Yale University
Department of Computer Science

A Categorical Approach to Distributed Systems Expressibility and Knowledge

Ruben Michel

YALEU/DCS/TR-669
January 1989

This work was supported in part by the National Science Foundation under grant DCR-8405478 and by the Office of Naval Research under contract N00014-82-K-0154.

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: TR 669	2. GOVT ACCESSION NO. / 3. RECIPIENT'S CATALOG NUMBER
4. TITLE *(and Subtitle)*: A CATAGORICAL APPROACH TO DISTRIBUTED SYSTEMS EXPRESSIBILITY AND KNOWLEDGE	5. TYPE OF REPORT & PERIOD COVERED: Technical Report
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Ruben Michel	8. CONTRACT OR GRANT NUMBER(s): ONR: N00014-82-K-0154
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Dept. of Computer Science, Yale University, 51 Prospect St., New Haven, CT 06520-2158	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS	12. REPORT DATE: January 1989
	13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS *(if different from Controlling Office)*: Office of Naval Research, 800 N. Quincy, Arlington, VA 22217	15. SECURITY CLASS. *(of this report)*: Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

distributed systems
expressibility of protocols
logics of knowledge
cateogry theory

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

Distributed protocols for unreliable networks are difficult both to design and to compare. In this work we introduce a knowledge-based algebraic approach to distributed systems that facilitates these two tasks.

Our approach reveals intrinsic connections between distributed systems, expressibility of protocols, logics of knowledge and category theory, providing a better intuitive and mathematical understanding of distributed systems.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

A Categorical Approach to Distributed Systems Expressibility and Knowledge

Ruben Michel

Abstract

Distributed protocols for unreliable networks are difficult both to design and to compare. In this work we introduce a knowledge-based algebraic approach to distributed systems that facilitates these two tasks.

Assuming a notion of comparable expressibility of faulty processors, we construct a category whose objects are protocols and whose morphisms are based on a known simulation of any protocol by the classical full-view protocol $\mathcal{FV}$. The full-view protocol is, in fact, the universal element in this category.

The categorical view leads to a hierarchy of protocols that has many appealing properties: As one goes up the hierarchy the protocols attain more knowledge and more common knowledge. As one goes down, the messages transmitted need not get longer. Within each isomorphism class in the hierarchy there exists a protocol with least average bit complexity at each round. To any set of protocols there corresponds a unique minimal protocol that can simulate all the elements in the set in parallel. This protocol is the categorical product of the elements in the set. Dually, every set of protocols has a unique maximal element dominated by each member of the set. We introduce universal predicates that provide a language for comparing different protocols. While the very expressive monotone universal predicates separate any protocol from the protocols it strictly dominates, they are unable to separate each infinite decreasing sequence of protocols from its limit protocol, which is guaranteed to exist. Any monotone universal predicate can be implemented with a protocol having the following three properties: First, it is as effective as $\mathcal{FV}$ for that predicate; second, it is minimal among the protocols that are as effective as $\mathcal{FV}$ for that predicate; and third, its average bit complexity is arbitrarily close to optimal.

Finally, we prove lower bounds. We show that, subject to comparable expressibility of faulty processors, $\mathcal{FV}$ attains strictly more common

This work was supported in part by the National Science Foundation under grant DCR-8405478 and by the Office of Naval Research under contract N00014-82-K-0154.

knowledge than any other protocol. We prove that protocols that attain Simultaneous Byzantine Agreement as early as possible require, in the worst case, exponential communication in the number of faulty processors.

To summarize, our approach reveals intrinsic connections between distributed systems, expressibility of protocols, logics of knowledge and category theory, providing us thereby with a better intuitive and mathematical understanding of distributed systems.

1 Introduction

Designing distributed protocols for unreliable networks is complex. Comparing the performance of protocols running in the same unreliable network seems no easier. In this work we introduce a novel algebraic approach to distributed systems that facilitates the two above mentioned tasks.

To exemplify the motivation for this work as well as the disarray currently existing in distributed systems consider the following two problems:

Problem 1. Suppose that you have a standard synchronous distributed system with n processors, and suppose that you have two protocols dictating the behavior of the correct processors. In the first protocol, each processor informs all the other processors at each computational round of the messages it has received in the following order: First, it transmits the message it received from processor one, next the message it received from processor two, and so on up to the message it received from the n^{th} processor. The second protocol is very similar to the first, differing only in that each processor transmits its information in the *opposite* ordering, i.e., it first transmits the message it received from the n^{th} processor, next the message it received from the $(n-1)^{\text{th}}$ processor, and so on up to the message it received from the first processor.

Our question is: In what sense are these two protocols alike?

Problem 2. Consider the following two protocols in a system such as the one above. In the first protocol every correct processor transmits to all other processors at each round all the messages it has received so far. In the second protocol each correct processor transmits to all other processors at each round all the messages it has received from all processors but one.

Our intuition dictates to us that the first protocol is "stronger" than the second. Our questions are: Is the first protocol really stronger than the second? Is it always the case that whatever elementary information is agreed upon amongst the correct processors in a run of the second protocol is also agreed upon in a run of the first?

In this work we present rigorous answers for these problems as well as many

others.

In order to establish a solid basis for this work we begin by introducing our formal distributed model which closely resembles both the classical model defined in [PSL] and the well-accepted model in [LFF]. Our starting point is the well-known *full-view protocol*, $\mathcal{FV}$, or *full-information protocol* introduced in [PSL]. In this protocol, each processor transmits at every round all the information it has seen so far, that is, its external inputs together with all the messages it has both sent and received. A rather remarkable property of this protocol is that it can "simulate" the behavior of any other protocol. Although the existence of these simulations has been noted in the distributed system folklore, their implications have not been well understood. This might have occurred for several reasons: The models for distributed systems were not appropriately defined, the connection between the theory of distributed systems and logics of knowledge was not yet established, etc. In our algebraic setting, we introduce the category of Distributed Systems, DS, in which every object is the set of states of some protocol, and the morphisms are introduced through the simulations mentioned above.

The thesis of this work is that distributed systems, the expressibility of protocols, logics of knowledge and the categorical viewpoint are all intrinsically entangled. We will demonstrate this thesis repeatedly throughout this work.

Our most significant assumption in this work is the notion of comparable expressibility of faulty processors in runs of different protocols. Intuitively, we say that two states of different protocols are *comparable* if they express the same information insofar as their respective protocols allow. In other words, the two states of the different protocols are both images under the corresponding morphisms of a single state of $\mathcal{FV}$. For this definition to make sense we would have to prove that for any protocol $\mathcal{B}$, the morphism from the states of $\mathcal{FV}$ into the states of $\mathcal{B}$ is surjective. Fortunately, under natural assumptions that is the case.

The categorical approach induces a partial order on the set of protocols, giving rise to a *hierarchy* of protocols. This hierarchy has several aesthetic properties as follows. There are strong structural connections among its members. Suppose that protocol $\mathcal{B}$ *dominates* protocol $\mathcal{C}$ in the hierarchy. Categorically, this means that there exists a morphism from $\mathcal{B}$ to $\mathcal{C}$. The state of a processor at a round in the morphic image of a *run* of $\mathcal{B}$ is precisely the morphic image of the *state* of that processor at that round in the run of $\mathcal{B}$. As we go up the hierarchy, more and more facts become known and also more facts become commonly known. As we go down the hierarchy, the messages transmitted by the processors need not become longer. In each isomorphism class of protocols in the hierarchy there exists a protocol with least average bit complexity at each round. In every such class there also exists a protocol in which all the processors transmit simply their

corresponding states. Any arbitrary set of (possibly uncountably many) protocols has a unique *least* protocol that can simulate each of the protocols in the set in parallel or, in other words, there exists a minimal protocol dominating each of the elements of the set. Categorically, the existence of this minimal dominating protocol corresponds precisely to the existence of the categorical product of the members in the set. Dually, we construct for any given set of protocols the maximal protocol dominated by all the elements in the set.

Based on the morphisms, we introduce *universal predicates* that allow us, among other things, to compare the performance of the faulty processors in runs of different protocols. In previous works, runs of different protocols were compared through predicates such as the *basic predicates* of [M], which were not sufficiently expressive to capture the behavior of faulty processors.

We examine separation and density properties of the hierarchy. We prove that there are uncountably many isomorphism classes. We show that even though any protocol can be separated through monotone universal predicates from the protocols it strictly dominates, it is *not* the case that even the very expressive monotone universal predicates can separate a strictly decreasing sequence of protocols from its categorical limit protocol, which is guaranteed to exist. Informally, tasks that can be executed by each of the members in a decreasing sequence of protocols can also be executed by the limit of these protocols, which incidentally is lower in the knowledge hierarchy than any of the elements in the sequence. Any monotone universal predicate can be implemented with a protocol having the following three properties: First, it is as effective as $\mathcal{FV}$ for that predicate; second, it is minimal among the protocols that are as effective as $\mathcal{FV}$ for that predicate; and third, its average bit complexity is arbitrarily close to optimal.

Naturally, when faced with new algebraic structures, one looks for their *universal elements*. Not surprisingly, the universal element in the category of protocols is the full-view protocol.

We complete this work by proving lower bounds on the information that must be conveyed by protocols that attain certain goals and on the message complexity of these protocols. Specifically, we prove that the protocols in the isomorphism class of $\mathcal{FV}$ attain strictly more common knowledge than any other protocol. In other words, for any protocol that is *not* isomorphic to $\mathcal{FV}$, there exist a run of $\mathcal{FV}$ and a monotone universal predicate, such that the predicate *is* common knowledge at some round in that run, but is *not* common knowledge at that same round in the morphic image of the run. Furthermore, we show that protocols that attain Simultaneous Byzantine Agreement as early as possible require, in the worst case, exponentially long messages in the number of faulty processors. Evidently, this result implies that protocols that attain the most common knowledge also require,

in the worst case, exponentially long messages.

To summarize, our approach reveals intrinsic connections between distributed systems, expressibility of protocols, logics of knowledge and category theory, providing us thereby with a better intuitive and mathematical understanding of distributed systems.

This paper is organized as follows: Section 2 introduces our formal distributed model. Section 3 presents a brief overview of the knowledge formalism used in this paper. Section 4 introduces the objects in the category DS and the morphisms from $\mathcal{FV}$ to any other protocol. Sections 5 and 6 examine structural and knowledge properties of these morphisms. Section 7 extends this examination to general morphisms in the category. Section 8 introduces the hierarchy of protocols and investigates several of its properties: The monotonicity of the message length, the best protocol in each isomorphism class, the minimal dominating protocol, the maximal dominated protocol, the universal predicates, the partitions of state trees induced by protocols, density and limit properties and, finally, minimal bit efficient protocols for predicates. Section 9 proves that $\mathcal{FV}$ attains strictly more common knowledge than any other protocol. Section 10 shows that protocols that attain Simultaneous Byzantine Agreement as early as possible require, in the worst case, exponential communication in the number of faulty processors. Finally, section 11 shows that $\mathcal{FV}$ is the universal element in the category of protocols.

2 The Model

In this section we introduce our model of computation, which closely resembles the models in both [PSL] and [LFF]. This formalization is based on the well-established approach used in automata theory for formalizing computing devices such as finite automata, pushdown automata, Turing machines, and so on.

We first introduce the notion of a *protocol.* Intuitively, the protocol completely specifies the actions of each of the correct processors. The protocol $\mathcal{B}$ is in fact a tuple,

$$\mathcal{B} = < \{\delta_p^{\mathcal{B}}\}, \{\mu_{p,q}^{\mathcal{B}}\}, \{\alpha_p^{\mathcal{B}}\}, \{ST_p^{\mathcal{B}}\}, \Sigma, \Gamma > .$$

The set of processors taking part in the protocol, P, is implicit in the definition of the protocol, and its cardinality is denoted by n. The external inputs to each of the processors are drawn from the set Σ and, similarly, the messages are elements in Γ. The set Γ includes two special messages: The *empty message*, denoted by $\emptyset$, standing for no transmission, and the *ungrammatical message*, $\perp$, representing a non-empty ungrammatical message.

Each processor p is a state machine whose states are in $ST_p^{\mathcal{B}}$. The *state transition function*

$$\delta_p^{\mathcal{B}} : ST_p^{\mathcal{B}} \times \Gamma^{n-1} \times \Sigma \rightarrow ST_p^{\mathcal{B}}$$

maps the previous state of processor p along with both the messages and the external input that p receives into its next state. The *message generator* $\mu_{p,q}^{\mathcal{B}} : ST_p^{\mathcal{B}} \rightarrow \Gamma$ generates the message that p transmits to q when it is at a given state. The sets $ST_p^{\mathcal{B}}$ include special states denoted by $\langle p, \emptyset, \ell \rangle$ and $\langle p, \perp, \ell \rangle$, where $\ell > 0$ denotes a round number. These states satisfy $\mu_{p,q}^{\mathcal{B}} \langle p, \emptyset, \ell \rangle = \emptyset$ and $\mu_{p,q}^{\mathcal{B}} \langle p, \perp, \ell \rangle = \perp$. Finally, the *action function* $\alpha_p^{\mathcal{B}}$ from $ST_p^{\mathcal{B}}$ into some other set determines the remaining actions of processor p at each of its states, such as printing, blinking, etc.

A note about notation: For $s_q \in ST_q^{\mathcal{B}}$, $\iota \in \Sigma$ and $p = p_i$, where $i \in \{1 \ldots n\}$, we let $\delta_p^{\mathcal{B}}(s_p \ldots \mu_{q,p}^{\mathcal{B}} s_q \ldots \iota)$ be a shorthand for the rather unappealing expression,

$$\delta_{p_i}^{\mathcal{B}} \left(s_{p_i}, \mu_{p_1,p_i}^{\mathcal{B}} s_{p_1}, \ldots, \mu_{p_{i-1},p_i}^{\mathcal{B}} s_{p_{i-1}}, \mu_{p_{i+1},p_i}^{\mathcal{B}} s_{p_{i+1}}, \ldots, \mu_{p_n,p_i}^{\mathcal{B}} s_{p_n}, \iota \right).$$

We usually assume throughout this work that the processors have access to both their own identification numbers and the global clock, which specifies the round number. Although these assumptions are not always needed, they often simplify both our notation and our treatment.

When there are no faulty processors, the protocol along with the external inputs fully determine the actions of each of the processors. However, faulty processors are usually allowed in models for distributed networks; thus, we formalize a *run* ξ *of the protocol* $\mathcal{B}$, as a tuple

$$\xi = < \mathcal{B}, IN, CA, AD > .$$

Here $\mathcal{B}$ denotes a protocol, and IN denotes the external inputs that the processors receive at each round; in other words, IN is a map

$$IN : P \times \mathcal{N} \rightarrow \Sigma$$

where $\mathcal{N}$ is the set of natural numbers, $\mathcal{N} = \{1, 2, \ldots\}$. The crashing assignment CA specifies the processors that are faulty along with each of the rounds at which they do not follow the protocol. Formally, $CA \subseteq P \times \mathcal{N}$. In order to avoid issues of processor recovery, we require that whenever $\langle p, \ell \rangle \in CA$, also $\langle p, k \rangle \in CA$ for all $k > \ell$. Finally, the adversary, AD, specifies for each faulty processor and recipient the state that the faulty processor uses in its transmission to that recipient. Thus, AD is a map $(p, q, \ell) \rightsquigarrow s$ where $\langle p, \ell \rangle \in CA$, $q \in P$ and $s \in ST_p^{\mathcal{B}}$.

Notice that the crashing assignment naturally partitions the set $P \times \mathcal{N}$ into two: If $\langle p, \ell \rangle \in CA$, then we say that p is *faulty* at ℓ, otherwise, when $\langle p, \ell \rangle \notin CA$, we say that p is *correct* at ℓ.

So far we have not specified how a run determines the actions of the processors, in the same way that a finite automaton, given as a tuple, does not directly specify the *computation* of that automaton. In distributed systems the parallel of the automaton's computation is the *message execution* of the run ξ, which we denote by $MX[\xi]$. The message execution of the run determines the sequence of events occurring in the network. In particular, it specifies when the processors receive the external inputs, how they modify their internal states, what messages they transmit, and what other actions they perform.

We formalize the message execution of a run ξ by examining the chronology of the actions. Initially, that is at time 0, each processor p is in its *unique* initial state $s_{p,0}$. At the beginning of round 1, p performs whatever action is required by evaluating $\alpha_p^{\mathcal{B}} s_{p,0}$. Next, at the middle of the first round, each of the processors receives the external input corresponding to that round, $\iota = IN[\xi](p, 1)$. Based on this external input and its initial state, processor p calculates its next state by evaluating $\delta_p^{\mathcal{B}}(s_{p,0}, \ldots \emptyset \ldots \iota)$, and it enters into this state at the end of round 1. Suppose inductively that we have already constructed the events occurring until the end of round $\ell - 1$. We now describe the actions at ℓ by concentrating first on a correct processor p at ℓ, that is $\langle p, \ell \rangle \notin CA$. Suppose that at the end of $\ell - 1$, processor p is at state $s_{\ell-1}$. At the beginning of round ℓ processor p transmits to each of the other processors q by evaluating $\mu_{p,q}^{\mathcal{B}} s_{\ell-1}$, and it performs the actions pertaining to that round by computing $\alpha^{\mathcal{B}} s_{\ell-1}$. At the middle of round ℓ processor p receives all the messages that were transmitted to it along with its external input, $IN[\xi](p, \ell)$. By the end of round ℓ, p will have concluded the calculation of its next state s_ℓ by evaluating

$$s_\ell = \delta_p^{\mathcal{B}}(s_{\ell-1} \ldots M[\xi](q, p, \ell) \ldots IN[\xi](p, \ell))$$

where $M[\xi](q, p, \ell)$ denotes the message that q transmits to p at ℓ in ξ. This completes the actions of the correct processor p at ℓ in ξ. What if p is faulty at ℓ, that is, $\langle p, \ell \rangle \in CA$? In this case the actions of p at ℓ are completely determined by the adversary: For each processor q, p will send q a message at ℓ as if p were at $\ell - 1$ in state $AD[\xi](p, q, \ell)$. Notice that since we allow the adversary to specify completely the actions taken by the faulty processors, we are in effect considering in this work the Byzantine case.

Finally, how does our model compare with the one in [LFF]? Although these two models seem identical at first sight, there is a basic conceptual difference in their definitions: Whereas in [LFF] the adversary is given by a set of *messages*

transmitted by the faulty processors, in our model the adversary is modelled by the *states* that the faulty processors use for transmitting. It is precisely this difference that allows the development of the categorical approach introduced in this paper.

3 Knowledge Formalism

In this section we introduce the knowledge formalism used in this work. A *predicate φ over $\mathcal{B}$* is a set of runs of $\mathcal{B}$, that is, $\varphi \subseteq \mathcal{R}^{\mathcal{B}}$. The predicate φ is *basic* if it depends only on the crashing assignment and the external inputs. As seen in future sections, basic predicates are useful for comparing the performance of different protocols. A predicate φ *holds at a run* ξ, denoted by $\xi \models \varphi$, if simply $\xi \in \varphi$.

We proceed to introduce the modal operators of knowledge and common knowledge. Let the processor p be correct at round ℓ in the run ξ. Say that p *knows the predicate φ at ℓ in ξ*, which we denote by $\xi \models K_{(p,\ell)}\varphi$, if φ holds at all runs which p cannot distinguish at ℓ from ξ. To make this definition more precise consider the following equivalence relation: Two runs of the *same* protocol are *(p,ℓ)-equivalent*, denoted by $\xi \overset{(p,\ell)}{\approx} \xi'$, if p is correct at ℓ in both, and it has exactly the same state at ℓ in both. Therefore, $\xi \models K_{(p,\ell)}\varphi$ iff p is correct at ℓ in ξ and $\xi' \models \varphi$ for all $\xi' \overset{(p,\ell)}{\approx} \xi$. Now, $\xi \models E_\ell\varphi$ means that every correct processor p at round ℓ in the run ξ knows the predicate φ.

Next, we inductively define the notion of *common knowledge*. Let the predicate $E_\ell^0\varphi$ be just φ, and for $m \geq 0$ let

$$\xi \models E_\ell^{m+1}\varphi \quad \text{iff} \quad \xi \models E_\ell(E_\ell^m\varphi).$$

Now, a predicate φ is *common knowledge* at ℓ in ξ, denoted by $\xi \models C_\ell\varphi$, if for every $m \geq 0$, $\xi \models E_\ell^m\varphi$.

We proceed to introduce a more useful characterization of common knowledge. Two runs of the same protocol are *similar at ℓ*, denoted by $\xi \overset{\ell}{\sim} \xi'$, if there exist a finite sequence of runs of that protocol $\{\xi_k\}$, for $k \in \{1 \ldots (m-1)\}$, and a finite sequence of processors $\{p_{i_k}\}$, where $k \in \{1 \ldots m\}$, such that:

$$\xi \overset{(p_{i_1},\ell)}{\approx} \xi_1 \overset{(p_{i_2},\ell)}{\approx} \ldots \overset{(p_{i_{m-1}},\ell)}{\approx} \xi_{m-1} \overset{(p_{i_m},\ell)}{\approx} \xi'.$$

It is apparent that $\overset{\ell}{\sim}$ is an equivalence relation. The following simple fact establishes a clear connection between knowledge and distributed systems:

Fact 1

$$\xi \models C_\ell\varphi \ \textit{ iff } \ \xi' \models \varphi \ \textit{ for all } \ \xi' \ \textit{ satisfying } \ \xi' \overset{\ell}{\sim} \xi.$$

4 The Morphism

In this section we introduce the category of distributed systems which we denote by DS. For the time being, each of its objects is a set consisting of all the states of some protocol. The more interesting part of this category are its morphisms which we now carefully introduce.

First recall the well-known *full-view protocol*, $\mathcal{FV}$, or *full-information protocol* introduced in [PSL]. In this protocol, each processor transmits at every round all the information it has seen so far, that is, its external inputs together with all the messages it has both sent and received. A rather remarkable property of this protocol is that it can "simulate" the behavior of any other protocol. This simulation provides in turn the morphisms of the category of protocols.

What is the meaning of the statement that the full-view protocol can simulate any other protocol? Well, this simulation is no more than a structure-preserving map from each of the states of $\mathcal{FV}$ into the states of any other protocol $\mathcal{B}$, or, stated differently, it is a morphism from the object corresponding to $\mathcal{FV}$, into the object corresponding to $\mathcal{B}$.

For introducing this morphism we first need a crisp representation for the states of the full-view protocol. In the following definition we show that each such state can be given by an n-ary tree which we call a *state tree.*

Definition 1 *A* state tree s of processor p at round ℓ *is given by the following recursive definition:*
Base case, $\ell = 0$*:* s *is a single node tree labelled* $s^{\mathcal{FV}}_{p,0}$.
Inductive step, $\ell > 0$*:* s *is either a single node tree labelled* $\langle p, \emptyset, \ell \rangle$ *or* $\langle p, \perp, \ell \rangle$*, or it is a tree whose root is labelled* $\langle p, \iota, \ell \rangle$*, with* $\iota \in \Sigma$*, and for each* $q \in P$*,* s *has a unique principal subtree which is a state tree of* q *at* $\ell - 1$.

The semantics of the state tree are straightforward. In the base case, processor p is in its initial state at round 0. In the inductive step, p is at round $\ell > 0$ in either a non-transmitting state, an ungrammatical state, or a state recording its external input at that round and the state trees of all the processors at the previous round. Each such state tree can be recursively transformed into a state of the target protocol $\mathcal{B}$ by applying $\mathcal{B}$ on the information encoded in the state tree. The formal details are given in the following definition.

Definition 2 *The* morphism $h_s : ST^{\mathcal{FV}}_p \rightarrow ST^{\mathcal{B}}_p$ *is given by the following inductive definition on the structure of the state tree* $s \in ST^{\mathcal{FV}}_p$.
Base case, $\ell = 0$*: Here* $s = s^{\mathcal{FV}}_{p,0}$ *and* $h_s s = s^{\mathcal{B}}_{p,0}$.
Inductive step, $\ell > 0$*: If* s *is either* $\langle p, \emptyset, \ell \rangle$ *or* $\langle p, \perp, \ell \rangle$*, then* $h_s s = s$*. Otherwise,*

s is a tree whose root is labelled $\langle p, \iota, \ell \rangle$*, with* $\iota \in \Sigma$*, and for each* $q \in P$*, s has a unique principal subtree,* s_q*, which is a state tree of* q *at round* $\ell - 1$*. Then,*

$$h_s s = \delta_p^{\mathcal{B}}(h_s s_p \ldots \mu_{q,p}^{\mathcal{B}} h_s s_q \ldots h_\iota \iota).$$

where h_ι *is a map* $\Sigma^{\mathcal{FV}} \rightarrow \Sigma^{\mathcal{B}}$.

We proceed to extend the domain of h_s. Let the *relative state from processor p to processor q at round* ℓ *in the run* θ, denoted by $S[\theta](p, \ell - 1 \mid p \xrightarrow{\ell} q)$, be given by

$$S[\theta](p, \ell - 1 \mid p \xrightarrow{\ell} q) = \begin{cases} S[\theta](p, \ell - 1) & \text{if } p \text{ is correct at } \ell \text{ in } \theta \\ S[\theta](p, \ell - 1) & \text{if } p \text{ is faulty at } \ell \text{ in } \theta \text{ and } p = q \\ AD[\theta](p, q, \ell) & \text{if } p \text{ is faulty at } \ell \text{ in } \theta \text{ and } p \neq q \end{cases}$$

Now we may view the adversary AD in a run ρ of $\mathcal{FV}$ as a map

$$(p, q, \ell) \rightsquigarrow S[\rho](p, \ell - 1 \mid p \xrightarrow{\ell} q)$$

where $\langle p, \ell \rangle$ is an element in the crashing assignment CA of ρ and q is a processor in P. The adversary $h_a AD$ is obtained by applying the morphism h_s to each of the states constituting AD. More formally, $h_a AD$ is the map

$$(p, q, \ell) \rightsquigarrow h_s S[\rho](p, \ell - 1 \mid p \xrightarrow{\ell} q)$$

defined for each (p, q, ℓ) in the domain of AD.

The definition of h_a naturally leads to the function $h_r : \mathcal{R}^{\mathcal{FV}} \rightarrow \mathcal{R}^{\mathcal{B}}$, mapping the run ρ of $\mathcal{FV}$ given by

$$\rho = < \mathcal{FV}, IN, CA, AD >$$

into the run

$$h_r \rho = < \mathcal{B}, h_\iota IN, CA, h_a AD >$$

of $\mathcal{B}$. We show in theorem 2 that, subject to some natural conditions, the morphism h_r maps $\mathcal{R}^{\mathcal{FV}}$ *onto* $\mathcal{R}^{\mathcal{B}}$. Consequently, we will sometimes view the category DS as a category in which every object is a set of all *runs* of some protocol, as opposed to a set of all *states* of some protocol, and whose morphisms are the maps h_r. More abstractly and without any danger of being imprecise, we will usually view the objects in DS as just protocols, and the morphism in the category as whichever map h_s, h_a or h_r we happen to have in mind at the moment. Moreover, we will usually blur the distinction between the maps h_s, h_ι, h_a and h_r by referring to them with the generic name h, unless precision or clarity dictate otherwise. In the following sections we examine several structural and knowledge properties of the maps introduced above.

5 Structural Properties of the Morphism

In this section we examine structural properties of the morphisms introduced above. We prove that the operators h and S commute, and that the morphism h is surjective.

5.1 The Commutativity of h and S

In this subsection we show that the operators h and S commute.

Theorem 1

$$h\,S = S\,h$$

Proof: Let ρ be a run of $\mathcal{FV}$ and let $\xi = h\rho$. We want to show that

$$hS[\rho](p, \ell - 1 \mid p \xrightarrow{\ell} q) = S[\xi](p, \ell - 1 \mid p \xrightarrow{\ell} q).$$

The case where p is faulty at round ℓ follows directly from the definition of the run ξ. Consider therefore the case where p is correct at ℓ. The proof proceeds by induction on ℓ. We only consider the following case: Let

$$S[\rho](r, \ell - 2 \mid r \xrightarrow{\ell-1} p \xrightarrow{\ell} q)$$

denote the state that p informs q at ℓ in ρ that r transmitted to p at $\ell - 1$. For notational convenience we will place the arguments of $\delta_p^{\mathcal{B}}$ vertically rather than horizontally in this proof.

$$\begin{aligned}
&hS[\rho](p, \ell - 1 \mid p \xrightarrow{\ell} q) = \\
&= \delta_p^{\mathcal{B}} \begin{pmatrix} hS[\rho](p, \ell - 2 \mid p \xrightarrow{\ell-1} p \xrightarrow{\ell} q) \\ \mu_{r,p}^{\mathcal{B}} hS[\rho](r, \ell - 2 \mid r \xrightarrow{\ell-1} p \xrightarrow{\ell} q) \\ hIN[\rho](p, \ell - 1 \mid p \xrightarrow{\ell} q) \end{pmatrix} \\
&= \delta_p^{\mathcal{B}} \begin{pmatrix} hS[\rho](p, \ell - 2 \mid p \xrightarrow{\ell-1} p) \\ \mu_{r,p}^{\mathcal{B}} hS[\rho](r, \ell - 2 \mid r \xrightarrow{\ell-1} p) \\ hIN[\rho](p, \ell - 1) \end{pmatrix} \\
&= \delta_p^{\mathcal{B}} \begin{pmatrix} S[\xi](p, \ell - 2 \mid p \xrightarrow{\ell-1} p) \\ \mu_{r,p}^{\mathcal{B}} S[\xi](r, \ell - 2 \mid r \xrightarrow{\ell-1} p) \\ IN[\xi](p, \ell - 1) \end{pmatrix} \\
&= S[\xi](p, \ell - 1) \\
&= S[\xi](p, \ell - 1 \mid p \xrightarrow{\ell} q). \qquad \blacksquare
\end{aligned}$$

5.2 The Surjectivity of the Morphism

In this subsection we show that, subject to some natural restrictions on the way protocols are designed, the morphisms in the category DS are surjective. We need a definition in order to formalize the restrictions mentioned above.

Definition 3 *A state $s \in ST_p^{\mathcal{B}}$ is* $\langle p, \ell \rangle$ reachable for the protocol $\mathcal{B}$, *if either* $\ell = 0$ *and* $s = s_{p,0}^{\mathcal{B}}$, *or, for* $\ell > 0$, $s \in \{\langle p, \emptyset, \ell \rangle, \langle p, \perp, \ell \rangle\}$, *or, finally, if there exist for each* $q \in P$ *a* $\langle q, \ell - 1 \rangle$ *reachable state* s_q *for* $\mathcal{B}$ *and an input* $\iota \in \Sigma^{\mathcal{B}}$ *so that*

$$s = \delta_p^{\mathcal{B}}(s_p \ldots \mu_{q,p}^{\mathcal{B}} s_q \ldots \iota).$$

We denote by $ST_{p,\ell}^{\mathcal{B}}$ *the set of all* $\langle p, \ell \rangle$ *reachable states for* $\mathcal{B}$.

The intuition here is that a state which is *not* reachable gives the adversary unnecessary freedom to confuse the correct processors. Moreover, given any protocol, we can always inductively discard its non-reachable states, thereby generating a new protocol with message generators and state transition functions as in the original one, but having only reachable states. Thus, one should design protocols with only reachable states; such protocols are called *concise protocols.* In the following lemma we give a precise characterization of the concise protocols.

Lemma 1 *A protocol* $\mathcal{B}$ *is concise iff for any morphism* h *with surjective* h_ι, $ST_p^{\mathcal{B}} = hST_p^{\mathcal{F}\mathcal{V}}$ *for each processor* $p \in P$.

Proof: $\Rightarrow$ We prove first that for each p,

$$ST_p^{\mathcal{B}} \subseteq hST_p^{\mathcal{F}\mathcal{V}}.$$

Let $s^{\mathcal{B}}$ be a state in $ST_p^{\mathcal{B}}$. Since $\mathcal{B}$ is concise, there exists some round ℓ for which $s^{\mathcal{B}} \in ST_{p,\ell}^{\mathcal{B}}$. We now show that there exist some $s^{\mathcal{F}\mathcal{V}} \in ST_{p,\ell}^{\mathcal{F}\mathcal{V}}$ such that

$$s^{\mathcal{B}} = h_s s^{\mathcal{F}\mathcal{V}},$$

where the map h_ι corresponding to h is an arbitrary surjective function. The proof proceeds by induction on ℓ. We consider only the case where there exist an input $\iota^{\mathcal{B}} \in \Sigma^{\mathcal{B}}$ and states $s_q \in ST_{q,\ell-1}^{\mathcal{B}}$ such that

$$s^{\mathcal{B}} = \delta_p^{\mathcal{B}}(s_p \ldots \mu_{q,p}^{\mathcal{B}} s_q \ldots \iota^{\mathcal{B}}).$$

By the inductive hypothesis there exists for each $q \in P$ some state $s_q^{\mathcal{F}\mathcal{V}} \in ST_{q,\ell-1}^{\mathcal{F}\mathcal{V}}$ such that

$$s_q^{\mathcal{B}} = h_s s_q^{\mathcal{F}\mathcal{V}},$$

and by the surjectivity of h_ι there exists some input $\iota^{\mathcal{FV}} \in \Sigma^{\mathcal{FV}}$ such that

$$\iota^{\mathcal{B}} = h_\iota \iota^{\mathcal{FV}}.$$

Let $s^{\mathcal{FV}}$ be the state of $\mathcal{FV}$ with root labelled $\langle p, \iota^{\mathcal{FV}}, \ell \rangle$, whose principal subtrees are $s_q^{\mathcal{FV}}$, for each $q \in P$. By definition, $s^{\mathcal{FV}} \in ST_{p,\ell}^{\mathcal{FV}}$, and furthermore

$$s^{\mathcal{B}} = h_s s^{\mathcal{FV}}.$$

The containment $ST_p^{\mathcal{B}} \supseteq hST_p^{\mathcal{FV}}$ follows similarly.
$\Leftarrow$ We now prove that if $ST_p^{\mathcal{B}} = hST_p^{\mathcal{FV}}$ for each p, then $\mathcal{B}$ is concise. Fix a processor p, and let $s^{\mathcal{B}} \in ST_p^{\mathcal{B}}$. For demonstrating that $\mathcal{B}$ is concise we must show that $s^{\mathcal{B}} \in ST_{p,\ell}^{\mathcal{B}}$ for some ℓ.

First, since $ST_p^{\mathcal{B}} = hST_p^{\mathcal{FV}}$, there exists some $s^{\mathcal{FV}} \in ST_p^{\mathcal{FV}}$ such that

$$s^{\mathcal{B}} = hs^{\mathcal{FV}}.$$

We prove now by induction on the height ℓ of the tree $s^{\mathcal{FV}}$ that $s^{\mathcal{B}} \in ST_{p,\ell}^{\mathcal{B}}$. We only consider the following case: Let $s_q^{\mathcal{FV}}$ be the principal subtree of $s^{\mathcal{FV}}$ corresponding to q. Notice that $s_q^{\mathcal{FV}} \in ST_{q,\ell-1}^{\mathcal{FV}}$; therefore, by the inductive hypothesis, the state $s_q^{\mathcal{B}} = hs_q^{\mathcal{FV}}$ satisfies $s_q^{\mathcal{B}} \in ST_{q,\ell-1}^{\mathcal{B}}$. Let $\iota^{\mathcal{FV}}$ denote the external input of p at round ℓ in $s^{\mathcal{FV}}$, and let $\iota^{\mathcal{B}} = h\iota^{\mathcal{FV}}$. By the definition of h we have

$$s^{\mathcal{B}} = \delta_p^{\mathcal{B}}(s_p^{\mathcal{B}} \ldots \mu_{q,p}^{\mathcal{B}} s_q^{\mathcal{B}} \ldots \iota^{\mathcal{B}})$$

showing that $s^{\mathcal{B}} \in ST_{p,\ell}^{\mathcal{B}}$. ∎

We now state the main theorem of this subsection, which asserts that the morphism h_r is surjective if the corresponding map h_ι is surjective and $\mathcal{B}$ is concise.

Theorem 2 *Assume that the protocol $\mathcal{B}$ is concise, and let h_r be a morphism with surjective h_ι. Then h_r is also surjective.*

6 Knowledge Properties of the Morphism

We begin this section with the following key lemma which asserts that (p,ℓ)-equivalence is preserved under morphisms, i.e., if two runs are (p,ℓ)-equivalent, then their morphic images are also (p,ℓ)-equivalent.

Lemma 2 *If $\rho' \overset{(p,\ell)}{\approx} \rho$, then $h\rho' \overset{(p,\ell)}{\approx} h\rho$.*

Proof: Assume that $\rho' \overset{(p,\ell)}{\approx} \rho$. Then $S[\rho'](p,\ell) = S[\rho](p,\ell)$ and therefore by theorem 1, $S[h\rho'](p,\ell) = S[h\rho](p,\ell)$. Hence, $h\rho' \overset{(p,\ell)}{\approx} h\rho$. ∎

Equipped with this lemma we proceed to prove several knowledge properties of the morphism h. We first assert that, under natural assumptions, whatever is known or commonly known in the morphic image of a run of $\mathcal{FV}$ is also known or commonly known in the run itself.

Theorem 3

1. *Let φ be a predicate over $\mathcal{B}$, and let the predicate $h^{-1}\varphi$ be given by*

$$h^{-1}\varphi = \{\rho \in \mathcal{R}^{\mathcal{FV}} \mid h\rho \models \varphi\}.$$

Then,

   (a) *If $h\rho \models K_{(p,\ell)}\varphi$ then $\rho \models K_{(p,\ell)}h^{-1}\varphi$.*

   (b) *If $h\rho \models C_\ell\varphi$ then $\rho \models C_\ell h^{-1}\varphi$.*

2. *Assume that the map h_ι corresponding to h_r is injective, and let φ be a* basic *predicate over $\mathcal{FV}$. Then,*

   (a) *If $h\rho \models K_{(p,\ell)}h\varphi$ then $\rho \models K_{(p,\ell)}\varphi$.*

   (b) *If $h\rho \models C_\ell h\varphi$ then $\rho \models C_\ell\varphi$.*

Proof:

1.a Let the run ρ' satisfy $\rho' \overset{(p,\ell)}{\approx} \rho$. By lemma 2, $h\rho' \overset{(p,\ell)}{\approx} h\rho$. But $h\rho' \models \varphi$ since $h\rho \models K_{(p,\ell)}\varphi$. Thus, $\rho' \models h^{-1}\varphi$, and therefore $\rho \models K_{(p,\ell)}h^{-1}\varphi$.

1.b Similar to 1.a.

2.a Assume as before that the run ρ' satisfies $\rho' \overset{(p,\ell)}{\approx} \rho$. Hence, $h\rho' \overset{(p,\ell)}{\approx} h\rho$. Again $h\rho \models K_{(p,\ell)}h\varphi$ implies $h\rho' \models h\varphi$. Thus, there exists some $\rho'' \models \varphi$ so that $h\rho' = h\rho''$. It follows that $CA[\rho'] = CA[\rho'']$ and $h_\iota IN[\rho'] = h_\iota IN[\rho'']$. Recall now that h_ι is injective, thus $IN[\rho'] = IN[\rho'']$. Finally, since φ is a basic predicate and since ρ' and ρ'' have identical crashing assignments and external inputs, we have $\rho' \models \varphi$ and therefore $\rho \models K_{(p,\ell)}\varphi$.

2.b Similar to 2.a. ∎

We examine now the effect of the morphisms on *basic* predicates. To this end, we view the morphism h_r as a map $\mathcal{P}(\mathcal{R}^{\mathcal{FV}}) \rightarrow \mathcal{P}(\mathcal{R}^{\mathcal{B}})$ given by

$$\varphi^{\mathcal{FV}} \rightsquigarrow \{h_r\rho \mid \rho \models \varphi^{\mathcal{FV}}\},$$

where $\mathcal{P}(\mathcal{R}^{\mathcal{B}})$ denotes the power set of $\mathcal{R}^{\mathcal{B}}$. An interesting property of h_r when viewed in this way is that, subject to some natural conditions, h_r maps basic predicates over $\mathcal{FV}$ *onto* basic predicates over $\mathcal{B}$. We prove this fact using the following two lemmas:

Lemma 3 *Let the protocol $\mathcal{B}$ be concise and assume that the map h_t corresponding to h_r is bijective. Then h_r maps basic predicates over the full-view protocol* into *basic predicates over $\mathcal{B}$.*

Proof: Let $\varphi^{\mathcal{FV}}$ be a basic predicate over the full-view protocol. Let the run ρ of $\mathcal{FV}$, given by

$$\rho =< \mathcal{FV}, IN, CA, AD >$$

satisfy $\rho \models \varphi^{\mathcal{FV}}$. Let ξ' be the run of $\mathcal{B}$ given by

$$\xi' =< \mathcal{B}, hIN, CA, AD' > .$$

Recall that by assumption the protocol $\mathcal{B}$ is concise and h_t is surjective. Therefore, by theorem 2, there exists a run ρ' of $\mathcal{FV}$ such that $\xi' = h\rho'$. Thus,

$$CA[\rho'] = CA[\xi'] = CA[\rho]$$

and also

$$h_t IN[\rho'] = IN[\xi'] = h_t IN[\rho].$$

But h_t is also injective, hence

$$IN[\rho'] = IN[\rho].$$

Recall also that $\varphi^{\mathcal{FV}}$ is a basic predicate and $\rho \models \varphi^{\mathcal{FV}}$. Hence $\rho' \models \varphi^{\mathcal{FV}}$ and therefore $\xi' = h\rho' \models h\varphi^{\mathcal{FV}}$. ∎

Lemma 4 *Let $\varphi^{\mathcal{B}}$ be a basic predicate over $\mathcal{B}$. Then the predicate $h^{-1}\varphi^{\mathcal{B}}$ given by*

$$h^{-1}\varphi^{\mathcal{B}} = \{\rho \in \mathcal{R}^{\mathcal{FV}} \mid h\rho \models \varphi^{\mathcal{B}}\}$$

is a basic predicate over $\mathcal{FV}$.

We now summarize the two lemmas given above in the following theorem which asserts that the map h_r when viewed as a map from predicates into predicates, maps basic predicates *onto* basic predicates.

Theorem 4 *Let the protocol $\mathcal{B}$ be concise and assume that the map h_t corresponding to h_r is bijective. Then h_r maps basic predicates* onto *basic predicates.*

7 Morphisms Among General Protocols

We begin this section by introducing a relation between protocols, denoted by $\succeq$, which is based on the morphisms from the full-view protocol into any other protocol introduced in previous sections. This relation leads in turn to morphisms from protocols possibly different from the full-view protocol into other protocols, thereby generating the category of distributed systems, DS.

In this section $\mathcal{B}$ and $\mathcal{C}$ denote two concise protocols in n participants. We introduce now the relation $\succeq$ and the morphism $h\langle \mathcal{B}, \mathcal{C} \rangle$.

Definition 4 *Let the function h_ι map $\Sigma^{\mathcal{B}} \to \Sigma^{\mathcal{C}}$. Say that $\mathcal{B}$* dominates *$\mathcal{C}$ with respect to h_ι, denoted by $\mathcal{B} \succeq_{h_\iota} \mathcal{C}$, if the map $h_r\langle \mathcal{B}, \mathcal{C} \rangle : \mathcal{R}^{\mathcal{B}} \to \mathcal{R}^{\mathcal{C}}$ given by*

$$h_r\langle \mathcal{B}, \mathcal{C} \rangle = h_r\langle \mathcal{FV}, \mathcal{C} \rangle h_\iota (h_r\langle \mathcal{FV}, \mathcal{B} \rangle)^{-1}$$

is well-defined.

Notice that a necessary and sufficient condition for the map $h_r\langle \mathcal{B}, \mathcal{C} \rangle$ to be well-defined is that the map

$$h_s\langle \mathcal{B}, \mathcal{C} \rangle = h_s\langle \mathcal{FV}, \mathcal{C} \rangle h_\iota (h_s\langle \mathcal{FV}, \mathcal{B} \rangle)^{-1}$$

be also well-defined.

We proceed to show that the morphism $h\langle \mathcal{B}, \mathcal{C} \rangle$ inherits many of the properties of the morphism h from $\mathcal{FV}$ to other protocols. Hereafter ρ, ξ and η denote runs of $\mathcal{FV}$, $\mathcal{B}$ and $\mathcal{C}$ respectively. For simplicity of notation we denote the morphisms $h\langle \mathcal{FV}, \mathcal{B} \rangle$ and $h\langle \mathcal{FV}, \mathcal{C} \rangle$ by $h^{\mathcal{B}}$ and $h^{\mathcal{C}}$ respectively. We now prove that, whenever the morphism $h\langle \mathcal{B}, \mathcal{C} \rangle$ exists, the operators $h\langle \mathcal{B}, \mathcal{C} \rangle$ and S commute.

Theorem 5 *Assume that $\mathcal{B} \succeq_{h_\iota} \mathcal{C}$. Then*

$$h\langle \mathcal{B}, \mathcal{C} \rangle \; S = S \; h\langle \mathcal{B}, \mathcal{C} \rangle.$$

Proof: Let ξ be an arbitrary run of $\mathcal{B}$. We want to show that

$$h\langle \mathcal{B}, \mathcal{C} \rangle S[\xi](p, \ell - 1 \mid p \xrightarrow{\ell} q) = S[h\langle \mathcal{B}, \mathcal{C} \rangle \xi](p, \ell - 1 \mid p \xrightarrow{\ell} q).$$

By construction, this assertion is true when p is faulty at ℓ. Consider now the case where p is correct at ℓ. Hence,

$$S[\xi](p, \ell - 1 \mid p \xrightarrow{\ell} q) = S[\xi](p, \ell - 1).$$

Notice that the sets $h\langle \mathcal{B}, \mathcal{C}\rangle S[\xi](p, \ell - 1)$ and $h\langle \mathcal{B}, \mathcal{C}\rangle \xi$ are both singleton sets, thus, we feel free to identify each of these sets with their corresponding element. Let $s = S[h\langle \mathcal{B}, \mathcal{C}\rangle \xi](p, \ell - 1)$ and let ρ be a run of $\mathcal{FV}$ such that $h^{\mathcal{B}}\rho = \xi$. By theorem 1, $h^{\mathcal{B}} S[\rho](p, \ell - 1) = S[\xi](p, \ell - 1)$. Thus,

$$S[\rho](p, \ell - 1) \in (h^{\mathcal{B}})^{-1} S[\xi](p, \ell - 1).$$

Applying theorem 1 again,

$$s = S[h^{\mathcal{C}} h_\iota \rho](p, \ell - 1) = h^{\mathcal{C}} h_\iota S[\rho](p, \ell - 1).$$

Thus, we have $s \in h^{\mathcal{C}} h_\iota (h^{\mathcal{B}})^{-1} S[\xi](p, \ell - 1)$, and therefore

$$s = h\langle \mathcal{B}, \mathcal{C}\rangle S[\xi](p, \ell - 1). \qquad \blacksquare$$

In section 4 we used the equality

$$h^{\mathcal{B}} s = \delta_p^{\mathcal{B}}(h^{\mathcal{B}} s_p \ldots \mu_{q,p}^{\mathcal{B}} h^{\mathcal{B}} s_q \ldots h_\iota \iota)$$

for introducing the morphism $h^{\mathcal{B}}$ from $\mathcal{FV}$ to $\mathcal{B}$. We now show that this equality extends to arbitrary morphisms $h\langle \mathcal{B}, \mathcal{C}\rangle$, and moreover that it characterizes them.

Theorem 6 *Assume that $s_q^{\mathcal{B}} \in ST_{q,\ell}^{\mathcal{B}}$ for all processors $q \in P$, and let h stand for the morphism $h\langle \mathcal{B}, \mathcal{C}\rangle = h^{\mathcal{C}} h_\iota (h^{\mathcal{B}})^{-1}$. Then,*

$$h\delta_p^{\mathcal{B}}(s_p^{\mathcal{B}} \ldots \mu_{q,p}^{\mathcal{B}} s_q^{\mathcal{B}} \ldots \iota) = \delta_p^{\mathcal{C}}(h s_p^{\mathcal{B}} \ldots \mu_{q,p}^{\mathcal{C}} h s_q^{\mathcal{B}} \ldots h_\iota \iota).$$

Furthermore, any map $h' : ST_p^{\mathcal{B}} \to ST_p^{\mathcal{C}}$ satisfying the following equalities

$$\begin{aligned} h'\delta_p^{\mathcal{B}}(s_p^{\mathcal{B}} \ldots \mu_{q,p}^{\mathcal{B}} s_q^{\mathcal{B}} \ldots \iota) &= \delta_p^{\mathcal{C}}(h' s_p^{\mathcal{B}} \ldots \mu_{q,p}^{\mathcal{C}} h' s_q^{\mathcal{B}} \ldots h_\iota \iota) \\ h' s &= s \quad \text{for } s \in \{\langle p, \emptyset, \ell\rangle, \langle p, \perp, \ell\rangle\} \\ h' s_{p,0}^{\mathcal{B}} &= s_{p,0}^{\mathcal{C}} \end{aligned}$$

is identical to h.

Proof: By lemma 1 there are states $s_q^{\mathcal{FV}} \in ST_{q,\ell}^{\mathcal{FV}}$ such that $s_q^{\mathcal{B}} = h s_q^{\mathcal{FV}}$. Let $s^{\mathcal{FV}}$ be the state tree whose root is labelled $\langle p, \iota, \ell + 1\rangle$ and whose q^{th} principal subtree is $s_q^{\mathcal{FV}}$. Let $s^{\mathcal{B}} = h^{\mathcal{B}} s^{\mathcal{FV}}$. Then, on one hand,

$$s^{\mathcal{B}} = \delta_p^{\mathcal{B}}(s_p^{\mathcal{B}} \ldots \mu_{q,p}^{\mathcal{B}} s_q^{\mathcal{B}} \ldots \iota)$$

and, on the other hand,

$$\begin{aligned} hs^{\mathcal{B}} &= h^{\mathcal{C}} h_\iota s^{\mathcal{F}\mathcal{V}} \\ &= \delta_p^{\mathcal{C}}(h^{\mathcal{C}} h_\iota s_p^{\mathcal{F}\mathcal{V}} \ldots \mu_{q,p}^{\mathcal{C}} h^{\mathcal{C}} h_\iota s_q^{\mathcal{F}\mathcal{V}} \ldots h_\iota \iota) \\ &= \delta_p^{\mathcal{C}}(hs_p^{\mathcal{B}} \ldots \mu_{q,p}^{\mathcal{C}} hs_q^{\mathcal{B}} \ldots h_\iota \iota). \end{aligned}$$

The second part of the theorem holds since h and h' satisfy both the same recursive equation and the same initial conditions. ∎

Next, we assert that, subject to some natural conditions, the morphisms $h\langle \mathcal{B}, \mathcal{C} \rangle$ introduced above are surjective.

Theorem 7 *Assume that $\mathcal{B} \succeq_{h_\iota} \mathcal{C}$, that is, the morphism $h\langle \mathcal{B}, \mathcal{C} \rangle = h^{\mathcal{C}} h_\iota (h^{\mathcal{B}})^{-1}$ is well-defined. Then, $h\langle \mathcal{B}, \mathcal{C} \rangle$ is surjective if h_ι is.*

Finally, we examine knowledge properties of the morphism $h\langle \mathcal{B}, \mathcal{C} \rangle$. We show that lemma 2 in section 6 which asserts that (p, ℓ)-equivalence is preserved under morphism holds when the morphism $h\langle \mathcal{B}, \mathcal{C} \rangle$ replaces the morphism $h^{\mathcal{B}}$.

Lemma 5 *If $\xi' \overset{(p,\ell)}{\approx} \xi$, then $h\langle \mathcal{B}, \mathcal{C} \rangle \xi' \overset{(p,\ell)}{\approx} h\langle \mathcal{B}, \mathcal{C} \rangle \xi$.*

Applying this lemma we may now extend theorem 3 using the more general morphism $h\langle \mathcal{B}, \mathcal{C} \rangle$.

Theorem 8 *Let h denote the morphism $h\langle \mathcal{B}, \mathcal{C} \rangle = h^{\mathcal{C}} h_\iota (h^{\mathcal{B}})^{-1}$.*

1. *Let φ be a predicate over $\mathcal{C}$. Then,*
   (a) *If $h\xi \models K_{(p,\ell)}\varphi$ then $\xi \models K_{(p,\ell)} h^{-1}\varphi$.*
   (b) *If $h\xi \models C_\ell \varphi$ then $\xi \models C_\ell h^{-1}\varphi$.*
2. *Assume that the map h_ι is injective, and let φ be a* basic *predicate over $\mathcal{B}$. Then,*
   (a) *If $h\xi \models K_{(p,\ell)} h\varphi$ then $\xi \models K_{(p,\ell)}\varphi$.*
   (b) *If $h\xi \models C_\ell h\varphi$ then $\xi \models C_\ell \varphi$.*

8 The Protocol Hierarchy

In this section we show that, subject to comparable expressibility of adversaries, there is a natural and aesthetic hierarchy of protocols in distributed systems.

We restrict our attention to concise protocols over the same input set Σ, and we consider the relation $\mathcal{B} \succeq \mathcal{C}$ which stands for $\mathcal{B} \succeq_I \mathcal{C}$, where I denotes the identity on Σ. The notion of comparability needs some clarification: Intuitively, two states of different protocols are *comparable* if they express the same information insofar as their respective protocols allow. Thus, the states $s^{\mathcal{B}}$ and $s^{\mathcal{C}}$ of the protocols $\mathcal{B}$ and $\mathcal{C}$ respectively are *expressibly comparable* if for some state $s^{\mathcal{F}\mathcal{V}}$ of $\mathcal{F}\mathcal{V}$ both

$$s^{\mathcal{B}} = h^{\mathcal{B}} s^{\mathcal{F}\mathcal{V}} \quad \text{and} \quad s^{\mathcal{C}} = h^{\mathcal{C}} s^{\mathcal{F}\mathcal{V}}.$$

Comparability of adversaries and of runs is defined similarly. For example, the runs ξ and η of the protocols $\mathcal{B}$ and $\mathcal{C}$ respectively are *comparable* if for some run ρ of $\mathcal{F}\mathcal{V}$ both $\xi = h^{\mathcal{B}}\rho$ and $\eta = h^{\mathcal{C}}\rho$. Notice that many of the results developed in the previous sections relied on this notion of comparability, and so will the forthcoming assertions.

We begin by introducing a partial order on the set of protocols. First, we show that the relation $\succeq$ is transitive and therefore that it induces a preorder on the category of protocols.

Lemma 6 *The relation $\succeq$ is transitive. Furthermore, if $\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$ are three protocols such that $\mathcal{A} \succeq \mathcal{B}$ and $\mathcal{B} \succeq \mathcal{C}$, then*

$$h\langle \mathcal{B}, \mathcal{C} \rangle \; h\langle \mathcal{A}, \mathcal{B} \rangle = h\langle \mathcal{A}, \mathcal{C} \rangle.$$

The standard succeeding step is to transform this preorder into a partial order by identifying elements satisfying both $\succeq$ and $\preceq$. Thus, say that the protocols $\mathcal{B}$ and $\mathcal{C}$ are *isomorphic* if both $\mathcal{B} \succeq \mathcal{C}$ and $\mathcal{B} \preceq \mathcal{C}$.

As expected, it follows from theorem 8 that if two protocols are isomorphic, then a processor knows a predicate at a round in a run iff that same processor knows the morphic image of that predicate in the morphic image of the run. A similar statement holds also for common knowledge. Furthermore, theorem 8 indicates how knowledge and common knowledge evolve as we go up and down the hierarchy. In future sections we will examine this issue more closely.

8.1 The Monotonicity of the Message Length

We now show that as we go down the protocol hierarchy, the messages transmitted by the processors need not get longer.

Theorem 9 *Assume that $\mathcal{B} \succeq \mathcal{C}$ and let $h = h\langle \mathcal{B}, \mathcal{C} \rangle$. Then there exists a protocol $\mathcal{D}$ isomorphic to $\mathcal{C}$ such that for every state $s^{\mathcal{B}} \in ST_p^{\mathcal{B}}$ and processor q,*

$$|\mu_{p,q}^{\mathcal{D}} h s^{\mathcal{B}}| \leq |\mu_{p,q}^{\mathcal{B}} s^{\mathcal{B}}|.$$

Proof: The only difficulty in this theorem is the construction of the protocol $\mathcal{D}$. Let the set of states in $\mathcal{D}$ be precisely as in $\mathcal{C}$. Next, define the message generators. Let $s^{\mathcal{C}}$ be an arbitrary state of processor p in $\mathcal{C}$. Then, $\mu_{p,q}^{\mathcal{D}} s^{\mathcal{C}}$ is a shortest message of the form $\mu_{p,q}^{\mathcal{B}} s^{\mathcal{B}}$, for states $s^{\mathcal{B}} \in ST_p^{\mathcal{B}}$ satisfying $s^{\mathcal{C}} = h s^{\mathcal{B}}$. Finally, the state transition functions are given by,

$$\delta_p^{\mathcal{D}}(s_p^{\mathcal{C}} \ldots \mu_{q,p}^{\mathcal{D}} s_q^{\mathcal{C}} \ldots \iota) = \delta_p^{\mathcal{C}}(s_p^{\mathcal{C}} \ldots \mu_{q,p}^{\mathcal{C}} s_q^{\mathcal{C}} \ldots \iota).$$

This definition guarantees by theorem 6 that the protocols $\mathcal{D}$ and $\mathcal{C}$ are isomorphic. We proceed to show that

Lemma 7 *The protocol $\mathcal{D}$ is well-defined.*

Proof: By theorem 7, the message generators are well-defined. We now prove that also the state transition functions are well-defined. Assume that for every $q \neq p$

$$\mu_{q,p}^{\mathcal{D}} s_q^{\mathcal{C}} = \mu_{q,p}^{\mathcal{D}} s_q'^{\mathcal{C}}.$$

We have to show that for every state $s_p^{\mathcal{C}}$ and $\iota \in \Sigma$,

$$\delta_p^{\mathcal{D}}(s_p^{\mathcal{C}} \ldots \mu_{q,p}^{\mathcal{D}} s_q^{\mathcal{C}} \ldots \iota) = \delta_p^{\mathcal{D}}(s_p^{\mathcal{C}} \ldots \mu_{q,p}^{\mathcal{D}} s_q'^{\mathcal{C}} \ldots \iota).$$

First, by theorem 7, there are states $s_q^{\mathcal{B}}$ and $s_q'^{\mathcal{B}}$ in $ST_q^{\mathcal{B}}$ such that both $s_q^{\mathcal{C}} = h s_q^{\mathcal{B}}$ and $s_q'^{\mathcal{C}} = h s_q'^{\mathcal{B}}$, and, furthermore, for $q \neq p$,

$$\mu_{q,p}^{\mathcal{D}} s_q^{\mathcal{C}} = \mu_{q,p}^{\mathcal{B}} s_q^{\mathcal{B}} \quad \text{and} \quad \mu_{q,p}^{\mathcal{D}} s_q'^{\mathcal{C}} = \mu_{q,p}^{\mathcal{B}} s_q'^{\mathcal{B}}.$$

Thus, we have

$$\delta_p^{\mathcal{B}}(s_p^{\mathcal{B}} \ldots \mu_{q,p}^{\mathcal{B}} s_q^{\mathcal{B}} \ldots \iota) = \delta_p^{\mathcal{B}}(s_p^{\mathcal{B}} \ldots \mu_{q,p}^{\mathcal{B}} s_q'^{\mathcal{B}} \ldots \iota)$$

and furthermore, applying the morphism h to both sides and by theorem 6,

$$\delta_p^{\mathcal{C}}(s_p^{\mathcal{C}} \ldots \mu_{q,p}^{\mathcal{C}} s_q^{\mathcal{C}} \ldots \iota) = \delta_p^{\mathcal{C}}(s_p^{\mathcal{C}} \ldots \mu_{q,p}^{\mathcal{C}} s_q'^{\mathcal{C}} \ldots \iota).$$

Therefore

$$\delta_p^{\mathcal{D}}(s_p^{\mathcal{C}} \ldots \mu_{q,p}^{\mathcal{D}} s_q^{\mathcal{C}} \ldots \iota) = \delta_p^{\mathcal{D}}(s_p^{\mathcal{C}} \ldots \mu_{q,p}^{\mathcal{D}} s_q'^{\mathcal{C}} \ldots \iota). \qquad \blacksquare$$

We complete the proof of the theorem by noticing that, by definition,

$$|\mu_{p,q}^{\mathcal{D}} h s^{\mathcal{B}}| \leq |\mu_{p,q}^{\mathcal{B}} s^{\mathcal{B}}|. \qquad \blacksquare$$

8.2 The Best Protocol in the Isomorphism Class

In this section we constructively show that in each isomorphism class of protocols there exists a protocol whose average bit complexity *at each round* is the smallest in the isomorphism class. For this assertion to make sense, we assume that all the protocols in the isomorphism class have exactly the same message set Γ.

We assume throughout this section that each processor continuously records the round number and that the set of external inputs Σ is finite. We also assume without loss of generality that each message M in $\Gamma \setminus \{\emptyset, \perp\}$ has a *length* $|M|$, which is an integer greater than or equal to 0, and that the messages in Γ are indexed according to their lengths, that is,

$$|M_1| \geq |M_2| \geq |M_3| \geq \cdots$$

where $M_i \in \Gamma \setminus \{\emptyset, \perp\}$.

Notice that there might exist states $s \in ST^{\mathcal{B}}_{r,k}$ for which there exists no run $\xi \in \mathcal{R}^{\mathcal{B}}$ such that

$$S[\xi](r,k) = s.$$

Since we will only count the number of bits transmitted by the correct processors, we define the set of *achievable states of processor r at round k in protocol $\mathcal{B}$*, $AST^{\mathcal{B}}_{r,k}$, to be the set of states $s \in ST^{\mathcal{B}}_{r,k}$ for which there *exists* a run $\xi \in \mathcal{R}^{\mathcal{B}}$ such that

$$S[\xi](r,k) = s.$$

Pick an arbitrary isomorphism class of protocols, and let $\mathcal{B}$ be a protocol in that class. Fix a pair of distinct processors r and p. We first set a lower bound on the cardinality of the set

$$\{\mu^{\mathcal{C}}_{r,p} s \mid s \in AST^{\mathcal{C}}_{r,p}\}$$

where $\mathcal{C}$ is an arbitrary protocol isomorphic to $\mathcal{B}$.

Consider the following equivalence relation on the states in $AST^{\mathcal{B}}_{r,k}$. For states s_r and s'_r in $AST^{\mathcal{B}}_{r,k}$, say that $s_r \overset{\mathcal{B},p}{\simeq} s'_r$ iff for all $q \neq r$, $s_q \in ST^{\mathcal{B}}_{q,k}$, and $\iota \in \Sigma$,

$$\delta^{\mathcal{B}}_p(s_p \ldots \mu^{\mathcal{B}}_{q,p} s_q \ldots \mu^{\mathcal{B}}_{r,p} s_r \ldots \iota) = \delta^{\mathcal{B}}_p(s_p \ldots \mu^{\mathcal{B}}_{q,p} s_q \ldots \mu^{\mathcal{B}}_{r,p} s'_r \ldots \iota).$$

Thus, if the states s_r and s'_r satisfy $s_r \overset{\mathcal{B},p}{\simeq} s'_r$, we may safely construct a protocol $\mathcal{C}$ isomorphic to $\mathcal{B}$ satisfying

$$\mu^{\mathcal{C}}_{r,p} s_r = \mu^{\mathcal{C}}_{r,p} s'_r.$$

Conversely, suppose that $s_r \overset{\mathcal{B},p}{\not\simeq} s'_r$, but for some protocol $\mathcal{C}$ isomorphic to $\mathcal{B}$

$$\mu^{\mathcal{C}}_{r,p} s_r = \mu^{\mathcal{C}}_{r,p} s'_r.$$

Then, on one hand, for all $q \neq r$, $s_q \in ST^{\mathcal{C}}_{q,k}$ and $\iota \in \Sigma$,

$$\delta^{\mathcal{C}}_p(s_p \ldots \mu^{\mathcal{C}}_{q,p} s_q \ldots \mu^{\mathcal{C}}_{r,p} s_r \ldots \iota) = \delta^{\mathcal{C}}_p(s_p \ldots \mu^{\mathcal{C}}_{q,p} s_q \ldots \mu^{\mathcal{C}}_{r,p} s'_r \ldots \iota),$$

and, on the other hand, since $s_r \overset{\mathcal{B},p}{\not\simeq} s'_r$, for some $s_q \in ST^{\mathcal{B}}_{q,k}$, $q \neq r$, and $\iota \in \Sigma$,

$$\delta^{\mathcal{B}}_p(s_p \ldots \mu^{\mathcal{B}}_{q,p} s_q \ldots \mu^{\mathcal{B}}_{r,p} s_r \ldots \iota) \neq \delta^{\mathcal{B}}_p(s_p \ldots \mu^{\mathcal{B}}_{q,p} s_q \ldots \mu^{\mathcal{B}}_{r,p} s'_r \ldots \iota).$$

But $\mathcal{B}$ is isomorphic to $\mathcal{C}$, thus the morphism $h\langle \mathcal{C}, \mathcal{B} \rangle$ exists; hence, the equality above together with the inequality below it contradict theorem 6. We summarize this discussion in the following lemma,

Lemma 8 *For every protocol $\mathcal{B}$, there exists a protocol $\mathcal{C}$ isomorphic to $\mathcal{B}$ such that*

$$s_r \overset{\mathcal{B},p}{\simeq} s'_r \quad \textit{implies} \quad \mu^{\mathcal{C}}_{r,p} s_r = \mu^{\mathcal{C}}_{r,p} s'_r.$$

Furthermore, for any protocol $\mathcal{C}$ isomorphic to $\mathcal{B}$

$$\mu^{\mathcal{C}}_{r,p} s_r = \mu^{\mathcal{C}}_{r,p} s'_r \quad \textit{implies} \quad s_r \overset{\mathcal{B},p}{\simeq} s'_r.$$

We now introduce notation for defining the notion of average bit complexity of a protocol. Consider a run ξ of protocol $\mathcal{B}$. Denote by $\psi(\xi, \ell)$ the number of bits transmitted by the correct processors up to and including round ℓ in ξ. For notational convenience we denote the state of a faulty processor r at round k in a run ξ by $\top = S[\xi](r,k)$, and we let

$$|\mu^{\mathcal{B}}_{r,p} \top| = 0.$$

With this notation we have

$$\psi(\xi, \ell) = \sum_{r,p,1<k<\ell} |\mu^{\mathcal{B}}_{r,p} S[\xi](r,k)|.$$

Next, in order to talk about average bit complexity, we introduce the probability framework. Two runs of protocol $\mathcal{B}$ are *identical at round ℓ*, denoted by $\xi \overset{\ell}{=} \xi'$, if both runs have identical external inputs and adversaries up to and including

round ℓ. An *initial* A *of* $\mathcal{R}^{\mathcal{B}}$ is a subset of $\mathcal{R}^{\mathcal{B}}$ such that for some run $\xi \in \mathcal{R}^{\mathcal{B}}$ and round number ℓ,

$$A = \{\xi' \mid \xi' \stackrel{\ell}{=} \xi\}.$$

A subset of $\mathcal{R}^{\mathcal{B}}$ is *measurable* if it is an element in the σ-algebra generated by the initials, see [H] section 5. Finally, a *probability measure on* $\mathcal{R}^{\mathcal{B}}$ is just a probability measure defined on this σ-algebra.

Since the function $\psi(\cdot, \ell)$ is measurable we may define the *average bit complexity of protocol* $\mathcal{B}$ *at round* ℓ, denoted by $\mathsf{ABC}(\mathcal{B}, \ell)$, as

$$\mathsf{ABC}(\mathcal{B}, \ell) = \mathsf{E}\psi(\xi, \ell) = \int \psi(\xi, \ell) d\nu(\xi)$$

where ν is a probability measure on $\mathcal{R}^{\mathcal{B}}$.

Definition 5 *The protocol* $\mathcal{B}$ has the least average bit complexity *in its isomorphism class if for all protocols* $\mathcal{C}$ *isomorphic to* $\mathcal{B}$,

$$\mathsf{ABC}(\mathcal{B}, \ell) \leq \mathsf{ABC}(\mathcal{C}, \ell) \quad \textit{for all } \ell.$$

Protocols such as $\mathcal{B}$ *are called LAC protocols.*

We proceed to show that in every isomorphism class there exists a *LAC* protocol. For a subset B of $AST^{\mathcal{B}}_{r,k}$, let the *indicator function of* B *in* $\mathcal{R}^{\mathcal{B}}$ be the function $\chi_B : \mathcal{R}^{\mathcal{B}} \to \{0, 1\}$ given by

$$\chi_B(\xi) = \begin{cases} 1 & \text{if } S[\xi](r,k) \in B \\ 0 & \text{otherwise} \end{cases}$$

Define the *weight function* δ by

$$\delta B = \mathsf{E}\chi_B(\xi).$$

For the state $s \in AST^{\mathcal{B}}_{r,k}$, let $[s]$ denote the equivalence class of s with respect to the relation $\stackrel{\mathcal{B},p}{\simeq}$. Let $[s_1], [s_2], \ldots, [s_m]$ be all the distinct equivalence classes in $AST^{\mathcal{B}}_{r,k}$ with respect to that relation, and assume without loss of generality that

$$\delta[s_1] \geq \delta[s_2] \geq \cdots \geq \delta[s_m].$$

Let $\mathcal{A}$ be the protocol whose set of states coincides with the ones in $\mathcal{B}$ such that

$$\mu^{\mathcal{A}}_{r,p} s'_j = M_j \quad \text{for all } s'_j \in [s_j].$$

The central theorem in this section asserts that the protocol $\mathcal{A}$ has the least average bit complexity in its isomorphism class, which is, of course, also $\mathcal{B}$'s isomorphism class. We need some more terminology. For the states $s_r, s'_r \in AST^{\mathcal{C}}_{r,k}$, let

$$s_r \overset{\mathcal{C},p}{\leftrightarrow} s'_r \qquad \text{iff} \qquad \mu^{\mathcal{C}}_{r,p} s_r = \mu^{\mathcal{C}}_{r,p} s'_r.$$

Notice that $\overset{\mathcal{C},p}{\leftrightarrow}$ is an equivalence relation; thus, denote by $\langle s \rangle$ the equivalence class in $AST^{\mathcal{C}}_{r,k}$ of the state s, and by $\tau(\mathcal{C}, r, p, k)$ the set of all equivalence classes induced by this relation on $AST^{\mathcal{C}}_{r,k}$.

Lemma 9

$$\mathsf{E}|\mu^{\mathcal{C}}_{r,p} S[\eta](r,k)| = \sum |\mu^{\mathcal{C}}_{r,p}\langle s \rangle| \delta \langle s \rangle$$

where the summation ranges over $\langle s \rangle \in \tau(\mathcal{C}, r, p, k)$.

Lemma 10 *Let $\mathcal{A}$ be the protocol constructed above, and let $\mathcal{C}$ be an arbitrary protocol isomorphic to $\mathcal{A}$. Then,*

$$\sum_{\langle s \rangle \in \tau(\mathcal{C},r,p,k)} |\mu^{\mathcal{C}}_{r,p}\langle s \rangle| \delta \langle s \rangle \geq \sum_{\langle s \rangle \in \tau(\mathcal{A},r,p,k)} |\mu^{\mathcal{A}}_{r,p}\langle s \rangle| \delta \langle s \rangle.$$

Proof: If the partitions $\tau(\mathcal{C}, r, p, k)$ and $\tau(\mathcal{A}, r, p, k)$ of $AST^{\mathcal{C}}_{r,k}$ are identical, this inequality is simple. Otherwise, by lemma 8, $\tau(\mathcal{C}, r, p, k)$ is a refinement of $\tau(\mathcal{A}, r, p, k)$. Consider first the case where the probabilities corresponding to $\tau(\mathcal{A}, r, p, k)$ and $\tau(\mathcal{C}, r, p, k)$ are given by the following two sorted sequences

$$\begin{array}{ccccccccccccc}
\alpha_1 & \geq & \cdots & \geq & \alpha_{i-1} & \geq & \alpha_i & \geq & \alpha_{i+1} & \geq & \cdots & \geq & \alpha_j \\
\alpha_2 & \geq & \cdots & \geq & \alpha_i & \geq & \beta & \geq & \alpha_{i+1} & \geq & \cdots & \geq & \alpha_j \quad \geq \quad \gamma
\end{array}$$

where $\alpha_1 = \beta + \gamma$.

Denote by Δ the difference between the average lengths of the partitions $\tau(\mathcal{C}, r, p, k)$ and $\tau(\mathcal{A}, r, p, k)$. We argue that $\Delta \geq 0$. Indeed, if ℓ_k denotes the length of the k^{th} message, then

$$\begin{aligned}
\Delta &\geq \sum_{k=2}^{i} \alpha_k(\ell_{k-1} - \ell_k) + \beta\ell_i + \gamma\ell_{j+1} - \alpha_1\ell_1 \\
&\geq \alpha_1\ell_i - \alpha_1\ell_1 - \alpha_2 \sum_{k=2}^{i} (\ell_k - \ell_{k-1}) \\
&\geq \alpha_1(\ell_i - \ell_1) - \alpha_2(\ell_i - \ell_1) \\
&\geq 0.
\end{aligned}$$

The more general case follows by induction. ∎

Finally, we state and prove the central result in this section.

Theorem 10 *In each isomorphism class of protocols there exists a protocol that has the least average bit complexity in its isomorphism class.*

Proof: Pick an isomorphism class of protocols, and let $\mathcal{B}$ be a protocol in that class. We show that the protocol $\mathcal{A}$ constructed above has the least average bit complexity in $\mathcal{B}$'s isomorphism class. Indeed, for any protocol $\mathcal{C}$ isomorphic to $\mathcal{B}$ and every round ℓ,

$$\mathsf{E}\psi(\eta, \ell) \geq \mathsf{E}\psi(\alpha, \ell)$$

where η and α denote runs of $\mathcal{C}$ and $\mathcal{A}$ respectively. ∎

8.3 The Minimal Dominating Protocol

Suppose that you have two protocols $\mathcal{B}$ and $\mathcal{C}$ for performing two different tasks, and that you want to design a third protocol that will perform these two tasks simultaneously. The full-view protocol can, of course, serve that purpose. However, in light of theorem 9, a more appealing approach would suggest using the *minimal* protocol dominating $\mathcal{B}$ and $\mathcal{C}$ in order to perform these two tasks.

Several questions immediately emerge: Is there a minimal protocol dominating $\mathcal{B}$ and $\mathcal{C}$? If such a protocol exists, can it be given explicitly? Is it unique? Does category theory provide a natural approach to this problem? In this subsection we answer all these questions in the affirmative.

We begin with the categorical approach. The crucial observation is that the existence of the minimal dominating protocol is precisely equivalent to the existence of the categorical *product* of $\mathcal{B}$ and $\mathcal{C}$. Thus, uniqueness up to isomorphism is guaranteed by the uniqueness of the product.

We now apply our set theoretic intuition to construct the product of two protocols. Let $\mathcal{B} \Pi \mathcal{C}$ be the protocol whose state transition functions are $\delta_p^{\mathcal{B}} \times \delta_p^{\mathcal{C}}$, whose message generators are $\mu_{p,q}^{\mathcal{B}} \times \mu_{p,q}^{\mathcal{C}}$ and whose inputs are drawn from Σ.

The main theorem of this section asserts that $\mathcal{B} \Pi \mathcal{C}$ is the product of $\mathcal{B}$ and $\mathcal{C}$. Thus, $\mathcal{B} \Pi \mathcal{C}$ is also the minimal protocol dominating $\mathcal{B}$ and $\mathcal{C}$.

Theorem 11 *The protocol $\mathcal{B} \Pi \mathcal{C}$ is the product of $\mathcal{B}$ and $\mathcal{C}$.*

Before proving this theorem we examine the morphism $h^{\mathcal{B}\Pi\mathcal{C}}$. Not surprisingly, we have,

Lemma 11

$$h_s^{\mathcal{B}\Pi\mathcal{C}} = h_s^{\mathcal{B}} \times h_s^{\mathcal{C}}$$

Proof: By induction on the depth of the state tree of the full-view protocol to which $h_s^{\mathcal{B}\Pi\mathcal{C}}$ is applied. ∎

An immediate result of this lemma is that the protocol $\mathcal{B} \Pi \mathcal{C}$ dominates both $\mathcal{B}$ and $\mathcal{C}$. Furthermore, this lemma leads naturally to the notion of the *product of runs*. We concentrate now on the proof of theorem 11.

Proof: Let $\mathcal{D}$ be any protocol such that $\mathcal{D} \succeq \mathcal{B}, \mathcal{C}$. By definition, both $h^{\mathcal{B}}\ (h^{\mathcal{D}})^{-1}$ and $h^{\mathcal{C}}\ (h^{\mathcal{D}})^{-1}$ are well-defined. Thus, again by definition,

$$(h^{\mathcal{B}} \times h^{\mathcal{C}})\ (h^{\mathcal{D}})^{-1}$$

is well-defined. Finally, by lemma 11,

$$h^{\mathcal{B}\Pi\mathcal{C}}\ (h^{\mathcal{D}})^{-1}$$

is also well-defined and therefore $\mathcal{D} \succeq \mathcal{B} \Pi \mathcal{C}$. ∎

We leave it for the reader to verify that if at some round in a run of $\mathcal{B} \Pi \mathcal{C}$ each of the processors p is in a state in $h^{\mathcal{B}\Pi\mathcal{C}} ST_p^{\mathcal{F}\mathcal{V}}$, then at the succeeding round each of them will still be in a state of that form, showing thereby that the set of states of $\mathcal{B} \Pi \mathcal{C}$, $\{h^{\mathcal{B}\Pi\mathcal{C}} ST_p^{\mathcal{F}\mathcal{V}}\}$, is closed.

We conclude this section by examining the partition that the states of $\mathcal{B} \Pi \mathcal{C}$ induce on the set of states of the full-view protocol. Optimally, this partition should be the coarsest refinement of the partitions induced on the set of states of the full-view protocol by $\mathcal{B}$ and $\mathcal{C}$. Fortunately, this is precisely the situation.

Corollary 1 *The partition on the set of states of the full-view protocol induced by the product of two protocols is the coarsest refinement of the partitions induced by each of the protocols.*

Previously we considered the product of two protocols. Due to various subtleties we now give the products of an arbitrary indexed set of protocols. Following conventions, let I be an index set, and consider the indexed family of protocols

$$\{\mathcal{B}^\alpha \mid \alpha \in I\}.$$

For every state $s \in ST_p^{\mathcal{F}\mathcal{V}}$, let the *state function* $f_s : I \to \cup ST_p^\alpha$ be given by $\alpha \rightsquigarrow h^\alpha s$, where ST_p^α and h^α denote $ST_p^{\mathcal{B}^\alpha}$ and $h^{\mathcal{B}^\alpha}$ respectively. The states of the

product protocol are equivalence classes of state functions induced by the relation $\equiv$, which is defined by $f_{s_1} \equiv f_{s_2}$ iff these two functions are equal pointwise. The definition of the state transition functions as well as the proof that this protocol is indeed the product of the family $\{\mathcal{B}^\alpha\}$ is left for the reader. We concentrate on properties of this protocol. Notice first that the set of states of the product protocol is *countable* regardless of the cardinality of the index set I, since the set of state trees of the full-view protocol is countable. For this very reason, we may encode each of the messages generated by the message generators using just *finitely* many bits. Thus, the product protocol of an arbitrary set of protocols is a perfectly legitimate protocol. Finally, the partition of the state trees of the full-view protocol induced by the product protocol is optimal. In fact,

$$(h^\Pi)^{-1}[f_s] = \cap (h^\alpha)^{-1}[f_s](\alpha).$$

8.4 The Maximal Dominated Protocol

Our success in determining the minimal protocol dominating a given set of protocols tempts us naturally to seek the maximal protocol dominated by a set of protocols. Categorically, this means searching for the coproduct of a set of protocols.

Before attempting to construct this coproduct, we settle the issue of its existence by recalling the following theorem from lattice theory: A partially ordered set with a least element such that every non-vacuous subset has a least upper bound is a complete lattice. Here a *complete lattice* is a partially ordered set in which every subset has both a supremum and an infimum, see [J1] page 436.

Suppose that we are given two protocols $\mathcal{B}$ and $\mathcal{C}$. We proceed to determine their coproduct by introducing the protocol $\mathcal{B} \amalg \mathcal{C}$. Consider first the relation $\asymp$ on $\Im = h^{\mathcal{B}\amalg\mathcal{C}} ST_p^{\mathcal{F}\mathcal{V}}$ given by $\langle s_1^{\mathcal{B}}, s_1^{\mathcal{C}}\rangle \asymp \langle s_2^{\mathcal{B}}, s_2^{\mathcal{C}}\rangle$ iff either $s_1^{\mathcal{B}} = s_2^{\mathcal{B}}$ or $s_1^{\mathcal{C}} = s_2^{\mathcal{C}}$, where both $\langle s_1^{\mathcal{B}}, s_1^{\mathcal{C}}\rangle$ and $\langle s_2^{\mathcal{B}}, s_2^{\mathcal{C}}\rangle$ are in $\Im$. It is clear that $\asymp$ is both reflexive and symmetric; however, it is not transitive. Consider therefore the transitive closure $\overset{*}{\asymp}$ of $\asymp$, and denote by $[s^{\mathcal{B}}, s^{\mathcal{C}}]$ the equivalence class of $\langle s^{\mathcal{B}}, s^{\mathcal{C}}\rangle$ induced by $\overset{*}{\asymp}$. The states corresponding to processor p in $\mathcal{B} \amalg \mathcal{C}$ are just these equivalence classes.

Assume for the moment that we could define a protocol with these sets of states such that

$$h_s^{\mathcal{B}\amalg\mathcal{C}} s^{\mathcal{F}\mathcal{V}} = [h_s^{\mathcal{B}} s^{\mathcal{F}\mathcal{V}}, h_s^{\mathcal{C}} s^{\mathcal{F}\mathcal{V}}].$$

Then, we argue that $\mathcal{B} \amalg \mathcal{C}$ would be the coproduct of $\mathcal{B}$ and $\mathcal{C}$.

Lemma 12 *Assume that* $h_s^{\mathcal{B}\amalg\mathcal{C}} s^{\mathcal{F}\mathcal{V}} = [h^{\mathcal{B}} s^{\mathcal{F}\mathcal{V}}, h^{\mathcal{C}} s^{\mathcal{F}\mathcal{V}}]$ *and let* $\mathcal{D}$ *be an arbitrary protocol satisfying both* $\mathcal{B} \succeq \mathcal{D}$ *and* $\mathcal{C} \succeq \mathcal{D}$. *Then* $\mathcal{B} \amalg \mathcal{C} \succeq \mathcal{D}$.

Proof: Let s_1 and s_2 be any two states in $ST_p^{\mathcal{FV}}$ and assume that

$$h^{\mathcal{B}\amalg\mathcal{C}}s_1 = h^{\mathcal{B}\amalg\mathcal{C}}s_2.$$

Thus, there exists a sequence of states $s^i \in ST_p^{\mathcal{FV}}$, for $i = 1 \ldots m$, such that $s^1 = s_1$, $s^m = s_2$, and

$$h^{\mathcal{B}\amalg\mathcal{C}}s^1 \asymp h^{\mathcal{B}\amalg\mathcal{C}}s^2 \asymp \cdots \asymp h^{\mathcal{B}\amalg\mathcal{C}}s^m.$$

Therefore, for each $i \in \{1 \ldots m-1\}$, either

$$h^{\mathcal{B}}s^i = h^{\mathcal{B}}s^{i+1} \quad \text{or} \quad h^{\mathcal{C}}s^i = h^{\mathcal{C}}s^{i+1}.$$

But recall now that by assumption both $\mathcal{B} \succeq \mathcal{D}$ and $\mathcal{C} \succeq \mathcal{D}$. Hence, we have $h^{\mathcal{D}}s^i = h^{\mathcal{D}}s^{i+1}$ for each $i \in \{1 \ldots m-1\}$, and therefore

$$h^{\mathcal{D}}s_1 = h^{\mathcal{D}}s_2. \qquad \blacksquare$$

Encouraged by this result we proceed to construct the protocol $\mathcal{B} \amalg \mathcal{C}$ for which we hope the equality $h_s^{\mathcal{B}\amalg\mathcal{C}}s^{\mathcal{FV}} = [h_s^{\mathcal{B}}s^{\mathcal{FV}}, h_s^{\mathcal{C}}s^{\mathcal{FV}}]$ will hold. First, define the message generator $\mu_{p,q}^{\mathcal{B}\amalg\mathcal{C}}$ by assigning, for fixed p and q, different messages to each of the countably many triplets $([s^{\mathcal{B}}, s^{\mathcal{C}}], p, q)$.

Next, define the transition function $\delta_p^{\mathcal{B}\amalg\mathcal{C}}$ by

$$\begin{aligned}\delta_p^{\mathcal{B}\amalg\mathcal{C}}([s_p^{\mathcal{B}}, s_p^{\mathcal{C}}] \ldots \mu_{q,p}^{\mathcal{B}\amalg\mathcal{C}}[s_q^{\mathcal{B}}, s_q^{\mathcal{C}}] \ldots \iota) = \\ [\delta_p^{\mathcal{B}}(s_p^{\mathcal{B}} \ldots \mu_{q,p}^{\mathcal{B}} s_q^{\mathcal{B}} \ldots \iota), \delta_p^{\mathcal{C}}(s_p^{\mathcal{C}} \ldots \mu_{q,p}^{\mathcal{C}} s_q^{\mathcal{C}} \ldots \iota)]\end{aligned}$$

We proceed to persuade ourselves that this definition meets our expectations by first showing that the protocol $\mathcal{B} \amalg \mathcal{C}$ is well-defined.

Lemma 13 *The protocol $\mathcal{B} \amalg \mathcal{C}$ is well-defined.*

Proof: In order to show that $\mathcal{B} \amalg \mathcal{C}$ is well-defined we only have to prove that the transition functions $\delta_p^{\mathcal{B}\amalg\mathcal{C}}$ are well-defined. To this end assume that for each processor $q \in P$,

$$[s_q^{\mathcal{B}}, s_q^{\mathcal{C}}] \overset{*}{\asymp} [s_q'^{\mathcal{B}}, s_q'^{\mathcal{C}}].$$

We have to prove that

$$\begin{aligned}[\delta_p^{\mathcal{B}}(s_p^{\mathcal{B}} \ldots \mu_{q,p}^{\mathcal{B}} s_q^{\mathcal{B}} \ldots \iota), \delta_p^{\mathcal{C}}(s_p^{\mathcal{C}} \ldots \mu_{q,p}^{\mathcal{C}} s_q^{\mathcal{C}} \ldots \iota)] = \\ [\delta_p^{\mathcal{B}}(s_p'^{\mathcal{B}} \ldots \mu_{q,p}^{\mathcal{B}} s_q'^{\mathcal{B}} \ldots \iota), \delta_p^{\mathcal{C}}(s_p'^{\mathcal{C}} \ldots \mu_{q,p}^{\mathcal{C}} s_q'^{\mathcal{C}} \ldots \iota)].\end{aligned}$$

This equality holds by definition if for each processor $q \neq r$,

$$[s_q^{\mathcal{B}}, s_q^{\mathcal{C}}] = [s_q'^{\mathcal{B}}, s_q'^{\mathcal{C}}]$$

and for r we have,

$$[s_r^{\mathcal{B}}, s_r^{\mathcal{C}}] \asymp [s_r'^{\mathcal{B}}, s_r'^{\mathcal{C}}].$$

The extension for the more general case is immediate. ∎

We now show that the equality we wanted $h_s^{\mathcal{B}\amalg\mathcal{C}}$ to satisfy indeed holds.

Lemma 14

$$h_s^{\mathcal{B}\amalg\mathcal{C}} = [h_s^{\mathcal{B}}, h_s^{\mathcal{C}}]$$

Proof: By induction on the depth of the state tree of the full-view protocol to which $h_s^{\mathcal{B}\amalg\mathcal{C}}$ is applied. ∎

This completes the construction of the coproduct of two protocols. We conclude this subsection by noticing that the partition that the states of $\mathcal{B} \amalg \mathcal{C}$ induce on the set of states of the full-view protocol is precisely the finest coarsening of the partitions induced on the set of states of the full-view protocol by $\mathcal{B}$ and $\mathcal{C}$. This is the best we can hope for.

8.5 Universal Predicates

In order to examine properties of runs of a given protocol $\mathcal{B}$ we usually employ predicates which are merely subsets of $\mathcal{R}^{\mathcal{B}}$. Unfortunately, predicates designed in this way often rely heavily on the protocol under consideration, thus, it is usually difficult to use them in order to compare the performance of different protocols. To overcome this difficulty, several researchers have considered special types of predicates that are protocol independent. Since the performance of the adversary depends heavily on the protocol, these predicates cannot be used to compare *adversaries* in runs of different protocols. However, precisely the adversaries are the central component that introduce action in runs; thus, predicates that are protocol independent cannot, by their very definition, capture this central ingredient in distributed systems.

In this subsection we employ our categorical approach in order to design *universal predicates* that allow us to examine in a natural way the effect of the different components in the specification of runs, including, in particular, the effect of the adversaries.

Elementary universal predicates are subsets of $\mathcal{R}^{\mathcal{FV}}$. We say that *the run ξ of $\mathcal{B}$ satisfies the universal predicate φ*, denoted by

$$\xi \models_U \varphi,$$

if $(h^{\mathcal{B}})^{-1}\xi$ is a subset of φ which is denoted by $(h^{\mathcal{B}})^{-1}\xi \models \varphi$. We now make these predicates more expressive by introducing standard logic connectives and modal operators of knowledge and common knowledge.

Definition 6 *A* universal predicate *is one of the following:*

1. *An elementary universal predicate.*
2. *Either $\varphi \vee \psi$, $\varphi \wedge \psi$ or $\neg\varphi$, where both φ and ψ are universal predicates.*
3. *$K_{(p,\ell)}\varphi$ or $C_\ell\varphi$ where φ is a universal predicate.*

The semantics of these predicates are as follows:

1. If φ is an elementary universal predicate, then

$$\xi \models_U \varphi \qquad \text{iff} \qquad (h^{\mathcal{B}})^{-1}\xi \models \varphi.$$

2. $\xi \models_U \varphi \vee \psi$ iff either $\xi \models_U \varphi$ or $\xi \models_U \psi$. The other cases are treated similarly.

3. $\xi \models_U K_{(p,\ell)}\varphi$ iff for each run ξ' of $\mathcal{B}$ such that $\xi' \overset{(p,\ell)}{\approx} \xi$, $\xi' \models_U \varphi$. Similarly, $\xi \models_U C_\ell\varphi$ iff for each run ξ' of $\mathcal{B}$ such that $\xi' \overset{\ell}{\sim} \xi$, $\xi' \models_U \varphi$.

Say that a predicate is *monotone* if the negation symbol does not appear in it. In the sequel we examine the role of monotone universal predicates in distributed systems. Our first proposition shows, in the flavor of theorem 8, that whenever a monotone universal predicate holds in the morphic image of a run, then it holds in the run itself. We show thereafter that elementary universal predicates are expressive enough to separate the protocol hierarchy.

Theorem 12 *Assume that $\mathcal{B} \succeq \mathcal{C}$, that ξ is a run of $\mathcal{B}$, and that φ is a monotone universal predicate. Then $h\langle\mathcal{B},\mathcal{C}\rangle\xi \models_U \varphi$ implies $\xi \models_U \varphi$.*

Proof: The proof proceeds by induction on the structure of the predicate. The only interesting case here is the base case, where φ is an elementary universal

predicate, which we now prove. Assume that $h\langle \mathcal{B}, \mathcal{C}\rangle \xi \models_U \varphi$. Then, by definition, $(h^{\mathcal{C}})^{-1} h\langle \mathcal{B}, \mathcal{C}\rangle \xi \models \varphi$. But we have

$$(h^{\mathcal{C}})^{-1} h\langle \mathcal{B}, \mathcal{C}\rangle \xi = (h^{\mathcal{C}})^{-1} h^{\mathcal{C}} (h^{\mathcal{B}})^{-1} \xi \supseteq (h^{\mathcal{B}})^{-1} \xi.$$

Thus, $(h^{\mathcal{B}})^{-1}\xi \models \varphi$, and therefore $\xi \models_U \varphi$. ∎

We proceed to separate the protocol hierarchy using monotone universal predicates. Evidently, the basic predicates are completely useless for that purpose. Note first that if $\mathcal{B}$ is *strictly* bigger than $\mathcal{C}$, that is $\mathcal{B} \succ \mathcal{C}$, then there exists a run ξ of $\mathcal{B}$ such that the set $(h^{\mathcal{C}})^{-1}(h\langle \mathcal{B}, \mathcal{C}\rangle \xi)$ *strictly* contains the set $(h^{\mathcal{B}})^{-1}\xi$. Now consider the elementary universal predicate $\varphi = (h^{\mathcal{B}})^{-1}\xi$. This predicate satisfies $\xi \models_U \varphi$; however, $h\langle \mathcal{B}, \mathcal{C}\rangle \xi \not\models_U \varphi$. Thus, φ separates the protocols $\mathcal{B}$ and $\mathcal{C}$.

8.6 Dominance and Conveying

We now examine the effect of the dominance relation on the notion of *conveying* introduced in [M]. Intuitively, a processor p conveys a fact to another processor q if p is certain that q will know that fact if it trusts p.

Definition 7 *Let ξ be a run of the protocol $\mathcal{B}$, and assume that p is correct at ℓ in ξ. Let φ be a predicate such that*

$$\xi \models K_{(p,\ell-1)}\varphi.$$

Then, p conveys φ to q at ℓ in ξ *if*

$$\xi \models K_{(p,\ell-1)} K_{(q,\ell)} \left(\textit{if } p \textit{ is correct at } \ell, \textit{ then } \varphi\right).$$

We now prove that if the protocol $\mathcal{B}$ dominates the protocol $\mathcal{C}$, then whenever the processor p conveys the monotone universal predicate φ to q in the morphic image of a run of $\mathcal{B}$, it conveys that same predicate in the run itself. A similar assertion holds if the predicate is basic.

Theorem 13 *Assume that $\mathcal{B} \succeq \mathcal{C}$, and let $h = h\langle \mathcal{B}, \mathcal{C}\rangle$. Let φ be a monotone universal predicate, and ξ a run of $\mathcal{B}$. If p conveys φ to q at ℓ in $h\xi$, then p conveys that same predicate to q at ℓ in ξ.*

Proof: Let ξ' and ξ'' be any two runs of $\mathcal{B}$ such that $\xi' \overset{(p,\ell-1)}{\approx} \xi$, $\xi'' \overset{(q,\ell)}{\approx} \xi'$ and $\xi'' \models_U$ "p is correct at ℓ".

Then by lemma 5, we have both $h\xi' \overset{(p,\ell-1)}{\approx} h\xi$ and $h\xi'' \overset{(q,\ell)}{\approx} h\xi'$. Furthermore, we also have $h\xi'' \models_U$ "p is correct at ℓ", showing that $h\xi'' \models_U \varphi$. But φ is monotone, thus, by theorem 12, $\xi'' \models_U \varphi$. Therefore, p also conveys φ to q at ℓ in ξ. ∎

The case where φ is a basic predicate follows similarly. In particular, we have the natural property that processors in corresponding runs of isomorphic protocols convey precisely the same monotone universal predicates at each round.

8.7 Partitions of State Trees Induced by Protocols

In subsection 5.2 we showed that the states of every concise protocol induce a partition of the states of the full-view protocol. This statement naturally leads to the question: Given a partition of the states of the full-view protocol, can it be identified with the partition induced by some concise protocol? In this subsection we give a precise characterization of these partitions. As an immediate corollary of this characterization we show that each protocol can be matched with an isomorphic one in which all the message generators are just the identity function. Thus, non-trivial message generators do not introduce new elements in the hierarchy of protocols.

Let $\wp$ be a partition of the states of the full-view protocol. Our intuition is that $\wp$ can be identified with a concise protocol iff it is the image of some morphism. A partition $\wp$ is *protocol induced* if the following holds for every set of states $s_p^i \in ST_{p,\ell_i}$, for $i = 1,2$, rounds $\ell_i \geq 0$, and processors $p \in P$: If s_p^1 and s_p^2 are in the same equivalence class for all p, then the states s^1 and s^2 are also in the same equivalence class, where s^i is the state tree whose p^{th} principal subtree is s_p^i, and the roots of s^1 and s^2 carry precisely the same labels. We now have

Theorem 14 *The partition $\wp$ can be identified with a concise protocol iff $\wp$ is protocol induced.*

Proof: Suppose that $\wp$ can be identified with a concise protocol $\mathcal{B}$, that is, the states s^1 and s^2 of the full-view protocol belong to the same equivalence class iff $h^{\mathcal{B}} s^1 = h^{\mathcal{B}} s^2$. Then, by the definition of $h^{\mathcal{B}}$, $\wp$ is protocol induced.

Conversely, let $\wp$ be protocol induced, and denote by $[s]$ the equivalence class of the state s of the full-view protocol in $\wp$. We now define the protocol $\mathcal{B}$ corresponding to $\wp$. Not surprisingly, its states are the equivalence classes in $\wp$. Its message generators are identities and its state transition functions are given by

$$\delta_p^{\mathcal{B}}([s_p]\ldots[s_q]\ldots\iota) = [s]$$

where s is a depth ℓ state tree whose root is labelled $\langle p, \iota, \ell \rangle$, and whose q^{th} principal subtree is a depth $\ell - 1$ state tree drawn from the equivalence class s_q. Finally, $\delta_p^{\mathcal{B}}$ is well-defined since $\wp$ is protocol induced. ∎

We conclude this subsection by noticing that the message generators in the protocol constructed in the sufficient part of the proof above are all identities. Thus, we have

Corollary 2 *For every protocol there exists an isomorphic protocol with identity message generators.*

Consequently, while designing protocols it is sufficient to consider only protocols with identity message generators. Each such protocol can then be transformed into an isomorphic one with least average bit complexity by the construction in section 8.2.

8.8 Density and Limit Properties

We begin this section by counting the number of isomorphism classes of protocols. While a simple counting argument shows that there are uncountably many protocols, such an argument seems too coarse for measuring the cardinality of the set of isomorphism classes. We evaluate this cardinality using diagonalization.

Theorem 15 *There are uncountably many isomorphism classes of protocols.*

Next, we examine density properties of the hierarchy. Given any two protocols $\mathcal{B}$ and $\mathcal{C}$ such that $\mathcal{B}$ strictly dominates $\mathcal{C}$, is there always a protocol strictly between them? The answer to that question is no. Indeed, just take for $\mathcal{B}$ an arbitrary protocol in which all the processors have exactly one state, excluding one processor which has two states. This protocol strictly dominates each protocol in which every processor has precisely one state. Furthermore, there certainly cannot exist any protocol strictly between these two.

However, we now show that there are both infinite strictly decreasing sequences of protocols and infinite strictly increasing sequences of protocols in the protocol hierarchy. Furthermore, such sequences exist even though the input set for the protocols contains precisely one element.

Consider the following strictly decreasing sequence of protocols. The ℓ^{th} element in the sequence, $\mathcal{B}^\ell$, is given intuitively as follows: Each of the processors remembers all the messages and external inputs it has received throughout the execution, excluding processor p. Processor p also remembers all the messages

and the inputs it has received; however, it forgets (through state transition) at each round $k \in \{1 \ldots \ell\}$ each of the messages that some other processor, say q, transmits to it at k. Now, it is readily verified that $\mathcal{B}^{\ell} \succ \mathcal{B}^{\ell+1}$. Similarly, we can construct strictly increasing sequences by letting processors remember more as the round number ℓ increases.

Having settled the issue of existence for both infinite strictly increasing and infinite strictly decreasing sequences, we proceed to examine their limit properties. We need in this section a technical assumption about the crashing assignment: Each faulty processor completely ceases transmitting within a bounded number of rounds after it becomes faulty.

We show that even the very expressive monotone universal predicates are not sufficiently strong for separating each infinite strictly decreasing sequence of protocols from its limit. In fact, we prove a stronger assertion. A subset of a poset is called a *chain* if every two elements in the subset are related. We prove that every chain of protocols cannot be separated from its direct limit through monotone universal predicates. Notice that in the hierarchy of protocols, as in any other poset, the direct limit of a chain of protocols coincides with the coproduct of its members.

Theorem 16 *Let $\{\mathcal{B}^{\alpha} \mid \alpha \in I\}$ be a chain of protocols with direct limit $\mathcal{B}^{\mathrm{II}}$. For a run ρ of $\mathcal{FV}$, let $\xi^{\alpha} = h^{\alpha}\rho$ and, similarly, let $\xi^{\mathrm{II}} = h^{\mathrm{II}}\rho$. Then for each monotone universal predicate φ such that*

$$\xi^{\alpha} \models_{U} \varphi \quad \text{for each } \alpha$$

also $\xi^{\mathrm{II}} \models_{U} \varphi$.

We begin by constructing the coproduct of the elements in a chain of protocols.

Lemma 15 *Let $C = \{\mathcal{B}^{\alpha} \mid \alpha \in I\}$ be a chain of protocols. Then the coproduct of the elements in C, denoted by $\mathcal{B}^{\mathrm{II}}$, is the protocol whose states are equivalence classes of state functions induced by the relation $\cong$, which is given by $f_{s1} \cong f_{s2}$ iff $f_{s1}\alpha = f_{s2}\alpha$ for some $\alpha \in I$, and whose state transition functions are given by*

$$\delta_p^{\mathrm{II}}(f_{s_p} \ldots \mu_{q,p}^{\mathrm{II}} f_{s_q} \ldots \iota)\alpha = \delta_p^{\alpha}(f_{s_p}\alpha \ldots \mu_{q,p}^{\alpha} f_{s_q}\alpha \ldots \iota).$$

In the next two lemmas we give properties of the direct limit protocol in terms of the protocols in the chain.

Lemma 16 *Let $C = \{\mathcal{B}^{\alpha} \mid \alpha \in I\}$ be a chain of protocols. Then,*

$$(h^{\mathrm{II}})^{-1}\xi^{\mathrm{II}} = \cup (h^{\alpha})^{-1}\xi^{\alpha}.$$

Proof: Assume first that $\rho' \in (h^\alpha)^{-1}\xi^\alpha$. Then we have

$$h^{\mathrm{II}}\rho' = h^{\mathrm{II}}(h^\alpha)^{-1}\xi^\alpha = h^{\mathrm{II}}\rho = \xi^{\mathrm{II}}.$$

Hence, $\rho' \in (h^{\mathrm{II}})^{-1}\xi^{\mathrm{II}}$.

Conversely, assume that $\rho' \in (h^{\mathrm{II}})^{-1}\xi^{\mathrm{II}}$. We show the existence of an element $\alpha \in I$ such that $\rho' \in (h^\alpha)^{-1}\xi^\alpha$.

For $\langle p, \ell\rangle \in CA$ and $q \in P$, let s and s' denote $S[\rho](p, \ell - 1 \mid p \xrightarrow{\ell} q)$ and $S[\rho'](p, \ell - 1 \mid p \xrightarrow{\ell} q)$ respectively. Since $h^{\mathrm{II}}\rho' = h^{\mathrm{II}}\rho$, by theorem 1, $h^{\mathrm{II}}s' = h^{\mathrm{II}}s$; hence, by the definition of h^{II}, $f_{s'} \cong f_s$. Thus, for some $\beta \in I$, $f_{s'}\gamma = f_s\gamma$ whenever $\mathcal{B}^\gamma \preceq \mathcal{B}^\beta$, and by the definition of state functions $h^\gamma s' = h^\gamma s$. Finally, since faulty processors behave maliciously for only finitely many rounds, the minimum over all $\langle p, \ell\rangle \in CA$ and $q \in P$ of the corresponding $\mathcal{B}^\beta$'s exists. Denote this minimum protocol by $\mathcal{B}^\alpha$. Then, $h^\alpha\rho' = h^\alpha\rho$. ∎

Lemma 17 *For the run ρ of $\mathcal{FV}$ let $\xi^\alpha = h^\alpha\rho$ and $\xi^{\mathrm{II}} = h^{\mathrm{II}}\rho$. Assume $\xi^{\mathrm{II}} \overset{(p,\ell)}{\approx} \xi'^{\mathrm{II}}$. Then for some $\alpha \in I$ there exists some run ξ'^α such that both $\xi'^{\mathrm{II}} = h\langle \mathcal{B}^\alpha, \mathcal{B}^{\mathrm{II}}\rangle\xi'^\alpha$ and $\xi^\alpha \overset{(p,\ell)}{\approx} \xi'^\alpha$.*

Next, we prove the following technical assertion.

Lemma 18 *Let $C = \{\mathcal{B}^\alpha \mid \alpha \in I\}$ be a chain of protocols with coproduct $\mathcal{B}^{\mathrm{II}}$. Let $I = I_1 \cup I_2$, and let $C_i = \{\mathcal{B}^\alpha \mid \alpha \in I_i\}$, for $i = 1, 2$. Then, the coproduct of either C_1 or C_2 is $\mathcal{B}^{\mathrm{II}}$.*

Proof: Either $\mathcal{B}_1^{\mathrm{II}}$ or $\mathcal{B}_2^{\mathrm{II}}$ is dominated by all the elements in C; thus, at least one of them is $\mathcal{B}^{\mathrm{II}}$. ∎

Finally, we prove theorem 16.

Proof: The proof proceeds by induction on the structure of the universal predicate φ, that for any chain of protocols

$$C = \{\mathcal{B}^\alpha \mid \alpha \in I\}$$

and run ρ of $\mathcal{FV}$, if $\xi^\alpha = h^\alpha\rho \models_U \varphi$ for all $\alpha \in I$, then also $\xi^{\mathrm{II}} = h^{\mathrm{II}}\rho \models_U \varphi$.
Base case: Assume that φ is elementary. Since for each $\alpha \in I$, $\xi^\alpha \models_U \varphi$, we have $\cup(h^\alpha)^{-1}\xi^\alpha \subseteq \varphi$. Thus, by lemma 16, $(h^{\mathrm{II}})^{-1}\xi^{\mathrm{II}} \subseteq \varphi$ and therefore $\xi^{\mathrm{II}} \models_U \varphi$.
Inductive step: Assume first that $\varphi = \psi \vee \phi$. Since $\xi^\alpha \models_U \varphi$, either $\xi^\alpha \models \psi$ or $\xi^\alpha \models_U \phi$. Thus, there are index sets I_ψ and I_ϕ such that $I = I_\psi \cup I_\phi$, for $\alpha \in I_\psi$,

$\xi^\alpha \models_U \psi$, and, similarly, for $\alpha \in I_\phi$, $\xi^\alpha \models_U \phi$. By lemma 18 we may assume without loss of generality that the chain $\{\mathcal{B}^\alpha \mid \alpha \in I_\psi\}$ has coproduct $\mathcal{B}^{\mathrm{II}}$. Hence, by the inductive hypothesis, $\xi^{\mathrm{II}} \models_U \psi$. Therefore, $\xi^{\mathrm{II}} \models_U \varphi$. The case $\varphi = \psi \wedge \phi$ is simple.

Next, assume that $\varphi = K_{(p,\ell)}\psi$. We want to show $\xi^{\mathrm{II}} \models_U K_{(p,\ell)}\psi$, that is, for each $\xi'^{\mathrm{II}} \overset{(p,\ell)}{\approx} \xi^{\mathrm{II}}$, $\xi'^{\mathrm{II}} \models_U \psi$. By lemma 17 there exists an $\alpha \in I$ and a run ξ'^α for which both $\xi'^{\mathrm{II}} = h\langle \mathcal{B}^\alpha, \mathcal{B}^{\mathrm{II}} \rangle \xi'^\alpha$ and $\xi'^\alpha \overset{(p,\ell)}{\approx} \xi^\alpha$. But $\xi^\alpha \models_U K_{(p,\ell)}\psi$ thus $\xi'^\alpha \models_U \psi$, and by lemma 5 combined with the definition of knowledge, we have, in fact, that $\xi'^\beta \models_U \psi$ for each $\mathcal{B}^\beta \preceq \mathcal{B}^\alpha$. Therefore, by the inductive hypothesis and by lemma 18, $\xi'^{\mathrm{II}} \models_U \psi$. Thus, $\xi^{\mathrm{II}} \models_U \varphi$. The case where $\varphi = C_\ell\psi$ is treated similarly. ∎

8.9 Minimal Bit Efficient Protocols for Predicates

In this section we address the following problem: Given a monotone universal predicate, show that there exists a protocol that is as effective for that predicate as the full-view protocol, minimal among the protocols that are as effective as full-view for that predicate, and arbitrarily close to optimal in average bit complexity.

First, notice that by theorem 12 no protocol can outperform the full-view protocol, assuming comparable expressibility of adversaries. Indeed, if a predicate holds in the morphic image of a run of $\mathcal{FV}$, then that predicate holds in the run itself. Now, a protocol $\mathcal{B}$ is *effective* for a monotone universal predicate φ, if it is as effective for φ as the full-view protocol. Thus, $\mathcal{B}$ is effective for φ if whenever $\rho \models_U \varphi$ for a run ρ of $\mathcal{FV}$, also $h^{\mathcal{B}}\rho \models_U \varphi$.

The first issue that we examine is the existence of *minimal* effective protocols. More specifically, we show that for any given monotone universal predicate φ there exists an effective protocol $\mathcal{B}$ for φ, such that no protocol that $\mathcal{B}$ strictly dominates is effective for φ. This assertion is proved using Zorn's lemma.

Lemma 19 *Let φ be a monotone universal predicate and let Ω denote the set of effective protocols for φ, that is,*

$$\Omega = \{\mathcal{B} \mid \forall \rho \in \mathcal{R}^{\mathcal{FV}} \quad \rho \models_U \varphi \rightarrow h^{\mathcal{B}}\rho \models_U \varphi\}.$$

Then, there exists a minimal protocol in Ω.

Proof: We prove the existence of the minimal element in Ω by first showing that Ω is inductive, and by applying Zorn's lemma to this inductive set.

Let $C = \{\mathcal{B}^\alpha \in \Omega \mid \alpha \in I\}$ be a non-empty chain in Ω. We have to prove that C has a coproduct in Ω. As shown in lemma 15, C has a coproduct $\mathcal{B}^{\amalg}$ in the hierarchy. Furthermore, for every run ρ of $\mathcal{FV}$, if $\rho \models_U \varphi$, then $h^\alpha \rho \models_U \varphi$ for all $\alpha \in I$, and therefore, by theorem 16, $h^{\amalg}\rho \models_U \varphi$. Hence, $\mathcal{B}^{\amalg} \in \Omega$. ∎

Having settled the issue of existence of minimal effective protocols for predicates, we examine how efficient these protocols are in terms of average bit complexity. We consider two cases: In the first case the task defined by the universal predicate always terminates within a finite number of rounds, and, in the second case, the task always runs forever.

We proceed to formalize the notion of tasks that terminate within a finite number of rounds. Recall that two runs are *identical at round* ℓ, denoted by $\xi \overset{\ell}{=} \xi'$, if up to and including round ℓ both runs have identical external inputs and adversaries. A universal predicate φ *holds at a round* ℓ *in the run* ξ, if every run ξ' that is identical at round ℓ with ξ satisfies that predicate. Thus, φ holds at ℓ in ξ iff $\xi' \models_U \varphi$ for every run ξ' satisfying $\xi' \overset{\ell}{=} \xi$. A universal predicate φ is *finitely attainable for* $\mathcal{B}$ if for every run ξ of $\mathcal{B}$ there exists some round ℓ such that φ holds at round ℓ in ξ. Denote the least such round ℓ by $\mathsf{L}(\xi)$.

Consider a monotone universal predicate φ and let $\mathcal{B}$ be effective for that predicate. We proceed to show that a necessary and sufficient condition for φ to be finitely attainable for $\mathcal{FV}$ is that φ be finitely attainable for $\mathcal{B}$.

Lemma 20 *Let φ be a monotone universal predicate and let $\mathcal{B}$ be effective for φ. Then, φ is finitely attainable for $\mathcal{FV}$ iff it is finitely attainable for $\mathcal{B}$.*

Proof: Assume that φ is finitely attainable for $\mathcal{FV}$. Pick an arbitrary run ξ of $\mathcal{B}$. By theorem 2, there exists a run ρ of $\mathcal{FV}$ such that $\xi = h^{\mathcal{B}}\rho$. Since φ is finitely attainable for $\mathcal{FV}$, there exists some round ℓ such that φ holds at ℓ in ρ. Pick any run ξ' such that $\xi' \overset{\ell}{=} \xi$. A slight refinement of theorem 2 shows that there exists some run ρ' such that $\xi' = h^{\mathcal{B}}\rho'$ and $\rho' \overset{\ell}{=} \rho$. Hence, $\rho' \models_U \varphi$ and since $\mathcal{B}$ is effective for φ, $\xi' \models_U \varphi$.

Conversely, assume that φ is finitely attainable for $\mathcal{B}$. Pick a run ρ of $\mathcal{FV}$. Then, there exists a round ℓ such that for all ξ' satisfying $\xi' \overset{\ell}{=} h^{\mathcal{B}}\rho$, $\xi' \models_U \varphi$. Next, pick a run ρ' such that $\rho' \overset{\ell}{=} \rho$. Since $h^{\mathcal{B}}\rho' \overset{\ell}{=} h^{\mathcal{B}}\rho$, $h^{\mathcal{B}}\rho' \models_U \varphi$. Thus, by theorem 12, $\rho' \models_U \varphi$. ∎

Lemma 21 *Let the monotone universal predicate φ be finitely attainable for $\mathcal{FV}$ and let $\mathcal{B}$ and $\mathcal{C}$ be two effective protocols for φ such that $\mathcal{B} \succeq \mathcal{C}$. Then, for each run ξ of $\mathcal{B}$, $\mathsf{L}(\xi) = \mathsf{L}(h\xi)$.*

Proof: Notice that since $\mathcal{B}$ and $\mathcal{C}$ are effective protocols for φ, by lemma 20, the function L is well-defined on the runs of these two protocols. Next, notice that for proving that $\mathsf{L}(\xi) = \mathsf{L}(h\xi)$ for all runs ξ of $\mathcal{B}$, it is sufficient to show that for any run ρ of $\mathcal{FV}$, $\mathsf{L}(\rho) = \mathsf{L}(h^{\mathcal{B}}\rho)$. Finally, this last equality follows directly from the proof of lemma 20. ∎

Assume that the predicate φ is finitely attainable for $\mathcal{B}$ and consider the run ξ of $\mathcal{B}$. Let the *bit complexity of* ξ, $\Theta(\xi)$, be the number of bits transmitted by the correct processors in ξ up to and including round $\mathsf{L}(\xi)$.

Since the function Θ is measurable, we may define the *Average Protocol Complexity of protocol* $\mathcal{B}$ *with respect to* φ, denoted by $APC(\mathcal{B},\varphi)$, as

$$APC(\mathcal{B},\varphi) = \mathsf{E}\Theta(\xi) = \int \Theta(\xi)d\nu(\xi).$$

Next, we show that the probability measure on the runs of $\mathcal{FV}$ induces in a natural way a probability measure on the runs of any other protocol $\mathcal{B}$. First, we argue that

Lemma 22 *The morphism* $h\langle\mathcal{B},\mathcal{C}\rangle$ *is measurable.*

Notice that the initials can also be used to generate a *metrizable topology* on the set of runs, and that the morphisms are not only continuous in this topology but also Lipschitz. The most significant consequence of lemma 22 is that the probability measure $\nu^{\mathcal{FV}}$ on $\mathcal{R}^{\mathcal{FV}}$ induces in a natural way a probability measure on $\mathcal{R}^{\mathcal{B}}$. This measure is given by

$$\nu^{\mathcal{B}} = \nu^{\mathcal{FV}}(h^{\mathcal{B}})^{-1}.$$

Now, let the *Average Task Complexity* of a monotone universal predicate φ, denoted by $ATC(\varphi)$, be the minimum over each effective protocol for φ of the average protocol complexity of that protocol. Thus, ATC is given by

$$ATC(\varphi) = \min_{\mathcal{B} \text{ effective for } \varphi} APC(\mathcal{B},\varphi).$$

In the main theorem of this section we show that the average task complexity of every monotone universal predicate can be approximated as close as desired by *minimal* effective protocols for φ. Moreover, within each isomorphism class of every minimal effective protocol, we need only consider the protocol with least average bit complexity whose construction was given in section 8.2.

Theorem 17 *Let the monotone universal predicate φ be finitely attainable. Then, there exists a LAC minimal effective protocol for φ with average protocol complexity arbitrarily close to the average task complexity of φ.*

We first show that the average protocol complexity monotonically decreases with the relation $\succeq$ in Ω. We need the following technical result:

Lemma 23 *Assume that $\mathcal{B} \succeq \mathcal{C}$ and let $A \subseteq \mathcal{R}^{\mathcal{C}}$. Then,*

$$(h^{\mathcal{B}})^{-1} h^{\mathcal{B}} (h^{\mathcal{C}})^{-1} A = (h^{\mathcal{C}})^{-1} A.$$

Proof: Assume that $s \in (h^{\mathcal{B}})^{-1} h^{\mathcal{B}} (h^{\mathcal{C}})^{-1} A$. Then, there exists some s' such that both $h^{\mathcal{C}} s' \in A$ and $h^{\mathcal{B}} s = h^{\mathcal{B}} s'$. But $\mathcal{B} \succeq \mathcal{C}$, therefore $h^{\mathcal{C}} s = h^{\mathcal{C}} s'$. Hence, $s \in (h^{\mathcal{C}})^{-1} A$. ∎

Lemma 24 *Let the predicate φ be finitely attainable for $\mathcal{FV}$, and let $\mathcal{B}$ and $\mathcal{D}$ be any two effective protocols for φ such that $\mathcal{B} \succeq \mathcal{D}$. Then, for some protocol $\mathcal{C}$ isomorphic to $\mathcal{D}$,*

$$APC(\mathcal{B}, \varphi) \geq APC(\mathcal{C}, \varphi).$$

Proof: By theorem 9 there exists a protocol $\mathcal{C}$ isomorphic to $\mathcal{D}$ such that for every state $s^{\mathcal{B}} \in ST_p^{\mathcal{B}}$ and processor q,

$$|\mu_{p,q}^{\mathcal{B}} s^{\mathcal{B}}| \geq |\mu_{p,q}^{\mathcal{C}} h s^{\mathcal{B}}|$$

where the morphism h stands for $h\langle \mathcal{B}, \mathcal{C} \rangle$.
Pick now an arbitrary run ξ of $\mathcal{B}$. By lemma 21, $\mathsf{L}(\xi) = \mathsf{L}(h\xi)$; hence,

$$\Theta(\xi) \geq \Theta(h\xi) \geq 0.$$

For proving this lemma we now show that

$$\mathsf{E}\Theta(h\xi) = \mathsf{E}\Theta(\eta)$$

where the expectation on the left hand side is taken over runs ξ of $\mathcal{B}$ and on the right hand side over runs η of $\mathcal{C}$. In other words, we show that

$$\int_{\mathcal{R}^{\mathcal{B}}} \Theta(h\xi) d\nu^{\mathcal{B}}(\xi) = \int_{\mathcal{R}^{\mathcal{C}}} \Theta(\eta) d\nu^{\mathcal{C}}(\eta).$$

We use an argument similar to the one appearing in theorem C, section 39 in [H]. We only have to show that for any measurable set $A \subseteq \mathcal{R}^C$,

$$\int_{\mathcal{R}^B} \chi_A(h\xi) d\nu^B(\xi) = \int_{\mathcal{R}^C} \chi_A(\eta) d\nu^C(\eta)$$

where χ_A is the characteristic function of A in $\mathcal{R}^C$.
Indeed, by lemma 23

$$\begin{aligned} &\int_{\mathcal{R}^B} \chi_A(h\xi) d\nu^B(\xi) = \\ &= \nu^{\mathcal{F}\mathcal{V}}((h^B)^{-1} h^B (h^C)^{-1} A) \\ &= \nu^{\mathcal{F}\mathcal{V}}((h^C)^{-1} A) \\ &= \int_{\mathcal{R}^C} \chi_A(\eta) d\nu^C(\eta) \end{aligned}$$ ∎

Next, we state two simple lemmas, the first about inductive posets and the second about subsets of the real line.

Lemma 25 *Let Ω be an inductive poset, and let $\mathcal{M}$ be the set of all minimal elements in Ω. Then for each $\omega \in \Omega$ there exists an $m \in \mathcal{M}$ such that $\omega \succeq m$.*

Lemma 26 *Let A be a subset of $[0, \infty)$. Then, A has either a least point or a least cluster point.*

We complete now the proof of theorem 17.
Proof: Notice first that by lemma 24 the average protocol complexity monotonically decreases with $\succeq$ in Ω. Second, notice that by lemma 25 each protocol that is effective for φ dominates a minimal effective protocol for that predicate. Thus, in evaluating the average task complexity of φ, it is sufficient to consider only minimal effective protocols for that predicate whose existence is guaranteed by lemma 19. Furthermore, within each isomorphism class of minimal effective protocols one need only consider the *LAC* protocol constructed in section 8.2.

Let A be the set of average protocol complexities of *LAC* minimal effective protocols for φ. By lemma 26, A has either a least point or a least cluster point. Thus, there exists a *LAC* minimal effective protocol for φ, with average protocol complexity arbitrarily close to the average task complexity of φ. ∎

Finally, we consider the case where the task defined by φ runs forever. Here the bit complexity of the run ξ, $\Phi(\xi)$, might be given by the limsup over the round number ℓ of the (possibly weighted) average number of bits transmitted per round

by the correct processors in the first ℓ rounds of ξ. Recall that the lim sup of a sequence of measurable functions is measurable, see [H], section 20, theorem A; hence, Φ is also measurable and, therefore, our approach extends to this case as well.

9 $\mathcal{FV}$ is Optimal for Common Knowledge

In this section we show that, assuming comparable expressibility of adversaries, the full-view protocol along with all the other protocols in its isomorphism class attain *strictly* more common knowledge about monotone universal predicates than any other protocol in the hierarchy.

We impose some restrictions on the behavior of faulty processors in runs considered both in this section and in the following one. An arbitrary processor is either correct at all rounds, or otherwise, it is correct up to some round, after which it *malfunctions*, and thereafter it does not transmit at all. The actions of malfunctioning processors are fully determined by the adversary as in previous sections. We also bound the number of faulty processors by the parameter t. For avoiding degenerate cases we require that $t < n - 1$, where n denotes the number of faulty processors in the system.

Notice that whenever p and q are faulty at the same round ℓ in a run, the correct processors will never know what information p transmitted to q at ℓ. To eliminate this redundancy in the definition of runs, we assume in this section that for such pair of processors p and q at round ℓ,

$$AD(p, q, \ell) = \emptyset.$$

For the same reasons we assume that the faulty processors do not receive external inputs. These assumptions guarantee that in any pair of distinct runs of $\mathcal{FV}$, there always exists a processor and a round, such that either this processor is correct at that round in both runs and in different states in both runs, or the processor is correct at that round in one of the runs and faulty in the other. We now state and prove the main theorem of this section.

Theorem 18 *Let $\mathcal{B}$ be an arbitrary protocol such that $\mathcal{FV} \succ \mathcal{B}$. Then there exists a run ρ of $\mathcal{FV}$ and a monotone universal predicate φ such that*

$$\rho \models_U C_\ell \varphi$$

but

$$h\rho \not\models_U C_\ell \varphi.$$

Proof: Since $\mathcal{FV} \succ \mathcal{B}$, there exists a run ξ' of $\mathcal{B}$ such that

$$|h^{-1}\xi'| > 1.$$

Thus, there exist two runs ρ' and ρ'' of $\mathcal{FV}$ such that $h\rho' = h\rho'' = \xi'$, but $\rho' \neq \rho''$. Since $IN[\rho'] = IN[\rho'']$ and $CA[\rho'] = CA[\rho'']$, there exists a processor p and a round $\ell-1$ such that p is correct at $\ell-1$ in both ρ' and ρ'', and the states $s' = S[\rho'](p, \ell-1)$ and $s'' = S[\rho''](p, \ell-1)$ satisfy $s' \neq s''$.

We proceed to construct the monotone universal predicate φ. First, let

$$\psi = \{\rho \in \mathcal{R}^{\mathcal{FV}} \mid S[\rho](p, \ell-1) = s'\}.$$

Then, the monotone universal predicate φ is given by

$$\varphi = K_{(p,\ell-1)}\psi.$$

In order to guarantee that φ will be common knowledge in some run of $\mathcal{FV}$, we modify slightly the run ρ'. Denote by j the number of processors that are faulty up to and including round $\ell-1$ in ρ'. Pick now an arbitrary set of $t-j$ correct processors at $\ell-1$ in ρ', different from p, and let each of them become faulty at $\ell-1$ in ρ. Each of the malfunctioning processors at $\ell-1$ in ρ will transmit precisely as in ρ' with one exception: Let $q \neq p$ be a correct processor at ℓ in ρ whose existence is guaranteed since $n-1 > t$. We let each malfunctioning processor at $\ell-1$ in ρ transmit the empty message to q at $\ell-1$. Hence, by the end of round $\ell-1$, processor q will have seen precisely t empty messages, and therefore the views of the correct processors at ℓ in ρ are common knowledge there.

We prove in the following two assertions that the run ρ that we just constructed along with the universal predicate φ indeed satisfies the two requirements stated in the theorem.

Lemma 27

$$\rho \models_U C_\ell\varphi$$

Proof: Let the run ρ_m of $\mathcal{FV}$ be given by

$$\rho = \rho_0 \overset{(p_{i_1},\ell)}{\approx} \rho_1 \overset{(p_{i_2},\ell)}{\approx} \ldots \overset{(p_{i_m},\ell)}{\approx} \rho_m.$$

We have to show that $\rho_m \models_U \varphi$, that is $\rho_m \models_U K_{(p,\ell-1)}\psi$.

By the construction of ρ, the state of each correct processor at ℓ in ρ completely specifies the state of every other processor at ℓ in that run. By straightforward induction we have that p_{i_m} is correct at ℓ in both ρ and ρ_m, and also that

$$S[\rho_m](p_{i_m}, \ell) = S[\rho](p_{i_m}, \ell).$$

Now, since ρ_m is a run of $\mathcal{FV}$ and p is correct at $\ell - 1$ in ρ_m,

$$S[\rho_m](p, \ell - 1) = s'.$$

Thus, every run ρ'_m such that $\rho'_m \overset{(p,\ell-1)}{\approx} \rho_m$ also satisfies

$$S[\rho'_m](p, \ell - 1) = s'$$

showing that $\rho'_m \models_U \{\rho \mid S[\rho](p, \ell - 1) = s'\}$, or written differently, $\rho'_m \models_U \psi$. Consequently, $\rho_m \models_U K_{(p,\ell-1)}\psi$ and also $\rho \models_U C_\ell\varphi$ as required. ∎

Lemma 28

$$h\rho \not\models_U C_\ell\varphi$$

Proof: We prove this lemma by showing that $h\rho \not\models_U \varphi$. First, recall that $h\rho' = h\rho''$. Second, notice that p receives precisely the same messages up to and including round $\ell - 1$ in the runs ρ and ρ'. Hence, $h\rho \overset{(p,\ell-1)}{\approx} h\rho''$. Third, $\rho'' \not\models \psi$, since $\psi = \{\rho \mid S[\rho](p, \ell - 1) = s'\}$ and $s' \neq s''$. Thus, $h\rho'' \not\models_U \psi$, implying that $h\rho \not\models_U K_{(p,\ell-1)}\psi$ and, therefore, $h\rho \not\models_U \varphi$. ∎

This completes the proof of the theorem. ∎

Say that a protocol is *best for common knowledge* if it attains as much common knowledge about monotone universal predicates as any other protocol, assuming comparable expressibility of adversaries. In this light, theorem 18 asserts that best protocols for common knowledge not only exist, but, in fact, they are isomorphic to the full-view protocol. Theorem 18 does not estimate, however, the bit complexity of best protocols for common knowledge. An exponential lower bound for this complexity follows directly from theorem 20 in the following section.

Corollary 3 *Best protocols for common knowledge require, in the worst case, exponentially in t long messages.*

10 The Lower Bound for SBA

Assume that the message set of the protocols in the hierarchy is the set of finite binary strings. We show that, subject to comparable expressibility of adversaries,

every protocol that is as effective for Simultaneous Byzantine Agreement (SBA) as $\mathcal{FV}$ requires in the worst case exponential in t communication.

SBA is a variant of the classical Byzantine Agreement problem introduced in [PSL]. Assume that the external inputs at round 0 are in $\{0,1\}$. Say that a protocol *attains SBA* if the following four conditions hold in every run of that protocol:

- ⋆ Every correct processor commits to either 0 or 1.
- ⋆ All the correct processors commit to the same value.
- ⋆ If the external inputs at round 0 are identical, then all the correct processors commit to this common input.
- ⋆ The correct processors commit simultaneously, that is, at the same round.

The notion of effectiveness for SBA is motivated by the following result,

Theorem 19 *Let ρ be a run of $\mathcal{FV}$. If SBA is attained at ℓ in $h\rho$, then it is also attained at that same round in ρ.*

Proof: Assume by contradiction that SBA is *not* attained at ℓ in ρ. Then there are two runs ρ' and ρ'' of $\mathcal{FV}$ such that both $\rho \overset{\ell}{\sim} \rho'$ and $\rho \overset{\ell}{\sim} \rho''$, and whereas the initial inputs in ρ' are all 1, the initial inputs in ρ'' are all 0.

But since $\sim$ is the transitive closure of the relations $\approx$ and by lemma 2, $h\rho \overset{\ell}{\sim} h\rho'$ and, similarly, $h\rho \overset{\ell}{\sim} h\rho''$. Thus, SBA is also *not* attained at ℓ in $h\rho$–a contradiction. ∎

A protocol $\mathcal{B}$ is *effective* for SBA if the converse of theorem 19 holds, that is, whenever SBA is attained at a round in a run ρ of $\mathcal{FV}$, then it is also attained at that same round in the morphic image $h^{\mathcal{B}}\rho$ of that run. Throughout this section h will stand for $h^{\mathcal{B}}$. We first prove the following technical result:

Lemma 29 *Let $\mathcal{B}$ be an effective protocol for SBA. Let $s^{\mathcal{B}}$ be a state of $\mathcal{B}$ for which there exist a run ρ of $\mathcal{FV}$, a processor $p \in P$, and a round number $\ell > 1$, such that p is correct at ℓ in ρ, the state $s = S[\rho](p, \ell - 1)$ satisfies $s^{\mathcal{B}} = hs$ and SBA is* not *attained at $\ell + 1$ in ρ. Then, the following two hold:*

1. $|h^{-1}s^{\mathcal{B}}| = 1$.
2. *For every $r \neq p$ correct at $\ell + 1$,*

$$h\rho \models K_{(r,\ell)} \text{“}\textit{if } p \textit{ is correct at } \ell\textit{, then its state at } \ell - 1 \textit{ is } s^{\mathcal{B}}\text{”}.$$

Proof:
1. Assume by contradiction that $|h^{-1}s^{\mathcal{B}}| > 1$, that is, there exists a state s' of $\mathcal{FV}$ such that both $s' \neq s$ and $hs' = hs = s^{\mathcal{B}}$.

Since SBA is not attained at $\ell+1$ in ρ, there exists (by appendix C in [M] and section 6.1 in [MT]) a run ρ' of $\mathcal{FV}$ such that $\rho' \overset{\ell+1}{\sim} \rho$, which differs from ρ in at most the following: First, at least one processor other than p is malfunctioning at ℓ in ρ'. Second, denote by j the number of malfunctioning processors at ℓ in ρ. Then, there exists a set of cardinality $t-j-1$ of correct processors in ρ distinct from p, which are malfunctioning at ℓ in ρ'. Third, p is correct at ℓ in ρ', but does not transmit at all at $\ell+1$ in ρ'. Finally, all the malfunctioning processors at ℓ in ρ' transmit at ℓ in ρ' precisely as in ρ at ℓ with one possible exception: Each of them transmits the empty message to some other processor $q \neq p$ that is correct at $\ell+1$. The existence of such a processor q is guaranteed since $n-1 > t$.

Recall that by assumption SBA is *not* attained at $\ell+1$ in ρ, thus, since $\rho' \overset{\ell+1}{\sim} \rho$, SBA is also *not* attained at $\ell+1$ in ρ'. Next, since $\mathcal{FV} \succeq \mathcal{B}$, SBA is also *not* attained at $\ell+1$ in $h\rho'$.

We now construct a run ρ'' which differs from ρ' only in that p malfunctions at ℓ in ρ'' and the adversary corresponding to p at that round is given by:

$$\begin{aligned} S[\rho''](p, l-1 \mid p \xrightarrow{\ell} q) &= s' \\ S[\rho''](p, l-1 \mid p \xrightarrow{\ell} u) &= s \qquad \text{for } u \neq p, q \end{aligned}$$

The crucial property satisfied by ρ'' is

Lemma 30 *SBA* is *attained at* $\ell+1$ *in* ρ''.

Proof: First, notice that all the correct processors at $\ell+1$ in ρ'' receive t empty messages and therefore they know that the information conveyed through the non-empty messages they receive at $\ell+1$ is trustworthy. Furthermore, each of them knows that precisely t processors are faulty at ℓ. Indeed, q conveys to all of them at $\ell+1$ that it received $t-1$ empty messages at ℓ, and that p conveyed to it at ℓ that p's state at $\ell-1$ was s'. Next, there exists at least one other correct processor $u \neq q$ at $\ell+1$, and u conveys to all processors at $\ell+1$ that p conveyed to it that p's state at $\ell-1$ was s.

But $s \neq s'$, thus, all the correct processors know that p malfunctioned at ℓ, since p transmitted asymmetrical messages. Therefore, the states of the correct processors at $\ell+1$ in ρ'' are common knowledge there, and SBA is attained at $\ell+1$ in ρ''. ∎

However,

Lemma 31 *SBA is* not *attained at* $\ell+1$ *in* $h\rho''$.

Proof: Since $hs = hs'$, exactly the same messages are transmitted in the runs $h\rho'$ and $h\rho''$. Now, since SBA is *not* attained at $\ell+1$ in $h\rho'$, SBA is also *not* attained at $\ell+1$ in $h\rho''$. ∎

Finally, it follows from these two assertions that $\mathcal{B}$ is *not* effective for SBA–a contradiction.

2. We have to prove that for every processor $r \neq p$ correct at $\ell+1$,

$$h\rho \models K_{(r,\ell)}\text{"if } p \text{ is correct at } \ell, \text{ then its state at } \ell-1 \text{ is } s^{\mathcal{B}}\text{"}.$$

Assume by contradiction that the above is false. Thus, there exists a run ρ' of $\mathcal{FV}$ such that $h\rho' \overset{(r,\ell)}{\approx} h\rho$, p is correct at ℓ in $h\rho'$, but the state $s'^{\mathcal{B}}$ given by $s'^{\mathcal{B}} = S[h\rho'](p,\ell-1)$ satisfies $s'^{\mathcal{B}} \neq s^{\mathcal{B}}$. Informally, as far as r knows at ℓ in $h\rho$, p could have been correct at ℓ, and in either state $s^{\mathcal{B}}$ or $s'^{\mathcal{B}}$ or some other state. Now, denote by s and s' the states $S[\rho](p,l-1)$ and $S[\rho'](p,\ell-1)$ respectively, and notice that since $s^{\mathcal{B}} \neq s'^{\mathcal{B}}$, also $s \neq s'$. The proof, and in particular the construction of ρ'', proceed now exactly as in part one. ∎

Equipped with this lemma we now prove the central result in this section which asserts that every effective protocol for SBA requires, in the worst case, exponential in t communication.

Theorem 20 *Any effective protocol for SBA requires, in the worst case, exponentially in* t *long messages.*

Proof: Let $\mathcal{B}$ be an effective protocol for SBA. We will show that there exists a run of that protocol in which some correct processor transmits exponentially in t many bits before committing to the decision value of SBA.

Consider the following basic construction for runs of $\mathcal{FV}$: Assume that $t \geq 5$ and let f be given by

$$f = \lfloor (t-3)/2 \rfloor.$$

Let the processors p_{2i-1} and p_{2i} malfunction at round i, for $i = 1,\ldots,f$, but let each of them follow $\mathcal{FV}$ there. The processor p_{2f+1} will malfunction at $f+1$ and it will also transmit according to $\mathcal{FV}$ there to all the other processors excluding p_{n-1} to which it will convey some, and possibly many, actual lies as follows: Processor p_{2f+1} will forge at $f+1$ a content for every chain of processors given by

$$p_{i_1} \rightarrow p_{i_2} \rightarrow \ldots \overset{f}{\rightarrow} p_{i_f}$$

and satisfying either $p_{i_j} = p_{2j-1}$ or $p_{i_j} = p_{2j}$, for each $j = 1, \ldots, f$. Two remarks are worth mentioning here: First, the state of p_{2f+1} at f determined by each set of forged content for these chains is in $ST^{\mathcal{F}\mathcal{V}}_{p_{2f+1},f}$. Second, there are exponentially in t many such chains, say c^t, for some $c > 1$, and therefore 2^{c^t} possible states for p_{2f+1} at round f.

Denote by ρ_j, for $j \in \{1 \ldots 2^{c^t}\}$, the run whose first f rounds are as described above, and in which p_{2f+1} transmits to p_{n-1} at $f+1$ by applying the message generator to the j^{th} state. Finally, let the processors p_{2f+2} and p_{2f+3} malfunction at $f+2$ and $f+3$ respectively, and let both of them follow $\mathcal{F}\mathcal{V}$ while malfunctioning. Notice that the number of faulty processors up to and including $f+3$ in each of the runs ρ_j is $2f+3$, and since

$$2f + 3 \leq 2\lfloor (t-3)/2 \rfloor + 3 \leq t$$

at most t processors fail in each run ρ_j. Finally, let the processors p_{n-1} and p_n be correct at all rounds in each of these runs.

Lemma 32 *The states $S[h\rho_j](p_{n-1}, f+1)$ for $j \in \{1 \ldots 2^{c^t}\}$ are all distinct.*

Proof: Assume by contradiction that

$$S[h\rho_j](p_{n-1}, f+1) = S[h\rho_k](p_{n-1}, f+1)$$

for $j, k \in \{1 \ldots 2^{c^t}\}$. Then, by theorem 1,

$$hS[\rho_j](p_{n-1}, f+1) = hS[\rho_k](p_{n-1}, f+1) = S[h\rho_j](p_{n-1}, f+1)$$

and therefore

$$|h^{-1}S[h\rho_j](p_{n-1}, f+1)| > 1.$$

But the protocol $\mathcal{B}$ is effective for SBA and the state $S[h\rho_j](p_{n-1}, f+1)$ satisfies the conditions specified in lemma 29–a contradiction to part 1 of that lemma. ∎

Lemma 33 *There exists an integer $j \in \{1 \ldots 2^{c^t}\}$ such that processor p_{n-1} transmits at least c^t bits to processor p_n at $f+2$ in $h\rho_j$.*

Proof: Assume by contradiction that for all $j \in \{1 \ldots 2^{c^t}\}$

$$|M[h\rho_j](p_{n-1}, p_n, f+2)| < c^t.$$

Since there are at most $2^{c^t} - 2$ such messages, but 2^{c^t} distinct runs ρ_j, by the pigeonhole principle there are two indices j and k in $\{1 \ldots 2^{c^t}\}$ such that

$$M[h\rho_j](p_{n-1}, p_n, f+2) = M[h\rho_k](p_{n-1}, p_n, f+2).$$

Now, since corresponding processors, distinct from p_{n-1}, in the runs $h\rho_j$ and $h\rho_k$ are in identical states at round $f+1$ we have

$$S[h\rho_j](p_n, f+2) = S[h\rho_k](p_n, f+2).$$

But by lemma 32,

$$S[h\rho_j](p_{n-1}, f+1) \neq S[h\rho_k](p_{n-1}, f+1)$$

hence, processor p_n does *not* know at $f+2$ in $h\rho_j$ the state of p_{n-1} at $f+1$. But this contradicts part 2 of lemma 29 applied to the state $S[h\rho_j](p_{n-1}, f+1)$. ∎

This assertion completes the proof of theorem 20. ∎

11 The Universality of $\mathcal{FV}$

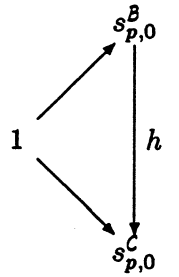
In this section we prove that the full-view protocol satisfies the universal property. In order to make the universality assertion precise, we need to specify carefully the categories and the functors that we employ.

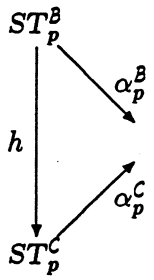
Definition 8 *Let* DS *denote the category whose objects are protocols and whose morphisms are maps* $\mathsf{H} : \mathcal{B} \to \mathcal{C}$ *such that* $\mathsf{H} = \langle h, h_\iota \rangle_\iota$ *for*

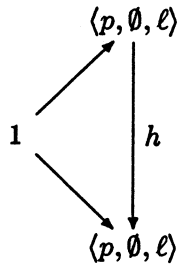
$$h : ST_p^{\mathcal{B}} \to ST_p^{\mathcal{C}} \qquad \text{and} \qquad h_\iota : \Sigma^{\mathcal{B}} \to \Sigma^{\mathcal{C}},$$

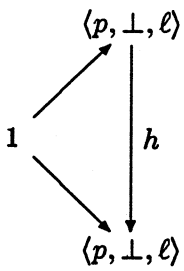
and the following diagrams commute:

$$\begin{array}{ccc} \prod ST_q^{\mathcal{B}} \times \Sigma^{\mathcal{B}} & \xrightarrow{\quad\mathcal{B}\quad} & ST_p^{\mathcal{B}} \\ \Big\downarrow{\scriptstyle h^n \times h_\iota} & & \Big\downarrow{\scriptstyle h} \\ \prod ST_q^{\mathcal{C}} \times \Sigma^{\mathcal{C}} & \xrightarrow[\quad\mathcal{C}\quad]{} & ST_p^{\mathcal{C}} \end{array}$$









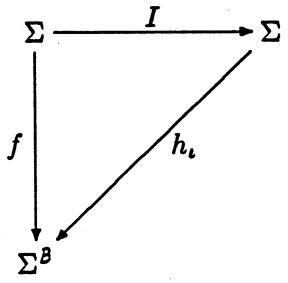
We leave it for the reader to verify that the category DS given in this definition indeed satisfies the axioms of category theory. Furthermore, notice that by theorem 6, this definition coincides with definition 4. Next, consider the category INPUT whose objects are input sets and whose morphisms are maps $\Sigma^{\mathcal{B}} \rightarrow \Sigma^{\mathcal{C}}$, where $\Sigma^{\mathcal{B}}$ and $\Sigma^{\mathcal{C}}$ are objects in INPUT. Finally, consider the very forgetful functor $\mathsf{F} : \mathsf{DS} \rightarrow \mathsf{INPUT}$ given in the obvious way. Now the universality statement can be made precise.

Theorem 21 *For $\Sigma \in$ INPUT, let $\mathcal{FV}$ denote the full-view protocol with input set Σ, and let I denote the identity morphism from Σ to itself. Then, the pair $\langle \mathcal{FV}, I \rangle$ is universal from Σ to the functor* F.

Proof: Let $\mathcal{B}$ be an object in DS with input set $\Sigma^{\mathcal{B}}$, and let $f \in \hom(\Sigma, \Sigma^{\mathcal{B}})$. We must show that there exists a unique morphism

$$\mathsf{H} = \langle h, h_\iota \rangle : \mathcal{FV} \rightarrow \mathcal{B}$$

such that the following diagram commutes.



Since H is a morphism in DS we have the following recursive equations

$$hs^{\mathcal{FV}}_{p,\ell} = \delta^{\mathcal{B}}_p \left(hs^{\mathcal{FV}}_{p,\ell-1} \ldots \mu^{\mathcal{B}}_{q,p} hs^{\mathcal{FV}}_{q,\ell-1} \ldots h_\iota \iota \right)$$

where the root of the state tree $s^{\mathcal{FV}}_{p,\ell}$ is labelled $\langle p, \iota, \ell \rangle$ and the q^{th} principal subtree of $s^{\mathcal{FV}}_{p,\ell}$ is $s^{\mathcal{FV}}_{q,\ell-1}$. The initial conditions for these recursive equations are given by

$$hs^{\mathcal{FV}}_{p,0} = s^{\mathcal{B}}_{p,0}.$$

Hence, by induction on the round number ℓ, the function h exists and is unique; therefore, the morphism H also exists and is unique. ∎

Acknowledgements: I am indebted to Neil Immerman for very substantial contributions throughout the development of this work. I thank Jerrold Leichter for untiring support. I am grateful to Michael Fischer and George Seligman. I also thank Joyce Gastel, David Greenberg, Chun-Chung Hsieh, Gregory Kozlovsky, Abhiram Ranade, Zeév Rudnick and Allan Woods. I thank Michael Barr for allowing me to use his TEX routines for category theory. I thank Moshe Vardi for encouraging me to find an alternate definition for the notion of ck-informative introduced in [M].

References

[AM] Arbib M., Manes E. "Arrows, Structures, and Functors. The Categorical Imperative." *Academic Press*, (1975).

[H] Halmos P. "Measure Theory." *D. Van Nostrand Company, Inc.* (1961).

[J1] Jacobson N. "Basic Algebra 1." *Freeman and Company*, San Francisco, (1974).

[J2] Jacobson N. "Basic Algebra 2." *Freeman and Company*, San Francisco, (1980).

[LFF] Lynch N., Fischer M., Fowler R. "A Simple and Efficient Byzantine Generals Algorithm." *Proc. of the 2nd Symp. on Reliability in Distributed Software and Database Systems.*, (1982), 46–52.

[M] Michel R. "Efficient Protocols for Common Knowledge and Simultaneous Byzantine Agreement." *YALEU/DCS/TR-603*, Yale University, (Feb. 1988).

[ML] Mac Lane S. "Categories for the Working Mathematician." *Springer-Verlag*, (1971).

[MT] Moses Y., Tuttle M. "Programming Simultaneous Actions Using Common Knowledge." *Algorithmica*, Springer-Verlag, New York, (1988), 3: 121–169.

[PSL] Pease M., Shostak R., Lamport L. "Reaching Agreement in the Presence of Faults." *JACM, 27:2.*, (1980), 228–234.