薬物相互作用の形式オントロジー

Drug Interaction Ontology

米崎直樹 † 泉 直子 †† 秋山卓見 †

Naoki YONEZAKI Naoko IZUMI Takumi AKIYAMA

† 東京工業大学大学院情報理工学研究科計算工学専攻

Dept. of Information and Computer Science,

yonezaki@fmx.cs.titech.ac.jp

akiyama@fmx.cs.titech.ac.jp

†† 十文字学園女子大学社会情報学部社会情報学科

Dept. of Information and Computer Science,

nizumi@jumonji-u.ac.jp

薬学において, 反応の相互作用は, 薬物などの分子, DNA などの生体物質, さらに大きくは臓器における現象に至るまで, 共通する言葉を用いて説明されることが多い. この共通性の中には, 異なるレベルの現象を統一的に理解するオントロジーが潜んでいると考えられる. 本論文では, このような相互作用を, 活性, 阻害の観点から統一的に記述する形式オントロジーを提案する. 本オントロジーは 1 階論理の推論規則の集合で与えられるが, その背景となるモデルの意味は, 線形論理に翻訳された反応の振る舞いの性質として与えられる. このような意味に対して, オントロジーを規定する推論規則が完全であることを示し, またその言語拡張を行った推論規則が健全であることを示す. 最後に世界最大の薬害として知られている 5-FU とソリブジンの飲み合わせによる薬害の存在を, このオントロジーの推論体系を用いて証明可能であることを例として示す.

キーワード: オントロジー 線形論理 活性 阻害 薬物相互作用 動作意味論

1 まえがき

生物情報に関する静的な属性に基づいた情報検索および, 未知の有効な薬剤の効果や副作用の発見において, 分子や薬物, 生体物質などの相互作用に関する情報を用いることが注目されている. しかし, その情報を表現する言葉の持つ意味は, 実際の化学的反応が持つ性質に基づいているにも係わらず, その詳細を知ることが出来ないことと, その作用の効果の捉え方が一様でないため曖昧である. [1]

一方, 様々な異なる物質レベルで共通の言葉が使われる. たとえば, 薬学における反応の相互作用は, 薬物などの分子, DNA などの生体物質, さらに大きくは臓器における現象に至るまで, 共通する言葉を用いて説明されることが多い. この共通性の中には, 異なるレベルの現象を統一的に理解するオントロジーが潜んでいると考えられる. 本論文では, このような相互作用の記述に汎用的に用いられる, 活性作用, 阻害作用の概念を統一的に記述する形式オントロジーを提案する.

オントロジーとは, 平たく言えば概念の関係である. 概念には通常いくつかの語彙が対応するため, 語彙と語彙の関係と捕らえる立場もある. またその関係が持つ意味を, 単なる事実のみで与えるだけでなく, その概念間の関係規則を与えることによって, 与えることも行われる. 後者は通常 1 階述語論理の公理あるいは推論規則の集合で示される. 例えば, is 関係に関するオントロジー [2],[3] や, メレオロジーと呼ばれる部分と全体に関する体系 [4] などがある.

オントロジーをこのように形式化した場合には, そのモデルの存在を確認したい. 通常は外延的集合論的解釈をモデルとするが, 本論文が対象とする作用に関する概念は, 本質的に動的な内容を含んでいる. そのためその解釈はこの作用の基となる動作モデルの特徴として解釈したい. ここでは, 複数の反応の原料となる物質や場所などの取り合いにより, 活性作用や阻害作用が発生するという現象に注目し, 資源を意識した論理である線形論理をモデルの記述に用いる. 線形論理の推論の実行を, 反応過程と捕らえ, その性質を概念の意味とするのである. このように意味モデルが備わった体系としてオントロジーを与えるため, これを特に形式オントロジーと言う.

以下 2 章ではまず, 実際の阻害と, 活性概念に関す

る推論の例を示し, このようなオントロジーを形式化しておくことが現実的にも重要であることを示す.

次に 3 章では, これらの概念を陽に表現する形式言語を導入し, そこで通常用いられる推論体系について紹介する. 更に 4 章では, 裏に隠れた反応, より一般的な言葉で言えば, ある生成物が作られるための条件集合と結果との関係を基礎に, 結果が生成されるための条件の取り合いを線形論理の推論でモデル化し, その意味を定義する.

5 章では, オントロジーを規定する推論体系がこの意味に関して完全であるが, 必ずしも健全ではないことを示し, 6 章でこれを健全とする言語拡張とそれに伴って拡張された推論体系を与え, 7 章で健全性の証明を与える.

8 章で, 世界最大の薬害として知られている 5FU とソリブジンの飲み合わせによる薬害の存在 [5] が, このオントロジーの公理を用いることにより, 推論可能であることを例として示し, 最後に 9 章でまとめる.

2 薬害における推論の例

抗ウィルス性の薬ソリブジンと抗癌役 5-FU の飲み合わせによる薬害は既知の事実から推測することにより, 確実に防止できたと言われている. ここで用いられた推論は以下のようなものである.

既知の反応は次の通りである.

反応 R_1 ソリブジン (SRV) は腸内で細菌 (TP) により分解され BVU を生成.

反応 R_2 BVU は DPD と反応し, DPDBUV を生成.

反応 R_3 DPD は 5-FU とも反応し, H2FUras を生成.

反応 R_4 5-FU は OPRTase と反応し, FDUP を生成.

反応 R_5 FDUP は RR と反応し, FdUMP を生成.

反応 R_6 FdUMP は ts と反応し, miCTC を生成.

反応 R_7 ts は dUMP と反応し, dTMP を生成.

反応 R_8 dTMP は DNA を合成する.

これらの一連の反応を図 1 に示す. この事実から次の推論が可能である.

推論 1 反応 R_1 の生成物 BVU が反応 R_2 の材料であるため, R_1 は R_2 を活性する. （R_1 が活発になると R_1 の生成物の濃度が増し, それを材料物質とする反応 R_2 が活発になる. ）

推論 2 反応 R_2 と R_3 が共通の材料 DPD を持ち, 材料の取り合いが起こるため, R_2 は R_3 を阻害する. （R_2 が活発になると, 逆に R_3 は阻害され, R_3 が活発になると, R_2 は阻害される. ）同様に, R_3 は R_4 を阻害する.

推論 3 推論 1, 2 より, 次のことがわかる. R_1 が活発になると, R_2 も活発になり, R_3 は阻害され, R_4 が活発になる. つまり, 反応 R_1 が R_4 を活性化する.

推論 4 反応 R_4 から R_8 の事実から推論 1,2,3 と同様に推論することにより, 反応 R_4 が R_8 を阻害することがわかる.

推論 4 の結果は抗癌薬 5-FU の薬効「5-FU は癌細胞の DNA 合成を阻害すると同時に正常細胞の生成も阻害すること」を表しており, これは既知の事実であった. 推論 3 は「SRV は 5-FU の反応 R_4 の効果を増大させる」ことを表している. これら二つから, 「SRV は DNA 合成を阻害する効果を増大させ, 癌細胞を死滅させる以上の効果を発揮し, 患者を死に至らしめる」ことが推論できる.

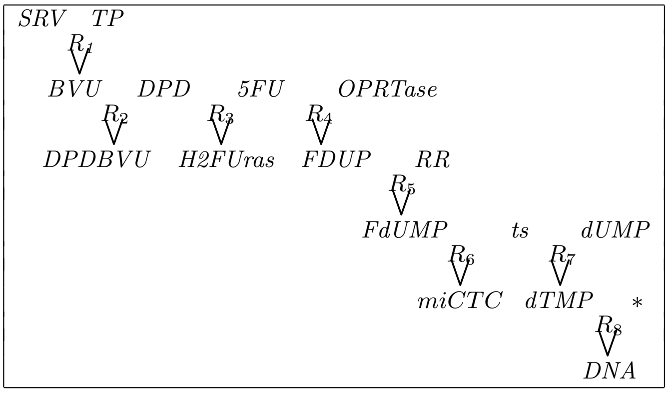

図 1: SRV と 5-FU の例

この図 1 において, $\begin{array}{c} a\ \ b \\ R \\ \bigvee \\ c \end{array}$ では, a, b は反応物質又は, 触媒, あるいは反応場所等, 反応 R に必要なものを示し, c は, 反応の生成物を示す.

3 形式化

本節では, 前章で述べたような推論を行えるように, 阻害関係, 活性化関係を形式化する. つまり, 反

応の‘阻害関係’, ‘活性化関係’ を記述するための言語 L およびその推論規則 RulesR を定義する．

阻害, 活性という言葉は, 本節で定義する反応間の用語としてだけではなく, 薬物間や薬物と反応の間の用語として用いられることもある．しかし, 意味の本質は反応間の関係にあるため, 本節ではその関係に限って定義する．薬物間の関係や薬物と反応の間の関係については, 反応間の関係をその参加者の関係と見直すことによって得られる．その推論は付録 A に示される．

3.1 言語 L

ここでは, 反応間の阻害活性関係を記述するための言語 L を定義する．

定義 1 論理式を記述するためのシンボル集合を以下のように定義する．

$\mathcal{O}$	:	物質を表す定数集合
$\mathcal{P}re$	:	以下の述語からなる述語集合
		Inh, $Prom$, $React$
$\mathcal{R}$	:	反応名を表す定数集合

End

定義 2 $x, y, z \in \mathcal{O}$ と $R, R' \in \mathcal{R}$ とする．このとき, 式は以下のとおりである．

$$Inh(R, R'), Prom(R, R'), React(R, x, y, z)$$

これらの式からなる集合を言語 L とする． End

これらの式の直感的な意味は次の通りである．
$Inh(R, R')$: 反応 R が反応 R' を阻害する．
$Prom(R, R')$: 反応 R が反応 R' を活性化する．
$React(R, x, y, z)$: 反応 R は物質 x と物質 y から物質 z を作る反応である．

以下では, $React(R, x, y, z)$ は, 前節と同様に図 2 のように 2 入力、1 出力の逆三角形で表す．n 入力 m 出力の反応に一般化することは容易であるが, ここでは記述の複雑さを避けるために 2 入力 1 出力について議論を行う．また、その入力は必ずしも物質である必要はなく反応の条件なるものなら何でも良い．

3.2 推論規則 RulesR

前節で示した推論を形式化した言語 L に関する推論規則（スキーマ）を定義する． これらの推論規則

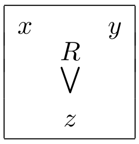

図 2: 反応 $React(R, x, y, z)$

の集合を RulesR と呼ぶことにする．以下に現れるパラメータは全てメタ変数である．

1. $\dfrac{React(R, x, y, w)}{React(R, y, x, w)}$

2. $\dfrac{React(R_1, x, y, w_1) \quad React(R_2, y, z, w_2)}{Inh(R_1, R_2)}$

3. $\dfrac{React(R_1, x, y, w_1) \quad React(R_2, w_1, z, w_2)}{Prom(R_1, R_2)}$

4. $\dfrac{Inh(R, R') \quad Inh(R', R'')}{Prom(R, R'')}$

5. $\dfrac{Inh(R, R') \quad Prom(R', R'')}{Inh(R, R'')}$

6. $\dfrac{Prom(R, R') \quad Inh(R', R'')}{Inh(R, R'')}$

7. $\dfrac{Prom(R, R') \quad Prom(R', R'')}{Prom(R, R'')}$

推論規則 1 は, 反応に使用される二つの物質には順序がないことを示している．推論規則 2 は, 物質（ここでは y）を取り合う二つの反応（図 3）は互いに阻害関係にあることを示している．推論規則 3 は, 反応

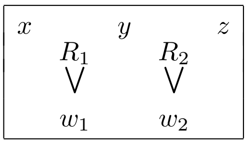

図 3: 同じ物質 y を取り合う反応

R_2 が反応 R_1 の生成物を使用する場合（図 4）, R_1 は R_2 を活性化することを示している．推論規則 4 は, R が R' を阻害し, R' が R'' を阻害関係にあるとき, R は R'' を活性化することを表している．推論規則 5 は, R が R' を阻害し, R' が R'' を活性する時, R が R'' を阻害することを表している．推論規則 6 は, R が R' を活性化し, R' が R'' を阻害する時, R は R'' を阻害する事を表している．推論規則 7 は, R が R' を, R' が R'' をがそれぞれ活性化する時, R が R'' を活性化することを表している．

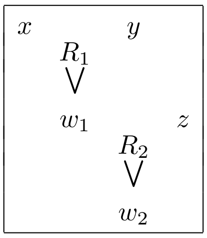

図 4: 反応が他の反応の生成物を使用する場合

4　言語 L の意味論

前章で与えられた推論規則で表される推論は, 薬物や生体物質の反応について, 通常良く使われているものである．これは, 前節の例で説明したように自然な推論規則のように思われるが, その意味を統一的に与えることは自明なことではない.

背景に物質の反応があることから, 反応の活性, 阻害が反応速度の相対的な変化であることは明らかである．反応速度は, 物質濃度の時間微分で与えられる．しかし全ての反応に対して, 濃度の初期値を与え, 連立の微分方程式を意味として与えることは現実的ではない.

そこでここでは, 2 つの反応に必要な要件の取り合いをもとに, 阻害, 活性の概念の意味を定義する．すなわち, 反応速度の増減などという連続的な値の概念を用いずに, 意味を定義する．この記述の道具として線形論理の部分言語を用いる.

まず最初に線形論理を簡単に紹介する．次に, 線形論理を用いて動的モデルを定義し, 最後に, その動的モデルの性質として, 前節で定義した式の意味を与える.

4.1　線形論理

線形論理 [6] は, 量的関係や消費の概念を記述することのできる論理体系である．線形論理の演算子のうち, 本システムで用いるのは乗法的 and である $\otimes$, 含意である $\multimap$, $\otimes$ の繰り返しである !, そして $\forall$ と $\exists$ の 5 つである.

$A \otimes B$ は,「A と B の両方」という意味を表している．$A \multimap B$ は「A を消費して B を生成する」という意味を表している．$!A$ は $A \otimes A \otimes \cdots$ という, $\otimes$ による A の 0 個以上の繰り返しを表している．また, $\forall x F(x)$ は通常の $\forall$ の意味に近く,「任意の x について $F(x)$」という意味であり, $\exists$ は $\forall$ の双対である.

線形論理の推論規則のうち, $\otimes$, $\multimap$, !, $\forall$, $\exists$ に関するものを以下に示す.

$$\frac{A, B, \Gamma \vdash \Delta}{A \otimes B, \Gamma \vdash \Delta} \otimes L \qquad \frac{\Gamma \vdash \Delta, A \vdash \Gamma', \Delta', B}{\Gamma, \Gamma' \vdash \Delta, \Delta', A \otimes B} \otimes R$$

$$\frac{B, \Gamma \vdash \Delta \quad \Gamma' \vdash \Delta', A}{A \multimap B, \Gamma, \Gamma' \vdash \Delta, \Delta'} \multimap L \qquad \frac{A, \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \multimap B} \multimap R$$

$$\frac{\Gamma \vdash \Delta}{!A, \Gamma \vdash \Delta} !W \qquad \frac{!A, !A, \Gamma \vdash \Delta}{!A, \Gamma \vdash \Delta} !C \qquad \frac{A, \Gamma \vdash \Delta}{!A, \Gamma \vdash \Delta} !L$$

$$\frac{F(t), \Gamma \vdash \Delta}{\forall x F(x), \Gamma \vdash \Delta} \forall L \qquad \frac{\Gamma \vdash \Delta, F(a)}{\Gamma \vdash \Delta, \forall x F(x)} \forall R$$

(t は任意)　　(a は結論部には出現しない)

$$\frac{F(a), \Gamma \vdash \Delta}{\exists x F(x), \Gamma \vdash \Delta} \exists L \qquad \frac{\Gamma \vdash \Delta, F(t)}{\Gamma \vdash \Delta, \exists x F(x)} \exists R$$

(a は結論部には出現しない)　　(t は任意)

4.2　動的モデル

本節では動的モデルを定義する．まず最初に, モデル定義に用いる一階線形論理 $\mathcal{L}$ を定義し, その後, それを用いて動的モデルを定義する.

定義 3　1. $\mathcal{L}$ の言語は, 以下の通りである.

$\mathcal{OL} \stackrel{def}{=} \{\bar{x} \| x \in \mathcal{O}\}$: 物質を表す述語集合
$\mathcal{RL} \stackrel{def}{=} \{\bar{R} \| R \in \mathcal{R}\}$: 反応名を表す定数集合
$\mathcal{IO} \stackrel{def}{=} \{i, o\}$: 入力（必要な前件, 物質など）と出力（生成物）を表す定数集合

2. $\bar{x} \in \mathcal{OL}$, $\alpha \in \mathcal{IO}$, $\bar{R} \in \mathcal{RL}$ に対し, $\bar{x}(\alpha, \bar{R})$ を線形論理の言語 $\mathcal{L}$ の原子命題式とする.

3. 命題式と通常の線形論理の演算子 ($\otimes$, $\multimap$, !, $\forall$, $\exists$) からなる式は言語 $\mathcal{L}$ の命題式とする.　End

注意 1 以後, 意味論での話であることが明らかにわかる場合には, $\bar{x}$, $\bar{R}$ の代わりに x, R を使う.　End

注意 2 $x(i, R)$ は反応 R の原料物質 x を表し, $x(o, R)$ は反応 R の結果作られた生成物 x を表す．式 $!(x_1(i, R) \otimes x_2(i, R) \multimap y(o, R))$ は, 原料として x_1 と x_2 を使用して y を生成する反応 R が（必要なだけ）あることを表す.　End

定義 4 (動的モデル) 動的モデル Δ は, 反応を表す式 $!(x_1(i, R) \otimes x_2(i, R) \multimap y(o, R))$ からなる一階線形論理 $\mathcal{L}$ の式集合である.　End

例

$$\Delta = \left\{ \begin{array}{l} !(x(i, R_1) \otimes y(i, R_1) \multimap w_1(o, R_1)), \\ !(w_1(i, R_2) \otimes z(i, R_2) \multimap w_2(o, R_2)), \\ !(z(i, R_3) \otimes u(i, R_3) \multimap w_3(o, R_3)), \end{array} \right\}$$

とする．Δ は反応として R_1, R_2, R_3 がある状況を表す動的モデルである.

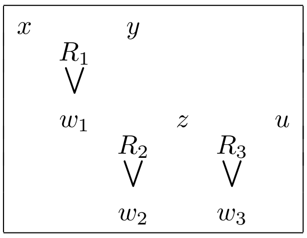

図 5: 動的モデル

4.3 動的モデルによる言語 L の意味

本節では, 動的モデル Δ を用いて, 意味関数

$\mathcal{I}_\Delta : L \to \mathcal{L}$ の証明に関する性質

を定義する. 最初に $React$ の意味を以下のように定義する.

1. $\mathcal{I}_\Delta(React(R,x,y,z)) \stackrel{def}{=}$
$$\Delta \vdash !(x(i,R) \otimes y(i,R) \multimap z(o,R))$$

次に, $Prom$ と Inh の意味記述に必要な Pathway を定義する. Pathway とは, いくつかの物が存在した時に連鎖して起きる一連の反応のことをいう. 反応の列 Seq が動的モデル Δ の元で Pathway であることを表す述語 $PW(Inputs, Reacts, Initial, Seq, y, z, Con, \Delta)$ を定義する. ここで各パラメータは, 次の物を表す.

$Inputs$ は Pathway Seq を構成する反応の入力物質の集合を, $Reacts$ は Seq に現れる反応の式と,「生成物は他の反応の原料となりうる」という基礎条件を表す式の集合を表す. $Initial$ は Seq の一連の反応のきっかけとなった最初の物質の集合を表し, 更に, y, z はこの Seq の終端となっている物を表す. これらは他の反応または Pathway と関連を持ち得る物であり, y, z から更に一連の反応を考えることにより, 新たな Pathway を作ることができる. このうち, y は最後の反応の原料のオブジェクトであり, z は結果作られたオブジェクトを表す. Con は Pathway を構成する時に隣接する二つの反応が共有するオブジェクトの集合を表す.

例えば, 図 5 の Pathway は,

$PW(\{\forall r x(i,r), \forall r y(i,r), \forall r z(i,r), \forall r u(i,r)\},$
$\{!(x(i,R_1) \otimes y(i,R_1) \multimap w_1(o,R_1)),$
$!(w_1(i,R_2) \otimes z(i,R_2) \multimap w_2(o,R_2)),$
$!(z(i,R_3) \otimes u(i,R_3) \multimap w_3(o,R_3)),$
$!(w_1(o,R_1) \multimap \forall r w_1(i,r)),$
$!(w_2(o,R_2) \multimap \forall r w_2(i,r)),$
$!(w_3(o,R_3) \multimap \forall r w_3(i,r)),$
$\{x,y\}, R_1 \cdot R_2 \cdot R_3, u, w_3, \{w_1, z\}, \Delta)$ で表される.

述語 PW は以下のように定義される.

定義 5 以下の 1.〜4. によって PW が真であることが分かる時のみ PW は真である. 但し, 以下の定義では $\cup$ は, 多重集合の和集合演算を表す.

1. $\Delta \vdash !(x(i,R) \otimes y(i,R) \multimap w_1(o,R)) \wedge$
$\Delta \vdash !(w_1(i,R') \otimes z(i,R') \multimap w_2(o,R'))$
$\Rightarrow PW(\{\forall r x(i,r), \forall r y(i,r), \forall r z(i,r)\},$
$\{!(x(i,R) \otimes y(i,R) \multimap w_1(o,R)),$
$!(w_1(i,R') \otimes z(i,R') \multimap w_2(o,R')),$
$!(w_1(o,R) \multimap \forall r w_1(i,r)),$
$!(w_2(o,R') \multimap \forall r w_2(i,r))\},$
$\{x,y\}, R \cdot R', z, w_2, \{w_1\}, \Delta)$

2. $\Delta \vdash !(x(i,R) \otimes y(i,R) \multimap w_1(o,R)) \wedge$
$\Delta \vdash !(y(i,R') \otimes z(i,R') \multimap w_2(o,R'))$
$\Rightarrow PW(\{\forall r x(i,r), \forall r y(i,r), \forall r z(i,r)\},$
$\{!(x(i,R) \otimes y(i,R) \multimap w_1(o,R)),$
$!(y(i,R') \otimes z(i,R') \multimap w_2(o,R')),$
$!(w_1(o,R) \multimap \forall r w_1(i,r)),$
$!(w_2(o,R') \multimap \forall r w_2(i,r))\},$
$\{x\}, R \cdot R', z, w_2, \{y\}, \Delta)$

3. $PW(Inputs, Reacts, Initial, Seq, y, z, Con, \Delta) \wedge$
$\Delta \vdash !(z(i,R) \otimes u(i,R) \multimap w(o,R))$
$\Rightarrow PW(Inputs \cup \{\forall r u(i,r)\},$
$Reacts \cup \{!(z(i,R) \otimes u(i,R) \multimap w(o,R)),$
$!(w(o,R) \multimap \forall r w(i,r))\},$
$Initial, Seq \cdot R, u, w, Con \cup \{z\}, \Delta)$

4. $PW(Inputs, Reacts, Initial, Seq, y, z, Con, \Delta) \wedge$
$\Delta \vdash !(y(i,R) \otimes u(i,R) \multimap w(o,R))$
$\Rightarrow PW(Inputs \cup \{\forall r u(i,r)\},$
$Reacts \cup \{!(y(i,R) \otimes u(i,R) \multimap w(o,R)),$
$!(w(o,R) \multimap \forall r w(i,r))\},$
$Initial, Seq \cdot R, u, w, Con \cup \{y\}, \Delta)$ End

ここで, 後の議論を簡単にするため Seq 中の同一の R の出現については異なる出現としてラベル付けされているとし, Reacts 中の式における R の出現にもそのラベルが反映されているものとする.

このとき, Inh と $Prom$ の意味を以下の 2,3 のように与える. 但し, $Oset(P)$, $head(Seq)$, $last(Seq)$ は次のように定義される.

P_j が全ての j について $P_j = x_j(\alpha, R_j)$ の形をしている時,
$Oset(P_1 \otimes P_2 \otimes ... \otimes P_m) \stackrel{def}{=} \{x_1, x_2, ..., x_m\}$

$Seq = S_1 \cdot S_2 \cdot .....S_{n-1} \cdot S_n$ であるとき,

$head(Seq) \stackrel{def}{=} S_1$
$last(Seq) \stackrel{def}{=} S_n$

2. $\mathcal{I}_\Delta(Inh(R, R')) \stackrel{def}{=}$
$\exists Inputs \exists Reacts \exists Initial \exists Seq \exists y \exists z \exists Con($
$PW(Inputs, Reacts, Initial, Seq, y, z, Con, \Delta) \wedge$
$head(Seq) = R \wedge last(Seq) = R' \wedge$
$\forall m \forall P_1 ... \forall P_m($
$(\ Inputs \cup Reacts \vdash P_1 \otimes ... \otimes P_m \wedge$
$Initial \cap Oset(P_1 \otimes ... \otimes P_m) = \phi \wedge$
$Con \cap Oset(P_1 \otimes ... \otimes P_m) = \phi)$
$\Rightarrow y \in Oset(P_1 \otimes ... \otimes P_m))$

3. $\mathcal{I}_\Delta(Prom(R, R')) \stackrel{def}{=}$
$\exists Inputs \exists Reacts \exists Initial \exists Seq \exists y \exists z \exists Con($
$PW(Inputs, Reacts, Initial, Seq, y, z, Con, \Delta) \wedge$
$head(Seq) = R \wedge last(Seq) = R' \wedge$
$\forall m \forall P_1 ... \forall P_m($
$(\ Inputs \cup Reacts \vdash P_1 \otimes ... \otimes P_m \wedge$
$Initial \cap Oset(P_1 \otimes ... \otimes P_m) = \phi \wedge$
$Con \cap Oset(P_1 \otimes ... \otimes P_m) = \phi)$
$\Rightarrow y \notin Oset(P_1 \otimes ... \otimes P_m))$

$\mathcal{I}_\Delta(Inh(R, R'))$ の直感的な意味は, Pathway に係わる反応で実行可能な反応が全て実行された時, R が反応するならば, R' が反応できず, 逆に R が反応しないならば, R' が反応可能であることである.

同様に $\mathcal{I}_\Delta(Prom(R, R'))$ の直感的な意味は, 同様な状況で R が反応するならば, R' も反応でき, 逆に R が反応しないならば, R' も反応できないことである.

5 完全性

本システムの完全性は, 以下のように表される.

定理 1 (完全性)
$\forall \Delta(\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(\phi)) \Longrightarrow \Gamma \vdash_{\mathrm{RulesR}} \phi$
ここで, $\Gamma = \gamma_1 ... \gamma_n$ の時,
$\mathcal{I}_\Delta(\Gamma) \stackrel{def}{=} \mathcal{I}_\Delta(\gamma_1), \mathcal{I}_\Delta(\gamma_2), ... \mathcal{I}_\Delta(\gamma_n)$ END

この証明のために, いくつかの補題を示す.

5.1 補題

補題 1 $\forall \Delta(\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(React(R, x, y, z))) \Longrightarrow \Gamma \vdash_{\mathrm{RulesR}} React(R, x, y, z)$

(証明)
$\mathcal{I}_\Delta(React(R, x, y, z))$ は定義より
$\Delta \vdash !(x(i, R) \otimes y(i, R) \multimap z(o, R))$ である.

一方, RulesR には $React$ を推論できる推論規則は (1) 以外に存在しない. したがって, この補題の後件部は $React(R, x, y, z) \in \Gamma$ 又は, $React(R, y, x, z) \in \Gamma$ と等価である. ここで, 対偶を示す. つまり,

$$React(R, x, y, z) \notin \Gamma \ and \ React(R, y, x, z) \notin \Gamma \Longrightarrow$$
$$\mathcal{I}_\Delta(\Gamma) \not\models \Delta \vdash !(x(i, R) \otimes y(i, R) \multimap z(o, R))$$

を示す.
$React(R, x, y, z) \notin \Gamma$ であるので, Γ には $Inh(R, R')$, $Prom(R, R')$, $React(R', x', y', z')$ のみが含まれる. 但し $R' \neq R$ である. この時, $Inh(R, R')$ $Prom(R, R')$ の定義より, $Inh(R, R')$, $Prom(R, R')$ からは $\Delta \vdash !(x(i, R) \otimes y(i, R) \multimap z(o, R))$ は含意されない. よって,
$\mathcal{I}_\Delta(\Gamma) \not\models \Delta \vdash !(x(i, R) \otimes y(i, R) \multimap z(o, R))$
である. End

補題 2 $\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(Inh(R, R')) \Longrightarrow$
$\Gamma \vdash_{\mathrm{RulesR}} Inh(R, R')$
$\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(Prom(R, R')) \Longrightarrow$
$\Gamma \vdash_{\mathrm{RulesR}} Prom(R, R')$

(証明)
前件を仮定し, 後件を示す.
$\mathcal{I}_\Delta(Inh(R, R'))$,$\mathcal{I}_\Delta(Prom(R, R'))$ の定義より, 前件を仮定すれば,
$PW(Inputs, Reacts, Initial, Seq, y, z, Con, \Delta) \wedge$
$head(Seq) = R \wedge last(Seq) = R'$
が成立することがわかる. この pathway Seq の長さ $|Seq|$ に関する帰納法によって証明する.

i) $|Seq| = 2$ のとき

a) $\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(Inh(R, R'))$ を仮定する.
補題 3 より, 以下を満たす x, w_1, w_2, y, z が存在する.

$$\mathcal{I}_\Delta(\Gamma) \models \Delta \vdash !(x(i, R) \otimes w_1(i, R) \multimap w_2(o, R)),$$
$$\Delta \vdash !(w_1(i, R') \otimes y(i, R') \multimap z(o, R'))$$

上式と補題 1 より,

$\Gamma \vdash_{\mathrm{RulesR}} React(R, x, w_1, w_2), React(R', w_1, y, z)$

となる. したがって規則 2 より

$$\Gamma \vdash_{\mathrm{RulesR}} Inh(R, R')$$

b) $\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(\mathit{Prom}(R, R'))$ を仮定する.
補題 3 より, 以下を満たす x, w_1, w_2, y, z が存在する.

$$\begin{aligned}\mathcal{I}_\Delta(\Gamma) \quad \models \quad & \Delta \vdash !(x(i,R) \otimes w_1(i,R) \multimap w_2(o,R)), \\ & \Delta \vdash !(w_2(i,R') \otimes y(i,R') \multimap z(o,R'))\end{aligned}$$

上式と補題 1 より,

$$\Gamma \vdash_{\mathrm{RulesR}} \mathit{React}(R, x, w_1, w_2), \mathit{React}(R', w_2, y, z)$$

となる. したがって規則 3 より

$$\Gamma \vdash_{\mathrm{RulesR}} \mathit{Prom}(R, R')$$

a), b) より, $|\mathit{Seq}| = 2$ のとき成立.

ii) $|\mathit{Seq}| = k$ で成り立つと仮定して $|\mathit{Seq}| = k+1$ のとき $(k \geq 2)$

a) $\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(\mathit{Inh}(R, R'))$ を仮定する.
このとき, i) と同様に, $\mathcal{I}_\Delta(\mathit{Inh}(R, R'))$ の定義より
$PW(\mathit{Inputs}, \mathit{Reacts}, \mathit{Initial}, \mathit{Seq}, y, z, \mathit{Con}, \Delta)$
が言える. いま, $|\mathit{Seq}| \geq 3$ であるので PW の定義 3, 4 より $\mathcal{I}_\Delta(\Gamma)$ から以下が成り立つ.

$PW(\mathit{Inputs}', \mathit{Reacts}', \mathit{Initial}, \mathit{Seq}', w_1, w_2, \mathit{Con}', \Delta)$
および以下の 1, 2 のいずれかが成立する
x, w_1, w_2, y, z が存在する. (a)

1.$\Delta \vdash !(w_2(i,R') \otimes y(i,R') \multimap z(o,R'))$
ただし$\mathit{Seq} = (\mathit{Seq}' \cdot R')$
$\mathit{last}(\mathit{Seq}') = R''$
$\mathit{Inputs} = \mathit{Inputs}' \cup \{\forall r(y(i,r))\}$
$\mathit{Reacts} = \mathit{Reacts}' \cup$
$\{!(w_2(i,R') \otimes y(i,R') \multimap z(o,R'))\}$
$\mathit{Con} = \mathit{Con}' \cup \{w_2\}$

2.$\Delta \vdash !(w_1(i,R') \otimes y(i,R') \multimap z(o,R'))$
ただし$\mathit{Seq} = (\mathit{Seq}' \cdot R')$
$\mathit{last}(\mathit{Seq}') = R''$
$\mathit{Inputs} = \mathit{Inputs}' \cup \{\forall r(y(i,r))\}$
$\mathit{Reacts} = \mathit{Reacts}' \cup$
$\{!(w_1(i,R') \otimes y(i,R') \multimap z(o,R'))\}$
$\mathit{Con} = \mathit{Con}' \cup \{w_1\}$

1.) いま, $\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(\mathit{Inh}(R, R'))$ が仮定されている. ここで, $\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(\mathit{Inh}(R, R''))$ である (補題 4). よって帰納法の仮定より

$$\Gamma \vdash_{\mathrm{RulesR}} \mathit{Inh}(R, R'') \qquad (1)$$

また,
$\mathcal{I}_\Delta(\Gamma) \models \Delta \vdash !(w_2(i,R') \otimes y(i,R') \multimap z(o,R'))$
であるので
$\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(\mathit{React}(R', w_2, y, z))$ となる. これと補題 1 より

$$\Gamma \vdash_{\mathrm{RulesR}} \mathit{React}(R', w_2, y, z) \qquad (2)$$

さらに, $\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(\mathit{Inh}(R, R''))$ より, ある w_3 が存在して $\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(\mathit{React}(R'', w_3, w_1, w_2))$ である. よって補題 1 より

$$\Gamma \vdash_{\mathrm{RulesR}} \mathit{React}(R'', w_3, w_1, w_2) \qquad (3)$$

(2), (3) から規則 3 によって $\Gamma \vdash_{\mathrm{RulesR}} \mathit{Prom}(R'', R')$ である. これと (1) から規則 5 によって $\Gamma \vdash_{\mathrm{RulesR}} \mathit{Inh}(R, R')$ である.

2.) 補題 5, 規則 2, 規則 6 を用いて 1. とほぼ同様に $\Gamma \vdash_{\mathrm{RulesR}} \mathit{Inh}(R, R')$ が言える.

b) $\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(\mathit{Prom}(R, R'))$ を仮定する.
a) と同様に示す. $\mathcal{I}_\Delta(\mathit{Prom}(R, R'))$ の定義より $PW(\mathit{Inputs}, \mathit{Reacts}, \mathit{Initial}, \mathit{Seq}, y, z, \mathit{Con}, \Delta)$
が言える. いま, $|\mathit{Seq}| \geq 3$ であるので PW の定義 3, 4 より $\mathcal{I}_\Delta(\Gamma)$ から上記の (a) と同じ性質が成り立つ.

1.) いま, $\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(\mathit{Prom}(R, R'))$ が仮定されている. ここで, $\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(\mathit{Prom}(R, R''))$ である (補題 6). よって帰納法の仮定より

$$\Gamma \vdash_{\mathrm{RulesR}} \mathit{Prom}(R, R'') \qquad (4)$$

また,
$\mathcal{I}_\Delta(\Gamma) \models \Delta \vdash !(w_2(i,R') \otimes y(i,R') \multimap z(o,R'))$
であるので
$\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(\mathit{React}(R', w_2, y, z))$
となる. これと補題 1 より

$$\Gamma \vdash_{\mathrm{RulesR}} \mathit{React}(R', w_2, y, z) \qquad (5)$$

さらに, $\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(\mathit{Inh}(R, R''))$ より, ある w_3 が存在して $\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(\mathit{React}(R'', w_3, w_1, w_2))$ である. よって補題 1 より

$$\Gamma \vdash_{\mathrm{RulesR}} \mathit{React}(R'', w_3, w_1, w_2) \qquad (6)$$

(5), (6) から規則 3 によって $\Gamma \vdash_{\mathrm{RulesR}} Prom(R'', R')$ である．これと (4) から規則 7 によって $\Gamma \vdash_{\mathrm{RulesR}} Prom(R, R')$ である．

2.) 補題 7, 規則 2, 規則 4 を用いて 1. とほぼ同様に $\Gamma \vdash_{\mathrm{RulesR}} Prom(R, R')$ が言える．

i), ii) より示せた． End

補題 3 補題 2 の証明において $|Seq| = 2$ のとき, 以下を満たす x, w_1, w_2, y, z が存在する．

i) $\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(Inh(R, R'))$ ならば

$$\begin{aligned}\mathcal{I}_\Delta(\Gamma) \ \models \ & \Delta \vdash !(x(i,R) \otimes w_1(i,R) \multimap w_2(o,R)),\\ & \Delta \vdash !(w_1(i,R') \otimes y(i,R') \multimap z(o,R'))\end{aligned}$$

ii) $\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(Prom(R, R'))$ ならば

$$\begin{aligned}\mathcal{I}_\Delta(\Gamma) \ \models \ & \Delta \vdash !(x(i,R) \otimes w_1(i,R) \multimap w_2(o,R)),\\ & \Delta \vdash !(w_2(i,R') \otimes y(i,R') \multimap z(o,R'))\end{aligned}$$

(証明)
$|Seq| = 2$ であるので PW の定義 1, 2 より以下のいずれかを満たす x, w_1, w_2, y, z が存在する．

i) $\mathcal{I}_\Delta(\Gamma) \models \Delta \vdash !(x(i,R) \otimes w_1(i,R) \multimap w_2(o,R)),$
$\Delta \vdash !(w_1(i,R') \otimes y(i,R') \multimap z(o,R'))$

ii) $\mathcal{I}_\Delta(\Gamma) \models \Delta \vdash !(x(i,R) \otimes w_1(i,R) \multimap w_2(o,R)),$
$\Delta \vdash !(w_2(i,R') \otimes y(i,R') \multimap z(o,R'))$

i) のとき,

$$\begin{aligned}Inputs &= \{\forall r(x(i,r)), \forall r(w_1(i,r)), \forall r(y(i,r))\}\\ Reacts &= \{!(x(i,R) \otimes w_1(i,R) \multimap w_2(o,R)),\\ &\quad !(w_1(i,R') \otimes y(i,R') \multimap z(o,R'))\}\end{aligned}$$

このとき $\mathcal{I}_\Delta(Inh(R, R'))$, $\mathcal{I}_\Delta(Prom(R, R'))$ の前件部を満たす $m, P_1, \ldots, P_m$ は

$m = 2, \quad P_1 = y(i, R'), P_2 = w_2(o, R)$ のみである．

よって $Oset(P_1 \otimes P_2) = \{y, w_2\}$ となり, $\mathcal{I}_\Delta(Prom(R, R'))$ の定義の後件部 $y \notin Oset(P_1 \otimes P_2)$ を満たさない. (2)

ii) のとき,

$$\begin{aligned}Inputs &= \{\forall r(x(i,r)), \forall r(w_1(i,r)), \forall r(y(i,r))\}\\ Reacts &= \{!(x(i,R) \otimes w_1(i,R) \multimap w_2(o,R)),\\ &\quad !(w_2(i,R') \otimes y(i,R') \multimap z(o,R'))\}\end{aligned}$$

このとき $\mathcal{I}_\Delta(Inh(R, R'))$, $\mathcal{I}_\Delta(Prom(R, R'))$ の前件部を満たす $m, P_1, \ldots, P_m$ は

$m = 1, \quad P_1 = z(o, R')$

の 1 通りである．よって $Oset(P_1) = \{z\}$ となる．これは $\mathcal{I}_\Delta(Inh(R, R'))$ の定義の後件部
$y \in Oset(P_1)$ を満たさない. (3)

(1) と (2) の対偶からこの補題の前半が, (1) と (3) の対偶から後半がただちに導出される． End

補題 4 補題 2 の証明中の ii-a-1 の条件が成り立つとき, $\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(Inh(R, R''))$

(証明)
補題 2 中の仮定より,
$PW(Inputs', Reacts', Initial, S', w_1, w_2, Con, \Delta) \wedge head(S') = R \wedge last(S') = R''$ が成立する． よって $\mathcal{I}_\Delta(Inh(R, R''))$ の定義中の $\forall m \forall P_1 \ldots$ 以降の前件部が成り立つような $m, P_1, P_2 \ldots$ に対して, $\forall m \forall P_1 \ldots$ 以降の後件部が成り立つことを言えばよい． そのような $m, P_1, P_2 \ldots$ は, $\mathcal{I}_\Delta(Inh(R, R'))$ の定義における $\forall m \forall P_1 \ldots$ 以降の前件部を成り立たせるような $m, P_1, P_2 \ldots$ に含まれている． よって $\mathcal{I}_\Delta(Inh(R, R'))$ の定義の中の $\forall m \forall P_1 \ldots$ 以降の後件部を仮定して $\mathcal{I}_\Delta(Inh(R, R''))$ の定義の中の $\forall m \forall P_1 \ldots$ 以降の後件部を言えばよい．

$\mathcal{I}_\Delta(Inh(R, R'))$ の定義の中の $\forall m \forall P_1 \ldots$ 以降の後件部を仮定するので

$y \in Oset(P_1 \otimes \cdots \otimes P_m)$

が成り立つ．

従って, $w_1 \in Oset(P_1 \otimes \cdots \otimes P_m)$ であり, $\mathcal{I}_\Delta(Inh(R, R''))$ の定義の後件部が成り立つので $\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(Inh(R, R''))$ が成立する．

補題 5 補題 2 の証明中の ii-a-2 の条件が成り立つとき, $\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(Prom(R, R''))$

補題 6 補題 2 の証明中の ii-b-1 の条件が成り立つとき, $\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(Prom(R, R''))$

補題 7 補題 2 の証明中の ii-b-2 の条件が成り立つとき, $\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(Inh(R, R''))$

(証明)
以上 3 つの補題は, 補題 4 と同様にして示される．

5.2 完全性

補題 1, 補題 2 によって

$$\mathcal{I}_{\Delta}(\Gamma) \models \mathcal{I}_{\Delta}(\phi) \Longrightarrow \Gamma \vdash_{\mathrm{RulesR}} \phi$$

が成立する.

6 推論規則の詳細化

次の Pathway(図 6) では,

$React(R_1, x, y, w_1)$, $React(R_2, w_1, v, w_2)$, $React(R_3, u, w_1, w_3)$ の 3 つの反応あり, $Prom(R_1, R_2)$, $Inh(R_2, R_3)$ が成立している. 推論規則 6 より, $Inh(R_1, R_3))$ が導出されるが, $\mathcal{I}(Inh(R_1, R_3))$ は真にはならない. 従って, 推論規則は健全ではない.

それは, 次の理由に示されるように $R_1 \cdot R_2 \cdot R_3$ は Pathway にならないからである.

$PW(\{\forall rx(i, r), \forall ry(i, r), \forall rv(i, r)\},$
$\{!(x(i, R_1) \otimes y(i, R_1) \multimap w_1(o, R_1)),$
$!(w_1(i, R_2) \otimes v(i, R_2) \multimap w_2(o, R_2))\},$
$\{x, y\}, R_1 \cdot R_2, v, w_2\}\}, \Delta)$ —— (1)

(1) の Pathway に関して, 定義 5 より, ある適当な v', w', w'' に対して,

$\vdash !(v(i, R_3) \otimes v'(i, R_3) \multimap w'(o, R_3))$, 又は,
$\vdash !(w_2(i, R_3) \otimes w'(i, R_3) \multimap w''(o, R_3))$,

である時のみに $R_1 \cdot R_2 \cdot R_3$ は Payhway となる. しかし, 反応 R_3 は, $React(R_3, u, w_1, w_3)$ であるので, 上のどちらの場合にも当てはまらないため, $R_1 \cdot R_2 \cdot R_3$ は Payhway とはならず, $\mathcal{I}_{\Delta}(I(R_1, R_3))$ は真とはならない.

実際, R_1 の反応が活性化すれば, R_3 は活性化されはすれども, 阻害はされない.

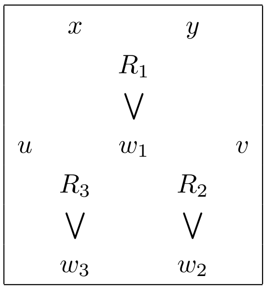

図 6: 問題の Pathway

$Inh(R, R')$, $Prom(R, R')$ どのオブジェクトが Pathway の次の反応に使うことができるのかを記述し, Pathway を作ることができるときのみに規則が成り立つように, 述語のパラメータを増すことにより言語拡張を行い推論規則を詳細化する必要がある.

6.1 健全な公理のための詳細化

定義 6 $x, y \in \mathcal{O}$ と $R, R' \in \mathcal{R}$ に対して, 言語 L' の原始式を以下ように定義する:
$Inh(R, x, R', y)$, $Prom(R, x, R', y)$ End

$Inh(R, x, R', y)$ は, R が R' を阻害する Pathway において R に隣接する反応はオブジェクト x を共有し, R' に隣接する反応はオブジェクト y を共有することを表す.

$Prom(R, x, R', y)$ について, R が R' を活性化する Pathway において同様の関係がある.

これらを用いた推論規則は, 以下のようになる.

$$\frac{Inh(R, x, R', y) \quad Inh(R', u, R'', v)}{Prom(R, x, R'', v)}\ y \neq u$$

これは, R と R' の Pathway で使われる R' が共有しているオブジェクト y と R' と R'' の Pathway で使われる R' が共有しているオブジェクト u が異なる時に R から R'' に至る Pathway が存在することを表す.

6.2 推論規則の詳細化

詳細化された推論規則を挙げる. これらの推論規則の集合を今後 RulesR' と呼ぶことにする. 但し, 次に出てくるパラメタは全てメタ変数である.

1' $$\frac{React(R, x, y, w)}{React(R, y, x, w)}$$

2' $$\frac{React(R_1, x, y, w_1) \quad React(R_2, y, z, w_2)}{Inh(R_1, y, R_2, y)}$$

3' $$\frac{React(R_1, x, y, w_1) \quad React(R_2, w_1, z, w_2)}{Prom(R_1, w_1, R_2, w_1)}$$

4' $$\frac{Inh(R, x, R', y) \quad Inh(R', u, R'', v)}{Prom(R, x, R'', v)}\ y \neq u$$

5' $$\frac{Inh(R, x, R', y) \quad Prom(R', u, R'', v)}{Inh(R, x, R'', v)}\ y \neq u$$

6' $$\frac{Prom(R, x, R', y) \quad Inh(R', u, R'', v)}{Inh(R, x, R'', v)}\ y \neq u$$

7' $$\frac{Prom(R, x, R', y) \quad Prom(R', u, R'', v)}{Prom(R, x, R'', v)}\ y \neq u$$

6.2.1 意味

ここでは, 言語 L' の意味を線形論理の導出に関する性質として与える.

定義 7 $\mathcal{I}_\Delta : L' \rightarrow$ ($\mathcal{L}$ をオブジェクトドメインとして含む一階述語メタ論理式) を次のように定義する. 動的モデルの意味にアンダーラインをした部分を追加する.

1. $\mathcal{I}_\Delta(Inh(R, \phi_1, R', \phi_2)) \stackrel{def}{=}$
   $\exists Inputs \exists Reacts \exists Initial \exists Seq \exists y \exists z \exists Con($
   $PW(Inputs, Reacts, Initial, Seq, y, z, Con, \Delta) \wedge$
   $head(Seq) = R \wedge last(Seq) = R' \wedge$
   $\underline{Initial = Inset(R) - \{\phi_1\}} \wedge$
   $\underline{\{y\} = Inset(R') - \{\phi_2\}} \wedge$

   $\forall m \forall P_1 ... \forall P_m($
   $(\ Inputs \cup Reacts \vdash P_1 \otimes P_2 \otimes ... \otimes P_m \wedge$
   $\ Initial \cap Oset(P_1 \otimes P_2 \otimes ... \otimes P_m) = \phi \wedge$
   $\ Con \cap Oset(P_1 \otimes P_2 \otimes ... \otimes P_m) = \phi \wedge$
   $\Rightarrow (\ (\ y \in Oset(P_1 \otimes P_2 \otimes ... \otimes P_m)))$

   ここで, $Inset(R)$ は, $x(i, R) \otimes y(i, R) \multimap w(o, R)$ に対して, $Inset(R) \stackrel{def}{=} \{x, y\}$ と定義する.

2. また, 同様に, $\mathcal{I}_\Delta(Prom(R, x, R', y)) \stackrel{def}{=}$
   ...
   ...(上記前半部分と同様)

   $\Rightarrow (\ (\ y \notin Oset(P_1 \otimes P_2 \otimes ... \otimes P_m)))$ End

7 健全性

定理 2 Γ を一階述語論理の言語 L の論理式の集合とし, $RulesR'$ を推論規則の集合とする. この時, 次が成り立つ.
(健全性) $\Gamma \vdash_{RulesR'} \varphi \Longrightarrow \forall \Delta(\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(\varphi))$
End

(証明)

φ の証明ステップに関する帰納法により証明する. ベースケースは明らかであり, 帰納ステップはほぼ同様なので代表として規則 4' に関する帰納ステップを証明する.

帰納法の仮定として,

$$\Gamma \vdash_{RulesR'} Inh(R_1, \phi_1, R_2, \phi_2')$$
$$\Longrightarrow \forall \Delta(\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(Inh(R_1, \phi_1, R_2, \phi_2'))) \quad (1)$$
$$\Gamma \vdash_{RulesR'} Inh(R_2, \phi_2, R_3, \phi_3')$$
$$\Longrightarrow \forall \Delta(\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(Inh(R_2, \phi_2, R_3, \phi_3'))) \quad (2)$$

(但し, $\phi_2 \neq \phi_2'$ とする)

が成り立つとし,

$\Gamma \vdash_{RulesR'} Prom(R_1, \phi_1, R_3, \phi_3') \Longrightarrow \forall \Delta(\mathcal{I}_\Delta(\Gamma) \models \mathcal{I}_\Delta(Prom(R_1, \phi_1, R_3, \phi_3'))$

が成り立つことを示す.

$\Gamma \vdash_{RulesR'} Prom(R_1, \phi_1, R_3, \phi_3')$

であり, この証明の最後のステップで, 規則 4' が用いられたとすると,(1), (2) の前件が成立し, $\mathcal{I}_\Delta(\Gamma)$ を仮定すると, 以下の (1')(2') が成立する.

$$\mathcal{I}_\Delta(Inh(R_1, \phi_1, R_2, \phi_2')) \quad (1')$$
$$\mathcal{I}_\Delta(Inh(R_2, \phi_2, R_3, \phi_3')) \quad (2')$$

このことより, (1')(2') に関する Pathway が存在する.

(1') に関する Pathway を

$PW(Inputs_1, Reacts_1, Initial_1, Seq_1, y_1, z_1, Con_1, \Delta) \wedge$
$head(Seq_1) = R_1 \wedge last(Seq_1) = R_2 \wedge$
$Initial_1 = Inset(R_1) - \phi_1 \wedge$
$\{y_1\} = Inset(R_2) - \phi_2'$

が成立するものとし,

(2') の Pathway を

$PW(Inputs_2, Reacts_2, Initial_2, Seq_2, y_2, z_2, Con_2, \Delta)$
$head(Seq_2) = R_2 \wedge last(Seq_2) = R_3 \wedge$
$Initial_2 = Inset(R_2) - \phi_2 \wedge$
$\{y_2\} = Inset(R_3) - \phi_3'$

が成立するものとする.

$PW(Inputs_1, Reacts_1, Initial_1, Seq_1, y_1, z_1, Con_1, \Delta)$ において最後に結合されている部分は, ϕ_2' であり, $PW(Inputs_2, Reacts_2, Initial_2, Seq_2, y_2, z_2, Con_2, \Delta)$ において最初に結合されている部分は, ϕ_2 である. これらは異なるので, 以下で証明する補題 8 より,

$head(Seq_3) = R_1 \wedge last(Seq_3) = R_3 \wedge$
$Initial_3 = Inset(R_1) - \phi_1 \wedge$
$\{y_3\} = Inset(R_3) - \phi_3' \wedge$
$Inputs_3 = Inputs_1 \cup Inputs_2 - \{\forall \phi_2'(i, r)\} \wedge$
$Reacts_3 = Reacts_1 \cup Reacts_2 \wedge$
$Con_3 = Con_1 \cup Con_2 \wedge$
$Initial_3 = Initial_1$

を満たす Pathway

$PW(Inputs_3, Reacts_3, Initial_3, Seq_3, y_3, z_3, Con_3, \Delta)$

が存在する. さて, この Pathway が,

$(\ Inputs_3 \cup Reacts_3 \vdash P_1 \otimes P_2 \otimes ... \otimes P_m \wedge$
$\ Initial_3) \cap Oset(P_1 \otimes P_2 \otimes ... \otimes P_m) = \phi \wedge$
$\ Con_3 \cap Oset(P_1 \otimes P_2 \otimes ... \otimes P_m) = \phi$ (3)

を満たす時, 次の二つの場合においてそれぞれ以下が成立する.

1. $y_1 \in Con_2$ の時 (1') より,

   $Inputs_1 \cup Reacts_1 \vdash P_1 \otimes P_2 \otimes ... \otimes P_n \otimes y_1(i, R_2) \wedge$
   $Initial_1 \cap Oset(P_1 \otimes P_2 \otimes ... \otimes P_n \otimes y_1(i, R_2)) = \phi \wedge$
   $Con_1 \cap Oset(P_1 \otimes P_2 \otimes ... \otimes P_n \otimes y_1(i, R_2)) = \phi$
   であり,
   $Inputs_2 \cup Reacts_2 \vdash \phi'_2(i, R_2) \otimes P_{n+1} \otimes P_{n+2} \otimes ... \otimes P_m \wedge$
   $Con_2 \cap Oset(\phi'_2(i, R_2) \otimes P_{n+1} \otimes P_{n+2} \otimes ... \otimes P_m) = \phi \wedge \ \phi'_2 \in Initial_2$
   なる $n (2 \leq n \leq m-1)$ が存在する.
   $Initial_2 = \{\phi'_2\}$ であり, この時, $Initial_2 \in Oset(\phi'_2(i, R_2) \otimes P_{n+1} \otimes P_{n+2} \otimes ... \otimes P_m)$

   であるから,(2) および補題 9. より,

   $y_2 \notin Oset(\phi'_2(i, R_2) \otimes P_{n+1} \otimes P_{n+2} \otimes ... \otimes P_m)$

   である. また, y_2 は $React_2$ 中の反応のみに関連するものであるから

   $y_2 \notin Oset(P_1 \otimes P_2 \otimes \quad ... \otimes P_n \otimes y_1(i, R_2))$ である.

   従って, $y_2 \notin Oset(P_1 \otimes P_2 \otimes \quad ... \otimes P_m)$ である.

2. $y_1 \notin Con_2$ の時, 同様にして,

   $Inputs_1 \cup Reacts_1 \vdash P_1 \otimes P_2 \otimes ... \otimes P_n \wedge$
   $Initial_1 \cap Oset(P_1 \otimes P_2 \otimes ... \otimes P_n) = \phi \wedge$
   $Con_1 \cap Oset(P_1 \otimes P_2 \otimes ... \otimes P_n) = \phi$ であり,
   $Inputs_2 \cup Reacts_2 \vdash \phi'_2(i, R_2) \otimes P_n \otimes P_{n+1} \otimes ... \otimes P_m \wedge$
   $Con_2 \cap Oset(\phi'_2(i, R_2) \otimes P_n \otimes P_{n+1} \otimes ... \otimes P_m) = \phi \wedge \ P_n = y_1(i, R_2)$
   なる $n (2 \leq n \leq m-1)$ が存在する.
   この時, $Initial_2 = \{\phi'_2, y_1\}$ であり, $Initial_2 \in Oset(\phi'_2(i, R_2) \otimes P_n \otimes P_{n+1} \otimes ... \otimes P_m)$

   であるから,1. の場合と同様に

   $y_2 \notin Oset(P_1 \otimes P_2 \otimes \quad ... \otimes P_m)$ が成立する. End

補題 8 反応の列 $R_1 \cdot R_2 \cdot ... \cdot R_n$ (但し, $R_i \neq R_{i+1}$) が Pathway であるための必要十分条件は,

1. $Inset(R_i) \cap Inset(R_{i+1}) \neq \phi$ 又は,
   $outset(R_i) \cap Inset(R_{i+1}) \neq \phi$
2. R_i の input 又は output と共有されている R_{i+1} のオブジェクトは R_{i+1} と R_{i+2} の共有オブジェクトではない.

ことである.
（証明）Pathway の構成に関する帰納法による. End

補題 9 $PW(Inputs, Reacts, Initial, Seq, y, z, Con, \Delta) \wedge head(Seq) = R \wedge last(Seq) = R'$
なる Pathway Seq に関して, 阻害と活性の意味について次が成り立つ.

$$
\begin{aligned}
&\forall m \forall P_1 ... \forall P_m (\\
&(\ Inputs \cup Reacts \vdash P_1 \otimes ... \otimes P_m \wedge \\
&\ \ Initial \cap Oset(P_1 \otimes ... \otimes P_m) = \phi \wedge \\
&\ \ Con \cap Oset(P_1 \otimes ... \otimes P_m) = \phi) \\
&\qquad\qquad \Rightarrow y \notin Oset(P_1 \otimes ... \otimes P_m)) \\
&\Rightarrow \\
&\forall m \forall P_1 ... \forall P_m (\\
&(\ Inputs \cup Reacts \vdash P_1 \otimes ... \otimes P_m \wedge \\
&\ \ Initial \cap Oset(P_1 \otimes ... \otimes P_m) \neq \phi \wedge \\
&\ \ Con \cap Oset(P_1 \otimes ... \otimes P_m) = \phi) \\
&\qquad\qquad \Rightarrow y \in Oset(P_1 \otimes ... \otimes P_m))
\end{aligned} \tag{1}
$$

$$
\begin{aligned}
&\forall m \forall P_1 ... \forall P_m (\\
&(\ Inputs \cup Reacts \vdash P_1 \otimes ... \otimes P_m \wedge \\
&\ \ Initial \cap Oset(P_1 \otimes ... \otimes P_m) = \phi \wedge \\
&\ \ Con \cap Oset(P_1 \otimes ... \otimes P_m) = \phi) \\
&\qquad\qquad \Rightarrow y \in Oset(P_1 \otimes ... \otimes P_m)) \\
&\Rightarrow \\
&\forall m \forall P_1 ... \forall P_m (\\
&(\ Inputs \cup Reacts \vdash P_1 \otimes ... \otimes P_m \wedge \\
&\ \ Initial \cap Oset(P_1 \otimes ... \otimes P_m) \neq \phi \wedge \\
&\ \ Con \cap Oset(P_1 \otimes ... \otimes P_m) = \phi) \\
&\qquad\qquad \Rightarrow y \notin Oset(P_1 \otimes ... \otimes P_m))
\end{aligned} \tag{2}
$$

(証明) Pathway の構成に関する帰納法による.

8 薬の飲みあわせの危険性の判定に関する応用

生体内の反応は多様で複雑であるが, ここでは既知の反応や阻害や活性に関する情報を用いて, 薬害の可能性を推論する方法について述べると共に, 実際の例に対する推論を示す.

8.1 薬物AとBの摂取が危険であるということの形式定義

$R1$: 生体内で生体の維持に必須の反応
$R2$: 薬物Aを原材料または, 触媒として用いる反応
$R3$: 薬物Bを原材料または, 触媒として用いる反応

とした時, 以下の条件が成立する時, 薬物AとBの両方を摂取することは, 生命にとって危険である可能性がある.

ある x,y,z,w について,
$React(R2, A, x, y), React(R3, B, z, w)$
とするとき,
$Inh(R2, R1), Prom(R3, R2)$...(1)
より精密には,
$\phi_4 \neq \phi_1$ である, ある $\phi_1, \phi_2, \phi_3, \phi 4$ について
$Inh(R2, \phi_1, R1, \phi_2)$, $Prom(R3, \phi_3, R2, \phi_4)$...(1')
あるいは,
$Inh(R2, R1), Inh(R3, R1)$...(2)
より精密には,
ある $\phi_1, \phi_2, \phi_3, \phi 4$ について
$Inh(R2, \phi_1, R1, \phi_2)$, $Inh(R3, \phi_3, R1, \phi_4)$...(2')

性質 (1)((1')) から条件 (2)((2')) は, 推論規則　を用いて常に推論可能であるが, 逆は真ではない. 条件 (2) の $Inh(R2, R1)$ の Pathway は $Inh(R3, R1)$ の Pathway は, 必ずしも重複している必要は無いが, 条件 (1) の場合は, 必ず, 部分的に重複した Pathway が存在する. ((1'), (2') についても同様.) このような場合の方が, 一つの薬剤単独ではその効果が見えないために, 見落としが多く危険な場合が多い. 冒頭に述べた例はそのような実例である.

8.2 実例についての証明

以下性質 (1) を, 冒頭の例について証明する. まず, 既知の事実として, 以下を仮定する.

$$\begin{array}{l}
React(R_1, SRV, TP, BVU)\\
React(R_2, BVU, DPD, DPDBVU)\\
React(R_3, DPD, 5FU, H2FUras)\\
React(R_4, 5FU, OPRTase, FDUP)\\
React(R_5, FDUP, RR, FdUMP)\\
React(R_6, FdUMP, ts, miCTC)\\
React(R_7, ts, dUMP, dTMP)\\
React(R_8, dTMP, *, DNA)
\end{array}$$

これらを用いて条件 (1) を導く証明図を付録 B に示す.

9 あとがき

阻害と活性という, 非常に汎用的な動作間の関係を, 形式オントロジーとして定義し, その完全性, 健全性について論じた.

この形式オントロジーでは, その意味の基礎を, あるイベントが持つその動作を行うために必要な前件と, その結果得られる後件とを考え, その条件の取り合いや供給関係に置いている. 動作の影響を, これらの関係を基本として, 量的な概念を持ち込むこと無しに抽象化し, 阻害, 活性概念の意味を与えた. この意味モデルは, 経済の原理や社会現象, 例えば, 「公定歩合の利上げは, 株価を抑制する. 」あるいは, 「高学歴化は, 出生率を抑制する. 」などの意味を解釈するのにも共通の枠組みとして用いることができる.

これはちょうど, CCS や CSP などのプロセス代数で, 並行処理の意味を, 厳密な量的時間関係や, 同時刻性などという概念持ち込むこと無しに, 単純に部分プロセスのあらゆる可能なシャッフリングによる全順序化を可能なモデルとして許すことを基に定義するときに考えられた抽象化に近い考え方である.

ここでは, 線形論理により資源に関連した性質が簡便に記述出来るため, その証明を動作のモデルとし, その持つ性質として, 概念の意味を定義することが出来た. もちろん, 資源に関連した性質が表現出来れば良いから, ペトリネットを用いても同じような意味論が展開出来るであろう. このような動的な意味に関連した形式オントロジーの研究は今後重要なオントロジーの研究分野となると考えている.

この研究では, この意味モデルを用いて, この推論規則の健全性, 完全性について論じた. 最初にあげた推論規則の集合は, 完全ではあるが, 必ずしも健全では無い場合がある. しかし薬学の分野では, 良く用いられる規則である. 医学や薬学の場合, 少しでも可能性のある新しい知識の発見は貴重である. もし反応の間に阻害や活性関係の可能性があるのなら, その本当の真偽は実験により確認すればよいからである. そしてもし実験によりそれが正しいことが示されれば, 実際的に大きな意味を持つ.

本論文では言語拡張を行うことにより, より厳密な推論が可能な体系についても示し, この健全性を証明した.

本研究が, 形式オントロジー研究の新しい展開に資すると共に, 生物情報学への情報科学からの本質的貢献となるよう, 理論, 応用両面について発展させて行きたい.

謝辞: 薬物相互作用について手ほどきを頂いた, 理化学研究所の小長谷明彦教授, 吉川澄美博士に感謝致します．尚, 本研究は 21 世紀 COE プログラム「大規模知識資源の体系化と活用基盤構築」のサポートを受けています．

参考文献

[1] Sumi Yoshikawa, Kenji Satou and Akihiko Konagaya, Drug Interaction Ontology (DIO) for Inferences of Possible Drug-drug Interactions, The proceeding of Info Med 2004, 2004.

[2] C. Lejewski, Zu Lesniewskis Ontologie, Ratio 1, 1958.

[3] B. Sobocinski, Successive Simplifications of the Axioms Systems ofLesniewski's Ontology, Plosh Logic 1920-1939, Oxford ,1967.

[4] David Lewis, Parts of Classes, Blackwell, ISBN 0-631-17655-1, 1991.

[5] 渡部烈 小倉健一郎 西山貴仁, ソリブジン薬害発生の分子毒性学的メカニズムとジヒドロピリミジン・デヒドロゲナーゼの遺伝的欠損, YAKUGAKU ZASSHI 122 (8) pp.527-535 (2002) ©The Pharmaceutical Society of Japan, 2002.

[6] Jean-Yves Girard, Yves Lafont, Paul Taylor, Proofs and Types, Cambridge University Press

付録

A 拡張された形式化

薬学の分野では「阻害する」「活性化する」という言葉を, 薬物間の用語としても使い, 反応間の用語としても使い, また, 薬物と反応の間の関係にも使う．例えば,「ソリブジン (SRV) は制癌剤 (5-FU) を活性化する.」というときは, 薬物間の関係として使われている．「5-FU は DNA 合成を阻害する」といったときは, 薬物と反応の間の関係に使われている．「BVU は DPD と反応して DPDBVU を生成することにより, DPD の 5-FU 解毒作用の活性を低下させる」といったときは, 物質・物質の関係であるがこれに関与する反応が明示されている．

本節では, 3 節で定義した言語 L およびその推論規則を拡張し, 薬物間の阻害, 活性関係を表現するための形式化を行う．反応と薬物の間の推論規則についてこれらから類推することは容易である．

言語 L の拡張

定義 8 言語 L に対して, 原始式を以下のように拡張する:

$$Inh(R,R'), Prom(R,R'), Inh_{\alpha,\beta}(x,y),$$
$$Prom_{\alpha,\beta}(x,y), React(R,x,y,z)$$

但し, $\alpha,\beta \in \{i,o\}$ とする．

End

追加された原始式に対する直感的な意味を以下のように与える．ただし, α や β が i であるとき, その物質は反応の入力物質であることを, o であるとき, その物質は反応の出力物質であることを意味している．

$Inh_{\alpha,\beta}(x,y)$	物質 x が物質 y を阻害する．
$Prom_{\alpha,\beta}(x,y)$	物質 x が物質 y を活性化する．

たとえば, $\alpha=i, \beta=i$ であるとき $Inh_{i,i}(x,y)$ は, ある反応の入力物質 x が, ある他の反応の入力物質 y を阻害することを意味している．$Inh_{o,i}(x,y)$ は, ある反応の出力物質 x が, ある他の反応の入力物質 y を阻害することを意味している．

推論規則 Rules の拡張

拡張された言語 L に関する推論規則を定義する．3 節で定義した RulesR に加えて, 物質間の阻害, 活性に関する以下の推論規則を導入する．

ただし以下では, $A,B \Rightarrow C,D$ と書いて推論規則 $\dfrac{A \quad B}{C}$ と $\dfrac{A \quad B}{D}$ を表すことにする．

1. $React(R_1,x,y,w_1), React(R_2,y,z,w_2) \Rightarrow$
   $Inh(R_1,R_2), Inh_{ii}(x,z), Inh_{oi}(w_1,z),$
   $Inh_{io}(x,w_2), Inh_{oo}(w_1,w_2)$

2. $React(R_1,x,y,w_1), React(R_2,w_1,z,w_2) \Rightarrow$
   $Prom_{ii}(x,z), Prom_{io}(x,w_2)$

3. $Inh_{ii}(y,z), React(R,x,y,w) \Rightarrow$
   $Prom_{ii}(x,z), Prom_{oi}(w,z)$

4. $Inh_{io}(y,w_2), React(R,x,y,w_1) \Rightarrow$
   $Prom_{io}(x,w_2), Prom_{oo}(w_1,w_2)$

5. $Inh_{ii}(y,z), React(R,x,y_1,y) \Rightarrow Inh_{ii}(x,z)$

6. $Inh_{io}(y,w_2), React(R,x,y_1,y) \Rightarrow Inh_{io}(x,w_2)$

7. $Inh_{io}(y,w_2), React(R,w_2,x,w_3) \Rightarrow$
   $Inh_{ii}(y,x), Inh_{io}(y,w_3)$

8. $Inh_{oo}(w_1, w_2), React(R, w_2, x, w_3) \Rightarrow$
   $Inh_{oi}(w_1, x), Inh_{oo}(w_1, w_3)$

9. $Inh_{ii}(y, z), React(R, z, x, w_3) \Rightarrow$
   $Prom_{ii}(y, x), Prom_{io}(y, w_3)$

10. $Inh_{oi}(w_1, z), React(R, z, x, w_3) \Rightarrow$
   $Prom_{oi}(w_1, x), Prom_{oo}(w_1, w_3)$

---

11. $Prom_{ii}(y, z), React(R, x, y, w) \Rightarrow$
   $Inh_{ii}(x, z), Inh_{oi}(w, z)$

12. $Prom_{io}(y, w_2), React(R, x, y, w_1) \Rightarrow$
   $Inh_{io}(x, w_2), Inh_{oo}(w_1, w_2)$

13. $Prom_{ii}(y, z), React(R, x, y_1, y) \Rightarrow Prom_{ii}(x, z)$

14. $Prom_{io}(y, w_2), React(R, x, y_1, y) \Rightarrow$
   $Prom_{io}(x, w_2)$

15. $Prom_{io}(y, w_2), React(R, w_2, x, w_3) \Rightarrow$
   $Prom_{ii}(y, x), Prom_{io}(y, w_3)$

16. $Prom_{oo}(w_1, w_2), React(R, w_2, x, w_3) \Rightarrow$
   $Prom_{oi}(w_1, x), Prom_{oo}(w_1, w_3)$

17. $Prom_{ii}(y, z), React(R, z, x, w_3) \Rightarrow$
   $Inh_{ii}(y, x), Inh_{io}(y, w_3)$

18. $Prom_{oi}(w_1, z), React(R, z, x, w_3) \Rightarrow$
   $Inh_{oi}(w_1, x), Inh_{oo}(w_1, w_3)$

---

19. $x_1 \neq x_2$ の時,

   $Inh(R_1, R_2), React(R_1, x_1, y_1, z_1), React(R_2, x_2, y_2, z_2)$
   $\Rightarrow$
   $Inh_{ii}(x_1, x_2), Inh_{oo}(z_1, z_2),$
   $Inh_{io}(x_1, z_2), Inh_{oi}(z_1, x_2)$

20. $x_1 \neq x_2$ の時,

   $Inh(R_1, R_2), React(R_1, x_1, y_1, z_1), React(R_2, x_2, y_2, z_2)$
   $\Rightarrow$
   $Inh_{ii}(x_1, x_2), Inh_{oo}(z_1, z_2),$
   $Inh_{io}(x_1, z_2), Inh_{oi}(z_1, x_2)$

---

21. $Inh_{\alpha i}(x, y), Prom_{i\beta}(y, w) \Rightarrow Prom_{\alpha\beta}(x, w)$

22. $Inh_{\alpha i}(x, y), Inh_{i\beta}(y, w) \Rightarrow Inh_{\alpha\beta}(x, w)$

23. $Prom_{\alpha i}(x, y), Prom_{i\beta}(y, w) \Rightarrow Inh_{\alpha\beta}(x, w)$

24. $Prom_{\alpha i}(x, y), Inh_{i\beta}(y, w) \Rightarrow Prom_{\alpha\beta}(x, w)$

B 証明図

$$\frac{React(R_1, SRV, TP, BVU) \quad React(R_2, BVU, DPD, DPDBVU)}{Prom(R_1, R_2)}$$

$$\frac{React(R_2, BVU, DPD, DPDBVU) \quad React(R_3, DPD, 5FU, H2FUras)}{Inh(R_2, R_3)}$$

$$\frac{React(R_3, DPD, 5FU, H2FUras) \quad React(R_4, 5FU, OPRTase)}{Inh(R_3, R_4)}$$

$$\frac{React(R_4, 5FU, OPRTase) \quad React(R_5, FDUP, RR, FdUMP)}{Prom(R_4, R_5)}$$

$$\frac{React(R_5, FDUP, RR, FdUMP) \quad React(R_6, FdUMP, ts, miCTC)}{Prom(R_5, R_6)}$$

$$\frac{React(R_6, FdUMP, ts, miCTC) \quad React(R_7, ts, dUMP, dTMP)}{Inh(R_6, R_7)}$$

$$\frac{React(R_7, ts, dUMP, dTMP) \quad React(R_8, dTMP, *, DNA)}{Prom(R_7, R_8)}$$

$$\frac{\dfrac{Prom(R_1, R_2) \quad Inh(R_2, R_3)}{Inh(R_1, R_3)} \quad Inh(R_3, R_4)}{Prom(R_1, R_4)}$$

$$\frac{\dfrac{Prom(R_4, R_5) \quad Prom(R_5, R_6)}{Prom(R_4, R_6)} \quad \dfrac{Inh(R_6, R_7) \quad Prom(R_7, R_8)}{Inh(R_6, R_8)}}{Inh(R_4, R_8)}$$