ソフトウェア工学 100－2
（1994．9．8）

位相的観点からの仕様記述

安藤敏彦[1], 加藤靖[1], 高橋薫[2]

[1] 仙台電波高専
〒989-31 仙台市青葉区上愛子字北原1番地

[2] (株) 高度通信システム研究所
〒989-32 仙台市青葉区南吉成6-6-3

概要

一つの仕様にシステムの大域的な性質と局所的な振舞いを同時に盛り込むのは困難である．それは，記述法によって記述できる抽象度が異なり，これらの性質，振舞いを唯一つの記述法で記述することができないからである．制約指向的な手法を用いて仕様を得る場合，個々の制約は必ずしも同じ抽象度で書かれるとは限らないので，様々な抽象度を統一的に扱うことが必要となる．本研究では，仕様空間に導入された位相がそのための基準となり得ることを示す．また，この位相は記述法の抽象度や仕様間の等価関係とも対応していることを示す．

Specification from Topological Point of View

Toshihiko ANDO[1], Yasushi KATO[1] and Kaoru TAKAHASHI[2]

[1]Sendai National College of Technology,
1, Kitahara, Kamiayashi, Aoba-ku, SENDAI,
989-31, JAPAN,
Tel: +81-22-392-4761,
Fax: +81-22-392-3359,
E-mail: tando@cc.sendai-ct.ac.jp,
kato@info.sendai-ct.ac.jp .

[2]AIC System Labs. Ltd.,
6-6-3, Minami-Yoshinari, Aoba-ku, SENDAI,
989-32, JAPAN,
Tel: +81-22-279-3310,
Fax: +81-22-279-3640,
E-mail: kaoru@aic.co.jp .

Abstract

It is difficult to describe global properties and local behaviors of a system in a specification at once. Each description method has each abstract level, so that these properties and behaviors can not be described with only one description method (language). In case of using a constraint-oriented approach, each constraint may have various abstract levels. To establish a specification method with a constraint-oriented approach, it is necessary to treat various description methods uniformly. In this report, we show that topology in a specification space can be a standard to do it. Such topology corresponds to the abstract levels of description methods and equivalent relations among specifications.

1 はじめに

我々は時制論理 [1, 2] で書かれた性質から制約指向スタイル [3] を持つ LOTOS [4] 仕様を求める合成法を提案している [5, 6, 7]. LOTOS などの形式記述技法 (Formal Description Technique, FDT) [8] は記述が厳密で曖昧さがないなどの長所を持つが, 記述が局所的であるためシステムの大まかな性質を陽に書くことができない. 一方, 論理を使えばそのような性質を直接書くことができる. 仕様記述の初期段階では, システム全体として満足されるべき要求がまず与えられ, その要求を仕様に盛り込むことが必要となる. 従って, 初めから FDT で仕様を書くよりも, まず論理によって要求すべき性質を記述し, それから FDT の記述に変換した方が有効であると思われる. すなわち, 制約指向的手法が有効である. 我々の合成法はその観点から開発されたものであり, 制約を時制論理で記述している.

ところで, この合成法で問題となっているのは, 与えられた制約に対し, それを満足する LOTOS 仕様は一意に決められないことである. これは, 論理が大域的な性質を, FDT が局所的な振舞いを記述するというように, 記述法によって記述できる抽象度が異なっているためである. この抽象度の違いによる問題は, 様々な記述法で制約を書けるよう, この合成法を拡張する場合にも重要となる. このように様々な抽象度の制約を同時に取り扱うためには何が必要だろうか.

ここで, ある制約を満足する仕様の集合を考える. そのような集合の全体は仕様全体を被覆する. 仕様全体を空間とみなせば, この集合族はその空間の位相空間となっていると考えられる. この観点によれば, 異なる記述法を比較する場合, 記述そのものを扱う必要がないので, 様々な記述法を統一的に扱うことができる.

本研究では, 記述方毎に仕様空間の位相が定まることを示し, 位相が様々な記述法を統一的に扱うための基準となり得ることを示す. また, 位相と記述法の抽象度, および仕様間の等価関係との関係についても議論する.

2 仕様のモデル

本研究では, 簡単のため, 仕様をイベントの系列と考え, 次のようなモデルを扱うことにする. 以下で, $\boldsymbol{R}_0 = \{x | x \geq 0\}$, Act を可観測イベントの集合, 2^{Act} を Act の冪集合とする.

定義 1 イベント系列

$\Lambda(\sigma)$ を $\boldsymbol{R}_0$ の疎な集合, すなわち, たかだか加算な集合とする. この時,

$\sigma : \Lambda(\sigma) \rightarrow 2^{Act}$

をイベント系列と言う. □

ここで, $\Lambda(\sigma)$ は $\{t_0 = 0, t_1, t_2, \cdots, t_n\}$ あるいは, $\{t_0 = 0, t_1, t_2, \cdots, \}$ $(0 < t_1 < t_2 < \cdots)$ と表され, 各元を時刻と言う. a_i は可観測イベントの集合で, 複数個のイベントが同時に発生できるものとする. $\Lambda(\sigma)$ を σ のインデックスと言う. ここで考えられているイベント系列とは, 時刻 0 から始まり, 不定期的に起こるイベントの系列であり, イベントの発生する時刻はインデックスにより定まっている. インデックスの全体を $Index$, イベント系列の全体を Σ とする.

イベント系列は 3 通りに表記される.

- 実時間表示

  イベント系列を, 時刻とその時に発生するイベントの組合せで表記する.

  $\sigma = (a_0 : 0, a_1 : t_1, a_2 : t_2, \cdots)$,

  ここで, $t_1, t_2, \cdots \in \boldsymbol{R}_0$ は $0 < t_1 < t_2 < \cdots$ である時刻である.

  例. $\sigma = (a_0 : 0, a_1 : 1.9, a_2 : 3.4, \cdots)$
- 整数時間表示

  時刻を量子的に整数時間で表記する.

  $int(\sigma) = (a_0 : 0, a_1 : i_1, a_2 : i_2, \cdots)$,

  ここで, $i_1, i_2, \cdots$ は $0 < i_1 < i_2 < \cdots$ である正整数である. この表記には, $i_1 \leq t_1 < i_1 + 1$, $\cdots$ であるような $\sigma = (a_0 : 0, a_1 : t_1, \cdots)$ が対応する.

  例. $int(\sigma) = (a_0 : 0, a_1 : 1, a_2 : 3, \cdots)$
- 順序表示

  イベントの発生時刻は無視し, 発生順序だけを表記する.

$ord(\sigma) = (a_0, a_1, a_2, \cdots)$

注意: 現在の FDT ではイベントの発生時刻を書くことはなく, 順序表示に対応している. しかし, 最近では LOTOS の拡張版である E-LOTOS [9] のように時間表現を盛り込むことが試みられている. 実時間表記, 整数時間表記はそれを考慮したものである.

3 イベント系列上の論理

2 節で示した表記法ではシステムの局所的な動作は記述できるが, イベント系列全体に渡るような性質を表現することはできない. そのような性質を表すには論理を使うことが有効である. この節では, そのような論理として, 瞬間論理 (ML) と時制論理 (TL) を定義する. ML はどの時刻でどんなイベントが起こるかを表現でき, TL はイベント系列上で「いつでも」あるいは「いつか」起こり得る性質を表現できる.

3.1 瞬間論理

ML の構文と意味は次のよう定義される. ただし, 次の定義において, ϕ, ϕ', ϕ_1, ϕ_2 を ML の論理式, $t \in \boldsymbol{R}_0$, $a \in 2^{Act}$ とする.

定義 2 *ML* の構文

ML の論理式は次のように構成される.

ϕ ::= $(t, a) \mid \neg\phi' \mid \phi_1 \wedge \phi_2 \mid$ **true** □

他の命題論理演算子も次のように定義される.

$\phi_1 \vee \phi_2 \stackrel{def}{=} \neg(\neg\phi_1 \wedge \neg\phi_2)$

$\phi_1 \Rightarrow \phi_2 \stackrel{def}{=} \neg(\phi_1 \wedge \neg\phi_2)$

false $\stackrel{def}{=} \neg$ **true**

定義 3 *ML* の意味

$\sigma \models_m$ **true** : 常に真.

$\sigma \models_m (t, a)$: $a \in \sigma(t)$

$\sigma \models_m \neg\phi'$: $(\sigma \models_m \phi')$ ではない.

$\sigma \models_m \phi_1 \wedge \phi_2$: $(\sigma \models_m \phi_1)$ かつ $(\sigma \models_m \phi_2)$. □

例 1 *ML* はイベント系列の各時刻で起こるイベントを表現する. 例えば, イベント系列 $\sigma = (a_0 : 0, a_1 : 1.9, a_2 : 3.4, \cdots)$ に対し,

$\sigma \models_m (1.9, a_1) \wedge (3.4, a_2)$

が成り立つ.

3.2 時制論理

TL は様相論理を時系列上で解釈したものである. 様相論理で用いる演算子 □ (must) や ◇ (may) は各々「いつでも」, 「いつか」と解釈される. TL の構文と意味は次のように定義される. ただし, ϕ, ϕ', ϕ_1, ϕ_2 は TL の論理式, $a \in 2^{Act}$, $\sigma = (a_0 : 0, a_1 : t_1, a_2 : t_2, \cdots)$ をイベント系列, $\Lambda(\sigma)$ をσ のインデックスとする.

定義 4 *TL* の構文

TL の論理式は次のように構成される.

ϕ ::= $a \mid \neg\phi' \mid \phi_1 \wedge \phi_2 \mid \phi_1 \mathcal{U} \phi_2 \mid$ **true** □

命題論理の演算子 $\vee$, $\Rightarrow$ および **false** は ML と同様に定義される. また, 時制演算子 □, ◇ は次のように定義される.

□ $\phi \stackrel{def}{=} \phi \mathcal{U}$ **false**

◇ $\phi \stackrel{def}{=}$ **true** $\mathcal{U} \phi$

定義 5 *TL* の意味

$t \in \boldsymbol{R}_0$ に対し, イベント系列 σ のある時刻 $t_i = t \in \Lambda(\sigma)$ が存在するならば, *TL* の論理式の意味は次のように定義される.

$\sigma, t \models_t$ **true** : t で真.

$\sigma, t \models_t a$: $a \in \sigma(t)$

$\sigma, t \models_t \neg\phi'$: $(\sigma, t \models_t \phi')$ ではない.

$\sigma, t \models_t \phi_1 \wedge \phi_2$: $(\sigma, t \models_t \phi_1)$ かつ $(\sigma, t \models_t \phi_2)$.

$\sigma, t \models_t \phi_1 \mathcal{U} \phi_2$: $\exists j \geq i, (\sigma, t_j \models_t \phi_2)$ かつ $i \leq \forall k < j, (\sigma, t_k \models_t \phi_1)$. □

特に, $t = 0$ の場合,

$\sigma \models_t \phi$

と表現し, ϕ を σ の時間的性質と言う.

例 2 イベント系列 $\sigma = (a_0 : 0, a_1 : 1.9, a_2 : 3.4, \cdots)$ に対し, "◇ a_2 " (いつか a_2 が起こる) は σ の時間的性質である.

4 仕様空間の位相

2節でイベント系列の3通りの表記法を示し，3節で2通りの論理を示した．これらは全て抽象度が異なる．例えば，順序表記で $ord(\sigma) = (a_0, a_1, a_2, \cdots)$ が与えられたとする．これを満足する実時間表記はいくらでも存在する．逆に，TLでは，$a_0 \Rightarrow \Diamond(a_1 \Rightarrow \Diamond(a_2 \Rightarrow \cdots))$ と決まる．

一般に，ある記述法による一つの記述を満足するイベント系列は必ずしも1つではなく，いくつかのイベント系列からなる集合を成す．その集合の大きさは記述法とその記述自身にもよるが，それらの全体はイベント系列全体を被覆する．この節では，各々の記述に対し，どのような集合が対応するかを示し，それらの集合からなる集合族がイベント系列全体 Σ (仕様空間と言う) を空間とみなした時に位相となることを示す．

4.1 各記述法に対応する位相

(1) 実時間表示

実時間表示したイベント系列 $\sigma = (a_0 : 0, a_1 : t_1, a_2 : t_2, \cdots)$ に対応する集合 $O(\sigma)$ は，

$$O(\sigma) = \{\sigma\}.$$

(2) 整数時間表示

整数時間表示 $int(\sigma) = (a_0 : 0, a_1 : i_1, a_2 : i_2, \cdots)$ に対応する集合 $O(int(\sigma))$ は，

$$\begin{aligned} &O(int(\sigma)) = \\ &\quad \{\sigma \in \Sigma \mid \exists_1 t_1, \exists_1 t_2, \cdots \in \Lambda(\sigma), i_1 \leq \\ &\quad t_1 < i_1 + 1,\ i_2 \leq t_2 < i_2 + 1, \sigma(0) = \\ &\quad a_0, \sigma(t_1) = a_1,\ \sigma(t_2) = a_2, \cdots\}. \end{aligned}$$

ここで，$\exists_1$ は唯一つ存在することを表す限定作用記号である．

(3) 順序表示

順序表示 $ord(\sigma) = (a_0, a_1, a_2, \cdots)$ に対応する集合 $O(ord(\sigma))$ は，

$$\begin{aligned} &O(ord(\sigma)) = \\ &\quad \{\sigma \in \Sigma \mid \exists_1 t_1, \exists_1 t_2, \cdots, \in \Lambda(\sigma), \\ &\quad 0 < t_1 < t_2, \cdots, \sigma(0) = a_0, \\ &\quad \sigma(t_1) = a_1, \sigma(t_2) = a_2, \cdots\}. \end{aligned}$$

(1)～(3) では，各集合は互いに素であるが，次のような解釈をすることにより，位相の条件を満足する集合族を構成することができる．すなわち，$O_1 = O(\sigma_1)$, $O_2 = O(\sigma_2)$ に対し，$O_1 \cup O_2$ は，初期状態から σ_1 または σ_2 が起こり得る仕様を意味すると考える．これは，FDT では分岐（または選択）に相当する．従って，(1) に対して次のような集合族 $\mathcal{O}_r$ を構成する．

- すべての $\sigma \in \Sigma$ に対して，$O(\sigma) \in \mathcal{O}_r$.
- 全ての $\Gamma \subset \Sigma$ に対し，$\bigcup_{\sigma \in \Gamma} O(\sigma) \in \mathcal{O}_r$.

構成の仕方から，$\mathcal{O}_r$ が Σ の位相となるのは明らかである．特にこの場合，$\mathcal{O}_r$ は Σ の部分集合全体に等しいので，離散位相となる．各集合をその位相の開集合と言う．同様に，整数時間表示，順序表示に対する集合族 $\mathcal{O}_{int}$, $\mathcal{O}_{ord}$ も構成でき，各々 Σ の位相となる．

(4)ML

ML の論理式に対して次の集合が対応する．ただし，ML の論理式 ϕ に対応する集合を $O(\phi)$ とする．

$$\begin{aligned} &O(\mathbf{true}) = \Sigma \\ &O((t, a)) = \{\sigma \in \Sigma \mid t \in \Lambda(\sigma), \sigma \models_m (t, a)\}. \\ &O(\neg\phi) = \Sigma - O(\phi). \\ &O(\phi_1 \wedge \phi_2) = O(\phi_1) \cap O(\phi_2). \end{aligned}$$

この場合，集合族 $\mathcal{O}_{ML} = \{O(\phi) \mid \phi$ は ML の論理式 $\}$ は Σ の位相となる．なぜなら，

- $\Sigma = O(\mathbf{true}), \emptyset = O(\mathbf{false}) \in \mathcal{O}_{ML}$
- $O_1, O_2 \in \mathcal{O}_{ML}$ ならば，$O_1 = O(\phi_1), O_2 = O(\phi_2)$ となる ML の論理式 ϕ_1, ϕ_2 が各々唯一つ存在し，$O_1 \cap O_2 = O(\phi_1) \cap O(\phi_2) = O(\phi_1 \wedge \phi_2) \in \mathcal{O}_{ML}$.
- $O_\lambda \in \mathcal{O}_{ML}$ $(\lambda \in \Lambda)$ ならば，$O_\lambda = O(\phi_\lambda)$ となる ϕ_λ が唯一つ存在し，$\bigcup_{\lambda \in \Lambda} O_\lambda = O(\bigvee_{\lambda \in \Lambda} \phi_\lambda) \in \mathcal{O}_{ML}$.

(5) TL

時間的性質 (TL の $t=0$ における論理式) に対する集合は次の通りである.

$O(\mathbf{true}) = \Sigma$
$O(a) = \{\sigma \in \Sigma \mid \sigma(0) = a\}$.
$O(\neg\phi) = \Sigma - O(\phi)$.
$O(\phi_1 \wedge \phi_2) = O(\phi_1) \cap O(\phi_2)$.
$$O(\phi_1 \,\mathcal{U}\, \phi_2) = \bigcup_{\Lambda_0 \in Index} \bigcup_{i \geq 0} \Big[\ O(t_i, \phi_2) \cap \bigcap_{i>j\geq 0} O(t_j, \phi_1)\ \Big]$$

ただし, $O(t,\phi) = \{\sigma \in \Sigma \mid t \in \Lambda(\sigma) = \Lambda_0, (\sigma, t \models_t \phi)\}$ である.

この場合も, 集合族 $\mathcal{O}_{TL} = \{O(\phi) \mid \phi$ は TL の論理式$\}$ は (4) と同様にして, Σ の位相となる.

4.2 位相間の関係

次に, 前節で構成した位相間の関係を示す. 位相間の関係として強弱関係を考える.

例 3 $\mathcal{O}_r$ と $\mathcal{O}_{int}$とを比較する. ここで, Σ に $\mathcal{O}_r$, $\mathcal{O}_{int}$ を導入した位相空間をそれぞれΣ^r, Σ^{int} とする. 今, 恒等写像 $1_\Sigma : \Sigma^r \to \Sigma^{int}$ を考えると, これは連続写像となる. ところが, その逆では連続とはならない. 従って, $\mathcal{O}^r$ は $\mathcal{O}^{int}$ より強い (これを $\mathcal{O}^r > \mathcal{O}^{int}$ と書く). これは $\mathcal{O}^r \supset \mathcal{O}^{int}$ からも明らかである.

例 4 $\mathcal{O}_{ML}$ と $\mathcal{O}_{TL}$とを比較する. もし, $O = O(\phi) \in \mathcal{O}_{TL}$ ならば, $O \in \mathcal{O}_{ML}$ である. なぜならば, 時間的性質 ϕ は, ML で記述できるからである. 従って, $\mathcal{O}_{ML} \supset \mathcal{O}_{TL}$ であるから, $\mathcal{O}_{ML}$ は $\mathcal{O}_{TL}$ より強い.

その他の組合せも考えると, $\mathcal{O}_r$, $\mathcal{O}_{int}$, $\mathcal{O}_{ord}$, $\mathcal{O}_{ML}$, $\mathcal{O}_{TL}$ の強弱関係は次のようになる.

$\mathcal{O}_r > \mathcal{O}_{int} > \mathcal{O}_{ord} > \mathcal{O}_{TL}$.
$\mathcal{O}_r > \mathcal{O}_{ML} > \mathcal{O}_{TL}$.

5 考察

(1) 記述法の抽象度と位相

4 節で得られた位相間の関係を記述法の抽象度と比較する. 例として, 実時間表記と整数時間表記とを考える. 整数時間表記 $int(\sigma_1) = (a_0 : 0, a_1 : 2, a_2 : 4, \cdots)$ が与えられた時, これを満足する実時間表記は無数に存在し, $\sigma_2 = (a_0 : 0, a_1 : 2.3, a_2 : 4.5, \cdots)$ もその一つである. 一方, 実時間表記 σ_2 が与えられた時, これに対する整数時間表記は $int(\sigma_1)$ 唯一つである. この場合, 実時間表記よりも整数時間表記の方が抽象度が高いと言える. これを 例 2 と比較すると, 抽象度の高い整数時間表記の方が弱い位相をなしていることが分かる.

ML と TL を比較しても同様な事が言え, 抽象度の高い TL の方が ML より弱い位相をなしている. 従って, 記述法から導かれる位相の強さは, その記述法の抽象度の低さ (具体性) を反映していると考えられる.

一般に, 低い抽象度の記述から, より高い抽象度の記述は一意に得られるが, その逆の場合は一意に決定されない. これは, 4 節で議論したように, 同じ空間に強い位相が導入された位相空間から弱い位相が導入された位相空間への恒等写像は連続となるが, その逆は必ずしも成り立たないことに対応している. 直観的には, 位相とは空間を部分集合で被覆する仕方であり, 強い位相とは, より細かく被覆しているものを指す. 従って, 強い位相の場合には異なる開集合に属していた 2 つの仕様が, 弱い位相では同じ開集合に属する場合も存在する. 開集合と記述は対応しているから, この場合, 強い位相の記述法では記述が異なるが, 弱い位相の記述法では同じ記述になる (図 1).

(2) 仕様の等価関係

LOTOS における強双模倣関係, 弱双模倣関係, トレース等価 [10] 等の仕様の等価関係も, 仕様空間の位相の観点から考えることができる. LOTOS の強双模倣関係とは, 2 つの仕様の対応する状態から始まるイベント系列が全く等しくなる関係であるが, 本研究においては, 順序表記が等しくなる事に相当する. この場合, 同じ表記に属する異なる仕様はその意味で等価であると考えられる.

一般に, 仕様空間にある位相が導入されている場合, 同じ開集合に属する 2 つの仕様は区別することができない. この時, これらの仕様はその

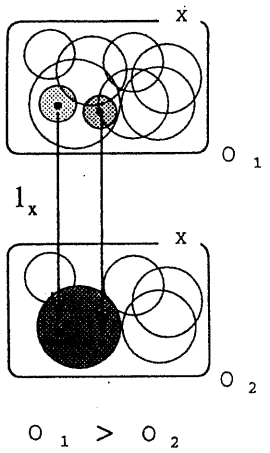

図 1: 仕様空間の位相と抽象度の関係

位相において等価であると考えられる. 従って, ある記述では等価であっても, より位相の強い記述においては等価でない場合も存在する. すなわち, 仕様の等価関係の強弱は位相の強さに対応する.

(3) 制約指向的手法に向けて

これまでの議論は, 与えられた記述法から導かれる位相に関するものであった. 一方, あらかじめ仕様空間に位相が与えられた場合については, 次のことが考えられる. 各開集合は各々ある記述に対応していた. 従って, 初めに位相が導入された場合, その開集合にある記述を対応させるような記述法を考えることができるかも知れない. その場合, 全順序となる位相の集合を与えることができれば, 初めは位相の弱い (抽象度の高い) 記述法で記述し, その後, 順に位相の強い性質で記述していくことにより, 階層的な仕様記述が行なえる. すなわち, 仕様記述に制約指向的な手法を自然に取り入れることができる.

6 結論

仕様空間に導入された位相を用いて, 様々な記述法を統一的に扱い得ることを示した. 位相の観点から様々な抽象度の制約を一様に扱えるようになれば, 制約指向的な手法が仕様記述においてより有効な方法となり得ると思われる. また, 等価関係の強弱も位相の観点から自然に説明することができる. 我々の提案している仕様合成法も本研究からの示唆により, 制約を様々な記述法で記述できるよう拡張できると思われる.

参考文献

[1] Ben-Ari, M., Manna, Z. and Pnueli, A., The Temporal Logic of Branching Time, *Proc. of 8th Annual ACM Symposium on Principles of Programming Languages*, pp. 164-176, 1981.

[2] Gotzhein, R., Temporal Logic and Applications - a Tutorial, *Computer Networks and ISDN System*, Vol.24, pp.203-218, North-Holland, 1992.

[3] Vissers, C. A., Scollo, G. and van Sinderen, M., Architecture and Specification Style in Formal Descriptions of Distributed Systems, *Protocol Specification, Testing and Verification VIII*, pp.189-204, North-Holland, 1988.

[4] ISO, Information Processing Systems - Open Systems Interconnection - LOTOS - A Formal Description Technique Based on the Temporal Ordering of Observational Behaviour, ISO8807, 1989.

[5] Ando, T., Kato, Y., Takahashi, K. and Noguchi, S., Compositional LOTOS Specification of Protocol Based on Temporal Logic, *Trans. IPSJ* vol. **34**, pp.1268-1280, 1993 (in Japanese).

[6] Ando, T., Kato, Y. and Takahashi, K., On Specifying Protocols Based on LOTOS and Temporal Logic, *IEICE Trans. Comm*, Vol.E77-B, pp.992-1006, 1994.

[7] Ando, T., Kato, Y. and Takahashi, K., Constraint-Oriented Specification using Temporal Logic, *Trans. IPSJ* (投稿中).

[8] Turner, K. J. (Ed.), *Using Formal Description Techniques - An Introduction to Estelle, LOTOS and SDL*, Wiley, 1993.

[9] Quemada, J. and Azcorra, A., Structuring Protocols using Exceptions in a LOTOS Extension, *Protocol Specification, Testing and Verification XII*, pp.81-96, North-Holland, 1992.

[10] Milner, R., *Communication and Concurrency*, Prentice Hall, 1989.