

シーケンス図に基づく通信システム仕様記述法 HSC と その支援環境

黄 錦 法[†] 白 鳥 則 郎[†]

シーケンス図は、通信システムの仕様を高い理解性と共にシステム全体を見通し良く開発していくことができるため、仕様要求段階で用いられることが多い。しかし、このような記述法には以下のような短所がある。①仕様が大きくなるにつれ、仕様全体の理解性が低下する。②図同士の関係を明解に表現することが困難である。③繰り返しと条件分岐の記述を簡潔・明瞭に表現することが困難である。そこで、本論文では、従来のシーケンス図を基本とし、シーケンス図の持つ長所を保存しつつ、その短所を改善する仕様記述法『HSC (Hierarchical Sequence Chart)』を設計、開発する。HSCの特徴は、通信システムの特徴を考慮した階層化とモジュール化の概念の導入により、仕様全体の理解性の向上を達成していることにある。また、HSCに基づいた仕様を効率的に記述できるように HSC 向きの仕様記述スタイルを考案すると共に、ユーザフレンドリィな支援環境を構築し、具体的な仕様記述を通してその有効性を示している。

A Specification Method HSC Based on Sequence Chart for Communication System and Its Support Environment

CHING-FA HUANG[†] and NORIO SHIRATORI[†]

The specification of communication systems can be described by sequence charts with easy understandability and clear visibility of the whole system. This leads to the use of sequence charts in the requirement stage of development of specifications. But the sequence chart has the following shortcomings: ① the understandability of the whole specification reduces as the size of the specification grows up. ② It is difficult to describe the relation among different sequence charts. ③ It is difficult to represent precisely and clearly the iteration and decision functions. In this paper, we propose a specification method HSC (Hierarchical Sequence Chart) based on the sequence chart. In HSC we make up the shortcomings of the sequence chart, preserving its merits. The main feature of HSC is the improvement of the understandability of the whole specification. This is accomplished by introducing hierarchy and modularity, the concepts which are considered to be the characteristics of communication systems. Further, for effectively describing the specification by HSC, we work out a specification style for HSC and create the user friendly support environment. The validity of the support environment is established through a concrete specification example.

1. はじめに

仕様の開発段階は、①仕様要求段階、②仕様記述段階、③仕様インプリメンテーションなどの段階に分けられる。LOTOS⁴⁾などの通信システムの仕様記述言語は仕様記述段階で適用され、記述内容の厳密性や検証性に重点をおいて開発されているため、理解性に難があり、その支援環境の構成が重要となっている^{5), 6)}。また、このような言語は、仕様要求段階で用いることには向いていない。そのため、システムの挙動を

表す構成要素間の信号(情報)のやりとりを視覚的に分かりやすく表現可能なシーケンス図に基づいた記述法^{1)~3)}が仕様要求段階で用いられてきている。

しかし、シーケンス図には以下のような短所がある。①仕様が大きくなるにつれ、仕様全体の理解性が低下する。②図同士の関係を明解に表現することが困難である。③繰り返しと条件分岐の記述を簡潔・明瞭に表現することが困難である。そこで、本論文では、従来のシーケンス図を基本とし、シーケンス図の持つ長所を保存しつつ、その短所を改善する仕様記述法『HSC (Hierarchical Sequence Chart)』を設計、開発する。また、HSCに基づいた仕様を効率的に記述できるように HSC 向きの仕様記述スタイルを考案する

[†] 東北大学工学部情報工学科
Department of Information Engineering, Faculty
of Engineering, Tohoku University

と共に、ユーザフレンドリな支援環境を構成する。

HSC では、仕様要求段階において要求仕様をトップダウン的に開発するための手法として、階層構造的に、特に大枠で通信システムの特徴を考慮し、3段階に分け、これらの間の関係を陽に記述し仕様の全体が分かりやすくなるよう、また、仕様の再利用がしやすいように工夫をしている。このような工夫により、HSC の特徴は、仕様をトップダウン的に開発でき、仕様全体の理解性の向上を達成し、さらに仕様の部品の再利用に基づいた仕様の開発が可能となっている。また、HSC 向きの仕様記述スタイルを支援するユーザフレンドリな支援環境を試作し、これを用いて仕様記述実験を行った。さらに、これらの記述例を通して他の関連する記述法と比較することにより、HSC の有効性を示す。

以下、2章では HSC の基本概念について述べ、3章では仕様記述法 HSC の具体的な設計を行う。4章では HSC の支援環境を試作する。5章では適用例と評価について述べる。

2. HSC の基本概念

2.1 設計方針

本論文で提案する HSC は、従来のシーケンス図を基本とし、シーケンス図の持つ長所を保存しつつ、その短所を改善するために、次の3点を基本方針とし設計されている。

(1) 階層化とモジュール化

通信システムの特徴を考慮した階層化とモジュール化の概念の導入により、従来のシーケンス図では記述が困難であった図同士の関係を陽に表現する。

(2) 繰り返しと条件分岐

繰り返しと条件分岐の表現を導入し、従来のシーケンス図ではうまく記述できなかったこれらの表現を明確に行う。

(3) 信号（情報）の間の関係

記述性と理解性をより高めるため、信号（情報）の性質を定義し、その間の関係を導入する。これにより、仕様における信号（情報）の間の関係や記述内容が把握しやすくなる。

2.2 準備

2.2.1 用語

HSC を設計する上で使用する次の2つの用語を導入する。

(1) ノード：端末、コンピュータ、交換機などの通

信システムの構成要素を抽象化したもの。

(2) メッセージ：ノード間で交換される制御やデータのための信号（情報）を抽象化したもの。

2.2.2 メッセージの分類

制御のためのメッセージを、動作の要求、結果の通知および状態の通知の観点から、次の3種類に分類する。

(1) 操作要求メッセージ：あるノードが他のノードに所定の動作を要求するメッセージ。例えば、通信したいノードが相手ノードに通信路の接続の動作を要求するメッセージなどがある。

(2) 結果通知メッセージ：操作要求メッセージを受信したノードが、それに対する結果を知らせるメッセージ。転送されたデータに対する受信確認のための応答も結果通知メッセージとして扱う。例えば、(1)で述べた例において通信路の接続の結果を知らせるメッセージなどがある。

(3) 状態通知メッセージ：あるノードが自分の状態を他のノードに知らせるメッセージ。例えば、(1)で述べた例において通信路の接続要求があった時、受信用バッファに関して空あるいはビジーなどの状態を知らせるメッセージなどがある。

なお、転送されるデータについては、受信確認の必要がある場合とない場合に応じて、それぞれ操作要求メッセージと状態通知メッセージとして扱う。

2.2.3 メッセージ間の関係

2.1 節の(3)で述べた信号（情報）の間の関係としてメッセージ間に次の2つの関係を導入する。

(1) 操作要求と結果通知のペアの関係

操作要求メッセージと結果通知メッセージは、通常ペアで用いられるため、操作要求メッセージと対応する結果通知メッセージの間にはペアの関係があると呼ぶ。例えば、図1に示すように操作要求メッセージ M_1 とその結果通知メッセージ M_3 の間にはペアの関係がある。また、操作要求メッセージ M_2 とその結果通知メッセージ M_4 の間にもペアの関係がある。これは、あるノードが相手ノードにメッセージを送信し

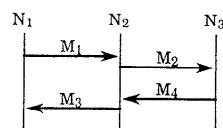


図1 操作要求と結果通知のペアの関係
Fig. 1 The pair relation of operation request message and result information message.

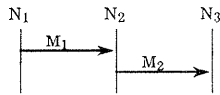


図 2 状態伝搬
Fig. 2 State propagation.

た結果、相手ノードからの返事を示す場合に利用される。例えば、受信確認の必要なデータの送信とその応答を表すために利用される。

(2) 状態伝搬の関係

状態通知メッセージを受信したノードが、その直後に、受信したメッセージに基づいて送信元以外のノードへ状態通知メッセージを送信する場合がある。この時、ノードが受信した状態通知メッセージとその直後に送信した状態通知メッセージの間には状態伝搬の関係があると呼ぶ。例えば、図 2 に示すようにノード N_2 は状態通知メッセージ M_1 を受信した直後に M_1 に基づいて状態通知メッセージ M_2 を送信する。この時、状態通知メッセージ M_1 と M_2 の間には状態伝搬の関係がある。これは、あるノードにおいて、受信したメッセージを、送信元へ応答を返さずに、他のノードに送信する場合に利用される。例えば、受信確認が必要のないデータの中継を表すために利用される。

2.3 仕様構造

HSC による仕様記述は、2.1 節の(1)の階層化とモジュール化に基づいて図 3 に示すように次の 3 つに階層化される。

(1) 仕様枠組

仕様記述は、図 3 に示すように、まず、いくつかのフェーズから構成される仕様枠組を記述する。例えば、基本電話サービスの仕様枠組は、接続、通話・切

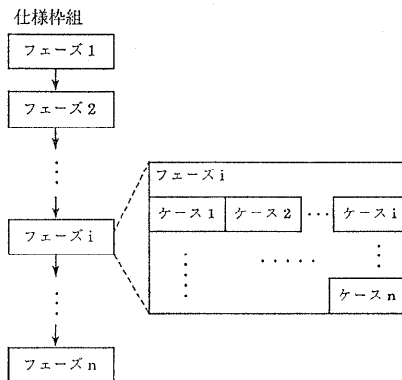


図 3 HSC の仕様構造
Fig. 3 Specification structure of HSC.

断の 2 つのフェーズから構成される。

(2) フェーズ

フェーズは、図 3 に示すようにいくつかのケースから構成される。例えば、(1)で述べた基本電話サービスの接続フェーズにおいて接続の成功や失敗などの場合の動作をケースとして記述する。さらに、ケース間の関係を 3.1.2 項で導入する選択と並列の関係をを用いて記述する。

(3) ケース

ケースの記述では、ノード間で交換されるメッセージのシーケンスをシーケンス図に基づいた 3.1.4 項で導入する形式で記述する。例えば、(2)で述べた例において、成功の場合のメッセージの具体的なやりとりのシーケンスを記述する。

3. 仕様記述法 HSC

本章では、2 章で与えた HSC の基本概念に基づき、仕様記述法 HSC の設計を行う。

3.1 HSC の言語仕様

HSC の仕様は、2.3 節で述べたように仕様枠組、フェーズおよびケースの 3 つの部分から構成される。ここでは、これらの 3 つについて具体的な言語仕様を与える。

3.1.1 仕様枠組

仕様枠組は図 4 のように表現される。まず、記述する仕様の名前(この例では“仕様名”)を SPEC の後に次のように宣言する。

SPEC 仕様名

また、仕様を構成する各フェーズはその実行順序に従って上から下へ向かって順番に記述される。なお、仕様枠組が複数のページにわたる場合は通常のフロー

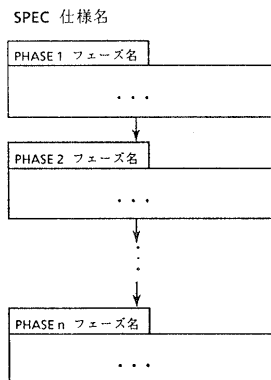


図 4 仕様枠組の表現
Fig. 4 Representation of the specification framework.

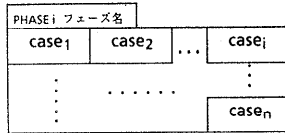


図 5 フェーズの表現
Fig. 5 Representation of the phase.

チャートと同様に円の中に数字を記入した接続記号を用いて表現する。

3.1.2 フェーズ

フェーズは、図 5 に示すようにいくつかのケースから構成される。フェーズの記述は、枠組と内容に分けられる。枠組の記述では、図 5 に示すように、**PHASE** の後にフェーズの番号（この例では “i”）とフェーズの名前（この例では “フェーズ名”）を宣言する。フェーズの内容はケースとケース間の関係を示すオペレータを用いて記述する。ここで、ケース間の関係として、次の定義する選択と並列の 2 つを導入する。

(1) 選択：各フェーズにおける複数のケースの中で 1 つだけが生起可能な場合を選択の関係と呼び、表 1 に示すオペレータで表現する。

条件分岐は、ケース間の選択の関係を用いて表現される。例えば、式 (1) では、フェーズ i におけるケース 1 からケース 4 の関係は選択オペレータで表現されている。この場合、フェーズ i ではケース 1 からケース 4 の中で分岐の条件に応じて 1 つだけが生起する。式 (1) の図式表現を図 6 の①に示す。

$$\text{フェーズ } i = \text{ケース } 1 \mid \text{ケース } 2 \mid \text{ケース } 3 \mid \text{ケース } 4$$

(1)

(2) 並列：各フェーズにおける複数のケースが同時に生起可能な場合を並列の関係と呼び、表 1 に示すオペレータで表現する。ここで、並列オペレータは選択オペレータより優先度が高いものとする。

以下に示す式 (2) のフェーズ k では並列オペレータ

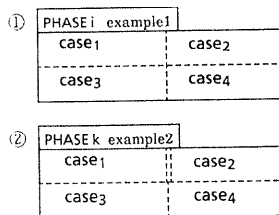


図 6 選択と並列オペレータの例
Fig. 6 Examples of 'or' and 'parallel' operations.

表 1 ケース間の関係のオペレータ
Table 1 Relational operators between cases.

関係	オペレータ
選択	$\mid$
並列	$\parallel$

と選択オペレータが混在している。この場合、フェーズ k ではケース 1 とケース 2 が同時に生起可能である。並列オペレータは選択オペレータより優先度が高いため、ケース 1 とケース 2 は 1 つのまとまったケースとして扱われる。従って、このひとまとまりのケース、ケース 3、ケース 4 の関係は選択オペレータで表現されており、これらのケースの中で 1 つだけが生起する。式 (2) の図式表現を図 6 の②に示す。

$$\text{フェーズ } k = \text{ケース } 1 \parallel \text{ケース } 2 \parallel \text{ケース } 3 \mid \text{ケース } 4 \quad (2)$$

フェーズは、さらにサブフェーズに段階的に分けることができる。各サブフェーズの記述は、これまで述べたフェーズの場合と同様である。ここで、ケースとサブフェーズのレベルは分離されており、これらを同一のフェーズ（サブフェーズ）内に記述することはできない。

3.1.3 ケース

ケースの記述は、その枠組と内容に分けられる。ケースの内容は 3.1.4 項で述べるシーケンス図を基本とした記法で表現する。本項では、枠組の記法について述べる。

ケースの枠組は、①番号、②名前、③行き先からなる。図 7 に示すように、まず、**CASE** の後にケースの番号（この例では “i”）とケースの名前（この例では “ケース名”）を宣言する。次に、枠組の右下隅にこのケースに続くフェーズを指定する。このような指定として、フェーズの遷移、フェーズの繰り返しおよび動作終了の 3 つを導入する。

(1) フェーズの遷移：次の動作として別のフェーズへ遷移することをフェーズの遷移と呼び、表 2 に示す記号で表現する。ここで、表中に示している k は遷移先のフェーズの番号を示す。ただし、遷移先が次のフ

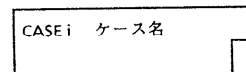


図 7 ケースの枠組の表現
Fig. 7 Representation of the framework of case.

表 2 ケースの行き先の記号
Table 2 Symbols for destination of case.

名 前	記 号
フェーズの遷移	$\downarrow k$
フェーズの繰り返し	$\hookrightarrow$
動作終了	$\times$

フェーズの場合に限り、 k を省略することができる。

(2) フェーズの繰り返し：次の動作として自分が属するフェーズを繰り返すことをフェーズの繰り返しと呼び、表 2 に示す記号で表現する。

(3) 動作終了：次の動作として仕様全体の動作が終了することを動作終了と呼び、表 2 に示す記号で表現する。

3.1.4 シーケンス図

シーケンス図の記述は、枠組と内容に分けられる。枠組は、図 8 に示すように、シーケンス図の名前として、PHASE の後にフェーズの番号（この例では“ k ”）とフェーズの名前（この例では“フェーズ名”）、その後に / を挿入し、さらに CASE の後にケースの番号（この例では“ j ”）とケースの名前（この例では“ケース名”）を宣言する。枠組の右下隅にはケースの行き先を書く。

シーケンス図の内容の記述として、次の 3 種類の記号を導入する（表 3）。

(1) ノード：表 3 に示すように縦線を用いてノードの動作を示す。ここで、縦線は時間軸を示し、特に指定がない場合は、上から下へと経過する（順序が任意の場合は、(3) で述べる任意順序を用いる）。縦線の上にはノードの名前（この例では“ N_i ”）が書かれる。

(2) メッセージ：2.2.2 項で導入したメッセージの種別を表 3 に示すように送信ノード（ N_i ）から受信ノード（ N_j ）へ向かう線の種類によって区別する。メッセージ自身は線の上に記述される。図 9 にメッセージの記述例を示す。この例では、NodeA から NodeB へ操作要求メッセージ M_1 が送られ、その後、NodeB から NodeA へ結果通知メッセージ M_2 が返

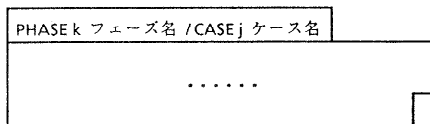


図 8 シーケンス図の枠組の表現
Fig. 8 Representation of the framework of sequence chart.

表 3 ノード、メッセージおよび任意順序の記号
Table 3 Symbols for node, messages, and arbitrary order.

名 前	記 号
ノード	N_i
メッ セー ジ	操作要求メッセージ $N_i \xrightarrow{M} N_j$
	結果通知メッセージ $N_i \xrightarrow{M} N_j$
	状態通知メッセージ $N_i \xrightarrow{M} N_j$
任意順序	N_i $M \rightarrow$

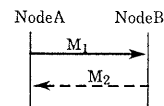


図 9 メッセージの記述例
Fig. 9 An example of message transfer.

Step 1 仕様枠組の記述

Step 1.1 仕様の名前の記述

Step 1.2 仕様枠組の内容の記述

仕様枠組を構成する各フェーズをそれぞれ Step 2 に従って記述する。

Step 2 フェーズの記述

Step 2.1 フェーズの枠組の記述

Step 2.2 フェーズの内容の記述

フェーズを構成する各ケースをそれぞれ Step 3 に従って記述する。ケース間には選択や並列の関係を与える。

Step 3 ケースの記述

Step 3.1 ケースの枠組の記述

Step 3.2 ケースの内容の記述

ケースの内容を Step 4 に従って記述する。

Step 4 シーケンス図の記述

Step 4.1 シーケンス図の枠組の記述

Step 4.2 シーケンス図の内容の記述

Step 4.2.1 ノードの記述

Step 4.2.2 メッセージのシーケンスの記述

図 10 HSC に基づく仕様記述スタイル

Fig. 10 Specification style based on HSC.

されることを示している。

(3) 任意順序：ノード間で交換されるメッセージの順番を規定せず、どのような順番で行ってもよいこと

を任意順序と呼ぶ。表3に示す任意順序の記号は、ノードの縦線に重ねて記述され、対応するメッセージが表示される。

3.2 仕様記述スタイル

HSC による仕様記述手順として、その特長である階層化とモジュール化を生かした仕様記述スタイルを図10のように構成する。

4. HSC の支援環境

本章では、HSC に基づいた仕様を効率的に記述できるユーザフレンドリイな支援環境を構成する。

4.1 設計目標

(1) HSC の特長に基づく仕様化技法の支援

次の3つの機能を提供する。1)階層化とモジュール化に基づく仕様記述スタイルの支援、2)フェーズ、ケースおよびシーケンス図などの図形自動配置の機能、3)ウィンドウの大きさに応じた仕様枠組の自動分割機能。

(2) 支援システムと操作に関する情報の提供

設計者が、特に HSC の初心者でも、仕様を記述できるように支援するため、支援システムの状態、HSC に関する情報およびシステムの操作に関する説明などをウィンドウを用いて提供する。

(3) ユーザフレンドリイインタフェースの提供

次の3つの機能を提供する。1)フェーズ、ケースおよびシーケンス図などの枠組の自動表示、2)ケース間の関係やケースの行き先などのシンボルのメニュー化、3)編集機能の選択にボタンとメニューなどを用いる。

(4) コメントの記述の支援

自然言語を用いて仕様の補助的な説明を行う機能。

4.2 提供する機能

4.1 節で述べた設計目標を達成するために、以下の4つの機能を提供する。

(1) 仕様化技法において用いられる機能

1) 仕様記述スタイルの支援

HSC に基づく仕様記述は仕様枠組、フェーズおよびケースの3つから構成されている。このような特長を生かすため、3.2 節で述べた仕様記述スタイルに基づいた仕様記述エディタを提供し、仕様枠組、フェーズおよびケースの記述を支援する。

2) 図形自動配置の機能

①フェーズの自動配置：仕様枠組を構成する各フェーズをそれぞれ一定の間隔において自動配置する。

また、フェーズの枠組をケースの数に応じて自動的に広げる。

②ケースの自動配置：フェーズの中にそれぞれのケースを左から右に、上から下に自動配置する。

③シーケンス図の自動配置：シーケンス図においてノード間で交換されるメッセージを一定の間隔において自動配置し、ノードを表す線の長さは接続されるメッセージの数に応じて伸縮し、シーケンス図の枠組も自動的に広げる。

3) 仕様枠組の自動分割機能

仕様枠組を構成する各フェーズをその大きさの変化に応じて自動的に配置し、一画面に収まらない場合は自動的に複数のページに割り付ける。

(2) 支援システムと操作に関する情報の自動表示機能

各時点において、システムの状態、システムが提供する機能および関連する HSC の用語や記号の説明をウィンドウ上に自動的に表示する。また、現在どのような操作が可能であるかの説明を自動的に表示する。

(3) ユーザフレンドリイインタフェースの機能

1) 枠組の自動表示

フェーズ、ケースやシーケンス図の記述に応じて必要な枠組を表示する。

2) シンボルの記述

フェーズにおけるケース間の関係やケースの行き先の記述の際には、それぞれ表1と表2に対応するメニューが表示される。また、シーケンス図の記述において、表3に示した3種類のメッセージと任意順序のシンボルがメニューとして画面に表示される。

3) 編集機能の選択

仕様枠組やシーケンス図、コメントの記述のため、個々の編集機能に対応するボタンやメニューを用意し、これらから選択する。

(4) コメントの記述支援の機能

ウィンドウを用いて自然言語による仕様の補助的な説明を行う機能を提供する。

4.3 支援環境の構成

支援環境はユーザインタフェースと仕様記述システムの2つの部分から構成されている(図11)。

(1) ユーザインタフェース

次の5つのウィンドウ、①仕様記述ウィンドウ、②シーケンス図サブウィンドウ、③システム情報ウィンドウ、④操作情報ウィンドウ、⑤コメントサブウィンドウを提供している。

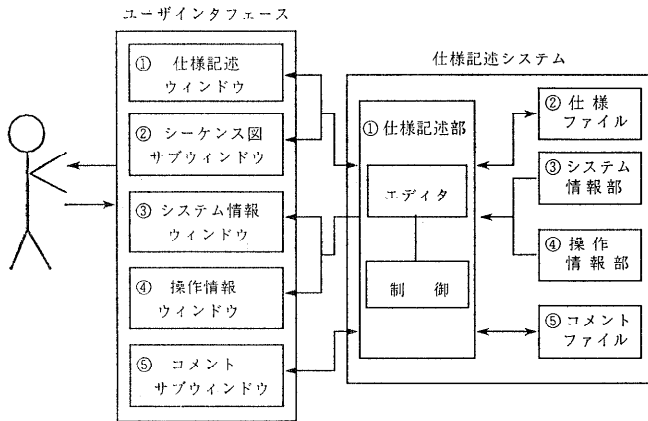


図 11 支援環境の構成

Fig. 11 Structure of specification environment.

(2) 仕様記述システム

仕様記述システムは図 11 に示すように次の 5 つの部分から構成されている。①仕様記述部：HSC のエディタとシステムの制御の機能を提供する。②仕様フ

ァイル：HSC により記述した仕様を格納する。③システム情報部：支援システムに関する情報を格納する。④操作情報部：システムの操作に関する情報を格納する。⑤コメントファイル：設計者が書いたコメントを格納する。

4.4 試作システム

本支援環境は、SUN ワークステーションにおいて X-window システム (Xview) および C 言語を用いて試作した。試作システムのソースプログラムは約 10000 行からなる。

〔記述例〕

試作システムの動作例として基本電話サービスの記述例を図 12 に示す。同図

には①仕様記述ウィンドウ (左上)、②シーケンス図サブウィンドウ (右上)、③システム情報ウィンドウ (左下)、④操作情報ウィンドウ (中央下)、⑤コメントサブウィンドウ (右下) が示されている。仕様記述

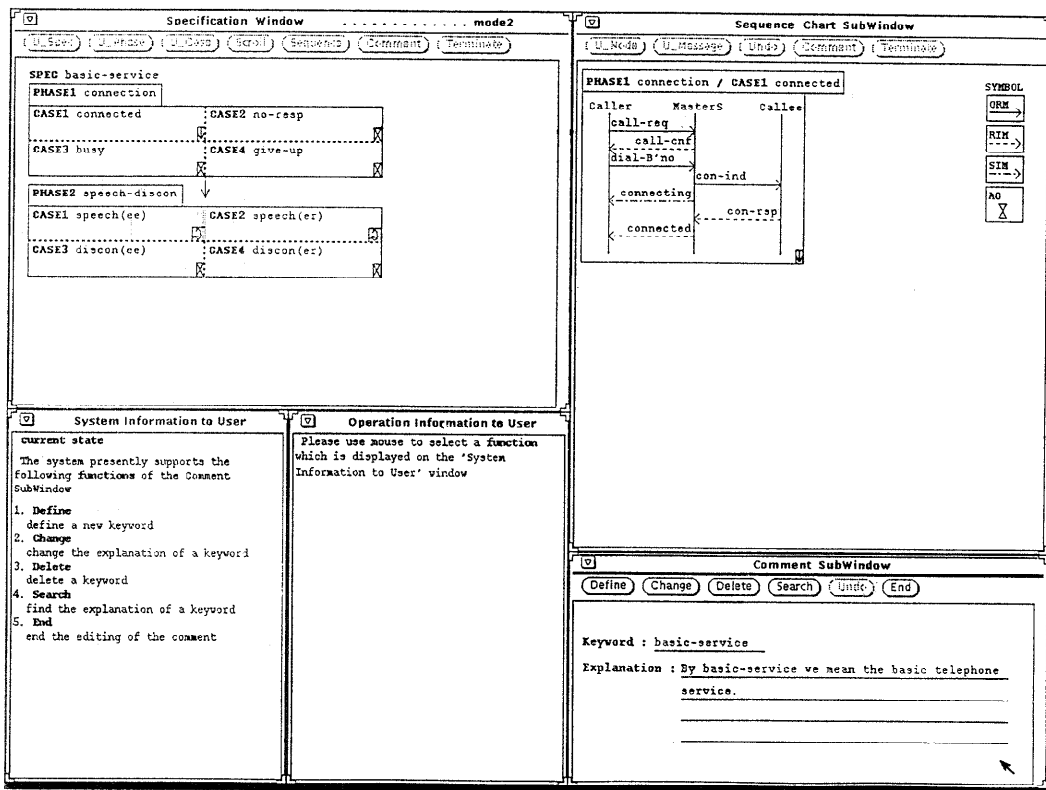


図 12 基本電話サービスの完成図の一例

Fig. 12 An example of specification environment for basic telephone service.

のそれぞれの局面に応じて、システムや操作に関する情報がそれぞれ③と④に自動的に表示される。設計者はこれらの説明や指示に基づいて①や②に仕様を記述していく。また、⑤には自然言語を用いて仕様の補助的な説明を記述する。

同図には、完成した基本電話サービスの仕様枠組とそのフェーズ1のケース1のシーケンス図がそれぞれ①と②に示されている。

仕様の入力速度については、仕様記述に関する操作の応答時間が約 0.5～

1.5 秒程度であり、設計者に支障を与えるほどの影響はないことが分かった。

5. 適用例と評価

5.1 適用例

HSC の適用例として、主として表4に示すサービスなどを記述した⁷⁾。

これらのサービスの記述を通して、HSC は通信システムの要求仕様の開発向きの記述法として有効であることを確認した。

5.2 評価

HSC の評価として、HSC と関係の深いシーケンス図、LSDL^{1),2)}および MSC³⁾をとりあげ、これらを用いた交換システムの基本電話サービスの記述(図13)を通して、記述能力、理解性と学習性、記述量および記述対象などについて比較を行う。

(1) 記述能力

1) 図同士の関係の記述

HSC では、シーケンス図、LSDL と MSC にはない階層化、モジュール化と図同士の関係を記述する機能を与えることにより、仕様全体の枠組を明確に表現できる。その結果、(2)の1)で述べる仕様全体の理解性の向上を達成している。

2) 条件分岐の記述

シーケンス図、LSDL および MSC では、条件分岐による各々のケースが別々の図に書かれる。そのため、条件分岐に関連した全体が分かりにくい。一方、HSC では、ケース間の選択の関係をを用いて条件分岐を記述するため、条件分岐に関連した全体が明確に分かりやすく表現できる。

表4 適用例の記述量

Table 4 The number of symbols used in the specifications of the examples.

サービス名 \ シンボル数	フェーズ数	サブフェーズ数	ケース数	メッセージ数
① OSIのトランスポート層サービス	2	0	13	30
② OSIの応用層のCCRサービス	3	0	17	102
③ 交換システムの基本電話サービス	2	0	8	31
④ 交換システムのCWサービス	4	2	27	83

3) 繰り返しの記述

シーケンス図と MSC では、繰り返しの表現はない。一方、LSDL では、ラベルと制御の移行点を設けることにより表現している。また、HSC でもフェーズの遷移やフェーズの繰り返しを用いて繰り返しの記述ができる。

(2) 理解性と学習性

1) 仕様全体の理解性

シーケンス図、LSDL および MSC では、仕様の構造として階層化の概念を与えていない。そのため、仕様が大きくなるにつれ理解性が低下する。一方、HSC では、仕様は階層的に仕様枠組、フェーズ、さらにケースが記述されるため、仕様の全体像が分かりやすくなっている。

2) 記述内容の把握

シーケンス図、LSDL および MSC では、主にメッセージのやりとりを記述している。これに加えて、HSC では、メッセージを分類し、それぞれ視覚的に表現し、また、メッセージ間の関係を導入することにより、シーケンス図の記述内容が理解しやすくなっている。

3) 学習性

LSDL と MSC では、従来のシーケンス図にいくつかの記号が加えられた文法となっている。HSC では、さらに仕様枠組、フェーズおよびケースのような階層化の概念や操作要求、結果通知および状態通知のようなメッセージの分類などを与えているため、文法としては、MSC と LSDL に比べて、やや複雑になっている。

なお、前述の文法の複雑さについては、支援環境において、HSC に基づく仕様記述スタイルの支援や、システム情報ウィンドウにおいて HSC の用語や記号

SPEC 基本電話サービス

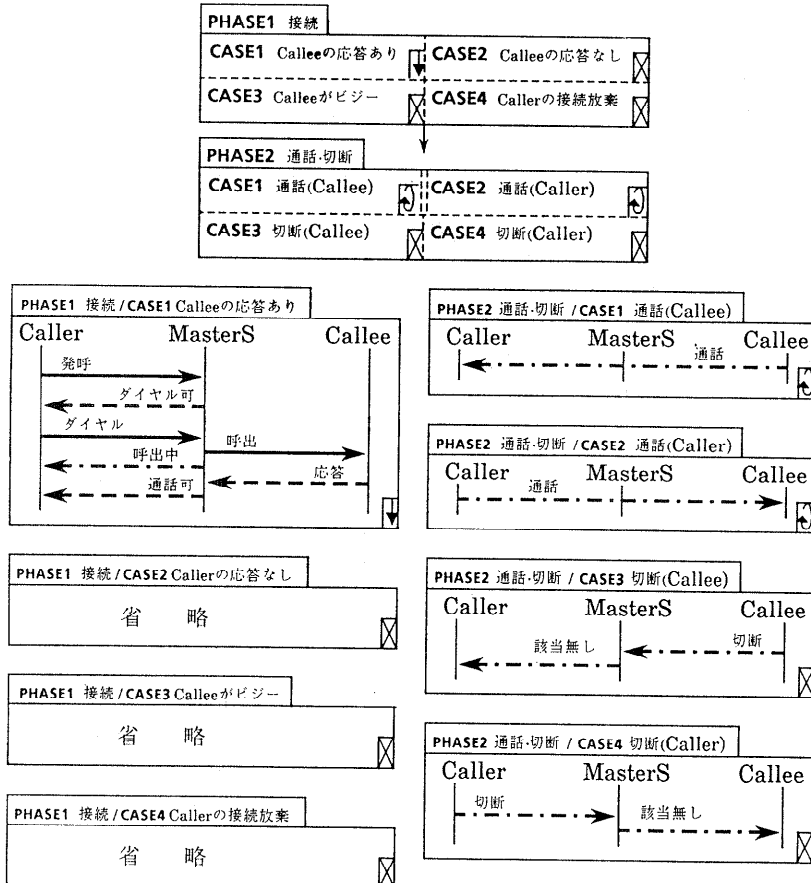


図 13 HSC による基本電話サービスの記述
Fig. 13 Basic telephone service specified in HSC.

の説明を行うことなどにより改善されている。

(3) 記 述 量

シーケンス図, LSDL および MSC は主にメッセージのやりとりを記述するため, 記述量はほぼ同じである。HSC の場合, シーケンス図, LSDL および MSC と比べて仕様枠組の分だけ記述量は多い。一方, シーケンスの一部をケースとしてモジュール化することにより, 重複して書かれるメッセージのやりとりの部分の記述量を減らすことが可能である。

(4) 記 述 対 象

HSC は, メッセージの交換が主体となる通信システムなどの仕様をトップダウン的にシステム全体を見通しよく記述する場合に有効である。しかし, データの記述, ノードの内部動作などの詳細な仕様およびメッセージの衝突や欠落などの異常処理の記述は対象と

しない。これらはソフトウェアの設計で考慮すべき問題である。この点については, シーケンス図, LSDL および MSC の場合も, HSC とほぼ同様である。ただし, MSC では, タイマの記述機能を与えているため, 時間に関する異常処理の記述が可能となっている。

6. お わ り に

本論文では, 従来のシーケンス図を基本とし, シーケンス図の持つ長所を保存しつつ, その短所を改善する仕様記述法 HSC を提案した。HSC の特徴は, 通信システムの特徴を考慮した階層化とモジュール化の概念の導入により, 仕様全体の理解性の向上を達成していることにある。また, HSC 向きの仕様記述スタイルを考案すると共に, SUN ワークステーション上に

ユーザフレンドリィな支援環境を試作した。HSC を用いたサービスの記述を通して、HSC は通信システムの要求仕様の開発向きの記述法として有効であることを確認した。また、階層化とモジュール化に基づいた HSC による仕様は、部品化がしやすく再利用による仕様記述にも適している。

今後の課題として、HSC で記述された仕様を部品化し再利用に基づく仕様の合成法や HSC の形式化へ向けた検討などが残されている。

謝辞 熱心に討論して頂いた東北大学白鳥研究室と野口研究室の皆様に感謝します。

参 考 文 献

- 1) Niitsu, Y. and Mizuno, O.: Interactive Specification Environment for Communication Service Software, *IEEE Journal on Selected Areas in Communications*, Vol. 8, No. 2, pp.181-188 (1990).
- 2) 水野 修, 新津善弘: メッセージシーケンス図入力による通信サービス仕様設計方式, 電子情報通信学会論文誌 B-I, Vol. J 74-B-I, No. 12, pp. 1042-1055 (1991).
- 3) CCITT: Message Sequence Chart, CCITT Study Group X (WP X/3), Experts meeting Rome (1991).
- 4) ISO: LOTOS: A Formal Description Technique Based on Temporal Ordering of Observed Behavior, ISO 8807 (1989).
- 5) 李 殷碩, 森 健一, 白鳥則郎, 野口正一: G-LOTOL の仕様化環境 SEGL の構成と試作, 情報処理学会論文誌, Vol. 32, No. 3, pp. 314-323

(1991).

- 6) Shiratori, N. and Lee, E.: An Integrated User-Friendly Specification Environment for LOTOS, *IEICE Trans. Commun.*, Vol. E 75-B, No. 10, pp. 931-941 (1992).

- 7) 黄 錦法, 高橋 薫, 白鳥則郎, 野口正一: 仕様記述法 HSC の適用と評価, 情報処理学会研究報告, SE 84-2 (1992).

(平成 4 年 11 月 2 日受付)

(平成 5 年 4 月 8 日採録)



黄 錦法

1964 年生。1991 年東北大学大学院博士前期課程修了。現在同大学院博士後期課程在学中。通信システム仕様記述法, 事例ベース推論による仕様再利用の研究に従事。



白鳥 則郎 (正会員)

昭和 21 年生。昭和 52 年東北大学大学院博士課程修了。同年、東北大学電気通信研究所勤務。昭和 59 年、同大学助教授(電気通信研究所)。平成 2 年、同大教授(工学部情報工学科)。情報通信システムの構成論, ソフトウェア開発法, ヒューマンインタフェースの研究に従事。情報処理学会 25 周年記念論文賞受賞。IEEE, 電子情報通信学会, 人工知能学会各会員。