i=1}^{k-1} [\frac{r}{s}(1-\frac{r}{s})^{i-1} \sum_{j=1}^{k-i} \frac{t}{u}(1-\frac{t}{u})^{j-1}] =$

$$= \sum_{i=1}^{k-1} \frac{r}{s}(1-\frac{r}{s})^{i-1} \frac{\frac{t}{u} - \frac{t}{u}(1-\frac{t}{u})^{k-i}}{1-(1-\frac{t}{u})} =$$

$= \sum_{i=1}^{k-1} \frac{r}{s}(1-\frac{r}{s})^{i-1}(1-(1-\frac{t}{u})^{k-i}) =$

$= \sum_{i=1}^{k-1} \frac{r}{s}(1-\frac{r}{s})^{i-1} - \sum_{i=1}^{k-1} \frac{r}{s}(1-\frac{r}{s})^{i-1}(1-\frac{t}{u})^{k-i} =$

$$= 1-(1-\frac{r}{s})^{k-1} - \frac{\frac{r}{s}(1-\frac{t}{u})^k}{1-\frac{r}{s}} \sum_{i=1}^{k-1} \left(\frac{1-\frac{r}{s}}{1-\frac{t}{u}}\right)^i$$

$$= 1-(1-\frac{r}{s})^{k-1} - \frac{\frac{r}{s}(1-\frac{t}{u})^k}{1-\frac{r}{s}} \left(\frac{\frac{1-\frac{r}{s}}{1-\frac{t}{u}} - \left(\frac{1-\frac{r}{s}}{1-\frac{t}{u}}\right)^k}{1-\frac{1-\frac{r}{s}}{1-\frac{t}{u}}}\right) =$$

$$= 1-(1-\frac{r}{s})^{k-1} - \frac{r}{s}(1-\frac{t}{u})^{k-1} \frac{1-\left(\frac{1-\frac{r}{s}}{1-\frac{t}{u}}\right)^{k-1}}{1-\frac{1-\frac{r}{s}}{1-\frac{t}{u}}}.$$

We know $lim_{k\to\infty} 1-(1-\frac{r}{s})^k = lim_{k\to\infty} p_k = 1$ and furthermore

$$lim_{k\to\infty} \frac{r}{s}(1-\frac{t}{u})^{k-1} \frac{1-\left(\frac{1-\frac{r}{s}}{1-\frac{t}{u}}\right)^{k-1}}{1-\frac{1-\frac{r}{s}}{1-\frac{t}{u}}} = 0, \text{ so again}$$

$lim_{k\to\infty} q_k = lim_{k\to\infty} p_k = 1$. This means that the probability of the delivery there and back within the k rounds goes to one if the time of the battle goes to infinity far in the future.

When we add the possibility of the messenger for a particular reason not delivering the message at all, again assuming that this probability is independent of how long the trip will take, we get two constants $c_1, c_2 \in (0,1)$ giving the probability c_1 of the message being eventually delivered from A to B and the probability c_2 of the message of confirmation eventually being delivered, given that the first one is delivered. We get:

$q_k =$

$= P(\text{messages delivered there and back within } k \text{ rounds}) =$

$= \sum_{i=1}^{k-1} P(\text{message delivered to B in } i^{th} \text{ round, confirmation to A within } k-i \text{ rounds}) =$

$= c_1 \sum_{i=1}^{k-1} [\frac{r}{s}(1-\frac{r}{s})^{i-1} c_2 \sum_{j=1}^{k-i} \frac{t}{u}(1-\frac{t}{u})^{j-1}] =$

$$= c_1 c_2 \sum_{i=1}^{k-1} \frac{r}{s}(1-\frac{r}{s})^{i-1} \frac{\frac{t}{u} - \frac{t}{u}(1-\frac{t}{u})^{k-i}}{1-(1-\frac{t}{u})} =$$

$= c_1 c_2 \sum_{i=1}^{k-1} \frac{r}{s}(1-\frac{r}{s})^{i-1}(1-(1-\frac{t}{u})^{k-i}) =$

$= c_1 c_2 \sum_{i=1}^{k-1} \frac{r}{s}(1-\frac{r}{s})^{i-1} - c_1 c_2 \sum_{i=1}^{k-1} \frac{r}{s}(1-\frac{r}{s})^{i-1}(1-\frac{t}{u})^{k-i} =$

$$= c_1 c_2 \left(1 - (1-\frac{r}{s})^{k-1} - \frac{\frac{r}{s}(1-\frac{t}{u})^k}{1-\frac{r}{s}} \sum_{i=1}^{k-1} \left(\frac{1-\frac{r}{s}}{1-\frac{t}{u}} \right)^i \right) =$$

$$= c_1 c_2 \left(1 - (1-\frac{r}{s})^{k-1} - \frac{\frac{r}{s}(1-\frac{t}{u})^k}{1-\frac{r}{s}} \left(\frac{\frac{1-\frac{r}{s}}{1-\frac{t}{u}} - \left(\frac{1-\frac{r}{s}}{1-\frac{t}{u}} \right)^k}{1-\frac{1-\frac{r}{s}}{1-\frac{t}{u}}} \right) \right) =$$

$$= c_1 c_2 \left(1 - (1-\frac{r}{s})^{k-1} - \frac{r}{s}(1-\frac{t}{u})^{k-1} \frac{1-\left(\frac{1-\frac{r}{s}}{1-\frac{t}{u}} \right)^{k-1}}{1-\frac{1-\frac{r}{s}}{1-\frac{t}{u}}} \right).$$

8 Strategies for coordinated attack

Now that we have come this far, we look at three possible strategies for general A:

1. general A chooses for the largest probability for a succesfull attack to take place.

2. general A chooses for the smallest probability of a failing attack (an attack by only one of the generals) taking place.

3. general A chooses for the largest probability that a succesfull attack takes place, given the fact that there will be an attack.

Whether general A had better ask for a confirmation of message delivery before attacking, depends on which strategy he or she follows.

8.1 Strategy 1

If general A doesn't ask for a confirmation before attacking in the k^{th} round, a succesfull attack depends solely on the delivery of the message from general A to general B. Then:

P(succesfull attack) =

$= P$(B receives message within k rounds) =

$= p_k =$

$= c_1 \sum_{i=1}^{k} \frac{r}{s}(1-\frac{r}{s})^{i-1}$.

When general A does ask for a message of confirmation before attacking in the planned round, we get:

P(succesfull attack) =

$= P$(message there and back delivered within k rounds) =

$= q_k =$

$= c_1 \sum_{i=1}^{k-1}[\frac{r}{s}(1-\frac{r}{s})^{i-1} c_2 \sum_{j=1}^{k-i} \frac{t}{u}(1-\frac{t}{u})^{j-1}]$.

This shows that when general A follows strategy 1, he will not ask for a confirmation, since in any case $p_k > q_k$ (even $p_{k-1} > q_{k-1}$). Namely if q_k is larger or equal then p_k (this implies also $q_k > p_{k-1}$), then at least for one i with $1 \leq i \leq k-1$ we must have that $c_2 \sum_{j=1}^{k-i} \frac{t}{u}(1-\frac{t}{u})^{j-1} \geq 1$. But $\forall 1 \leq i \leq k-1$:

$c_2 \sum_{j=1}^{k-i} \frac{t}{u}(1-\frac{t}{u})^{j-1} =$

$= c_2 \frac{\frac{t}{u} - \frac{t}{u}(1-\frac{t}{u})^{k-i}}{\frac{t}{u}} =$

$= c_2(1-(1-\frac{t}{u})^{k-i}) < 1$.

8.2 Strategy 2

If A doesn't ask for a confirmation from general B, the probability for a failing attack is the probability for B not to receive the message within k rounds:

P(failing attack) =

$= P$(B doesn't receive the message within k rounds) =

$= 1 - p_k$.

If A does ask for a confirmation from B of receiving the message and doesn't attack until this confirmation is delivered, we get:

P(failing attack) =

$= P$(B receives message within $k-1$ rounds)$-P$(A gets a confirmation of delivery within k rounds) =

$= p_{k-1} - q_k$.

Following this strategy, it depends on the parameters if it is better for A to ask for a confirmation from B or not. If $c_1 = \frac{3}{4}, \frac{r}{s} = \frac{1}{4}, k = 3$ we get without for a confirmation that

$P(\text{failing attack}) =$

$= 1 - p_k =$

$= 1 - (c_1 - c_1(1 - \frac{r}{s})^k)$

$= 1 - \frac{3}{4} + (\frac{3}{4})^4 \approx$

$\approx 0,57.$

We've seen earlier that $0 < q_k < p_{k-1}$, so with asking for a confirmation we get

$P(\text{failing attack}) =$

$= p_{k-1} - q_k < p_{k-1} \leq p_k \approx$

$\approx 1 - 0.57 = 0,43.$

In this example following strategy 1, A will ask for a confirmation.

But we could also have that $c_1 = \frac{3}{4}$, $\frac{r}{s} = \frac{3}{4}$, $\frac{t}{u} = \frac{11}{16}$ and $c_2 = \frac{1}{10}$. Then we had without asking for a confirmation

$P(\text{failing attack}) =$

$= 1 - \frac{3}{4} + \frac{3}{4}(\frac{1}{4})^3 \approx$

$\approx 0,26.$

and with asking for a confirmation

$P(\text{failing attack}) =$

$= p_{k-1} - q_k =$

$$= c_1 - c_1(1-\tfrac{r}{s})^{k-1} - c_1 c_2 \left(1 - (1-\tfrac{r}{s})^{k-1} - \tfrac{r}{s}(1-\tfrac{t}{u})^{k-1} \frac{1 - \left(\frac{1-\frac{r}{s}}{1-\frac{t}{u}} \right)^{k-1}}{1 - \frac{1-\frac{r}{s}}{1-\frac{t}{u}}} \right) \approx$$

$\approx 0,637.$

So with these parameters A will not ask for a confirmation.

8.3 Strategy 3

Without asking for a confirmation message we get

$P(\text{succesfull attack} \mid \text{attack takes place}) =$

$= \frac{P(\text{B receives message within } k \text{ rounds})}{P(\text{A attacks})} =$

$= \frac{p_k}{1} = p_k.$

With asking for the confirmation we get

$P(\text{succesfull attack} \mid \text{attack takes place}) =$

$$= \frac{P(\text{A receives confirmation within } k \text{ rounds})}{P(\text{B reveives message within } k-1 \text{ rounds})} =$$

$$= \frac{q_k}{p_{k-1}}.$$

In this case asking for a confirmation from A also depends on the values of the parameters. Let the parameters c_1, $\frac{r}{s}$ and k be the same as in the first example for strategy 2 and let $c_2 = c_1 = \frac{3}{4}$, $\frac{t}{u} = \frac{1}{4}$. Then we get without asking for a confirmation:

$P(\text{succesfull attack} \mid \text{attack takes place}) =$

$$= p_k =$$

$$= c_1 - c_1(1 - \tfrac{r}{s})^k$$

$$= \tfrac{3}{4} - \tfrac{3}{4}(1 - \tfrac{1}{4})^3 \approx$$

$$\approx 0,43.$$

Furthermore with asking for a confirmation, we get:

$P(\text{succesfull attack} \mid \text{attack takes place}) =$

$$= \frac{q_k}{p_{k-1}} =$$

$$= \frac{c_1 c_2 \left(1 - (1 - \frac{r}{s})^{k-1} - \frac{r}{s}(1 - \frac{t}{u})^{k-1} \frac{1 - \left(\frac{1 - \frac{r}{s}}{1 - \frac{t}{u}}\right)^{k-1}}{1 - \frac{1 - \frac{r}{s}}{1 - \frac{t}{u}}}\right)}{c_1 - c_1(1 - \frac{r}{s})^{k-1}} =$$

$$= \frac{\left(\frac{3}{4}\right)^2 \left(1 - \left(\frac{3}{4}\right)^2 - \left(\frac{1}{4}\right)^3 \frac{1 - 3^2}{1 - 3}\right)}{\frac{3}{4} - (\frac{3}{4})^3} \approx$$

$$\approx 0,64.$$

In this case following strategy 3 general A would choose for a confirmation of the delivery.

With the parameters of the second example for strategy 2, we get without asking for a confirmation

$P(\text{succesfull attack} \mid \text{attack takes place}) \approx 0,74$

and with asking for it

$P(\text{succesfull attack} \mid \text{attack takes place}) \approx 0,09$

so general A will definitely not ask for a confirmation here.

9 Different opinions of the generals

In all these examples we have only been looking for a decision by general A, but it is likely that when general A asks for a confirmation also general B on his turn has to decide whether or not he will attack before he has received a confirmation of his confirmation of receipt having been properly delivered to A. So A has to reason about the decision that B will make as well. It all can be made as complicated as you wish.

Of course the generals don't really know what the values of $c_1, c_2, \frac{r}{s}$ and $\frac{t}{u}$ are, they have to estimate them. It therefore could be that when the two generals have different knowledge about for example the landscape or the rate of preparedness of the enemy, they have very different opinions about the values of these parameters. Furthermore it is possible that the generals have wrong expectations about how the other general estimates the values of the parameters. When this is the case, it is possible that although they have in mind the same strategy, they come to different conclusions about asking for a confirmation or not. You only have to look at the examples for strategy 2 and 3 to see this. On the contrary strategy 1 cannot lead to misunderstandings about choosing for a confirmation or not, since independently of the parameters this never is a smart decision.

10 Starting the attack at different times

We said in the previous chapter that the generals could win only if they attacked exactly at the same moment, but it could be the case that they are victorious if the generals start the battle at different times, but within a not too large time interval. Let us say that they can win the battle if they start their attack within n rounds after each other. Then we can delete another item from σ^{ca}:

$I \vDash attacking_A \Leftrightarrow attacking_B$

For calculating the probabilities this hardly make a difference. The only thing you have to change is that you need to know the probability that the message is delivered within $k+n$ rounds, in stead of k rounds, so the probability of a succesfully attack becomes higher.

11 Conclusion and discussion

Using epistemic logic it is provable that a coordinated attack is never possible when you want to be hundred percent sure that there exists common knowledge, since common knowledge cannot be reached via communication.

The given proof of this seems very plausible to me and correct in the mathematical world and certainly points out a very serious problem in coordination.

But in a real situation the generals have to do something, even though common knowledge theoretically cannot be obtained.

When you look at probabilities of messages being properly delivered, there are some possibilities of a coordinated attack, although it can lead to much misunderstandings and it is hard to decide by what standards the other general is deciding and on which information he or she bases his or her conclusions.

I have only been looking at a small part of this infinitely complicated system. I am curious if other people can construct better probability models for this, which are more extended and defended. I am aware of the shortcomings of my model and I do not claim that this is *the* model for this. It is my own creation and I only would be glad if others can improve it.

References

[1] R. Fagin, J. Y. Halpern, Y. Moses, M. Y. Vardi
Reasoning about Knowledge
MIT Press (1995)
ISBN 0-262-06162-7

[2] J.-J. Ch. Meyer, W. van der Hoek
Epistemic Logic for AI and computerscience
Cambridge university press (1995)
ISBN 0-521-46014

[3] R. Verbrugge
Logics for Artificial Intelligence
Course Notes, University of Groningen

Index