TR-630

直観理論に基づく並列プロセス生成、検証のための体系 μ

川田　秀司、佐藤　洋祐、藤田　正幸

March, 1991

ICOT

Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
Minato-ku Tokyo 108 Japan

(03)3456-3191～5
Telex ICOT J32964

Institute for New Generation Computer Technology

直観論理に基づく並列プロセス生成，検証のための体系 μ

μ : A New Logic for Process Programming and Verification by Constructive Logics

川田秀司　佐藤洋祐　藤田正幸

Hideji Kawata　Yosuke Sato　Masayuki Fujita

(財) 新世代コンピュータ技術開発機構

Institute for New Generation Computer Technology

Abstract: There have been some researches[1,2,3] on program extraction from a proof on intuitionistic logic. Their programs, however, are merely defined as the λ term. If we want to consider programs as the processes by which computer communicates with the outside world, the the logical and the calculating powers of their methods are not enough. They have no expressions and operations relate to input and output. In order to deal with such programs, this paper introduces a new logic, μ. This logic can be used to synthesize communicating prosesses[6,7].

1 はじめに

プログラムを外界と通信を行うプロセスとして考えるとき，[1,2,3,4] によるプログラム生成方法で表現することは難しい．これは，この方法での論理式による仕様の表現に通信の概念がないなどの理由による．本稿では，[6,7] などで示される様な並列プロセス生成を目的として，新しい体系 μ を提案し，さらに実現可能性解釈（realizability interpretation）に対してのみ意味を持つ（論理上は区別されない）様な，補助記号と呼ばれる記号（論理記号に付加して使用する）を定義する．これらにより，プロセスの動作を記述が可能となる．また，補助記号を含んだ論理式に対し，Program Term という言語への実現可能性解釈と，証明図から実際に実現可能性解釈を満たす Program Term を生成するための規則（realizer extraction rule）を定義する．

ここで本稿で使用される記号の意味を定めておこう．

- $a_1..a_n$, $a_1...a_n$
  例えば，$n=5$ とすると，どちらも以下を意味する．

$$a_1a_2a_3a_4a_5$$

- $a_1,...a_n$, $a_1,....a_n$
  例えば，$n=5$ とすると，どちらも以下を意味する．

$$a_1,a_2,a_3,a_4,a_5$$

- $[a/x]F$, $[a_1/x_1,...,a_n/x_n]F$
  前者は，F に現れる x を全て a で置き換えたもの．後者は例えば $n=3$ とすると，F に現れる x_1 を a_1 で置き換え，その後，x_2 を a_2 で置き換え，更に，x_3 を a_3 で置き換えたもの．

- $\{1,2,3\}$
  要素が，3 つの要素，1, 2, 3 を元とする集合．

さて， R.Milner の CCS[6] を例に，プロセスの基本的動作を上げてみると以下のようになる．

1. $\alpha(x).P(x)$ … ポート α から x を入力し，その後 $P(x)$ が示す動作を行う．
2. $\bar{\alpha}(a).P$ … ポート α に a を出力しその後 P が示す動作を行う．
3. $P_1|P_2$ … P_1 が示す動作と P_2 が示す動作を並列に行う．
4. $P_1 + P_2$ … P_1 と P_2 のどちらか一方を非決定的に選択しその動作を行う．

これらの動作を論理式で表現するわけだが，ポイントはどこまでの情報が証明から得られる必要があるかということであろう．従来の方法での $F \wedge G$ の仕様としての意味を考えてみると，これは，

- F の仕様としての意味を満足する λ 式 と G の仕様としての意味を満足する λ 式の対．

である．これは A の示すプログラムと B の示すプログラムの両方を実行できる情報を持っているので，

- F の仕様としての意味を満足するプロセスの動作を行い，その後， G の仕様としての意味を満足するプロセスの動作を行うプロセス．
- F の仕様としての意味を満足するプロセスの動作と， G の仕様としての意味を満足するプロセスの動作を並列に行うプロセス．
- F と G のどちらかを非決定的に選択しその仕様としての意味を満足する動作を行うプロセス．

のどれと考えてもよい．そこで上記の論理上は $\wedge$ と区別されない補助記号を導入し，それぞれ， $F \wedge G$ （簡単のため，もとの記号を使用する）， $F \wedge_p G$， $F \wedge_n G$ で表現すればよいのである．
また，従来の方法での $\forall x F(x)$ の仕様としての意味は，

- 任意の t に対し， t を適用した結果が $F(t)$ の仕様を満たすものとなる λ 式

であるから，

- 実行過程が，「入力を行い，それを t とすると，その後，論理式 $F(t)$ が表現する仕様を満たすプログラムの実行したときと同様の動作を行う」ようなプログラム．

とすることも可能である．この直観的理由は，証明からのプログラム抽出時に $\lambda_x.f(x)$ （$f(t)$ は $F(t)$ の仕様としての意味を満足するプログラム）とする所を例えば，$(in, \lambda_x.f(x))$ とすれば良いからである（勿論，プログラム $(in, \lambda_x.f(x))$ の動作はあらかじめ定めておく）．同様に， $\exists x F(x)$ の仕様としての意味を，

- 実行過程が，「$F(t)$ を真とする t を出力し，その後，論理式 $F(t)$ の仕様としての意味を満足するプログラムの動作と同じ動作を行う」ようなプログラム．

とすることで，取り敢えず入出力は表現できる．問題は 1， 2 のポート名だが，これも論理で扱う必要はない．例えば 1 の仕様を $\forall_\alpha x F(x)$ （$F(x)$ は $P(x)$ の仕様を表すとする， α はポート名）で表現し，論理上は $\forall x F(x)$ と同一視して証明を行い，その証明からプログラムを抽出の際 $(in(\alpha).\lambda_x.f(x))$ （$f(x)$ は $F(x)$ の仕様を満足するプログラム）といったようにすることで十分であるわけである．
さて，色々な問題を論理式記述してみると，仕様としての $\forall$ には 2 つのレベルがあることが分かる．つまり，満たさなければならない条件のみを示す $\forall$ と，条件と同時に入力を示す $\forall$ があるのである．これと類似の問題は従来の方法においても議論されている [4,5]．ここでも補助記号を導入し，前者を $\dot{\forall}$，後者を $\forall$ と表記しよう． $\exists$ についても同様に，満たさなければならない条件を表すものを $\exists$，それと同時に出力を示すものを $\dot{\exists}$ としよう．

さて，上のように仕様を記述するとしたとき， 2 つの問題が生じる． 1 つは，入出力の回数が実行時に決まるようなプログラムの記述に関する問題である．つまり論理式の長さは有限であるから，そこに現れることのできる $\forall$， $\exists$ も有限である．よって対象とするプログラムの入出力の回数が最大値を持たないようなものは表現できないのである．
もう 1 つは，プログラムの停止性の問題である．従来の関数としてのプログラムを生成する方法では，作成されたプログラムの停止性が保証されていたが，これに対しプロセスとしてのプログラムでは何を保証すべきかということである．これらの問題を解決するために作成した論理体系が μ である．体系 μ とその実行可能性解釈は，以下の 2 つを満たす．

- 論理式の再帰表現ができる．

- 証明図より生成されたプログラムは良停止性を満たす.

ここで，プロセスとしてのプログラムに対する停止性と良停止性だが，その厳密な定義は次章以降で示すこととし，ここでは直観的意味を述べる．プログラムが停止性を満たすとは，そのプログラムを実行したとき，どの様な入力を行っても停止することとしよう．しかし，「外界とコミニケーションを行うプロセス」としてのプログラムは一般にこの性質を満たすべきであろうか．例えば，UNIX のコマンドシェルは "exit" を入力するまでは停止しない．つまり，上記の停止性を満たさない．よって，この様なプログラムを生成したいならば，この性質は強すぎることになる．そこで，停止性より弱い概念として良停止性を導入する．プログラムが良停止性を満たすということの直観的な意味は，どの様な入力を行っても，その後うまく入力を行えば，そのプログラムは停止するとするのである．
さて，論理式の再帰表現であるが，体系 μ では，次の様な表記法を採用する．

$$(\mu_{\Psi_{(n)}x_1..x_n}.F(x_1,...,x_n,\Psi_{(n)}a_{11}(x_1,...,x_n)..a_{1n}(x_1,...,x_n),...,\Psi_{(n)}a_{m1}(x_1,...x_n)..a_{mn}(x_1,..,x_n)))b_1..b_n$$

（以降簡単のため，$a_{ij}(x_1,..,x_n)$ は a_{ij} と表記する．）
これは，以下を満たす様な P を表す．

$$P(b_1,..,b_n) \Leftrightarrow F(b_1,..,b_n,P(a_{11}(b_1,...b_n),..,a_{1n}(b_1,...b_n)),..,P(a_{m1}(b_1,..,b_n),...a_{mn}(b_1,...,b_n)))$$

2 体系 μ

U をある集合とし，U^n 上には well-founded order の集合 I_n が定義されているとする．また，U_f を U 上で定義されている部分関数の集合とする．この U, U_f に対し，以下のように体系 μ を定める．

2.1 記号

- 個体定数 …… $c, d, e, ...(\epsilon U)$
- 述語定数 …… $=$，$<$，n 引数述語定数を $o_{(n)}, p_{(n)}, \cdots$
- 関数 ………… n 引数関数を $f_n, g_n, ...(\epsilon U_f)$
- 個体変数 …… $s, t, x, y, z, ...$
- 述語変数 …… n 引数述語変数を $\Psi_{(n)}, \Phi_{(n)}, \cdots$
- 論理演算子 … $\wedge, \vee, \Rightarrow, \forall, \exists, \neg, \mu, \natural$

ここで，$\natural$ であるが，

$$\natural x((x_1,...,x_n).F.\Psi_{(n)}.\mu_{\Psi_{(n)}y_1..y_n}.G.H(x)) \Leftrightarrow (\forall x_1,...,x_n(F \Rightarrow \Psi_{(n)}x_1..x_n) \Rightarrow \forall x[\Psi/\mu_{\Psi_{(n)}y_1..y_n}.G]H(x)) \wedge \exists x.H(x)$$

である．よって，論理を定義する上ではこの様な論理演算子を定義する必要はないが，後に定義する realizer extraction rule の都合上新たに導入する．

2.2 定義

上記の記号に対し，predicate, l-formula, formula，式を以下のように定義する．
体系 μ での論理式は formula を指し，l-formula（formula との違いは $\natural$ を含むかどうかのみ）は，は推論規則の上のみで扱われるものである．ここで，$a_1,...a_n$ は個体定数または個体変数とする．

1. object …… $a ::= x|c|f_n(a_1,..,a_n)$
2. predicate … $P_{(n)} ::= p_{(n)}|\Psi_{(n)}$
3. l-formula … $F ::= P_{(n)}a_1..a_n|\Psi_{(n)}a_1..a_n|\ell|F_1 \wedge F_2|F_1 \vee F_2|F_1 \Rightarrow F_2|\neg F|\forall xF|\exists xF|a\downarrow$
   $\natural x((x_1,..,x_n).G.\Psi_{(n)}.\mu_{\Psi_{(n)}y_1..y_n}.H.F)|(\mu_{\Psi_{(n)}x_1..x_n}.I)a_1..a_n$
   （ただし，I に現れる自由述語変数は $\Psi_{(n)}$ のみで $\Psi_{(n)}$ は，positive part（definition 2.1参照）のみに現れる．）
4. formula …… l-formula で，$\natural$，自由述語変数を含まないもの．

- $\forall xF(x), \exists xF(x)$，$(\mu_{\Psi_{(n)}x_1..x_n}.A)a_1..a_n$，$\natural x((x_1,..,x_n).G.\Phi_{(n)}.\mu_{\Phi_{(n)}y_1..y_n}.G'.F)$ で，A,F に現れる個体変数 x，$x_1,...,x_n$，述語変数 Ψ はそれぞれ束縛個体変数，束縛述語変数という．また，そうでない個体変数，述語変数を自由個体変数，自由述語変数と言う．

- 任意の formula $F(x_1,...,x_n)$ に対し，$\{x_1,...,x_n\}F(x_1,...,x_n)$ という表記を許す．これは，

$$\{x_1,...,x_n\}F(x_1,...,x_n)a_1..a_n = F(a_1,...,a_n)$$

  を満たすものと定義する．

Definition 2.1
任意の I-formula に対し， negative part， positive part を以下のように定義する．

- F において， F は positive part に出現すると定義する．
- F が I の posivite(negative) part に出現するとき， $I \wedge J$， $J \vee I$， $\forall xI$， $\exists xI$， $J \Rightarrow I$， $\natural x((x_1,...,x_n), G, \Psi_{(n)}, \mu_{\Psi_{(n)}y_1..y_n}.H, F)$， $(\mu_{\Psi_{(n)}x_1..x_n}.I)a_1..a_n$ において， F は positive(negative) part に出現すると定義する．
- J において， formula F が positive (negative) part に出現するとき， $\neg J, J \Rightarrow I$ において， F は negative (positive) part に出現すると定義する．

Definition 2.2
I-formula F が I-formula I の subformula で，どの $\neg$ の内側にも，どの $\Rightarrow$ の左側にも，さらにどの μ の内側にも現れていないとき， F は I の input part に出現すると定義する（F が $\forall xG(x)$ の形をしているとき， F が I の input part に出現するならば， I を仕様として見たとき，この $\forall$ は入力に対応する）．

Definition 2.3
$f(c_1,...,c_n)$ が値を持たないとき， $f(c_1,...,c_n) = *$ と定義する．

Definition 2.4
以下の関数を定義する．

- $\varpi((x_1,...,x_n), K, \Psi_{(n)}, \mu_{\Psi_{(n)}y_1..y_n}.H, F) = \begin{cases} F \text{ の input part に現れる } \forall xG(x) \text{ という形式の式を} \\ \natural x((x_1,...,x_n), K, \Psi_{(n)}, \mu_{\Psi_{(n)}y_1..y_n}.H, G(x)) \\ \text{に最外から順に変えたもの} \end{cases}$

2.3 推論規則

I-formula に関する推論規則を以下に記す．
この推論規則により導出される式が全て formula で構成されるとき（$\vdash$ の右側も，左側の各要素も全て formula であるとき），これを体系 μ の真の式と呼ぶ．

$$\frac{}{\Gamma \vdash F} \text{ (Axiom)} \qquad \text{（ただし，} \Gamma \text{ に，} F \text{ が現れているとする）}$$

$$\frac{}{\Gamma \vdash (a_1,...,a_n) \prec_i (b_1,...,b_n)} \text{ (partial-Intro)} \qquad \text{（}(a_1,...,a_n) \prec_i (b_1,...,b_n) \text{ であるとき（}\prec_i \in I_n\text{））}$$

$$\frac{\Gamma \vdash F \quad \Gamma \vdash G}{\Gamma \vdash F \wedge G} \text{ (And-Intro)} \qquad \frac{\Gamma \vdash F \wedge G}{\Gamma \vdash F} \text{ (And-Elim 1)}$$

$$\frac{\Gamma \vdash F \wedge G}{\Gamma \vdash G} \text{ (And-Elim 2)} \qquad \frac{\Gamma \vdash F}{\Gamma \vdash F \vee G} \text{ (Or-Intro 1)}$$

$$\frac{\Gamma \vdash F \vee G \quad \Gamma, F \vdash H \quad \Gamma, G \vdash H}{\Gamma \vdash H} \text{ (Or-Elim)} \qquad \frac{\Gamma \vdash G}{\Gamma \vdash F \vee G} \text{ (Or-Intro 2)}$$

$$\frac{\Gamma, F \vdash G}{\Gamma \vdash F \Rightarrow G} \text{ (Imp-Intro)} \qquad \frac{\Gamma \vdash F \Rightarrow G \quad \Gamma \vdash F}{\Gamma \vdash G} \text{ (Imp-Elim)}$$

$$\frac{\Gamma, y \downarrow \vdash F(y)}{\Gamma \vdash \forall x F(x)} \text{ (Univ-Intro)} \qquad \frac{\Gamma \vdash \forall x F(x) \quad \Gamma \vdash t \downarrow}{\Gamma \vdash F(t)} \text{ (Univ-Elim)}$$

（ただし， y は下式に含まれず， t は任意の U の元とする）

$$\frac{\Gamma \vdash F(t) \quad \Gamma \vdash t \downarrow}{\Gamma \vdash \exists x F(x)} \text{ (Exists-Intro)} \qquad \frac{\Gamma \vdash \exists x F(x) \quad \Gamma, F(y) \vdash G}{\Gamma \vdash G} \text{ (Exists-Elim)}$$

（ただし， y は下式に含まれず， t は任意の U の元とする）

$$\frac{\Gamma, \forall y_1 ..y_n((y_1, ..., y_n) <_{jn} (x_1, ..., x_n) \Rightarrow F(y_1, ..., y_n)) \vdash F(x_1, ..., x_n)}{\Gamma \vdash \forall x_1, ..., x_n F(x_1, ..., x_n)} \text{ (WF-Ind)}$$

（各 x_i は下式に現れない）

$$\frac{\Gamma \vdash G(b_1, ..., b_n, Ia_{11}..a_{1n}, ..., Ia_{m1}..a_{mn})}{\Gamma \vdash Ib_1..b_n} \text{ (Mu-1)} \qquad \frac{\Gamma \vdash Ib_1..b_n}{\Gamma \vdash G(b_1, ..., b_n, Ia_{11}..a_{1n}, ..., Ia_{m1}..a_{mn})} \text{ (Mu-2)}$$

（ただし， $I = \mu_{\Psi_{(n)} x_1..x_n}.G(x_1, ..., x_n, \Psi a_{11}..a_{1n}, ..., \Psi a_{m1}..a_{mn})$ とする． ）

$$\frac{\Gamma \vdash \exists x G(x) \quad \Gamma, \forall x_1 ...x_n(H \Rightarrow \Psi_{(n)} x_1..x_n) \vdash \forall x[\Psi_{(n)}/p_{(n)}]G(x)}{\Gamma \vdash \natural x((x_1, ..., x_n), H, \Psi_{(n)}, p_{(n)}, G(x))} \text{ (Natural-Intro)}$$

$$\frac{\Gamma, H(y_1, ..., y_n) \vdash \natural x((x_1 ..x_n), H(x_1, ..., x_n), \Psi_{(n)}, I, K(y_1, ..., y_n)) \quad \Gamma \vdash H(a_1, ..., a_n)}{\Gamma \vdash Ia_1..a_n} \text{ (Mu-Intro)}$$

（ただし， $H(y_1, ..., y_n) = \neg L(y_1, ..., y_n)$ の形をしていて，
$I = \mu_{\Psi_{(n)} x_1..x_n}.G(x_1, ..., x_n, \Psi a_{11}..a_{1n}, ..., \Psi a_{m1}..a_{mn})$
$K(y_1, ..., y_n) = G(y_1, ..., y_n, Ia_{11}..a_{1n}, ..., Ia_{m1}..a_{mn})$ とする． ）

$$\frac{\Gamma \vdash F \quad \Gamma \vdash \neg F}{\Gamma \vdash \bot} \text{ (Bot-Intro,neg-Elim)} \qquad \frac{\Gamma \vdash \bot}{\Gamma \vdash F} \text{ (Bot-Elim)}$$

$$\frac{\Gamma, F \vdash \bot}{\Gamma \vdash \neg F} \text{ (Neg-Intro)} \qquad \frac{}{\Gamma \vdash a = a} \text{ (eq 1)}$$

$$\frac{\Gamma \vdash a = b \quad \Gamma \vdash F(a)}{\Gamma \vdash F(b)} \text{ (eq 2)} \qquad \frac{\Gamma \vdash a < b \quad \Gamma \vdash b < c}{\Gamma \vdash a < b} \text{ (le)}$$

3 μ の恒真性

3.1 恒真性の定義

まず，前章での記号， object， predicate に対し， V-formula を定義する． V-formula は，恒真性を定義する際に使われる．

- V-formula
  $F ::= p_{(n)} a_1..a_n | f | F_1 \wedge F_2 | F_1 \vee F_2 | F_1 \Rightarrow F_2 | \neg F | \forall x F | \exists x F | \natural x F | \forall' x F | \exists' x F | \natural' x F | a \downarrow$

この定義より， formula は， V-formula である．

Definition 3.1
任意の V-formula に対し， negative part， positive part を I-formula と同様，以下のように定義する．

- F において， F は positive part に出現すると定義する．
- F が I の posivite(negative) part に出現するとき， $I \wedge J$， $J \vee I$， $\forall xI$， $\exists xI$， $J \Rightarrow I$ $\natural xI$， $\forall' xI$， $\exists' xI$， $\natural' xI$， $(\mu_{\Psi_{(n)}x_1..x_n}.I)a_1..a_n$ において， F は positive(negative) part に出現すると定義する．
- J において， formula F が positive(negative) part に出現するとき， $\neg J, J \Rightarrow I$ において， F は negative(positive) part に出現すると定義する．

Definition 3.2
V-formula F が V-formula I の subformula で，どの $\neg$ の内側にも，どの $\Rightarrow$ の左側にも，さらにどの μ の内側にも現れていないとき， F は I の input part に出現すると定義する（F が $\forall xG(x)$ の形をしているとき， F が I の input part に出現するならば， I を仕様として見たとき，この $\forall$ は入力に対応する）．

Definition 3.3
以下の関数を定義する．

- $\kappa((x_1,...,x_n),K,\Psi_{(n)},\mu_{\Psi_{(n)}y_1..y_n}.H,F) = \left\{ \begin{array}{l} F \text{ の input part に現れる } \forall xG(x) \text{ という形式の式を} \\ (\forall x_1,..x_n(K \Rightarrow \Psi_{(n)}x_1..x_n) \\ \qquad \Rightarrow \forall x[\Psi_{(n)}/\mu_{\Psi_{(n)}y_1..y_n}.H]G(x)) \wedge \exists xG(x) \\ \text{に最外から順に変えたもの} \end{array} \right.$

- $\theta_n(Ib_1..b_n) = \left\{ \begin{array}{lll} n=0 \text{ のとき} & \cdots & Ib_1..b_n \\ n \neq 0 \text{ のとき} & \cdots & \theta_{n-1}(Ib_1..b_n) \text{ に現れる } \Psi c_1..c_n \text{ の形の subformula を全て} \\ & & G(c_1,...,c_n,\Psi a_{11}..a_{1n},..,\Psi a_{m1}..a_{mn}) \text{ で置き換えたもの} \end{array} \right.$

  （ただし， $I = \mu_{\Psi_{(n)}x_1..x_n}.G(x_1,...,x_n,\Psi a_{11}..a_{1n},...,\Psi a_{m1}..a_{mn})$ とする．）

- $\gamma_n(Ib_1..b_n,p_{(n)}) = \theta_n(Ib_1..b_n)$ の Ψ を全て $p_{(n)}$ で置き換えたもの．

  （ただし， $I = \mu_{\Psi_{(n)}x_1..x_n}.G(x_1,...x_n,\Psi a_{11}..a_{1n},...,\Psi a_{m1}..a_{mn})$ とする．）

- $\rho_0(F) = \left\{ \begin{array}{l} F \text{ の input part に現れる } \forall xG(x) \text{ の形をした subformula を最外から順に } \natural' xG(x) \text{ に変え，} \\ \quad \text{その後， positive part で } \mu \text{ の内側に現れない } \forall \text{ を全て } \forall' \text{ に変え，さらに negative part で} \\ \quad \text{最外から順に } \natural' xG(x) \text{ に変え， } \mu \text{ の内側に現れない } \exists \text{ を全て } \exists' \text{ に変えたもの．} \end{array} \right.$

- $\rho(F) = \left\{ \begin{array}{l} F \text{ の positive part で } \mu \text{ の内側に現れない } \forall \text{ を全て } \forall' \text{ に変え，} \\ \quad \text{さらに negative part で } \mu \text{ の内側に現れない } \exists \text{ を全て } \exists' \text{ に変えたもの．} \end{array} \right.$

- $\nu(F) = \left\{ \begin{array}{l} F \text{ の negative part で } \mu \text{ の内側に現れない } \forall \text{ を全て } \forall' \text{ に変え，} \\ \quad \text{さらに positive part で } \mu \text{ の内側に現れない } \exists \text{ を全て } \exists' \text{ に変えたもの．} \end{array} \right.$

- $\eta(F_1,...,F_n \vdash F) = F_1 \Rightarrow (F_2 \Rightarrow (...(F_n \Rightarrow F)...))$

Definition 3.4
この体系において， $\mathcal{M} = < \mathbf{U}, U_f, V >$ を structrue と呼ぶ．ただし，

- $\mathbf{U} = < U, \cap_{n \in Nat} I_n \} >$ で， U はある集合， I_n は U^n 上の well-founded partial order の集合とする．
- U_f は， U 上の部分関数の集合とする．
- V は個体定数に対し U の元を割り当て， n 引数述語定数 $p_{(n)}$ に U^n の部分集合を割り当てる．また n 引数部分関数 f_n には， $U \cap \{*\}$ 上 n 引数関数を割り当てる．
- V は以下の要請を満たす．
  1. $V(f(a_1,..,a_n)) = V(f)(V(a_1),..,V(a_n))$
  2. $V(a_1),...,V(a_n)$ のどれかが $*$ のとき， $V(f)(V(a_1),...V(a_n)) = *$
  3. $V(\natural) = \{a | V(a) \neq *\}$

4. $<_i \epsilon I_n$ で $V(<_i) = U_1$ としたとき，
   $\{(V(a_1),...,V(a_n),V(b_1),..,V(b_n)) | (V(a_1),..,V(a_n) \prec_i (V(b_1),...,V(b_n))\} = U_1 \quad (\prec_i \epsilon I_n)$.

δ に現れる自由個体変数，任意の個体定数 $a \epsilon U$，n 引数自由述語変数を任意の n 引数述語定数 p_n に置き換えたもの δ' に対し，$\psi_{\mathcal{M},\phi,U,\phi}(\eta(\delta')) = 1$ であるとき，δ は structure M で恒真であるといい，$\mathcal{M} \models \delta$ と書く．ただし，ϕ は空集合とする.

ここで，モデルの都合上 $n \epsilon NaturalNumber$ である様な全ての n に対し，新たな述語定数 $\omega_{(n)}$ を導入する．ただし，この $\omega_{(n)}$ に関しては V の解釈は考えない.

$\psi_{\mathcal{M},M,N,H}(F)$ は，1, 0 の値をとり以下のように定義される．ただし，M はある n に対し，$M \subseteq U^n$，$N \subset U$，$H \subseteq M$ である（$U^0 = \{\phi\}$ とする．）.

- $H = \{\phi\}$，かつそのときに限り，

$$\psi_{\mathcal{M},M,N,H}(\omega_{(0)}) = 1$$

- $(V(x_1),...V(x_n)) \epsilon H$，かつそのときに限り，

$$\psi_{\mathcal{M},M,N,H}(\omega_{(n)} x_1 .. x_n) = 1 \quad (n > 0)$$

- $(V(a_1),...,V(a_n)) \epsilon V(p_{(n)})$，かつそのときに限り，

$$\psi_{\mathcal{M},M,N,H}(p_{(n)} a_1 .., a_n) = 1$$

- $\psi_{\mathcal{M},M,N,H}(F), \psi_{\mathcal{M},M,N,H}(G)$ のどちらも 1，かつそのときに限り，

$$\psi_{\mathcal{M},M,N,H}(F \wedge G) = 1$$

- $\psi_{\mathcal{M},M,N,H}(F), \psi_{\mathcal{M},M,N,H}(G)$ のどちらかが 1，かつそのときに限り，

$$\psi_{\mathcal{M},M,N,H}(F \vee G) = 1$$

- $\psi_{\mathcal{M},M,N,H}(F) = 0$，かつそのときに限り，

$$\psi_{\mathcal{M},M,N,H}(\neg F) = 1$$

- $\psi_{\mathcal{M},M,N,H}(G) = 1$. または，$\psi_{\mathcal{M},M,N,H}(F)$ が 0，かつそのときに限り，

$$\psi_{\mathcal{M},M,N,H}(F \Rightarrow G) = 1$$

- 全ての $a \epsilon U$ に対し，$\psi_{\mathcal{M},M,N,H}(F(a)) = 1$，かつそのときに限り，

$$\psi_{\mathcal{M},M,N,H}(\forall x F(x)) = 1$$

- ある $a \epsilon U$ が存在し，$\psi_{\mathcal{M},M,N,H}(F(a)) = 1$，かつそのときに限り，

$$\psi_{\mathcal{M},M,N,H}(\exists x F(x)) = 1$$

- 全ての $a \epsilon N$ に対し，$\psi_{\mathcal{M},M,N,H}(F(a)) = 1$，かつそのときに限り，

$$\psi_{\mathcal{M},M,N,H}(\forall' x F(x)) = 1$$

- ある $a \epsilon N$ が存在し，$\psi_{\mathcal{M},M,N,H}(F(a)) = 1$，かつそのときに限り，

$$\psi_{\mathcal{M},M,N,H}(\exists' x F(x)) = 1$$

- $K \subset U^n, (b_1, ..., b_n) \epsilon K$ を満たすある K があって，
  $(w_1, ... w_n) \epsilon K$ ならば，U の任意の有限部分集合 L に対し，
  ある自然数 s が存在し，全ての自然数 $t > s$ に対し，

$$\psi_{\mathcal{M},K,L,\phi}(\rho_0(\gamma_t(Iw_1..w_n, \omega_{(n)}))) = 1$$

  かつそのときに限り，

$$\psi_{\mathcal{M},M,N,H}(Ib_1..b_n) = 1$$

  （ただし，$I = \mu_{\Psi_{(n)}x_1..x_n}.G(x_1, ..., x_n, \Psi a_{11}..a_{1n}, .., \Psi a_{m1}..a_{mn})$ とする．また，各 a_{ij} は $(x_1, .., x_n)$ に依存する．）

- $\psi_{\mathcal{M},M,N,M}(\forall' x G(x)), \psi_{\mathcal{M},M,N,H}(\exists x G(x))$ のどちらも 1，かつそのときに限り，

$$\psi_{\mathcal{M},M,N,H}(\natural' x G(x)) = 1$$

全ての structrue で δ が恒真であるとき，δ は単に恒真であると言う．

以下簡単のため，$V(a)$ を a，$V(p_n)$ を p_n と書く．

Theorem 3.1
上で与えた恒真性の定義は well defined.

Proof. 帰納的に定義されていることをいえば良い．
V-formula F に対し，F の論理演算子の数を i（ただし，$\natural'$ は 2 と数える），F に現れる

$$\mu_{\Psi_{(m)}x_1..x_m}.G(x_1, ..., x_m, \Psi_{(m)}a_{11}..a_{1m}, ..., \Psi_{(m)}a_{l1}..a_{lm})$$

の形の数を j とし，(j, i) の辞書式 order を考えれば良い． □

3.2 Soundness

前章においてあたえた推論規則が前節の恒真性に対し，Sound であることをいう．

Lemma 3.1
任意の $\forall', \exists', \natural'$ を含まない V-formula F に対し，以下の 2 つは同値

- $\psi_{\mathcal{M},M,L,H}(F) = 1$ (L は U の部分集合)
- $\psi_{\mathcal{M},M,U,H}(F) = 1$

Lemma 3.2
任意の V-formula F に対し，F が $\natural$，$\natural'$ を含まないなら以下の 2 つは同値

- $\psi_{\mathcal{M},M,N,H}(F) = 1$
- $\psi_{\mathcal{M},\phi,N,H}(F) = 1$

（ただし，M は U^n の部分集合とする．）

Lemma 3.3
任意の V-formula F，その subformula でどの μ の内側に現れない J に関し，F，J が以下の条件を満たせば，J を任意の V-formula I に変えても，F の $\psi_{\mathcal{M},K,L,\phi}$ の値は変わらない．

- J が，positive part に現れるとき，

$$\psi_{\mathcal{M},K,L,\phi}(J) = 0, \psi_{\mathcal{M},K,L,\phi}(F) = 1$$

  または，

$$\psi_{\mathcal{M},K,L,\phi}(J) = 1, \psi_{\mathcal{M},K,L,\phi}(F) = 0$$

- J が、 negative part に現れるとき,

$$\psi_{\mathcal{M},K,L,\phi}(J) = 1, \psi_{\mathcal{M},K,L,\phi}(F) = 1$$

または,

$$\psi_{\mathcal{M},K,L,\phi}(J) = 0, \psi_{\mathcal{M},K,L,\phi}(F) = 0$$

Lemma 3.4
任意の V-formula F に関し,

$$\psi_{\mathcal{M},K,L,\phi}(F) = 1$$

ならば,

$$\psi_{\mathcal{M},K,L,K}(F) = 1$$

Lemma 3.5
任意の V-formula $F, F_1, ..., F_n$ に関し,

$$\psi_{\mathcal{M},\phi,U,\phi}(r(F_1, .., F_n \vdash F)) = 1$$

$$\Leftrightarrow \quad \forall i \epsilon \{1, 2, .., n\}(\psi_{\mathcal{M},\phi,U,\phi}(F_i) = 0) \vee \psi_{\mathcal{M},\phi,U,\phi}(F) = 1$$

Lemma 3.6
mu-Intro,Part-Elim,Part-Intro を除く各推論規則に対し，上式を $\delta_1, ..\delta_n$，下式が δ とするとき,

$$\forall i \epsilon \{1, 2, ..., n\}(\psi_{\mathcal{M},\phi,U,\phi}(r(\delta_i)) = 1) \Rightarrow \psi_{\mathcal{M},\phi,U,\phi}(r(\delta)) = 1$$

である.

Lemma 3.7
$\psi_{\mathcal{M},\phi,U,\phi}(F_i) = 1$ をみたす F_i から Mu-Intro,Part-Elim,Part-Intro を除く各推論規則を使って導出できる任意の I-formula F に対し,

$$\psi_{\mathcal{M},\phi,U,\phi}(F) - 1$$

Lemma 3.8
任意の V-formula F に対して,

$$\varpi((x_1, .., x_n), J, \Psi_{(n)}, I, F)$$

が,

$$\psi_{\mathcal{M},\phi,U,\phi}(F_i) = 1$$

をみたす F_i から推論規則で導出できるなら,

$$\kappa((x_1, .., x_n), J, \Psi_{(n)}, I, F)$$

は,

$$\psi_{\mathcal{M},\phi,U,\phi}(F_i) = 1$$

をみたす F_i から推論規則 Part-Intro,Part-Elim を使わずに導出できる.
（ただし， $I - \mu_{\Phi_{(n)} x_1 .. x_n}.G(x_1, ... x_n, \Psi a_{11} .. a_{1n}, .., \Psi a_{m1} .. a_{mn})$ とする. ）

Definition 3.5
V-formula F に対し， F に自由個体変数が含まれないとき， $(\mathrm{elim})_L$ の作業を， この formula に現れる $\forall$, $\exists$ に対し，最外から順に以下の作業を行なったものと定義する（ただし， μ の内側に現れるものは対象としない）. このとき，任意に選んだ a を $(\mathrm{elim})_L$ による割り付けと呼ぶ.

- positive part のものに関して,
    1. $\psi_{\mathcal{M},\phi,U,\phi}(\forall x D(x)) = 1$ のとき，任意の $a \epsilon L$ に対し， $D(a)$ に変える.

2. $\psi_{\mathcal{M},\phi,U,\phi}(\forall x D(x)) = 0$ のとき，定義よりある $a\epsilon U$ に対し，

$$\psi_{\mathcal{M},\phi,U,\phi}(G(a)) = 0$$

である．このような a を選んで $D(a)$ にかえる．

3. $\psi_{\mathcal{M},\phi,U,\phi}(\exists x D(x)) = 1$ のとき，定義よりある $a\epsilon U$ に対し，

$$\psi_{\mathcal{M},\phi,U,\phi}(D(a)) = 1$$

である．このような a を選んで $D(a)$ にかえる．

4. $\psi_{\mathcal{M},\phi,U,\phi}(\exists x D(x)) = 0$ のとき，任意の $a\epsilon L$ に対し，$D(a)$ に変える．

- negative part のものに関して，
  1. $\psi_{\mathcal{M},\phi,U,\phi}(\forall x D(x)) = 1$ のとき，任意の $a\epsilon L$ に対し，$D(a)$ に変える．
  2. $\psi_{\mathcal{M},\phi,U,\phi}(\forall x D(x)) = 0$ のとき，定義よりある $a\epsilon U$ に対し，

  $$\psi_{\mathcal{M},\phi,U,\phi}(G(a)) = 0$$

  である．このような a を選んで $D(a)$ にかえる．

  3. $\psi_{\mathcal{M},\phi,U,\phi}(\exists x D(x)) = 1$ のとき，定義よりある $a\epsilon U$ に対し，

  $$\psi_{\mathcal{M},\phi,U,\phi}(D(a)) = 1$$

  である．このような a を選んで $D(a)$ にかえる．

  4. $\psi_{\mathcal{M},\phi,U,\phi}(\exists x D(x)) = 0$ のとき，任意の $a\epsilon L$ に対し，$D(a)$ に変える．

Lemma 3.9
F を任意の formula とし，F に $(\mathrm{elim})_L$ （$L \subseteq U$）の任意の割り付けを行ったとする．その結果 F とその任意の subformula K （K は F 自身である場合を含む）が F'，K' になったとすると，これらは以下の性質を満たす．

- $\psi_{\mathcal{M},\phi,U,\phi}(K) = 1$ ならば，$\psi_{\mathcal{M},\phi,U,\phi}(K') = 1$
- $\psi_{\mathcal{M},\phi,U,\phi}(K) = 0$ ならば，$\psi_{\mathcal{M},\phi,U,\phi}(K') = 0$

Lemma 3.10
F は，任意の $G(I\,a_{11}..a_{1n},\ldots,I\,a_{m1}..a_{mn})$ の形をした（各 $I\,a_{in}..a_{mn}$ は positive part に現れる） formula であるとする．もし，

$$\psi_{\mathcal{M},K,U,\phi}(G) = 1$$

であり，この任意の $(\mathrm{elim})_L$ の割り付けを行った G' に対し，n 引数述語 $o_{(n)}$ が，G' に現れる各 $Ic_{i1}..c_{in}$ に対し，

- $\psi_{\mathcal{M},K,U,\phi}(Ic_{i1}..c_{in}) = 1$ ならば，

$$\psi_{\mathcal{M},K,U,\phi}(o_{(n)}c_{i1}..c_{in}) = 1$$

であるなら，G の positive part の $\forall$ を $\forall'$ に，negative part の $\exists$ を $\exists'$ に変えたものを G'' とすると，

$$\psi_{\mathcal{M},K,L,\phi}([o_{(n)}/I]G'') = 1$$

（ただし，$I = \mu_{\Psi_{(n)}x_1..x_n}.F(x_1,\ldots,x_n,\Psi a_{11}..a_{1n},\ldots,\Psi a_{m1}..a_{mn})$ とする．）

Proof. $o_{(n)}$ が上記の性質を満たすとし，G に対し，K を任意の subformula とする．今，K の外側までの $(\mathrm{elim})_L$ による任意の割り振りが定まったとする．これにより，K は自由変数を持たないものとなっている．これを K''' とおく．このとき，

- K が positive part に現れていて，$\psi_{\mathcal{M},K,U,\phi}(K''') = 1$ のとき，K''' に $(\mathrm{elim})_L$ の任意の割り付けを行い，さらに I を $o_{(n)}$ に変えたもの K' の $\psi_{\mathcal{M},K,L,\phi}$ による値が全て 1 ならば，$\psi_{\mathcal{M},K,L,\phi}$ による K'' の値は 1.

- K が negtive part に現れていて，$\psi_{\mathcal{M},K,U,\phi}(K''') = 0$ のとき，K''' に $(\mathrm{elim})_L$ の任意の割り付けを行い，さらに I を $o_{(n)}$ に変えたもの K' の $\psi_{\mathcal{M},K,L,\phi}$ による値が全て 0 ならば，$\psi_{\mathcal{M},K,U,\phi}$ による K'' の値は 0.

（K'' は G の subformula K の positive part の $\forall$ を $\forall'$ に，negative part の $\exists$ を $\exists'$ に 変え，さらに I を $o_{(n)}$ に変えたもの）
を証明すればよい.
K 構造の大きさ p による Induction. （ただし，$Ia_{i1}..a_{in}$ の大きさを 1 とする.）

- $p=1$ のとき OK.
- $p=l-1$ まで，OK とする．$p=l$ のとき，

1. $K = G_1 \wedge G_2, G_1 \vee G_2, G_1 \Rightarrow G_2, \neg G_1$ のとき，帰納法の仮定と恒真性の定義より OK.
2. $K = \forall x K_1(x)$ のとき，
   (a) $\psi_{\mathcal{M},K,U,\phi}(K) = 1$ で，positive part に現れるとき
   $\psi_{\mathcal{M},K,L,\phi}(\forall x K_1(x)) = 1$ であるから，全ての $a \epsilon L$ について，
   $$\psi_{\mathcal{M},K,L,\phi}(K_1(a)) = 1$$
   よって，全ての $a \epsilon U$ について，
   $$\psi_{\mathcal{M},K,L,\phi}(K_1(a)) = 1$$
   $\psi_{\mathcal{M},K,L,\phi}(K') = 1$ とすると，$(\mathrm{elim})_L$ の定義より，全ての $b \epsilon L$ について，
   $$\psi_{\mathcal{M},K,L,\phi}([o_{(n)}/I]K_1'(b)) = 1$$
   よって，帰納法の仮定より，全ての $b \epsilon L$ について，
   $$\psi_{\mathcal{M},K,L,\phi}([o_{(n)}/I]K_1''(b)) = 1$$
   よって，
   $$\psi_{\mathcal{M},K,L,\phi}(\forall' x[o_{(n)}/I]K_1''(x)) = 1$$
   OK.
   (b) $\psi_{\mathcal{M},K,U,\phi}(K) = 0$ で，negative part に現れるとき
   $\psi_{\mathcal{M},K,L,\phi}(K') = 0$ とすると，$(\mathrm{elim})_L$ の定義より，ある $a \epsilon L$ について，
   $$\psi_{\mathcal{M},K,L,\phi}([o_{(n)}/I]K_1'(a)) = 0$$
   かつ，
   $$\psi_{\mathcal{M},K,L,\phi}(K_1(a)) = 0$$
   帰納法の仮定より，
   $$\psi_{\mathcal{M},K,L,\phi}([o_{(n)}/I]K_1''(a)) = 0$$
   よって，
   $$\psi_{\mathcal{M},K,L,\phi}(\forall x[o_{(n)}/I]K_1''(x)) = 0$$
   OK.
3. $K = \exists x K_1(x)$ のとき，
   (a) $\psi_{\mathcal{M},K,U,\phi}(K) = 1$ で，positive part に現れるとき
   $\psi_{\mathcal{M},K,L,\phi}(K') = 1$ とすると，$(\mathrm{elim})_L$ の定義より，ある $a \epsilon L$ について，
   $$\psi_{\mathcal{M},K,L,\phi}([o_{(n)}/I]K_1'(a)) = 1$$
   かつ，
   $$\psi_{\mathcal{M},K,L,\phi}(K_1(a)) = 1$$

帰納法の仮定より、

$$\psi_{\mathcal{M},K,L,\phi}([o_{(n)}/I]K_1''(a)) = 1$$

よって、

$$\psi_{\mathcal{M},K,L,\phi}(\exists x[o_{(n)}/I]K_1''(x)) = 1$$

OK.

(b) $\psi_{\mathcal{M},K,U,\phi}(K) = 0$ で、negative part に現れるとき
$\psi_{\mathcal{M},K,L,\phi}(\exists x K_1(x)) = 0$ であるから、全ての $a \epsilon U$ について、

$$\psi_{\mathcal{M},K,L,\phi}(K_1(a)) = 0$$

$\psi_{\mathcal{M},K,L,\phi}(K') = 0$ とすると、$(\text{elim})_L$ の定義より、全ての $b \epsilon L$ に対し、

$$\psi_{\mathcal{M},K,L,\phi}([o_{(n)}/I]K_1'(b)) = 0$$

よって、帰納法の仮定より、

$$\psi_{\mathcal{M},K,L,\phi}([o_{(n)}/I]K_1''(b)) = 0$$

よって、

$$\psi_{\mathcal{M},K,L,\phi}(\exists' x[o_{(n)}/I]K_1''(x)) = 0$$

OK. □

Lemma 3.11
F を V-formula とする.

$$\psi_{\mathcal{M},K,L,H}(F) = 1$$

であるとき、$K \subseteq K'$、$H \subseteq H'$ ならば、

$$\psi_{\mathcal{M},K',L,H'}(F) = 1$$

Lemma 3.12
今、K を、

$$\psi_{\mathcal{M},\phi,U,\phi}(J(a_1,\ldots a_n)) = 1$$

を満たす全ての $(a_1,\ldots,a_n)$ により構成される集合とする.
任意の n 引数述語 P、任意の $(w_1,\ldots,w_1) \epsilon K$ に対し、

$$\psi_{\mathcal{M},\phi,U,\phi}(\kappa((x_1,\ldots x_n), J(x_1\ldots x_n), P, I, G(w_1,\ldots w_n, I\ a_{11}..a_{1n},\ldots, I\ a_{m1}..a_{mn}))) = 1$$

ならば、任意の t について、

$$\psi_{\mathcal{M},K,U,K}(\rho_0(\gamma_t(I w_1..w_n, \omega_{(n)}))) = 1$$

（ただし、$I = \mu_{\Psi_{(n)}x_1..x_n}.G(x_1,\ldots,x_n, \Psi a_{11}..a_{1n},\ldots, \Psi a_{m1}..a_{mn})$ とする.）

Proof. Lemma 3.11より、

$$\psi_{\mathcal{M},K,U,K}(\kappa((x_1,\ldots,x_n), J(x_1\ldots x_n), P, I, G(w_1,\ldots,w_n, I\ a_{11}..a_{1n},\ldots, I\ a_{m1}..a_{mn}))) = 1$$

で、P は任意であるので、P に $\omega_{(n)}$ を代入したものを F_1' とすると、

$$\psi_{\mathcal{M},\phi,U,\phi}(F_1') = 1$$

また、

$$\psi_{\mathcal{M},K,U,K}(\forall x_1,\ldots,x_n(J(x_1,\ldots,x_n) \Rightarrow \omega_{(n)}x_1..x_n)) = 1$$

である. よって、F_1' の最内に現れる

$$\forall x_1,\ldots,x_n(J(x_1,\ldots,x_n) \Rightarrow \omega_{(n)}x_1..x_n) \Rightarrow (\forall x[\omega_{(n)}/I_{(n)}]K(x) \wedge \exists x K(x))$$

を

$$\forall x[\omega_{(n)}/I_{(n)}]K(x) \wedge \exists x K(x)$$

に置き換えたものを F_1'' とすると，これは，$G(w_1,\ldots,w_n, I\,a_{11}..a_{1n},..,I\,a_{m1}..a_{mn})$ の positive part 以外には現れないので，Lemma 3.3より，

$$\psi_{\mathcal{M},K,U,K}(F_1'') = 1$$

この

$$\forall x[\omega_{(n)}/I_{(n)}]K(x)) \wedge \exists x K(x)$$

において，I は $K(x)$ の positive part 以外には現れないので，Lemma 3.3より，

$$\psi_{\mathcal{M},K,U,K}(\forall x[\omega_{(n)}/I_{(n)}]K(x) \wedge \exists x K(x)) = 1$$

ならば，

$$\psi_{\mathcal{M},K,U,K}(\forall x[\omega_{(n)}/I_{(n)}]K(x) \wedge \exists x[\omega_{(n)}/I]K(x)) = 1$$

また，$\exists'$ の恒真性の定義より，

$$\psi_{\mathcal{M},K,U,K}(\forall x[\omega_{(n)}/I_{(n)}]K(x) \wedge \exists x[\omega_{(n)}/I]K(x) = \psi_{\mathcal{M},K,U,K}(\exists' x K(x))$$

であるので，Lemma 3.3より，F_1'' の

$$\forall x[\omega_{(n)}/I_{(n)}]K(x)) \wedge \exists x K(x)$$

を

$$\exists' x K(x)$$

に置き換えたものを F_1''' とすると，

$$\psi_{\mathcal{M},K,U,K}(F_1''') = 1$$

この操作を繰り返し行うことにより，

$$\psi_{\mathcal{M},K,U,K}(\rho_0(\gamma_0(I w_1..w_n, \omega_{(n)}))) = 1$$

が得られる．さて，これは，任意の $(w_1,\ldots,w_n)\epsilon K$ で成り立つので，$\rho_0(\gamma_0(I w_1..w_n,\omega_{(n)}))$ の全ての $\omega_{(n)}a_1..a_n$ を $\rho_0(\gamma_0(I a_1..a_n,\omega_{(n)}))$ で置き換えたもの，すなわち $\rho_0(\gamma_1(I w_1..w_n,\omega_{(n)}))$ に対し，Lemma 3.3より，

$$\psi_{\mathcal{M},K,U,K}(\rho_0(\gamma_1(I w_1..w_n, \omega_{(n)}))) = 1$$

これを繰り返すことにより，全ての t に対し，

$$\psi_{\mathcal{M},K,U,K}(\rho_0(\gamma_t(I w_1..w_n, \omega_{(n)}))) = 1$$

OK. □

Lemma 3.13

$$\Gamma, J(y_1,\ldots,y_n) \vdash \varpi((x_1,\ldots,x_n), J(x_1,..x_n), P_{(n)}, G, F(y_1,\ldots,y_n, G\,a_{11}..a_{1n},..,G\,a_{m1}..a_{mn}))$$

$$\Gamma \vdash J(a_1,\ldots,a_n)$$

が推論規則により導出できるなら，

$$\psi_{\mathcal{M},\phi,U,\phi}(r(\Gamma \vdash G a_1..a_n)) = 1$$

(ただし，$G = \mu_{\Psi_{(n)}(x_1,\ldots,x_n)}.F(x_1,\ldots,x_n, \Psi_{(n)}a_{11}..a_{1n},\ldots,\Psi_{(n)}a_{m1}..a_{mn}))$)

Proof. Lemma3.5より，$\Gamma = F_1,..F_t$ とすると，

$$\forall i\epsilon\{1,2,..,t\}(\psi_{\mathcal{M},\phi,U,\phi}(F_i) = 1) \wedge \psi_{\mathcal{M},\phi,U,\phi}(J(y_1,..,y_n) = 1)$$

としてよい．J を n 引数の述語とし，K を J により定められる $\prod_n U$ の部分集合，$(w_1,..w_n)\epsilon K$ とする．仮定と Lemma3.8より，任意の n 引数の述語 P に対し，

$$\kappa((x_1,..,x_n), J(x_1..x_n), P, G, F(w_1,..,w_n, G\, a_{11}..a_{1n},..G\, a_{m1}..a_{mn}))$$

の，Part-Intro,Part-Elim を使わない証明図が存在する．よって，Lemma3.7より，

$$\psi_{\mathcal{M},\phi,U,\phi}(\kappa((x_1,..,x_n), J(x_1,..x_n), P, G, F(w_1,..,w_n, G\, a_{11}..a_{1n},.., G\, a_{m1}..a_{mn}))) = 1 \quad \cdots (1)$$

F には G 以外には μ を含まないとしてよい．

1. $F(w_1,..,w_n, G\, a_{11}..a_{1n},.., G\, a_{m1}..a_{mn})$ の positive part でかつどの μ の内側にも現れない $\forall$ がないとき．
   κ の定義，(1) より，

   $$\psi_{\mathcal{M},\phi,U,\phi}(F(w_1,..,w_n, G\, a_{11}..a_{1n},.., G\, a_{m1}..a_{mn}))) = 1$$

   よって，

   $$\psi_{\mathcal{M},\phi,U,\phi}(Gw_1..w_n) = 1$$

   $(w_1,..w_n)$ は，$J(w_1,..,w_n)$ を満たす任意のものであるので，OK.
2. $F(w_1,..,w_n, G\, a_{11}..a_{1n},.., G\, a_{m1}..a_{mn})$ の positive part でかつどの μ の内側にも現れない $\forall$ があるとき．
   ある $K' \subset U^n$，$a_1,..,a_n\epsilon K'$ があって，
   $(w_1,..,w_n)\epsilon K'$ ならば，U の任意の有限部分集合 L に対し，ある自然数 s が存在し，全ての $t > s$ に対し，

   $$\psi_{\mathcal{M},K',L,\phi}(\rho_0(\gamma_t(Gw_1..w_n, \omega_{(n)}))) = 1$$

   を証明する．

   仮定より，

   $$\psi_{\mathcal{M},\phi,U,\phi}(\kappa((x_1,..,x_n), J(x_1,..x_n), P, G, F(w_1,..,w_n, G\, a_{11}..a_{1n},.., G\, a_{m1}..a_{mn}))) = 1$$

   $\kappa(J(x_1..x_n), Px_1..x_n, Gx_1..x_n, F(w_1,..,w_n, G\, a_{11}..a_{1n},.., G\, a_{m1}..a_{mn})))$ に対し，任意にとった $(\text{elim})_L$ の割り付け結果を，$F'(w_1,..w_n, G\, c_{11}..c_{1n},.., G\, c_{m1}..c_{mn})$ とする．Lemma 3.9より，

   $$\psi_{\mathcal{M},\phi,U,\phi}(F'(w_1,..,w_n, G\, c_{11}..c_{1n},.., G\, c_{m1}..c_{mn})) = 1$$

   このとき，この formula に現れる各 $Gc_{i1}..c_{in}$ に対し，$c_{i1},..,c_{in}$ は定数．よって，これらに対しそれぞれ $\psi_{\mathcal{M},\phi,U,\phi}$ による値が定まる．$\psi_{\mathcal{M},\phi,U,\phi}(Gc_{i1}..c_{in})$ の値が 1 のものに対し，それぞれ，

   ある $K_i \subset U^n$，$(c_{i1},..,c_{in})\epsilon K_i$ となるような K_i があって，$(w_{i1},..,w_{in})\epsilon K_i$ ならば，U の任意の有限部分集合 L' に対し，ある自然数 s が存在し，全ての $t > s$ に対し，

   $$\psi_{\mathcal{M},K_i,L',\phi}(\rho_0(\gamma_t(Gw_{i1}..w_{in}, \omega_{(n)}))) = 1$$

   よって，Lemma 3.11より，

   $(w'_1,..,w'_n)\epsilon \cup_i K_i$ ならば，U の任意の有限部分集合 L' に対し，ある自然数 s が存在し，全ての $t > s$ に対し，

   $$\psi_{\mathcal{M},\cup_i K_i,L',\phi}(\rho_0(\gamma_t(Gw'_1..w'_n, \omega_{(n)}))) = 1 \quad \cdots \quad (2)$$

   ところで，$(w_1,..w_n)\epsilon K$ の取り方も，$(\text{elim})_L$ の割り付けも任意であったので，それぞれについて $\cup_i K_i$

が定まる．それら全ての和集合を Kk とすると，(2)，Lemma 3.11より，

$(w''_1,...,w''_n)\epsilon Kk$ ならば，U の任意の有限部分集合 L' に対し，ある自然数 s が存在し，全ての $t > s$ に対し，

$$\psi_{\mathcal{M},Kk,L',\phi}(\rho_0(\gamma_t(Gw''_1..w''_n,\omega_{(n)}))) = 1 \quad \cdots \quad (3)$$

よって，μ の恒真性の定義により，

$(p_1,...,p_n)\epsilon Kk$ であるものに対し，

$$\psi_{\mathcal{M},\phi,U,\phi}(Gp_1..p_n)) = 1 \quad \cdots \quad (3)'$$

さて，$F'(w_1,...,w_n,G\ c_{11}..c_{1n},...,G\ c_{m1}..c_{mn})$ に現れる各 $Gc_{i1}..c_{in}$ で，$\psi_{\mathcal{M},\phi,U,\phi}(Gc_{i1}..c_{in})$ の値が1であるものを $Gc_{i_j1}..c_{i_jn}$ $(j\epsilon J)$ とすると，J は有限．また，$F(w_1,..,w_n,G\ c_{11}..c_{1n},..,G\ c_{m1}..c_{mn})$ に現れる $\forall,\exists$ の数も L の元も有限なので $(\mathrm{elim})_L$ の割り付けはパターンは有限．よって，各 $(\mathrm{elim})_L$ の割り付けにより定められる $\{(c_{i_j1},...,c_{i_jn})\}_{j\epsilon J}$ の和集合を C とすると，やはり有限．よって，各 $(\mathrm{elim})_L$ の割り当てにより定まる (2) の，L' を L としたものでそれぞれで定まる s_L に対し，それらの最大値が存在するので，Lemma 3.11より，

全ての $(l_1,...,l_n)\epsilon C$，全ての $t > s_L$ に対し，

$$\psi_{\mathcal{M},Kk,L,\phi}(\rho_0(\gamma_t(Gl_1..l_n,\omega_{(n)}))) = 1 \quad \cdots \quad (4)$$

さて，

$$\psi_{\mathcal{M},\phi,U,\phi}(\kappa(J(x_1,..x_n),P,G,F(w_1,..,w_n,G\,a_{11}..a_{1n},..,G\,a_{m1}..a_{mn}))) = 1$$

であるので，Lemma3.2より，

$$\begin{aligned}&\psi_{\mathcal{M},Kk,U,\phi}(\kappa(J(x_1,..x_n),P,G,\\&\qquad F(w_1,...,w_n,G\,a_{11}..a_{1n},..,G\,a_{m1}..a_{mn}))) = 1 \quad \cdots \quad (5)\end{aligned}$$

今，(4) を満たすような任意の t に対し，

$$\kappa((x_1,...,x_n),J(x_1,..x_n),P,[BG,F(w_1,..,w_n,G\,a_{11}..a_{1n},...,G\,a_{m1}..a_{mn})))$$

の，どの様な $(\mathrm{elim})_L$ による割り付けを考えても，それによって定まる

$$F'(w_1,...,w_n,G\,c_{11}..c_{1n},..,G\,c_{m1}..c_{mn})$$

の各 $Gc_{i1}..c_{in}$ で $\psi_{\mathcal{M},Kk,U,\phi}$ による値が1ならば，$\rho_0(\gamma_t(Gc_{i1}..c_{in},\omega_{(n)}))$ の $\psi_{\mathcal{M},Kk,U,\phi}$ による値は1．よって，各 $Ga_{i1}..a_{in}$ を $\rho_0(\gamma_t(Ga_{i1}..a_{in},\omega_{(n)}))$ に変えさらに positive part に現れる $\forall$ を $\forall'$ に negative part に現れる $\exists$ を $\exists'$ に変えたものを F'' とすると，Lemma 3.10により，

$$\psi_{\mathcal{M},Kk,L,\phi}(F'') = 1 \quad \cdots \quad (6)$$

ここで，(1)，Lemma 3.12より，任意の t' について，

$$\psi_{\mathcal{M},K,U,K}(\rho_0(\gamma_{t'}(Gw_1..w_n,\omega_{(n)}))) = 1 \quad \cdots \quad (7)$$

よって，$\rho_0(\gamma_t(Gw_1..w_n,\omega_{(n)}))$ の $\omega_{(n)}$ を，任意の $(a_1,...,a_n)\epsilon U$ に対し，

$$\psi_{\mathcal{M},K,U,K}(J(a_1,...,a_n)) = \psi_{\mathcal{M},K,U,K}(o_{(n)}a_1..a_n)$$

となるような，今までどこにも現れていない $o_{(n)}$ と置き換えたものを $O(t,w_1,...,w_n)$ とおくと，

$$\psi_{\mathcal{M},K,U,K}(O(t,w_1,...,w_n)) = 1$$

これには，$\omega_{(n)}$ が現れていないので，

$$\psi_{\mathcal{M},K,U,\phi}(O(t,w_1,\dots w_n))=1$$

よって，Lemma 3.11，Lemma 3.1より，

$$\psi_{\mathcal{M},Kk\cup K,L,\phi}(O(t,w_1\dots,w_n))=1 \quad \cdots \quad (8)$$

このとき，(6) の $Pa_{i1}..a_{in}$ に $O(t,a_{i1}\dots,a_{in})$ を代入したものを F''' とすると，

$$\psi_{\mathcal{M},Kk\cup K,L,\phi}(F''')=1$$

ここで，F''' の最内に現れる

$$\forall x_1,\dots x_n(J(x_1,\dots,x_n)\Rightarrow O(t,x_1,\dots x_n))\Rightarrow(\forall x[\{y_1,\dots,y_n\}O(t,y_1,\dots y_n)/I_{(n)}]K(x)\wedge\exists xK'(x)$$

の形の subformula を考えると，

$$\psi_{\mathcal{M},Kk\cup K,L,\phi}(\forall x_1,\dots x_n(J(x_1,\dots x_n)\to O(t,x_1,\dots x_n))=1$$

であるから，

$$\begin{aligned}&\psi_{\mathcal{M},Kk\cup K,L,\phi}(\forall x_1,..,x_n(J(x_1,\dots,x_n)\Rightarrow O(t,x_1,\dots,x_n))\\&\qquad\Rightarrow(\forall x[\{y_1,\dots,y_n\}O(t,y_1,\dots y_n)/I_{(n)}]K(x)\wedge\exists xK'(x)))\\&=\psi_{\mathcal{M},Kk\cup K,L,\phi}(\forall x[\{y_1,\dots,y_n\}O(t,y_1,\dots,y_n)/I_{(n)}]K(x)\wedge\exists xK'(x))\end{aligned}$$

さて，今までの置き換えから，

$$\begin{aligned}&\psi_{\mathcal{M},Kk\cup K,L,\phi}(\forall x[\{y_1,\dots y_n\}O(t,y_1\dots y_n)/I_{(n)}]K(x))\\&\quad=\psi_{\mathcal{M},Kk\cup K,L,K}(\forall x[\omega_{(n)}/o_{(n)}][\{y_1,\dots y_n\}O(t,y_1\dots y_n)/I_{(n)}]K(x))\end{aligned}$$

また，

$$K'(x)=[\omega_{(n)}/o_{(n)}]([\{y_1,\dots y_n\}O(t,y_1,\dots y_n)/I_{(n)}]K(x))$$

よって，$\exists'$ の定義より，F''' 最内の

$$\forall x_1,\dots x_n(J(x_1,\dots x_n)\Rightarrow O(t,x_1,\dots,x_n))\Rightarrow(\forall x[\{y_1,\dots y_n\}O(t,y_1,\dots y_n)/I_{(n)}]K(x)\wedge\exists xK'(x)$$

を

$$\exists' xK'(x)$$

に変えたものを F'''' とすると，

$$\psi_{\mathcal{M},Kk\cup K,L,\phi}(F'''')=1$$

この置き換えを繰り返すことにより，

$$\psi_{\mathcal{M},Kk\cup K,L,\phi}(\rho_0(\gamma_{t+1}(Gw_1..w_n,\omega_{(n)})))=1$$

が得られる．これは，$t>s_L$ となる全ての t で成り立つ．

さて，これは任意の $(w_1,..,w_n)\epsilon K$ で成り立つ．また，$(w_1,..,w_n)\epsilon Kk$ のとき，(3)' より OK．よって，$K'=Kk\cup K$ とすれば，

$a_1,\dots,a_n\epsilon K'$ で，$(w_1,\dots,w_n)\epsilon K'$ ならば，U の任意の有限部分集合 L に対し，ある自然数 s が存在し，全ての $t>s$ に対し，

$$\psi_{\mathcal{M},K',L,\phi}(\rho_0(\gamma_t(Gw_1..w_n,\omega_{(n)})))=1$$

OK. □

Theorem 3.2
体系 μ の推論規則は恒真性に対し sound.

Proof. Lemma 3.6，Lemma 3.13により OK. □

4 realizability interpretation

この章では，体系 μ に対する realizability interpretation を定める．これは，formula にどのようなプログラムが対応するかを表現するもので，これにより体系 μ の formula はプログラムの仕様記述言語と見なすことができる．また，realizer extraction rule は，証明から実際にその formula に対応するプログラムを取り出す方法を定めたもので，これにより，証明はその仕様を満たすプログラム生成の手続きと考えることができる．
さて，体系 μ では，U をある集合であるとしか規定していない．これは，より多くの分野で μ を利用できることとするためである．以降，このことからの要請である，「realizer の処理系は U 上の関数は，全て計算可能である（その計算は停止し，かつ計算結果は一意に定まる）」という前提のもとに議論を進める．

4.1 Program Term

Program Term とは，λ Term に入出力の概念を付加した並列関数型プログラム記述言語である．次節で述べる realizer interpretaion はこの Program Term を対象に定義される．

4.1.1 Syntax

Definition 4.1
個体定数，個体変数，object の定義は，体系 μ と同様とする．

Definition 4.2
Program Term の syntax を以下のように帰納的に定める．

- 体系 μ の object は，Program Term における object である．
- a を object，n を自然数とすると，$(if, a = A)$，$proj(n)$ は select function である．
- nop は Program Term である．
- 任意の object は Program Term である．
- Program 変数は，Program Term である．
- Z を Program 変数，A, B, C, D, E を Program Term，s, t を select function，x を object 変数，i, j, n を自然数としたとき，(s, A)，$\lambda_x.A$，$\lambda_Z.A$，$proj(n)A$，(seq, A, B)，$(para, A, B)$，$(nondet, A, B)$，(Π, A)，(in, A)，$(nonin, A)$，(out, a, B)，$(nonout, a, B)$，$((rec, A), \lambda_Z.Za_1..a_n)$，$\mu_Z.A$ は Program Term である．

（ただし，seq,$para$,$proj$,in,$nondet$,$nonin$,out，$nonout$,rec は体系 μ の個体変数，関数とは厳密に区別されるものとする）
また，以下の省略表記を許す．

$$(in_n.A) \stackrel{def}{\equiv} (\overbrace{in, (in, ...(in}^{n}, A)..)), \qquad \lambda_{x_1,...,x_n}.A \stackrel{def}{\equiv} \lambda_{x_1}...\lambda_{x_n}.A$$

Definition 4.3
Program Term で A に現れる object 変数 x で，x を含む subterm に $\lambda_x.B$ の形のものがないとき，x は，a の自由 object 変数という．

Definition 4.4
Program Term で A に現れるプログラム変数 Z で，Z を含む subterm に $\lambda_Z.B$ の形のものがないとき，Z は，a の自由プログラム変数という．

Definition 4.5
Program Term A で A に自由 object 変数も，自由プログラム変数も存在しないなら，A は実行可能な Program Term という．

4.1.2 Operational Semantics

実行可能な Program Term の operational semantics は，λ 計算をもとに定義される．

Definition 4.6
Program Term A が与えられたとき，以下の計算を可能な限り行うとき，計算が終了するなら，その結果得られる Program Term A' を A の semi-Normal Form と呼ぶ．この計算を S 計算と呼ぶ．

1. subterm に $(\lambda_x.A(x))a$ の形のものがあれば，これを $A(a)$ に変える．
2. subterm に $(\lambda_Z.A(Z))B$ の形のものがあれば，これを $A(a)$ に変える．
3. subterm に $proj(j)(A_1, \ldots, A_i)$ の形のものがあるとき，$i > j$ ならこれを $A(j)$ にかえ，$i < j$ ならこれを nop にかえる．これは，他の規則より優先するとする．
4. subterm に U 上で定義されている関数で計算可能なものがある（その Term に個体変数が存在しない）なら，これを計算結果に置き換える．
5. subterm に $(if, a = B)$ があって，A, B に自由に現れる変数がなく，さらに A, B に対し書き換え規則が適用できないとする．このとき，A と B が Syntactical に一致するならこれを $proj(1)$ に，そうでないならこれを $proj(2)$ に変える．これは，他の規則より優先するとする（$proj$ と同じ優先順位）．
6. subterm に $(\mu_{Zx_1..x_n}.A)a_1..a_n$ があるとき，これを $A(\mu_{Zx_1..x_n}.A)a_1..a_n$ に変える．

Lemma 4.1
S 計算は Church Rosser 件を満たす．

Definition 4.7
Program Term A が与えられたとき，以下の計算を可能な限り行うとき，計算が終了するなら，その結果得られる Program Term A' を A の Normal Form と呼ぶ．また，この計算を N 計算と呼ぶ．

1. S 計算を行う．
2. subterm に $(nonout, B, A)$ があるなら，これを A に変える．
3. 上記の計算が終了したとき，subterm に $(proj(n), A, B)$ があるならこれを $proj(n)(A, B)$ にかえ，再び上記の計算を行う．

Lemma 4.2
N 計算は Church Rosser 性を満たす．

Definition 4.8
Program Term A に対し，N 計算がどのような計算順序で実行されても，かならず停止するならば，A は SNN（Storongly N Normarizable）という．

Program Term の Operational Semantics を N 計算を使って定義しよう．Program Term の Operational Semantics は，直観的には以下の様になる．

1. N 計算は，実行毎に可能な限り行う．
2. N 計算が停止したとき，以下の計算を行う．

   - (seq, A, B) ··· A を実行し，その後 B を実行する．
   - $(para, A, B)$ ··· A, B をそれぞれ実行する．
   - $(nondet, A, B)$ ··· A, B のどちらかを（非決定的に）実行する．
   - (II, A) ··· なにもしない（停止する）．
   - (in, A) ··· 入力を行ない，それを a とすると，その後 Aa を実行する．
   - (out, a, A) ··· a を出力し，その後 A を実行する．
   - $((rec, A), B)$ ··· $B(A(rec, A))$ を実行する．
   - 上記以外 ··· 停止する．

この動きを入出力に関する delivation tree を使って定義する．

Definition 4.9
Program Term の Operational Semantics は，以下の入出力に関する delivation Tree により定義される．τ は空入力,a は入力，$\bar{a}$ は出力を示し，$\perp$ は停止状態を示す．同一の入力に対し，複数の遷移が存在する（実際には τ のときのみ）ならばそのどれかが選択されるとする．また，N 計算は遷移毎に行われ，もし N 計算が停止しない（SNN でない）とき，そのノードからの遷移は存在しないとする．

$$\frac{}{(para, A, B) \xrightarrow{\tau} ((para, A, B), r)} \qquad \frac{}{(para, A, B) \xrightarrow{\tau} ((para, A, B), l)}$$

$$\frac{A \xrightarrow{a} A'}{((para, A, B), r) \xrightarrow{a} (para, A', B)} \qquad \frac{B \xrightarrow{a} B'}{((para, A, B), l) \xrightarrow{a} (para, A, B')}$$

$$\frac{}{(para, \perp, B) \xrightarrow{\tau} B} \qquad \frac{}{(para, A, \perp) \xrightarrow{\tau} A}$$

$$\frac{A \xrightarrow{a} A'}{(seq, A, B) \xrightarrow{a} (seq, A', B)} \qquad \frac{}{(seq, \perp, B) \xrightarrow{\tau} B}$$

$$\frac{}{(nondet, A, B) \xrightarrow{\tau} A} \qquad \frac{}{(nondet, A, B) \xrightarrow{\tau} B}$$

$$\frac{}{(in, A) \xrightarrow{a} Aa} \qquad \frac{}{(out, a, A) \xrightarrow{\bar{a}} A}$$

$$\frac{}{((rec, A), B) \xrightarrow{\tau} B(A(rec, A))} \qquad \frac{}{P \xrightarrow{\tau} \perp}$$

（P は上記のどれとも一致しない．）

Definition 4.10
I を無限 Index 集合とする．$i \in I$ に対し，$a_i \in U$ としたとき，$[a_1, a_2, \ldots\ldots]$ を入力列という．また，I が有限のとき有限入力列という．

Definition 4.11
A を Program Term，K を A の delivation Tree とする．K のノードに対し，停止ノードを以下のように定める．

- P が $\perp$ なら，P は停止ノード．
- P から遷移可能な全てのノードが停止ノードなら，P は停止ノード．

Definition 4.12
A を Program Term，K を A の delivation Tree とする．K のノード A が停止ノードのとき，A は TC (Termination without Control) である，もしくは A は停止性を満たすという．

Note 4.1
A が TC なら，A は任意入力列に対しその有限部分で停止する．

Definition 4.13
A を Program Term，K を A の delivation Tree とする．K のノードに対し，弱停止ノードを以下のように定める．

- P が $\perp$ なら P は弱停止ノード.
- P から適当な入力を行えば弱停止ノードに遷移できるなら, P は弱停止ノード.

Definition 4.14
A を Program Term, K を A の delivation Tree とする. K のノード A が弱停止ノードのとき, A は TIC (Termination by Immediate Control) である, もしくは A は弱停止性を満たすという.

Note 4.2
A が TIC なら, A はその動きに関し旨く入力を行えば停止する.

Definition 4.15
A を Program Term, K を A の delivation Tree とする. K のノードに対し, 良停止ノードを以下のように定める.

- 全ての有限入力列に対し, その列で P から達する可能性のあるノードが全て弱停止ノードならば P は良停止ノード.

Definition 4.16
A を Program Term, K を A の delivation Tree とする. K のノード A が良停止ノードのとき, A は TDC (Termination by Delayed Control) である, もしくは A は良停止性を満たすという.

Note 4.3
A が TDC なら, A はどの様に入力を行ったとしても, その後旨く入力を行えば停止する.

Definition 4.17
Program Term A が TDC のとき, $TDC(A)$ と書く.

Definition 4.18
A を Program Term, K を A の delivation Tree とする. K のノードで, $tic(K)$ を以下のように定める.

- $TIC(K)$ なら, $tic(K) = \omega$
- K が TIC で, 長さ n の任意の入力列で達することのできるノードが全て TIC で, 長さ $n+1$ のある入力列で達することができるノードに TIC でないものがあるとき, $tic(K) = n$.
- K が TIC でないとき, $tic(K) = -\omega$.

Lemma 4.3

(a) $TDC(A), TDC(B)$ ならば,

- $TDC(\langle para, A, B\rangle)$
- $TDC(\langle seq, A, B\rangle)$
- $TDC(\langle nondet, A, B\rangle)$

(b) $n < tic(A), tic(B)$ ならば,

- $n < tic(\langle para, A, B\rangle)$
- $n < tic(\langle seq, A, B\rangle)$
- $n < tic(\langle nondet, A, B\rangle)$

4.2 realizability interpretation

Definition 4.19
以下のように体系 μ の自由 object 変数の含まれていない formula に対し，realizability interpretation を定める（e が論理式 F の realizer であることを，(e **realize** F) と表記する）．ここで，$\forall$，$\exists$，$\natural$，$\wedge$ に対し，$\dot{\forall}$，$\dot{\exists}$，$\dot{\natural}$，$\wedge_p$，$\wedge_n$ を導入する．これらの推論規則は $\forall$，$\exists$，$\natural$，$\wedge$ と全く同様で，恒真性では区別しない．また，realizer は全て有限長とし，$Z_{\Psi_{(n)}}$（$\Psi_{(n)}$ は述語変数）をプログラム変数とする．

- e **realize** $a <_i b \Leftrightarrow e = nop \wedge TDC(e)$
- e **realize** $pa_1..a_n \Leftrightarrow e = nop \wedge TDC(e)$
- e **realize** $F \wedge G \Leftrightarrow e = ((seq, e_1, e_2) \wedge (e_1$ **realize** $F) \wedge (e_2$ **realize** $G)) \wedge TDC(e)$
- e **realize** $F \wedge_p G \Leftrightarrow (e = (para, e_1, e_2) \wedge (e_1$ **realize** $F) \wedge (e_2$ **realize** $G)) \wedge TDC(e)$
- e **realize** $F \wedge_n G \Leftrightarrow (e = (nondet, e_1, e_2) \wedge (e_1$ **realize** $F) \wedge (e_2$ **realize** $G)) \wedge TDC(e)$
- e **realize** $F \vee G \Leftrightarrow (e = (r, e_1, e_2) \wedge$
  $((r = \lambda_x.\lambda_y.x \wedge (e_1$ **realize** $F) \vee (r = \lambda_x.\lambda_y.y \wedge (e_2$ **realize** $G))))) \wedge TDC(e)$
- e **realize** $F \Rightarrow G \Leftrightarrow (e = (\Pi, \lambda_x.e_1(x))) \wedge$
  $(if \quad (P$ **realize** $F) \quad then \quad ((\lambda_x.e_1)P$ **realize** $G)) \wedge TDC(e)$
- e **realize** $\neg F \Leftrightarrow e = nop \wedge TDC(e)$
- e **realize** $\forall x F(x) \Leftrightarrow$
  $(e = (in, \lambda_x.e_1(x)) \wedge \forall x((\lambda_x.e_1)x$ **realize** $F(x))) \wedge TDC(e)$
- e **realize** $\dot{\forall} x F(x) \Leftrightarrow$
  $(e = (nonin, \lambda_x.e_1(x)) \wedge \forall x((\lambda_x.e_3(x))x$ **realize** $F(x))))$
  $\vee \quad \forall x(e_1(x)$ **realize** $F(x))) \wedge TDC(e)$
- e **realize** $\exists x F(x) \Leftrightarrow (e = (out, t, e_1(t)) \wedge (e_1(t)$ **realize** $F(t))) \wedge TDC(e)$
- e **realize** $\dot{\exists} x.F(x) \Leftrightarrow (e = (nonout, t, e_1(t))) \wedge (e_1(t)$ **realize** $F(t)) \wedge TDC(e)$
- e **realize** $Ia_1..a_n$
  $\Leftrightarrow (e$ **realize** $G(a_1, \ldots, a_n, Ia_{11}..a_{1n}, \ldots, Ia_{m1}..a_{mn}))$
  または，ある n 引数述語 L に対し，$(e = ((rec, \lambda_{Z_{\Psi_{(n)}}, x_1, ..x_n}.e_1), \lambda_Z.Za_1..a_n)$
  $\wedge \forall (y_1, \ldots y_n) \epsilon L([y_1/x_1, .., y_n/x_n]e_1$ **realize**$_L$
  $\eqsim ((x_1, ..x_n), H(x_1, .., x_n), \Psi_{(n)}, I, G(y_1, .., y_n, Ia_{11}..a_{1n}, .., Ia_{m1}..a_{mn})))) \wedge TDC(e)$
  （ただし，$L = \{(a_1, \ldots, a_n) | \vdash H(x_1, \ldots x_n)\}$
  $I = \mu_{\Psi_{(n)} x_1..x_n}.G(x_1, \ldots x_n, \Psi_{(n)}a_{11}..a_{1n}, \ldots, \Psi_{(n)}a_{m1}..a_{mn})$ とする．）

- e **realize**$_L$ $\Psi_{(n)} \Leftrightarrow e = Z_{\Psi_{(n)}}$
- e **realize**$_L$ $a <_i b \Leftrightarrow e = nop \wedge (*)$
- e **realize**$_L$ $pa_1..a_n \Leftrightarrow e = nop \wedge (*)$
- e **realize**$_L$ $F \wedge G \Leftrightarrow (e = (seq, e_1, e_2) \wedge (e_1$ **realize**$_L$ $F) \wedge (e_2$ **realize**$_L$ $G)) \wedge (*)$
- e **realize**$_L$ $F \wedge_p G \Leftrightarrow (e = (para, e_1, e_2) \wedge (e_1$ **realize**$_L$ $F) \wedge (e_2$ **realize**$_L$ $G)) \wedge (*)$
- e **realize**$_L$ $F \wedge_n G \Leftrightarrow (e = (nondet, e_1, e_2) \wedge (e_1$ **realize**$_L$ $F) \wedge (e_2$ **realize**$_L$ $G)) \wedge (*)$
- e **realize**$_L$ $F \vee G \Leftrightarrow (e = (r, e_1, e_2) \wedge$
  $((r = \lambda_x.\lambda_y.x \wedge (e_1$ **realize**$_L$ $F) \vee (r = \lambda_x.\lambda_y.y \wedge (e_2$ **realize**$_L$ $G))))) \wedge (*)$
- e **realize**$_L$ $F \Rightarrow G \Leftrightarrow (e = (\Pi, \lambda_x.e_1(x))) \wedge$
  $(if \quad (a$ **realize**$_L$ $F) \quad then \quad ((\lambda_x.e_1(x))(a)$ **realize**$_L$ $G)) \wedge (*)$
- e **realize**$_L$ $\neg F \Leftrightarrow e = nop \wedge (*)$
- e **realize**$_L$ $\forall x F(x) \Leftrightarrow$
  $(e = (in, \lambda_x.e_1(x)) \wedge \forall x((\lambda_x.e_3(x))x$ **realize**$_L$ $F(x))))$
  $\vee \quad \forall x(e_1(x)$ **realize**$_L$ $F(x))) \wedge (*)$

- e **realize**$_L$ $\dot{\forall} x F(x) \Leftrightarrow$
  $(e = (nonin, \lambda_x.e_1(x)) \wedge \forall x((\lambda_x.e_3(x))x$ **realize**$_L$ $F(x)))$
  $\qquad\qquad\qquad \vee \quad \forall x(e_1(x)$ **realize**$_L$ $F(x))) \wedge (*)$
- e **realize**$_L$ $\exists x F(x) \Leftrightarrow (e = (out, t, e_1(t)) \wedge (e_1(t)$ **realize**$_L$ $F(t))) \wedge (*)$
- e **realize**$_L$ $\dot{\exists} x.F(x) \Leftrightarrow (e = (nonout, t, e_1(t))) \wedge (e_1(t)$ **realize**$_L$ $F(t)) \wedge (*)$
- e **realize**$_L$ $\natural x((x_1,...x_n), H(x_1,...,x_n), \Psi_{(n)}, I, G(x)) \Leftrightarrow$

  $$(e = (in, \lambda_x.((if, x = k), e_1, e_2)) \wedge (e_1 \textbf{ realize}_L\ G(k)) \wedge (e_2 \textbf{ realize}_L\ [\Psi_{(n)}/I]G(x))) \wedge (*)$$

- e **realize**$_L$ $\dot{\natural} x((x_1,...,x_n), H(x_1,...x_n), \Psi_{(n)}, I, G(x)) \Leftrightarrow$

  $$(e = (nonin, \lambda_x.((if, x = k), e_1, e_2)) \wedge (e_1 \textbf{ realize}_L\ G(k)) \wedge (e_2 \textbf{ realize}_L\ [\Psi_{(n)}/I]G(x))) \wedge (*)$$

- e **realize**$_L$ $Ia_1..a_n$
  $\Leftrightarrow (e$ **realize** $Ia_1..a_n)$

ただし，$I = \mu_{\Psi_{(n)} x_1..x_n}.G(x_1,...,x_n, \Psi_{(n)}a_{11}..a_{1n},...,\Psi_{(n)}a_{m1}..a_{mn})$，また，$(*) = (*)_1 \wedge (*)_2 \wedge (*)_3$ で，それぞれ以下を表す．

$(*)_1$

このformulaに自由述語変数が含まれていないとき，$TDC(e)$．

$(*)_2$

このformulaに自由述語変数が含まれているとき，プログラム変数 $Z_{\Psi_{(n)}}$ をすべて，プログラム e （e はSNN）に置き換えたものを e' とすると，$(b_1,...,b_n) \epsilon L$ を満たす全ての $b_1,...,b_n$ について，

$$tic((e, \lambda_Z.Zb_1..b_n)) = \omega \quad (\Leftrightarrow TDC((e, \lambda_Z.Zb_1..b_n))) \Rightarrow TDC(e')$$

かつ，

$$tic((e, \lambda_Z.Zb_1..b_n)) \neq \omega \wedge tic((e, \lambda_Z.Zb_1..b_n)) \neq -\omega \Rightarrow tic(e') \geq tic((e, \lambda_Z.Zb_1..b_n))$$

特に，自由述語変数が $\natural$，$\dot{\natural}$ の内側以外に現れないとき，

$$tic((e, \lambda_Z.Zb_1..b_n)) \neq \omega \wedge tic((e, \lambda_Z.Zb_1..b_n)) \neq -\omega \Rightarrow tic(e') > tic((e, \lambda_Z.Zb_1..b_n))$$

で，e に関わらず，$TIC(e)$．

$(*_3)$

このFormulaに，

$$< \natural > x((x_1,...x_n), H(x_1,...x_n), \Psi_{(n)}, I, G(x))$$

の形のsubformulaが含まれているとき
$\vdash H(b_1..b_n)$ を満たす全ての $b_1,...,b_n$ に関し，

$$((Z_{\Psi_{(n)}}, \lambda_Z.Zb_1..b_n) \textbf{ realize}_L\ Ib_1..b_n)$$

と仮定すると，このformulaの

$$< \natural > x((x_1,..,x_n), H(x_1,...,x_n), \Psi_{(n)}, p_{(n)}, G(x))$$

の形のsubformulaを最外から順に $< \forall > xG(x)$ に変えたFormula F' に対し，

$$(e \textbf{ realize}_L\ F')$$

（ただし，$< \natural >$ は $\natural$ または $\dot{\natural}$ で，$\natural$ のとき，$< \forall >$ は $\forall$，そうでないとき $\dot{\forall}$ とする．）

Note 4.4
F に自由述語変数が含まれていないとき，

$$(e \textbf{ realize}_L\ F) \Leftrightarrow (e \textbf{ realize } F)$$

4.3 体系 μ の推論規則における realizer extraction rule

体系 μ において $F_1, ..., F_n \vdash F$ が証明されるなら，各 F_i の realizer が与えられれば，それをもとに F の realizer を構成することができる．よって，$\vdash F$ が証明されれば F の realizer を求めることができる．この証明は後節でおこなう．ここでは，実際にこの様な realizer の構成方法を定義する．この方法によって，取り出された Program Term の seminormal form が求めるものとなる．

Definition 4.20
$F_1, ..., F_n \vdash F$ に対し，F_i の realizer を示すプログラム変数を $Rv(F_i)$ と表記することとする．この $Rv(F_i)$ を realizing valiable と呼ぶ．また，F の realizer を $Ext(F)$ と表記する．

Definition 4.21
体系 μ において，formula のみで構成される式 $F_1, ..., F_n \vdash F$ の realizer を以下のように定める（e が式 δ の realizer であることを，(e **realize** δ) と表記する）．ただし，$F_1, ..., F_n \vdash F$ に含まれる自由 object 変数を，$x_1, ..., x_m$ とする．

- e **realize** $F_1, ..., F_n \vdash F \Leftrightarrow \forall x_1, ..., x_m(REAL(n, F_i, e_i) \Rightarrow (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.e)e_n..)e_2)e_1$ **realize** $F)$

また，一般に，式 δ $(= F_1, ..., F_n \vdash F)$ に対し，$F_1, ..., F_n \vdash F$ に含まれる自由 object 変数を，$x_1, ..., x_m$ とすると，

- e **realize**$_L$ $F_1, ... F_n \vdash F \Leftrightarrow$
$$\forall x_1, ... x_m(REAL_L(n, F_i, e_i) \Rightarrow (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.e)e_n..)e_2)e_1 \textbf{ realize}_L\ F)$$

ただし，

$REAL(n, F_i, e_i) \overset{def}{\Leftrightarrow}$
$\quad (e_1 \textbf{ realize } F_1) \wedge ((\lambda_{Rv(F_1)}.e_2)e_1 \textbf{ realize } F_2) \wedge .... \wedge ((\lambda_{Rv(F_1)}.(...(\lambda_{F_{n-1}}.e_n)e_{n-1}...)e_2)e_1 \textbf{ realize } F_n)$
$REAL_L(n, F_i, e_i) \overset{def}{\Leftrightarrow}$
$\quad (e_1 \textbf{ realize}_L\ F_1) \wedge ((\lambda_{Rv(F_1)}.e_2)e_1 \textbf{ realize}_L\ F_2) \wedge ... \wedge ((\lambda_{Rv(F_1)}.(...(\lambda_{F_{n-1}}.e_n)e_{n-1}...)e_2)e_1 \textbf{ realize}_L\ F_n)$

Definition 4.22
体系 μ の証明から realizer を構成する方法を以下のように各推論規則に対し帰納的に定義する．

- $Ext\left(\dfrac{}{\Gamma \vdash F}\right) \overset{def}{\equiv} Rv(F)$（ただし，$\Gamma$ に，F が現れているとする）
- $Ext\left(\dfrac{}{\Gamma \vdash (a_1, ..., a_n) <_{in} (b_1, ..., b_n)}\right) \overset{def}{\equiv} nop$
- $Ext\left(\dfrac{\Gamma \vdash F \quad \Gamma \vdash G}{\Gamma \vdash F < \wedge > G}\right) \overset{def}{\equiv} (< seq >, Ext(\Gamma \vdash F), Ext(\Gamma \vdash G))$
（ただし，$< \wedge >$ が $\wedge_p$ のとき $< seq >$ は $para$，$\wedge$ のとき seq，$\wedge_n$ のとき $nondet$ とする．）
- $Ext\left(\dfrac{\Gamma \vdash F \wedge G}{\Gamma \vdash F}\right) \overset{def}{\equiv} proj(2)Ext(\Gamma \vdash F \wedge G)$
- $Ext\left(\dfrac{\Gamma \vdash F \wedge G}{\Gamma \vdash G}\right) \overset{def}{\equiv} proj(3)Ext(\Gamma \vdash F \wedge G)$
- $Ext\left(\dfrac{\Gamma \vdash F}{\Gamma \vdash F \vee G}\right) \overset{def}{\equiv} (proj(1), (Ext(\Gamma \vdash F), nop))$
- $Ext\left(\dfrac{\Gamma \vdash G}{\Gamma \vdash F \vee G}\right) \overset{def}{\equiv} (proj(2), (nop, Ext(\Gamma \vdash F)))$

- $Ext\left(\dfrac{\Gamma \vdash F \vee G \quad \Gamma, F \vdash H \quad \Gamma, G \vdash H}{\Gamma \vdash H}\right)$
  $\stackrel{def}{\equiv} proj(1)Ext(\Gamma \vdash F \vee G)((\lambda_{Rv(F)}.Ext(\Gamma, F \vdash H))(proj(2)Ext(\Gamma \vdash F \vee G)))$
  $\qquad ((\lambda_{Rv(G)}.Ext(\Gamma, G \vdash H))(proj(3)Ext(\Gamma \vdash F \vee G)))$
- $Ext\left(\dfrac{\Gamma, F \vdash G}{\Gamma \vdash F \Rightarrow G}\right) \stackrel{def}{\equiv} (\Pi, \lambda_{Rv(F)}.Ext(\Gamma, F \vdash G))$
- $Ext\left(\dfrac{\Gamma \vdash F \Rightarrow G \quad \Gamma \vdash F}{\Gamma \vdash G}\right) \stackrel{def}{\equiv} (proj(2)Ext(\Gamma \vdash F \Rightarrow G))Ext(\Gamma \vdash F)$
- $Ext\left(\dfrac{\Gamma, a \downarrow \vdash F(x)}{\Gamma \vdash < \forall > xF(x)}\right) \stackrel{def}{\equiv} (< in >, \lambda_x.Ext(\Gamma \vdash F(x)))$
  （ただし，$< \forall >$が$\forall$のとき$< in >$はin，$< \forall >$が$\dot{\forall}$のとき$< in >$は$nonin$とする．）

- $Ext\left(\dfrac{\Gamma \vdash \forall xF(x) \quad \Gamma \vdash t \downarrow}{\Gamma \vdash F(t)}\right) \stackrel{def}{\equiv} (proj(2)Ext(\Gamma \vdash \forall xF(x)))t$
- $Ext\left(\dfrac{\Gamma \vdash F(t) \quad \Gamma \vdash t \downarrow}{\Gamma \vdash < \exists > xF(x)}\right) \stackrel{def}{\equiv} (< out >, t, Ext(\Gamma \vdash F(t)))$
  （ただし，$< \exists >$が$\exists$のとき$< out >$はout，$< \exists >$が$\dot{\exists}$のとき$< out >$は$nonout$とする．）
- $Ext\left(\dfrac{\Gamma \vdash \exists xF(x) \quad \Gamma, F(t) \vdash G}{\Gamma \vdash G}\right) \stackrel{def}{\equiv} (\lambda_{Rv(F(t))}.Ext(\Gamma, F(t) \vdash G))(proj(3)Ext(\Gamma \vdash \exists xF(x)))$
- $Ext\left(\dfrac{\Gamma, F \vdash \bot}{\Gamma \vdash \neg F}\right) \stackrel{def}{\equiv} nop$
- $Ext\left(\dfrac{\Gamma, \forall y_1, \dots y_n((y_1, \dots, y_n) <_{jn} (x_1, \dots, x_n) \Rightarrow F(y_1, \dots, y_n)) \vdash F(x_1, \dots, x_n)}{\Gamma \vdash < \forall > x_1, \dots, x_n F(x_1, \dots x_n)}\right)$
  $\stackrel{def}{\equiv} (< in >_1, (\lambda_{x_1}.(< in >_2, (\dots(\lambda_{x_{n-1}}.(< in >_n, (\lambda_{x_n}.$
  $\qquad (\mu z_{x_1..x_n}.(\lambda_Z.\lambda_{x_1..x_n}.((\lambda_{Rv(\xi)}.Ext(\zeta))(\lambda_{y_1..y_n}\lambda_x.Zy_1..y_n)))x_1..x_n)))))))$
  （ただし，$\xi = \forall y_1, \dots y_n.((y_1, \dots y_n) <_{jn} (x_1, \dots x_n) \Rightarrow F(y_1, \dots, y_n))$
  $\zeta = \Gamma, \forall y_1, ..y_n.((y_1, \dots, y_n) <_{jn} (x_1, \dots, x_n) \Rightarrow F(y_1, \dots y_n)) \vdash F(x_1, \dots, x_n)$，$< \forall >$の$n$番目を$< \forall >_n$のすると，$< \forall >_n$が$\dot{\forall}$のとき$< in >_i$は$nonin$，そうでなければ$in$とする．）

- $Ext\left(\dfrac{\Gamma \vdash G(b_1, \dots b_n, Ia_{11}..a_{1n}, \dots Ia_{m1}..a_{mn})}{\Gamma \vdash Ib_1..b_n}\right)$
  $\stackrel{def}{\equiv} Ext(\Gamma \vdash G(b_1, \dots, b_n, Ia_{11}..a_{1n}, \dots Ia_{m1}..a_{mn}))$
  （ただし，$I = \mu_{\Psi_{(n)}x_1..x_n}.G(x_1, \dots, x_n, \Psi a_{11}..a_{1n}, \dots, \Psi a_{m1}..a_{mn})$とする．）

- $Ext\left(\dfrac{\Gamma \vdash Ib_1..b_n}{\Gamma \vdash G(b_1, \dots b_n, Ia_{11}..a_{1n}, \dots Ia_{m1}..a_{mn})}\right)$
  $\stackrel{def}{\equiv} ((if, (proj(1), Ext(\xi)) = rec), (proj(2)Ext(\xi))((proj(2)(proj(1)Ext(\xi)))(proj(1)Ext(\xi))), Ext(\xi))$

  （ただし，$I = \mu_{\Psi_{(n)}x_1..x_n}.G(x_1, \dots, x_n, \Psi a_{11}..a_{1n}, \dots, \Psi a_{m1}..a_{mn})$
  $\xi = \Gamma \vdash Ib_1..b_n$とする．）
- $Ext\left(\dfrac{\Gamma \vdash \exists xG(x) \quad \Gamma, \forall x_1, \dots, x_n(H(x_1, \dots, x_n) \Rightarrow \Psi_{(n)}x_1..x_n) \vdash \forall x[\Psi_{(n)}/p_{(n)}]G(x)}{\Gamma \vdash < \natural > x((x_1, \dots x_n), H(x_1, \dots x_n), \Psi_{(n)}, p_{(n)}, G(x))}\right)$
  $\stackrel{def}{\equiv} (< in >, \lambda_x.((if, x = (proj(2)Ext(\varsigma)))$
  $\qquad (proj(3)Ext(\varsigma), (\lambda_{Rv(\zeta)}.Ext(\xi))(\lambda_{x_1..x_n}\lambda_x.(Z_{\Psi_{(n)}}, \lambda_Z.Zx_1..x_n)))))$
  （ただし，$< \natural >$が$< \dot{\natural} >$のとき$< in >$は$nonin$，$\natural$のとき$< in >$はinで，
  $\varsigma = \Gamma \vdash \exists xG(x)$
  $\xi = \Gamma, \forall x_1, ..x_n(H(x_1, \dots, x_n) \Rightarrow \Psi_{(n)}x_1..x_n) \vdash \forall x[\Psi_{(n)}/p_{(n)}]G(x)$
  $\zeta = \forall x_1, ..x_n(H(x_1, \dots, x_n) \Rightarrow \Psi_{(n)}x_1..x_n)$とする．）

- $Ext\left(\dfrac{\Gamma, H(y_1, \dots, y_n) \vdash \varpi((x_1, \dots, x_n), H(x_1, \dots x_n), \Psi_{(n)}, I, K(y_1, \dots y_n)) \quad \Gamma \vdash H(a_1, \dots, a_n)}{\Gamma \vdash Ia_1..a_n}\right)$
  $\stackrel{def}{\equiv} ((rec, \lambda_{Z_{\Psi_{(n)}}, y_1, \dots y_n}.(\lambda_{Rv(\xi)}.Ext(\varsigma))nop), (\lambda_Z.Za_1..a_n))$

（ただし，$I = \mu_{\Psi_{(n)}, x_1 \ldots x_n}.G(x_1, \ldots, x_n, \Psi a_{11}..a_{1n}, \ldots, \Psi a_{m1}..a_{mn})$
$K(y_1, \ldots, y_n) = G(y_1, \ldots, y_n, I a_{11}..a_{1n}, \ldots, I a_{m1}..a_{mn})$
$\varsigma = \Gamma. H(y_1, \ldots, y_n) \vdash \varpi((x_1, \ldots, x_n), H(x_1, \ldots, x_n), \Psi_{(n)}, I_{(n)}, K(y_1, \ldots, y_n))$
$\xi = H(y_1, \ldots, y_n)$ とする．）

- $Ext\left(\frac{}{\Gamma \vdash a = a}\right) \stackrel{def}{=} nop$
- $Ext\left(\frac{\Gamma \vdash a = b \quad \Gamma \vdash F(a)}{\Gamma \vdash F(b)}\right) \stackrel{def}{=} Ext(\Gamma \vdash F(a))$
- $Ext\left(\frac{\Gamma \vdash a < b \quad \Gamma \vdash b < c}{\Gamma \vdash a < b}\right) \stackrel{def}{=} nop$

Note 4.5
realizer extraction rule で導かれた $Ext(\Gamma \vdash F)$ には，$\Gamma \vdash F$ に現れない Object 自由変数は現れない．

4.4 Soundness

本節では，前節で与えた realizer extraction rule が realizer に対し，sound であることをいう．

Theorem 4.1
体系 μ の推論規則に対し，上式の正しい realizer が与えられれば，realizer 構成規則で構成される realizer は正しい．

Proof. 証明図の大きさに関する帰納法で証明する．以下を示せば良い．

(a) Mu-Intro,Natural-Intro を除く推論規則に対し，

i. 上式の数を m としたとき，上式をそれぞれ δ_i $(0 < i < m+1)$，下式を δ，各 $Ext(\delta_i)$ から realizer extraction rule により構成される Program Term を P としたとき，

$$\forall i \in \{1, 2, .., m\}((Ext(\delta_i) \textbf{ realize } \delta_i) \Rightarrow (P \textbf{ realize } \delta))$$

ii. 上式の数を m としたとき，δ_i $(0 < i < m+1)$，下式を δ，各 $Ext(\delta_i)$ から realizer extraction rule により構成される Program Term を P としたとき，全ての L に対し，

$$\forall i \in \{1, 2, .., m\}((Ext(\delta_i) \textbf{ realize}_L\ \delta_i) \Rightarrow (P \textbf{ realize}_L\ \delta))$$

(b) Mu-Intro の推論規則に対し，上式を δ_1, δ_2，下式を δ，各 $Ext(\delta_i)$ から realizer extraction rule により構成される Program Term を P としたとき，ある L に対し，

$$\forall i \in \{1, 2\}((Ext(\delta_i) \textbf{ realize}_L\ \delta_i) \Rightarrow (P \textbf{ realize } \delta))$$

(c) Natural-Intro の推論規則に対し，上式を δ_1, δ_2，下式を δ，各 $Ext(\delta_i)$ から realizer extraction rule により構成される Program Term を P としたとき，ある L に対し，

$$\forall i \in \{1, 2\}((Ext(\delta_i) \textbf{ realize}_L\ \delta_i) \Rightarrow (P \textbf{ realize}_L\ \delta))$$

以下にそれぞれの証明を記す．

(a) i，ii をそれぞれ証明する．

- Axiom
  realizer extraction rule，realizer の定義より明らか．OK．
- Partial Intro
  realizer extraction rule，realizer の定義より明らか．OK．

- And-Intro

(a) $\wedge$

Γ を $F_1, .., F_n$, $\Gamma \vdash F$, $\Gamma \vdash G$ に含まれる object 自由変数を $x_1, .., x_m$ とする.

- i

仮定より,

$$\forall x_1, .., x_m(Real(n, F_i, e_i) \Rightarrow (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F))e_n...)e_2)e_1 \textbf{ realize } F) ....(1)$$

$$\forall x_1, .., x_m(Real(n, F_i, e_i) \Rightarrow (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash G))e_n...)e_2)e_1 \textbf{ realize } G) ....(2)$$

realizer extraction rule より,

$$Ext(\Gamma \vdash F \wedge G) = (seq, Ext(\Gamma \vdash F), Ext(\Gamma \vdash G))$$

ここで,

$$\begin{aligned}&\forall x_1, ..., x_m(Real(n, F_i, e_i) \Rightarrow \\ &\quad (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(seq, Ext(\Gamma \vdash F), Ext(\Gamma \vdash G)))e_n...)e_2)e_1 \\ &\qquad = (seq, (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F))e_n...)e_2)e_1, \\ &\qquad\qquad (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash G))e_n...)e_2)e_1))\end{aligned}$$

であるから, (1), (2) より,

$$\begin{aligned}&\forall x_1, ..., x_m(Real(n, F_i, e_i) \Rightarrow (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(seq, Ext(\Gamma \vdash F), Ext(\Gamma \vdash G)))e_n...)e_2)e_1 \\ &\qquad = (seq, d_1, d_2) \wedge (d_1 \textbf{ realize } F) \wedge (d_2 \textbf{ realize } G))\end{aligned}$$

$x_1, ..., x_m$ を任意の Object とすると, 仮定より,

$$\begin{aligned}&Real(n, F_i, e_i) \Rightarrow \\ &\qquad TDC((..((\lambda_{Rv(F_1)}..._{Rv(F_n)}.Ext(\Gamma \vdash F))e_n...)e_2)e_1) \wedge \\ &\qquad\quad TDC((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash G))e_n...)e_2)e_1)\end{aligned}$$

であるので, Lemma 4.3より,

$$\begin{aligned}&Real(n, F_i, e_i) \Rightarrow \\ &\qquad TDC((seq, (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F))e_n...)e_2)e_1, \\ &\qquad\qquad (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash G))e_n...)e_2)e_1))\end{aligned}$$

よって OK.

ii

仮定より,

$$\forall x_1, ..., x_m(Real_L(n, F_i, e_i) \Rightarrow (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F))e_n...)e_2)e_1 \textbf{ realize}_L\ F) ....(1)$$

$$\forall x_1, ..., x_m(Real_L(n, F_i, e_i) \Rightarrow (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash G))e_n...)e_2)e_1 \textbf{ realize}_L\ G) ....(2)$$

realizer extraction rule より,

$$Ext(\Gamma \vdash F \wedge G) = (seq, Ext(\Gamma \vdash F), Ext(\Gamma \vdash G))$$

ここで，

$$\begin{aligned}\forall x_1, \ldots x_m(Real_L(n, F_i, e_i) \Rightarrow &(\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.(seq, Ext(\Gamma \vdash F), Ext(\Gamma \vdash G)))e_n \ldots)e_2)e_1 \\ &= (seq, (\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F))e_n \ldots)e_2)e_1, \\ &\quad (\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash G))e_n \ldots)e_2)e_1))\end{aligned}$$

であるから，(1)，(2) より，

$$\begin{aligned}\forall x_1, \ldots, x_m(Real_L(n, F_i, e_i) \Rightarrow &(\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.(seq, Ext(\Gamma \vdash F), Ext(\Gamma \vdash G)))e_n \ldots)e_2)e_1 \\ &= (seq, d_1, d_2) \wedge (d_1 \ \mathbf{realize}_L \ F) \wedge (d_2 \ \mathbf{realize}_L \ G)) \qquad \ldots(3)\end{aligned}$$

さて，

- $F \wedge G$ が自由述語変数を含んでいないとき

  F も G も自由述語変数を含んでいないので，仮定より，$Real_L(n, F_i, e_i)$ のとき，$x_1, \ldots, x_m$ を任意の Object とすると，

  $$TDC((\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F))e_n \ldots)e_2)e_1),$$
  $$TDC((\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash G))e_n \ldots)e_2)e_1)$$

  であるので，Lemma 4.3より，

  $$\begin{aligned}&Real_L(n, F_i, e_i) \Rightarrow \\ &\quad TDC((seq, (\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F))e_n)\ldots)e_n, \\ &\qquad (\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash G))e_n \ldots)e_2)e_1))\end{aligned}$$

  よって OK.

- $F \wedge G$ に自由述語変数が含まれているとき

$x_1, \ldots, x_m$ を任意の Object とし，

$$(\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F))e_n \ldots)e_2)e_1)$$

$$(\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash G))e_n \ldots)e_2)e_1$$

$$(\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash G))e_n \ldots)e_2)e_1))$$

のプログラム変数 $Z_{\Psi(x)}$ をすべて，プログラム e に置き換えたものを $e'_F, e'_G, e'_{G\wedge}$ とすると，仮定より，

$Real_L(n, F_i, e_i)$ のとき，$(b_1, \ldots, b_u) \epsilon L$ を満たす全ての $b_1, \ldots, b_u$ について，

$$tic((e, \lambda_Z.Zb_1..b_u)) = \omega \Rightarrow TDC(e'_F)$$

$$tic((e, \lambda_Z.Zb_1..b_u)) = \omega \Rightarrow TDC(e'_G)$$

$$tic((e, \lambda_Z.Zb_1..b_u)) \neq \omega \wedge tic((e, \lambda_Z.Zb_1..b_u)) \neq \omega \Rightarrow tic(e'_F) \geq tic((e, \lambda_Z.Zb_1..b_u))$$

$$tic((e, \lambda_Z.Zb_1..b_u)) \neq \omega \wedge tic((e, \lambda_Z.Zb_1..b_u)) \neq -\omega \Rightarrow tic(e'_G) \geq tic((e, \lambda_Z.Zb_1..b_u))$$

よって，Lemma 4.3，(3) より，

$$tic((e, \lambda_Z.Zb_1..b_u)) = \omega \quad (\Leftrightarrow TDC((e, \lambda_Z.Zb_1..b_u))) \Rightarrow TDC(e'_G)$$

$$tic((e, \lambda_Z.Zb_1..b_u)) \neq \omega \wedge tic((e, \lambda_Z.Zb_1..b_u)) \neq -\omega \Rightarrow tic(e'_F) \geq tic((e, \lambda_Z.Zb_1..b_u))$$

また，自由変数が⌊，⌋の内側以外に現れないとき，F，G おいても，自由変数が⌊，⌋の内側以外に現れないので，同様に，

$$tic((e, \lambda_Z.Zb_1..b_u)) \neq \omega \wedge tic((e, \lambda_Z.Zb_1..b_u)) \neq -\omega \Rightarrow tic(e'_F) > tic((e, \lambda_Z.Zb_1..b_u))$$

- この Formula に、

$$<\natural> x((x_1,...,x_u), H(x_1,...,x_u), \Psi_{(u)}, I, J(x))$$

の形の subformula が含まれているとき（$<\natural>$ は、$\natural$ もしくは、$\dot{\natural}$）
F,G の少なくともどちらか一方はこの形の subformula を含んでいる．今，

$\vdash H(b_1..b_u)$ を満たす全ての $b_1,..,b_u$ に関し，

$$((Z_{\Psi_{(u)}}, \lambda_Z.Zb_1..b_u))\ \mathbf{realize}_L\ \ Ib_1..b_u)$$

と仮定し，この F,G の

$$<\natural> x((x_1,...,x_u), H(x_1,..,x_u), \Psi_{(u)}, p_{(u)}, J(x))$$

の形の subformula を最外から順に $<\forall> xJ(x)$ に変えた Formula F',G' を考えると，仮定より（F，G でもし $\natural$，$\dot{\natural}$ を含んでなえれば $F=F'$，$G=G'$ であるので），$x_1,...,x_m$ を任意の Object とすると，

$$Real_L(n,F_i,e_i) \Rightarrow ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma\vdash F))e_n...)e_2)e_1)\ \mathbf{realize}_L\ \ F')$$

$$Real_L(n,F_i,e_i) \Rightarrow ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma\vdash G))e_n...)e_2)e_1\ \mathbf{realize}_L\ \ G')$$

（ただし，$<\natural>$ は $\natural$ または，$\dot{\natural}$ で，$\natural$ のとき，$<\forall>$ は $\forall$，そうでないとき $\dot{\forall}'$ とする．）
よって，(3) より，OK.

(b) $\wedge_p$、$\wedge_n$
$\wedge$ と同様．OK.

- And-Elim 1

(a) $\wedge$
Γ を $F_1,...,F_n$、$\Gamma\vdash F\wedge G$ に含まれる object 自由変数を $x_1,...,x_m$，$\Gamma\vdash F$ に含まれる object 自由変数を $x_1,...,x_s$ $(s\leq m)$ とする．

- i
仮定より，

$$\forall x_1,...,x_m(Real(n,F_i,e_i) \Rightarrow$$
$$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma\vdash F\wedge G))e_n...)e_2)e_1\ \mathbf{realize}\ F\wedge G) \qquad ......(1)$$

realizer extraction rule より，

$$Ext(\Gamma\vdash F) = proj(2)(Ext(\Gamma\vdash F))$$

ここで，F に含まれる object 自由変数は $x_1,...x_s$ だから，Note 4.5より，

$$\forall x_1,..,x_s(Real(n,F_i,e_i) \Rightarrow (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.proj(2)(\Gamma\vdash F\wedge G))e_n...)e_2)e_1$$
$$= proj(2)((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma\vdash F\wedge G))e_n...)e_2)e_1))$$

よって，(1)，$\wedge$ の realizer の定義より，
$\forall x_1,...x_s(Real(n,F_i,e_i) \Rightarrow$
$$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.proj(2)(\Gamma\vdash F\wedge G))e_n...)e_2)e_1 = d\wedge(d\ \mathbf{realize}\ F))$$

よって OK.

- ii
仮定より，

$$\forall x_1,...,x_m(Real_L(n,F_i,e_i) \Rightarrow$$
$$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F \wedge G))e_n...)e_2)e_1 \textbf{ realize}_L\ F \wedge G) \qquad ......(1)$$

realizer extraction rule より,

$$Ext(\Gamma \vdash F) = proj(2)(Ext(\Gamma \vdash F))$$

ここで, F に含まれる object 自由変数は $x_1,...,x_s$ だから, Note 4.5より,

$$\forall x_1,...,x_s(Real_L(n,F_i,e_i) \Rightarrow$$
$$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.proj(2)(\Gamma \vdash F \wedge G))e_n...)e_2)e_1$$
$$= proj(2)((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F \wedge G))e_n...)e_2)e_1))$$

よって, (2), $\wedge$ の realizer の定義より,

$$\forall x_1,...x_s(Real_L(n,F_i,e_i) \Rightarrow$$
$$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.proj(2)(\Gamma \vdash F \wedge G))e_n...)e_2)e_1 = d \wedge (d \textbf{ realize}_L\ F))$$

よって OK.

(b) $\wedge_p$, $\wedge_n$

$\wedge$ と同様. OK.

- And-Elim 2

And-Elim 1 と同様. OK.

- Or-Intro 1

Γ を $F_1,...F_n$, $\Gamma \vdash F \vee G$ に含まれる object 自由変数を $x_1,...x_m$, $\Gamma \vdash F$ に含まれる object 自由変数を $x_1,...,x_s$ $(s \leq m)$ とする.

i

仮定より,

$$\forall x_1,...x_s(Real(n,F_i,e_i) \Rightarrow (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F))e_n...)e_2)e_1 \textbf{ realize } F) ......(1)$$

realizer extraction rule より,

$$Ext(\Gamma \vdash F \vee G) = (proj(1),(Ext(\Gamma \vdash F),nop))$$

よって, (1), $\vee$ の realizer の定義より,

$$\forall x_1,...x_s(Real(n,F_i,e_i) \Rightarrow (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(proj(1),(Ext(\Gamma \vdash F \vee G),nop)))e_n...)e_2)e_1$$
$$= (proj(1),((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F))e_n...)e_2)e_1,nop))$$

ここで, (1), $\vee$ の realizer の定義より,

$$\forall x_1,...x_s(Real(n,F_i,e_i) \Rightarrow$$
$$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(proj(1),(Ext(\Gamma \vdash F),nop)))e_n...)e_2)e_1$$
$$= (proj(1),(d,nop)) \wedge (d \textbf{ realize } F)$$

また, Note 4.5より, $(proj(1),((\lambda_{Rv(F_1)..Rv(F_n)}.Ext(\Gamma \vdash F))e_1..e_n,nop))$ には $x_{s+1},...,x_m$ は含まれないので,

$$\forall x_1,...,x_m(Real(n,F_i,e_i) \Rightarrow$$
$$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(proj(1),(Ext(\Gamma \vdash F \wedge G),nop)))e_n...)e_2)e_1$$
$$= (proj(1),(d,nop)) \wedge (d \textbf{ realize } F)$$

ここで, $(proj(1),(d,nop))$ の N 計算の結果は, d の N 計算の結果と一致する. よって, 仮定よ

り $TDC(proj(1),(d,nop))$.
OK.

- ii

仮定より，

$$\forall x_1, \ldots x_s(Real_L(n, F_i, e_i) \Rightarrow (\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F))e_n \ldots)e_2)e_1 \ \mathbf{realize}_L\ F) \ldots\ldots(1)$$

realizer extraction rule より，

$$Ext(\Gamma \vdash F \vee G) = (proj(1).(Ext(\Gamma \vdash F), nop))$$

ここで，(1)，$\vee$ の realizer の定義より，

$$\begin{aligned}\forall x_1, \ldots, x_s(Real_L(n, F_i, e_i) &\Rightarrow (\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.(proj(1).(Ext(\Gamma \vdash F \vee G), nop)))e_n \ldots)e_2)e_1 \\ &= (proj(1).((\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F))e_n \ldots)e_2)e_1, nop))\end{aligned}$$

よって，(1)，$\vee$ の realizer の定義より，

$$\begin{aligned}&\forall x_1, \ldots x_s(Real_L(n, F_i, e_i) \Rightarrow \\ &\qquad (\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.(proj(1).(Ext(\Gamma \vdash F), nop)))e_n \ldots)e_2)e_1 \\ &\qquad\quad = (proj(1).(d, nop)) \wedge (d\ \mathbf{realize}_L\ F)\end{aligned}$$

また，Note 4.5より，$(proj(1).((\lambda_{Rv(F_1)\ldots Rv(F_n)}.Ext(\Gamma \vdash F))e_1..e_n, nop))$ には $x_{s+1}, \ldots x_m$ は含まれないので，

$$\begin{aligned}&\forall x_1, \ldots x_m(Real_L(n, F_i, e_i) \Rightarrow \\ &\qquad (\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.(proj(1).(Ext(\Gamma \vdash F \wedge G), nop)))e_n \ldots)e_2)e_1 \\ &\qquad\quad = (proj(1).(d, nop)) \wedge (d\ \mathbf{realize}_L\ F)\end{aligned}$$

ここで，$(proj(1).(d, nop)$ の N 計算の結果は，d の N 計算の結果と一致する．よって仮定より (*) も OK.
OK.

- Or-Intro 2
  Or-Intro 1 と同様．OK.
- Or-Elim
  Γ を $F_1, \ldots, F_n$，Γ に含まれる object 自由変数を $m_1, \ldots, m_i$，F に含まれる object 自由変数を $f_1, \ldots, f_j$，G に含まれる object 自由変数を $g_1, \ldots, g_k$，H に含まれる object 自由変数を $h_1, \ldots, h_l$ とする．

  - i

    仮定より，

$$\begin{aligned}&\forall m_1, \ldots, m_i, f_1, \ldots, f_j, g_1, \ldots, g_k(Real(n, F_i, e_i) \Rightarrow \\ &\qquad (\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F \wedge G))e_n \ldots)e_2)e_1 \ \mathbf{realize}\ F \wedge G) \qquad \ldots\ldots(1)\end{aligned}$$

$$\begin{aligned}&\forall m_1, \ldots, m_i, f_1, \ldots f_j, h_1, \ldots h_l(Real(n, F_i, e_i) \wedge ((\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.e_F)e_n \ldots)e_2)e_1 \ \mathbf{realize}\ F) \\ &\qquad \Rightarrow ((\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.(\lambda_{Rv(F)}.Ext(\Gamma, G \vdash H))e_F)e_n \ldots)e_2)e_1 \ \mathbf{realize}\ H)) \ldots\ldots(2)\end{aligned}$$

$$\begin{aligned}&\forall m_1, \ldots m_i, f_1, \ldots, f_j, h_1, \ldots, h_l(Real(n, F_i, e_i) \wedge ((\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.e_G)e_n \ldots)e_2)e_1 \ \mathbf{realize}\ F) \\ &\qquad \Rightarrow ((\lambda_{Rv(F_1)}.(\ldots(\lambda_{Rv(F_n)}.(\lambda_{Rv(G)}.Ext(\Gamma, G \vdash H))e_G)e_n \ldots)e_2)e_1 \ \mathbf{realize}\ H)) \ldots\ldots(3)\end{aligned}$$

realizer extraction rule より，

$$Ext(\Gamma \vdash H) = (proj(1)Ext(\Gamma \vdash F \vee G))(((\lambda_{Rv(F)}.Ext(\Gamma, F \vdash H))(proj(2)Ext(\Gamma \vdash F \vee G))),\\ ((\lambda_{Rv(G)}.Ext(\Gamma, G \vdash H))(proj(3)Ext(\Gamma \vdash F \vee G))))$$

ここで,

$$\forall m_1,..,m_i,f_1,..,f_j,g_1,...,g_k,h_1,...,h_l(Real(n,F_i,e_i) \Rightarrow\\ (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.proj(1)Ext(\Gamma \vdash F \vee G)(((\lambda_{Rv(F)}.Ext(\Gamma, F \vdash H))\\ (proj(2)Ext(\Gamma \vdash F \vee G))),((\lambda_{Rv(G)}.Ext(\Gamma, G \vdash H))(proj(3)Ext(\Gamma \vdash F \vee G)))))e_n...)e_2)e_1\\ = (proj(1)((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F \vee G))e_1..)e_2)e_1))(\\ ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\lambda_{Rv(F)}.Ext(\Gamma, F \vdash H))(proj(2)Ext(\Gamma \vdash F \vee G)))e_1...)e_2)e_1),\\ ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\lambda_{Rv(G)}.Ext(\Gamma, G \vdash H))(proj(3)(Ext(\Gamma \vdash F \vee G))))e_1...)e_2)e_1)))$$

よって,(1),$\vee$ の realizer の定義より,

$$\forall m_1,...,m_i,f_1,...,f_j,g_1,...,g_k,h_1,...,h_l(Real(n,F_i,e_i) \Rightarrow\\ ((Ext(\Gamma \vdash H) = proj(1)(((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\lambda_{Rv(F)}.Ext(\Gamma, F \vdash H))e')e_1...)e_2)e_1),\\ ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\lambda_{Rv(G)}.Ext(\Gamma, G \vdash H))nop)e_1,..)e_2)e_1)))\\ \vee(Ext(\Gamma \vdash H) = proj(2)(((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\lambda_{Rv(F)}.Ext(\Gamma, F \vdash H))nop)e_1...)e_2)e_1),\\ ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\lambda_{Rv(G)}.Ext(\Gamma, G \vdash H))e'')e_1...)e_2)e_1)))))$$

よって,(2),(3) より OK.

- ii

仮定より,

$$\forall m_1,...,m_i,f_1,...,f_j,g_1,...,g_k(Real_L(n,F_i,e_i) \Rightarrow\\ (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F \wedge G))e_n...)e_2)e_1 \ \mathbf{realize}_L \ F \wedge G) \qquad ......(1)$$

$$\forall m_1,...,m_i,f_1,...,f_j,h_1,...,h_l(Real_L(n,F_i,e_i) \wedge\\ ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.e_F)e_n...)e_2)e_1 \ \mathbf{realize}_L \ F)\\ \Rightarrow ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\lambda_{Rv(F)}.Ext(\Gamma, G \vdash H))e_F)e_n..)e_2)e_1 \ \mathbf{realize}_L \ H))......(2)$$

$$\forall m_1,...,m_i,f_1,...,f_j,h_1,...,h_l(Real_L(n,F_i,e_i) \wedge ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.e_G)e_n...)e_2)e_1 \ \mathbf{realize}_L \ F)\\ \Rightarrow ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\lambda_{Rv(G)}.Ext(\Gamma, G \vdash H))e_G)e_n..)e_2)e_1 \ \mathbf{realize}_L \ H))......(3)$$

realizer extraction rule より,

$$Ext(\Gamma \vdash H) = (proj(1)Ext(\Gamma \vdash F \vee G))(((\lambda_{Rv(F)}.Ext(\Gamma, F \vdash H))(proj(2)Ext(\Gamma \vdash F \vee G))),\\ ((\lambda_{Rv(G)}.Ext(\Gamma, G \vdash H))(proj(3)Ext(\Gamma \vdash F \vee G))))$$

ここで,

$$\forall m_1,...,m_i,f_1,...,f_j,g_1,...,g_k,h_1,...,h_l(Real_L(n,F_i,e_i) \Rightarrow\\ (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.proj(1)Ext(\Gamma \vdash F \vee G)(((\lambda_{Rv(F)}.Ext(\Gamma, F \vdash H))\\ (proj(2)Ext(\Gamma \vdash F \vee G))),((\lambda_{Rv(G)}.Ext(\Gamma, G \vdash H))(proj(3)Ext(\Gamma \vdash F \vee G)))))e_n...)e_2)e_1\\ = (proj(1)((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F \vee G))e_1..)e_2)e_1))(\\ ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\lambda_{Rv(F)}.Ext(\Gamma, F \vdash H))(proj(2)Ext(\Gamma \vdash F \vee G)))e_1...)e_2)e_1),\\ ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\lambda_{Rv(G)}.Ext(\Gamma, G \vdash H))(proj(3)(Ext(\Gamma \vdash F \vee G))))e_1...)e_2)e_1)))$$

よって，(1)，$\vee$ の realizer の定義より，

$$
\begin{aligned}
&\forall m_1, ..., m_i, f_1, ... f_j, g_1, ... g_k, h_1, ... h_l(Real_L(n, F_i, e_i) \Rightarrow \\
&\qquad ((Ext(\Gamma \vdash H) = proj(1)(((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\lambda_{Rv(F)}.Ext(\Gamma, F \vdash H))e')e_1 ...)e_2)e_1), \\
&\qquad\quad ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\lambda_{Rv(G)}.Ext(\Gamma, G \vdash H))nop)e_1 ...)e_2)e_1))) \\
&\qquad \vee(Ext(\Gamma \vdash H) = proj(2)(((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\lambda_{Rv(F)}.Ext(\Gamma, F \vdash H))nop)e_1 ...)e_2)e_1), \\
&\qquad\quad ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\lambda_{Rv(G)}.Ext(\Gamma, G \vdash H))e'')e_1 ...)e_2)e_1)))))
\end{aligned}
$$

よって，(2)，(3) より OK.

- Inp-Intro

Γ を $F_1, ... F_n$，$\Gamma, F \vdash G$ に含まれる object 自由変数を $x_1, ..., x_m$ とする．

i

仮定より，

$$
\begin{aligned}
&\forall x_1, ... x_m(Real(n, F_i, e_i) \Rightarrow \\
&\qquad (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma, F \vdash G))e_n ...)e_2)e_1 \textbf{ realize } G) \qquad \text{......(1)}
\end{aligned}
$$

realizer extraction rule より，

$$Ext(\Gamma \vdash F \Rightarrow G) = (\Pi, \lambda_{Rv(F)}.Ext(\Gamma, G \vdash F))$$

ここで，

$$
\begin{aligned}
&\forall x_1, ... x_m(Real(n, F_i, e_i) \Rightarrow \\
&\qquad (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\Pi, \lambda_{Rv(F)}.Ext(\Gamma \vdash F)))e_n ...)e_2)e_1 \\
&\qquad\quad = (\Pi, (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\lambda_{Rv(F)}.Ext(\Gamma \vdash F)))e_n ...)e_2)e_1))
\end{aligned}
$$

また，(e_F **realize** F) とすると，e_F には，$Rv(F_i)(0 < i < n+1)$ は含まれないので，

$$
\begin{aligned}
&\forall x_1, ... x_m(Real(n, F_i, e_i) \Rightarrow \\
&\qquad ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\lambda_{Rv(F)}.Ext(\Gamma \vdash F)))e_n ...)e_2)e_1)e_F \\
&\qquad\quad = (\lambda_{Rv(F_1)}.(...((\lambda_{Rv(F_n)}.\lambda_{Rv(F)}.Ext(\Gamma \vdash F))e_F)e_n ...)e_2)e_1)
\end{aligned}
$$

(1) より，

$$
\begin{aligned}
&\forall x_1, ... x_m(Real(n, F_i, e_i) \Rightarrow \\
&\qquad ((\lambda_{Rv(F_1)}.(...((\lambda_{Rv(F_n)}.\lambda_{Rv(F)}.Ext(\Gamma \vdash F))e_F)e_n ...)e_2)e_1 \textbf{ realize } G))
\end{aligned}
$$

OK.

ii

仮定より，

$$
\begin{aligned}
&\forall x_1, .., x_m(Real_L(n, F_i, e_i) \Rightarrow \\
&\qquad (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma, F \vdash G))e_n ...)e_2)e_1 \textbf{ realize}_L\ G) \qquad \text{......(1)}
\end{aligned}
$$

realizer extraction rule より，

$$Ext(\Gamma \vdash F \Rightarrow G) = (\Pi, \lambda_{Rv(F)}.Ext(\Gamma, G \vdash F))$$

ここで，

$$\forall x_1,...x_m(Real_L(n,F_i,e_i) \Rightarrow$$
$$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\Pi.\lambda_{Rv(F)}.Ext(\Gamma \vdash F)))e_n...)e_2)e_1$$
$$= (\Pi.(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\lambda_{Rv(F)}.Ext(\Gamma \vdash F)))e_n...)e_2)e_1))$$

また，(e_F **realize**$_L$ F) とすると，e_F には，$Rv(F_i)(0 < i < n+1)$ は含まれないので，

$$\forall x_1,...,x_m(Real_L(n,F_i,e_i) \Rightarrow$$
$$((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\lambda_{Rv(F)}.Ext(\Gamma \vdash F)))e_n...)e_2)e_1)e_F$$
$$= (\lambda_{Rv(F_1)}.(...((\lambda_{Rv(F_n)}.\lambda_{Rv(F)}.Ext(\Gamma \vdash F))e_F)e_n...)e_2)e_1)$$

(1) より，

$$\forall x_1,...x_m(Real_L(n,F_i,e_i) \Rightarrow$$
$$((\lambda_{Rv(F_1)}.(...((\lambda_{Rv(F_n)}.\lambda_{Rv(F)}.Ext(\Gamma \vdash F))e_F)e_n...)e_2)e_1 \text{ \textbf{realize}}_L\ G))$$

OK.

- Imp-Elim

Γ を $F_1,...,F_n$，$\Gamma,F \vdash G$ に含まれる object 自由変数を $x_1,...x_m$ とする．

- i

仮定より，

$$\forall x_1,...x_m(Real(n,F_i,e_i) \Rightarrow$$
$$(proj(2)((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F \Rightarrow G))e_n...)e_2)e_1)) \text{ \textbf{realize} } F \Rightarrow G) \quad .....(1)$$
$$\forall x_1,...x_m(Real(n,F_i,e_i) \Rightarrow$$
$$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F))e_n...)e_2)e_1 \text{ \textbf{realize} } F) \quad ......(2)$$

realizer extraction rule より，

$$Ext(\Gamma \vdash G) = (proj(2)(Ext(\Gamma,G \vdash F)))Ext(\Gamma \vdash F)$$

ここで，

$$\forall x_1,...,x_m(Real(n,F_i,e_i) \Rightarrow$$
$$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.((proj(2)(Ext(\Gamma \vdash F \Rightarrow)))Ext(\Gamma \vdash F)))e_n...)e_2)e_1$$
$$= e'((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F))e_n...)e_2)e_1) \wedge ((e \text{ \textbf{realize} } F) \rightarrow (e'e \text{ \textbf{realize} } G)$$

よって，(2) より OK.

- ii

仮定より，

$$\forall x_1,...x_m(Real_L(n,F_i,e_i) \Rightarrow$$
$$(proj(2)((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F \Rightarrow G))e_n...)e_2)e_1)) \text{ \textbf{realize}}_L\ F \Rightarrow G) \quad .....(1)$$
$$\forall x_1,...,x_m(Real_L(n,F_i,e_i) \rightarrow$$
$$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F))e_n...)e_2)e_1 \text{ \textbf{realize}}_L\ F) \quad ......(2)$$

realizer extraction rule より，

$$Ext(\Gamma \vdash G) = (proj(2)(Ext(\Gamma,G \vdash F)))Ext(\Gamma \vdash F)$$

ここで,

$$\forall x_1,...,x_m(Real_L(n,F_i,e_i)\Rightarrow$$
$$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.((proj(2)(Ext(\Gamma\vdash F\Rightarrow)))Ext(\Gamma\vdash F)))e_n...)e_2)e_1$$
$$=e'((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma\vdash F))e_n...)e_2)e_1)\wedge((e\ \mathbf{realize}_L\ F)\Rightarrow(e'e\ \mathbf{realize}_L\ G)$$

よって, (2) より OK.

- Univ-Intro

(a) $\forall$

Γ を $F_1,...,F_n$, $\Gamma\vdash F(x)$ に含まれる object 自由変数を $x,x_1,...x_m$ とする.

i

仮定より,

$$\forall x,x_1,...,x_m(Real(n,F_i,e_i)\Rightarrow$$
$$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma\vdash F(x)))e_n...)e_2)e_1\ \mathbf{realize}\ F(x)) \qquad ......(1)$$

realizer extraction rule より,

$$Ext(\Gamma\vdash\forall xF(x))=(in,\lambda_x.Ext(\Gamma\vdash F(x)))$$

(1) より,

$$\forall x_1,...,x_m(Real(n,F_i,e_i)\Rightarrow$$
$$((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(in,\lambda_x.Ext(\Gamma\vdash F(x))))e_n...)e_2)e_1$$
$$=(in,\lambda_x.e'))\wedge\forall t((\lambda_x.e')t\ \mathbf{realize}\ F(t)))$$

OK.

ii

仮定より,

$$\forall x,x_1,...,x_m(Real_L(n,F_i,e_i)\Rightarrow$$
$$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma\vdash F(x)))e_n...)e_2)e_1\ \mathbf{realize}_L\ F(x)) \qquad ......(1)$$

realizer extraction rule より,

$$Ext(\Gamma\vdash\forall xF(x))=(in,\lambda_x.Ext(\Gamma\vdash F(x)))$$

(1) より,

$$\forall t,x_1,...,x_m(Real_L(n,F_i,e_i)\Rightarrow$$
$$((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(in,x,\lambda_x.Ext(\Gamma\vdash F(x))))e_n...)e_2)e_1$$
$$=(in,\lambda_x.e'))\wedge\forall t((\lambda_x.e')t\ \mathbf{realize}_L\ F(t)))$$

OK.

(b) $\dot\forall$

$\forall$ と同様. OK.

- Univ-Elim

(a) $\forall$

Γ を $F_1, ..., F_n$, $\Gamma \vdash \forall x F(x)$ に含まれる object 自由変数を $x_1, ..., x_m$ とする．

- i

仮定より，

$$\forall x, x_1, ... x_m(Real(n, F_i, e_i) \Rightarrow (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash \forall x F(x)))e_n...)e_2)e_1 \text{ realize } \forall x F(x)) ......(1)$$

realizer extraction rule より，

$$Ext(\Gamma \vdash F(a)) = proj(2)(Ext(\Gamma \vdash \forall x.F(x)))$$

また，

$$\forall x_1, ... x_m(Real(n, F_i, e_i) \Rightarrow ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.proj(2)(Ext(\Gamma \vdash \forall x.F(x))))e_n...)e_2)e_1)a = proj(2)(((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash \forall x.F(x))))e_n...)e_2)e_1)a)$$

よって，(1)，$\forall$ の realizer の定義より，

$$\forall x_1, ..., x_m(\quad Real(n, F_i, e_i) \rightarrow (((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F(a)))e_n...)e_2)e_1 = (\lambda_x.e(x))a) \wedge \forall t.((\lambda_x.e(e))t \text{ realize } F(t))))$$

OK.

- ii

仮定より，

$$\forall x, x_1, ..., x_m(Real_L(n, F_i, e_i) \Rightarrow (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash \forall x F(x)))e_n...)e_2)e_1 \text{ realize}_L \forall x F(x)) ......(1)$$

realizer extraction rule より，

$$Ext(\Gamma \vdash F(a)) = proj(2)(Ext(\Gamma \vdash \forall x.F(x)))$$

また，

$$\forall x_1, ..., x_m(Real_L(n, F_i, e_i) \Rightarrow ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.proj(2)(Ext(\Gamma \vdash \forall x.F(x))))e_n...)e_2)e_1)a = proj(2)(((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash \forall x.F(x))))e_n...)e_2)e_1)a)$$

よって，(1)，$\forall$ の realizer の定義より，

$$\forall x_1, ..., x_m(Real_L(n, F_i, e_i) \Rightarrow (((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F(a)))e_n...)e_2)e_1 = (\lambda_x.e(x))a) \wedge \forall t.((\lambda_x.e(e))t \text{ realize}_L F(t))))$$

OK.

(b) $\dot{\forall}$

$\forall$ と同様．OK.

- Exist-Intro

  (a) $\exists$

  Γ を $F_1, ..., F_n$， $\Gamma \vdash F(t)$ に含まれる object 自由変数を $x_1, ..., x_m$ とする．

  - i

    仮定より，

    $$\forall x, x_1, ..., x_m(Real(n, F_i, e_i) \Rightarrow (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F(t)))e_n...)e_2)e_1 \textbf{ realize } F(t)).......(1)$$

    realizer extraction rule より，

    $$Ext(\Gamma \vdash \exists x F(x)) = (out, t, Ext(\Gamma \vdash F(t)))$$

    また，

    $$\begin{aligned}\forall x_1, ... x_m(Real(n, F_i, e_i) \rightarrow \\ (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(out, t, Ext(\Gamma \vdash F(t))))e_n...)e_2)e_1 \\ = (out, (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.t)e_n...)e_2)e_1, \\ (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F(t)))e_n...)e_2)e_1))\end{aligned}$$

    よって， (1)， $\exists$ の realizer の定義より OK.

  - ii

    仮定より，

    $$\forall x, x_1, ... x_m(Real_L(n, F_i, e_i) \Rightarrow (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F(t)))e_n...)e_2)e_1 \textbf{ realize}_L\ F(t)).......(1)$$

    realizer extraction rule より，

    $$Ext(\Gamma \vdash \exists x F(x)) = (out, t, Ext(\Gamma \vdash F(t)))$$

    また，

    $$\begin{aligned}\forall x_1, ..., x_m(Real_L(n, F_i, e_i) \Rightarrow \\ (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(out, t, Ext(\Gamma \vdash F(t))))e_n...)e_2)e_1 \\ = (out, (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.t)e_n...)e_2)e_1, \\ (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash F(t)))e_n...)e_2)e_1))\end{aligned}$$

    よって， (1)， $\exists$ の realizer の定義より OK.

  (b) $\dot{\exists}$

  $\exists$ と同様． OK.

- Exist-Elim

  (a) $\exists$

  Γ を $F_1, ..., F_n$， $\Gamma, F(t) \vdash G$ に含まれる object 自由変数を $x_1, ..., x_m$ とする．

  - i

    仮定より，

    $$\forall x, x_1, ..., x_m(Real(n, F_i, e_i) \Rightarrow (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash \exists F(x)))e_n...)e_2)e_1 \textbf{ realize } \exists x F(x)).......(1)$$

$\forall x, x_1, ..., x_m(Real(n, F_i, e_i) \wedge ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.e')e_n...)e_2)e_1 \textbf{ realize } F(t)) \Rightarrow$
$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.((\lambda_{Rv(F(t))}.Ext(\Gamma, F(t) \vdash G))e'))e_n...)e_2)e_1 \textbf{ realize } G)$......(2)

realizer extraction rule より，

$Ext(\Gamma, F(t) \vdash G) = (\lambda_{Rv(F(t))}.Ext(\Gamma, F(t) \vdash G))(proj(3)(Ext(\Gamma \vdash \exists x F(x))))$

(1),(2) より，

$$\forall x_1, ..., x_m(Real(n, F_i, e_i) \Rightarrow (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\lambda_{Rv(F(t))}.Ext(\Gamma, F(t) \vdash G)) (proj(3)(Ext(\Gamma \vdash \exists x F(x)))))e_n...)e_2)e_1 = e'' \wedge (e'' \textbf{ realize } G))$$

OK.

ii

仮定より，

$\forall x, x_1, ..., x_m(Real_L(n, F_i, e_i) \Rightarrow$
$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\Gamma \vdash \exists F(x)))e_n...)e_2)e_1 \textbf{ realize}_L \exists x F(x))$......(1)

$\forall x, x_1, ..., x_m(Real_L(n, F_i, e_i) \wedge ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.e')e_n...)e_2)e_1 \textbf{ realize}_L F(t)) \Rightarrow$
$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.((\lambda_{Rv(F(t))}.Ext(\Gamma, F(t) \vdash G))e'))e_n...)e_2)e_1 \textbf{ realize}_L G)$......(2)

realizer extraction rule より，

$Ext(\Gamma, F(t) \vdash G) = (\lambda_{Rv(F(t))}.Ext(\Gamma, F(t) \vdash G))(proj(3)(Ext(\Gamma \vdash \exists x F(x))))$

(1),(2) より，

$$\forall x_1, ..., x_m(Real_L(n, F_i, e_i) \Rightarrow (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.(\lambda_{Rv(F(t))}.Ext(\Gamma, F(t) \vdash G)) (proj(3)(Ext(\Gamma \vdash \exists x F(x)))))e_n...)e_2)e_1 = e'' \wedge (e'' \textbf{ realize}_L G))$$

OK.

(b) $\dot{\exists}$

$\exists$ と同様．OK.

- WF-Intro

(a) $< \forall > x_1, ..., x_n$ に $\dot{\forall}$ が現れないとき

Γ を $F_1, ..., F_t$，$\Gamma \vdash \forall x_1, ... x_n F(x_1, ..., x_n)$ に含まれる object 自由変数を $m_1, ..., m_s$ とし，

$$\nu = \forall y_1, ..y_n((y_1, ..., y_n) <_{jn} (x_1, ..., x_n) \Rightarrow F(y_1, ... y_n))$$

$$\zeta = \Gamma, \forall y_1 ..y_n.((y_1, ..., y_n) <_{jn} (x_1, ..., x_n) \Rightarrow F(y_1, ..., y_n)) \vdash F(x_1, ..., x_n)$$

とする．

i

仮定より，

$$\forall m_1,...,m_s,y_1,...,y_n,x_1,..,x_n(Real(t,F_i,e_i)\wedge((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.e')e_t...)e_2)e_1 \textbf{ realize } \nu)$$
$$\Rightarrow ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.(\lambda_{Rv(\nu)}.Ext(\zeta))e')e_t...)e_2)e_1 \textbf{ realize } F(x_1,...,x_n))......(1)$$

さて，realizer extraction rule より，

$$Ext(\Gamma\vdash\forall x_1,...,x_n F(x_1,...x_n))$$
$$=(in.(\lambda_{x_1}.(in.(...(\lambda_{x_{n-1}}.(in.(\lambda_{x_n}.$$
$$(\mu_{Z x_1..x_n}.(\lambda_Z.\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1...y_n}.(\lambda_x.Zy_1..y_n))))x_1..x_n)))))))$$

また，

$$(\mu_{Z x_1..x_n}.(\lambda_Z.\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1...y_n}.(\lambda_x.Zy_1..y_n))))x_1..x_n$$
$$=(\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1,...,y_n}.(\lambda_x.(\mu_{Z x_1..x_n}.(\lambda_Z.\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))$$
$$(\lambda_{y_1,...,y_n}.(\lambda_x.Zy_1..y_n))))))y_1..y_n)x_1..x_n \qquad ....(2)$$

$x_1,...,x_n$ を任意の object とし，$Real(t,F_i,e_i)$ を仮定する．
このとき，$\forall$ の realizer の定義，(2) から，

$$((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.(\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1,...,y_n}.(\lambda_x.(\mu_{Z x_1..x_n}.(\lambda_Z.\lambda_{x_1,...,x_n}.$$
$$(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1,...,y_n}.(\lambda_x.Zy_1..y_n))))))y_1..y_n)x_1..x_n)e_t...)e_2)e_1$$
$$\textbf{realize } F(x_1,...x_n))$$

を示せばよい．$<_{jn}$ は，well-founded な order なので，$(x_1,...,x_n)$ の $<_{jn}$ に対する大きさに関する帰納法で証明する．

[Base Case]

$(x_1,...x_n)$ は，全ての $(y_1,...,y_n)$ に関し，

$$(x_1,...x_n)\le(y_1,...,y_n)$$

よって realizer の定義より，全ての Program Term $P_1,...P_t$ に対し，

$$((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.(\lambda_{x_1,...,x_n}.(\lambda_{y_1,...,y_n}.(\lambda_x.(\mu_{Z x_1..x_n}.(\lambda_Z.\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))$$
$$(\lambda_{y_1,...,y_n}.(\lambda_x.Zy_1..y_n))))))y_1..y_n)x_1..x_n)P_t...)P_2)P_1 \textbf{ realize } \nu)$$

よって，(1) より，

$$((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.(\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1,...,y_n}.(\lambda_x.(\mu_{Z x_1..x_n}.(\lambda_Z.\lambda_{x_1,...,x_n}.$$
$$(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1,...,y_n}.(\lambda_x.Zy_1..y_n))))))y_1..y_n)x_1..x_n)e_t...)e_2)e_1$$
$$\textbf{realize } F(x_1,...x_n))$$

OK.

[Induction Case]

以下を仮定する．

$$\forall z_1,...,z_n((z_1,...,z_n)\lneq(x_1,...,x_n)\Rightarrow$$
$$((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.(\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1...y_n}.(\lambda_x.(\mu_{Z x_1..x_n}.(\lambda_Z.\lambda_{x_1,...,x_n}.$$
$$(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1...y_n}.(\lambda_x.Zy_1..y_n))))))y_1..y_n)z_1..z_n)e_t...)e_2)e_1$$
$$\textbf{realize } F(z_1,...z_n))$$

この式と，(2) より，

$$\forall z_1,...,z_n((z_1,...,z_n)\lneq(x_1,...x_n)\Rightarrow$$

$$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.(\mu_{Zx_1..x_n}.(\lambda_Z.\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))$$
$$(\lambda_{y_1,...,y_n}.(\lambda_x.Zy_1..y_n))))z_1..z_n)e_t...)e_2)e_1 \textbf{ realize } F(z_1,...,z_n))$$

よって，

$$((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.(\lambda_{y_1,...,y_n}\lambda_x.(\mu_{Zx_1..x_n}.(\lambda_Z.\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))$$
$$(\lambda_{y_1,...,y_n}.(\lambda_x.Zy_1..y_n))))z_1..z_n))e_t...)e_2)e_1 \textbf{ realize } \nu)$$

よって，(1) より，

$$((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.(\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1,...,y_n}.(\lambda_x.(\mu_{Zx_1..x_n}.(\lambda_Z.\lambda_{x_1,...,x_n}.$$
$$(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1,...,y_n}.(\lambda_x.Zy_1..y_n))))))y_1..y_n)x_1..x_n)e_t...)e_2)e_1$$
$$\textbf{realize } F(x_1,...,x_n))$$

OK.

– ii

仮定より，

$$\forall m_1,...,m_s,y_1,...,y_n,x_1,...,x_n(Real_L(t,F_i,e_i)\wedge((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.c')e_t...)e_2)e_1 \textbf{ realize}_L \nu)$$
$$\Rightarrow ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.(\lambda_{Rv(\nu)}.Ext(\zeta))c')e_t...)e_2)e_1 \textbf{ realize}_L F(x_1,...,x_n))......(1)$$

さて，realizer extraction rule より，

$$Ext(\Gamma\vdash\forall x_1,...,x_n F(x_1,...,x_n))$$
$$=(in.(\lambda_{x_1}.(in.(...,(\lambda_{x_{n-1}}.(in.(\lambda_{x_n}.$$
$$(\mu_{Zx_1..x_n}.(\lambda_Z.\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1,...,y_n}.(\lambda_x.Zy_1..y_n))))x_1..x_n)))))))$$

また，

$$(\mu_{Zx_1..x_n}.(\lambda_Z.\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1,...,y_n}.(\lambda_x.Zy_1..y_n))))x_1..x_n$$
$$=(\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1,...,y_n}.(\lambda_x.(\mu_{Zx_1..x_n}.(\lambda_Z.\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))$$
$$(\lambda_{y_1,...,y_n}.(\lambda_x.Zy_1..y_n))))))y_1..y_n)x_1..x_n \qquad ....(2)$$

$x_1,...,x_n$ を任意の object とし，$Real_L(t,F_i,e_i)$ を仮定する．
このとき，$\forall$ の realizer の定義，(2) から，

$$((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.(\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1,...,y_n}.(\lambda_x.(\mu_{Zx_1..x_n}.(\lambda_Z.\lambda_{x_1,...,x_n}.$$
$$(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1,...,y_n}.(\lambda_x.Zy_1..y_n))))))y_1..y_n)x_1..x_n)e_t...)e_2)e_1$$
$$\textbf{realize}_L F(x_1,...,x_n))$$

を示せばよい．$<_{jn}$ は，well-founded な order なので，$(x_1,...,x_n)$ の $<_{jn}$ に対する大きさに関する帰納法で証明する．

[Base Case]

$(x_1,...,x_n)$ は，全ての $(y_1,...,y_n)$ に関し，

$$(x_1,...,x_n)\leq(y_1,...,y_n)$$

よって realizer の定義より，全ての Program Term $P_1,...,P_t$ に対し，

$$((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.(\lambda_{x_1,...,x_n}.(\lambda_{y_1,...,y_n}.(\lambda_x.(\mu_{Zx_1..x_n}.(\lambda_Z.\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))$$
$$(\lambda_{y_1,...,y_n}.(\lambda_x.Zy_1..y_n))))))y_1..y_n)x_1..x_n)P_t...)P_2)P_1 \textbf{ realize}_L \nu)$$

よって、(1) より、

$$((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.(\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1,...,y_n}.(\lambda_x.(\mu_{Zx_1..x_n}.(\lambda_Z.\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1,...,y_n}.(\lambda_x.Zy_1..y_n))))))y_1..y_n)x_1..x_n)e_t...)e_2)e_1 \text{ } \mathbf{realize}_L F(x_1,...x_n))$$

OK.

[Induction Case]

以下を仮定する.

$$\forall z_1,...z_n((z_1,...z_n) \leq (x_1,...x_n) \Rightarrow ((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.(\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1,...,y_n}.(\lambda_x.(\mu_{Zx_1..x_n}.(\lambda_Z.\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1,...,y_n}.(\lambda_x.Zy_1..y_n))))))y_1..y_n)z_1..z_n)e_t...)e_2)e_1 \text{ } \mathbf{realize}_L F(z_1,...,z_n))$$

この式と、(2) より、

$$\forall z_1,...,z_n((z_1,...z_n) \leq (x_1,...x_n) \Rightarrow (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.(\mu_{Zx_1..x_n}.(\lambda_Z.\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1,...,y_n}.(\lambda_x.Zy_1..y_n))))z_1..z_n)e_t...)e_2)e_1 \text{ } \mathbf{realize}_L F(z_1,...,z_n))$$

よって、

$$((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.(\lambda_{y_1,...,y_n}\lambda_x.(\mu_{Zx_1..x_n}.(\lambda_Z.\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1,...,y_n}.(\lambda_x.Zy_1..y_n))))z_1..z_n))e_t...)e_2)e_1 \text{ } \mathbf{realize}_L \nu)$$

よって、(1) より、

$$((\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.(\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1,...,y_n}.(\lambda_x.(\mu_{Zx_1..x_n}.(\lambda_Z.\lambda_{x_1,...,x_n}.(\lambda_{Rv(\nu)}.Ext(\zeta))(\lambda_{y_1,...,y_n}.(\lambda_x.Zy_1..y_n))))))y_1..y_n)x_1..x_n)e_t...)e_2)e_1 \text{ } \mathbf{realize}_L F(x_1,...,x_n))$$

OK.

(b) $<\forall> x_1,...x_n$ に $\dot{\forall}$ が現れるとき

$<\forall> x_1,...,x_n$ に $\dot{\forall}$ が現れないときと同様. OK.

- Mu-1

  realizer extraction rule, Realizer の定義より明らか. OK.

- Mu-2

  $I = \mu_{\Psi_{(n)}x_1..x_n}.G(x_1,...x_n,\Psi a_{11}..a_{1n},...,\Psi a_{m1}..a_{mn})$

  $\xi = \Gamma \vdash Ib_1..b_n$

  とし、Γ を $F_1,...,F_t$, $\Gamma \vdash Ib_1..b_n$) に含まれる object 自由変数を $u_1,...u_k$ とする. I には、1-formula の定義から、自由述語変数は含まれない. よって、Note 4.4より、i を示せばよい.

  仮定より、

$$\forall u_1,...,u_k(Real(t,F_i,e_i) \Rightarrow (\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.Ext(\Gamma,\vdash Ib_1..b_n))e_t...)e_2)e_1 \text{ } \mathbf{realize} \text{ } Ib_1..b_n) \qquad ......(1)$$

realizer extraction rule より、

$$Ext(\Gamma \vdash G(b_1,..,b_n, Ia_{11}..a_{1n},...,Ia_{m1}..a_{mn}))$$

$$= ((if, (proj(1), Ext(\xi)) = rec), (proj(2)Ext(\xi))$$
$$((proj(2)(proj(1)Ext(\xi)))(proj(1)Ext(\xi))), Ext(\xi))$$

$u_1, \dots u_k$ を値を持つ任意の object とし， $Real(t, F_i, e_i)$ を仮定する．
このとき， realizer の定義から，ある $L \subset U^n$ について，

$$(\lambda_{Rv(F_1)}.(\dots(\lambda_{Rv(F_t)}.Ext(\Gamma, \vdash Ib_1..b_n))e_t\dots)e_2)e_1 = ((rec, \lambda_{Z_{\Psi_{(n)}}, x_1, .., x_n}.e', \lambda_Z.Zb_1..b_n)$$
$$\wedge \forall(y_1, \dots, y_n) \epsilon L([y_1/x_1, .., y_n/x_n]e' \ \mathbf{realize}_L$$
$$\varpi((x_1, ..x_n), H(x_1, .., x_n), \Psi_{(n)}, I, G(y_1, \dots, y_n, Ia_{11}..a_{1n}, \dots, Ia_{m1}..a_{mn})))$$

または，

$$((\lambda_{Rv(F_1)}.(\dots(\lambda_{Rv(F_t)}.Ext(\Gamma, \vdash Ib_1..b_n))e_t\dots)e_2)e_1 \ \mathbf{realize} \ G(b_1, \dots, b_n, Ia_{11}..a_{1n}, \dots, Ia_{m1}..a_{mn}))$$

後者のとき． OK． 前者のとき，

$$Ext(\Gamma \vdash G(b_1, \dots b_n, Ia_{11}..a_{1n}, \dots, Ia_{m1}..a_{mn})) = ((\lambda_{Z_{\Psi_{(n)}}, x_1, .., x_n}.e')(rec, \lambda_{Z_{\Psi_{(n)}}, x_1, .., x_n}.e')b_1..b_n$$

さて，

$$(((rec, \lambda_{Z_{\Psi_{(n)}}, x_1, .., x_n}.e'), \lambda_Z.Zb_1..b_n) \ \mathbf{realize}_L \ Ib_1..b_n)$$

よって， $\mathrm{realizer}_L$ の定義の $(*)_3$ より，

$$((\lambda_{Rv(F_1)}.(\dots(\lambda_{Rv(F_t)}.Ext(\Gamma, \vdash Ib_1..b_n))e_t\dots)e_2)e_1$$
$$\mathbf{realize}_L \ G(b_1, \dots, b_n), Ia_{11}..a_{1n}, \dots, Ia_{m1}..a_{mn}))$$

OK．

- Bot-Intro,Bot-Elim,Neg-Intro,eq 1,eq 2,le
  realizer extraction rule， realizer の定義より明らか． OK．

(b)

- ∃
  Γ を $F_1, \dots, F_t$， $\Gamma \vdash \exists x G(x)$ に含まれる object 自由変数を $y_1, \dots, y_m$ とし，
  $\nu = \forall x_1, \dots x_n(H(x_1, \dots, x_n) \Rightarrow \Psi_{(n)}x_1..x_n$
  $\zeta = \Gamma, \forall x_1, \dots x_n(H(x_1, \dots, x_n) \Rightarrow \Psi_{(n)}x_1..x_n) \vdash \forall x[\Psi_{(n)}/p_{(n)}]G(x)$
  $\ell = \Gamma \vdash <\exists> x((x_1, \dots x_n), H(x_1, \dots, x_n), \Psi_{(n)}, p_{(n)}, G(x))$
  とする．
  仮定より，

$$\forall y_1, \dots, y_m(Real(t, F_i, e_i) \Rightarrow$$
$$(\lambda_{Rv(F_1)}.(\dots(\lambda_{Rv(F_t)}.Ext(\Gamma \vdash \exists x G(x)))e_t\dots)e_2)e_1 \ \mathbf{realize}_L \ \exists x G(x)) \qquad \dots\dots(1)$$

$$\forall y_1, \dots, y_m(Real(t, F_i, e_i) \wedge ((\lambda_{Rv(F_1)}.(\dots(\lambda_{Rv(F_t)}.e')e_t\dots)e_2)e_1 \ \mathbf{realize}_L \ \nu) \Rightarrow$$
$$(\lambda_{Rv(F_1)}.(\dots(\lambda_{Rv(F_t)}.((\lambda_{Rv(\nu)}.Ext(\zeta))e'))e_t\dots)e_2)e_1 \ \mathbf{realize}_L \ \forall x[\Psi_{(n)}/p_{(n)}]G(x)) \dots\dots(2)$$

さて， realizer extraction rule より，

$$Ext(\ell) = (in, \lambda_x.((if, x = (proj(2)Ext(\varsigma)))(proj(3)Ext(\varsigma),$$
$$(\lambda_{Rv(\zeta)}.Ext(\xi))(\lambda_{x_1..x_n}\lambda_x.(Z_{\Phi_{(n)}}, \lambda_Z.Zx_1..x_n)))))$$

$y_1, \dots y_m$ を値を持つ任意の object とし、 $Real(t, F_t, e_t)$ を仮定する.

$$((\lambda_{x_1..x_n}\lambda_x.(Z_{\Psi_{(n)}}, \lambda_Z.Zx_1..x_n)) \ \mathbf{realize}_L \ \forall x_1, \dots, x_n(H(x_1, \dots, x_n) \Rightarrow \Psi_{(n)}x_1..x_n))$$

であるので、 ∃ の realizer の定義から,

$$\begin{aligned}&((\lambda_{Rv(F_1)}.(\dots(\lambda_{Rv(F_t)}.Ext(f))e_t\dots)e_2)e_1 \\ &\quad = (in, \lambda_x.((if, x = k), (e_a, e_b)))) \wedge (e_a \ \mathbf{realize}_L \ G(k)) \wedge (e_b \ \mathbf{realize}_L \ [\Psi_{(n)}/I]G(x)))\end{aligned}$$

さて、 $\mathrm{realize}_L$ の定義より,
e_a、 e_b のプログラム変数 $Z_{\Psi_{(n)}}$ をすべて、プログラム e に置き換えたものをそれぞれ e'_a, e'_b とすると、 $(b_1, \dots, b_n)\epsilon L$ を満たす全ての $b_1, \dots, b_n$ について,

$$tic((e, \lambda_Z.Zb_1..b_n)) = \omega \Rightarrow TDC(e'_a) \quad tic((e, \lambda_Z.Zb_1..b_n)) = \omega \Rightarrow TDC(e'_b)$$

$$tic((e, \lambda_Z.Zb_1..b_n)) \neq \omega \wedge tic((e, \lambda_Z.Zb_1..b_n)) \neq -\omega \Rightarrow tic(e'_a) \geq tic((e, \lambda_Z.Zb_1..b_n))$$

$$tic((e, \lambda_Z.Zb_1..b_n)) \neq \omega \wedge tic((e, \lambda_Z.Zb_1..b_n)) \neq -\omega \Rightarrow tic(e'_b) \geq tic((e, \lambda_Z.Zb_1..b_n))$$

よって、 $e_c = ((\lambda_{Rv(F_1)}.(\dots(\lambda_{Rv(F_t)}.Ext(f))e_t\dots)e_2)e_1$ とし、このプログラム変数 $Z_{\Psi_{(n)}}$ をすべて、プログラム e に置き換えたものを e'_c とすると、

$$tic((e, \lambda_Z.Zb_1..b_n)) = \omega \Rightarrow TDC(e'_c)$$

$$tic((e, \lambda_Z.Zb_1..b_n)) \neq \omega \wedge tic((e, \lambda_Z.Zb_1..b_n)) \neq -\omega \Rightarrow tic(e'_c) \geq tic((e, \lambda_Z.Zb_1..b_n))$$

また、自由述語変数が ⁝ の内側以外に現れないとき、

(a) $\exists xG(x)$ に ⁝ が現れないとき、 e に関わらず $TDC(e'_a)$ であるから e に関わらず $TIC(e_c)$. また、 e_c は、何を入力しても、 e_a か e_b になるので、仮定より,

$$tic((e, \lambda_Z.Zb_1..b_n)) \neq \omega \wedge tic((e, \lambda_Z.Zb_1..b_n)) \neq -\omega \Rightarrow tic(e'_c) > tic((e, \lambda_Z.Zb_1..b_n))$$

(b) $\exists xG(x)$ に ⁝ が現れるとき、 e に関わらず $TDC(e'_a)$ であるので、 e に関わらず $TIC(e_c)$. また、 e_c は、何を入力しても、 e_a か e_b になるので、仮定より,

$$tic((e, \lambda_Z.Zb_1..b_n) \neq \omega \wedge tic((e, \lambda_Z.Zb_1..b_n) \neq -\omega \Rightarrow tic(e'_c) > tic((e, \lambda_Z.Zb_1..b_n)$$

また、仮定より, $\vdash H(b_1..b_n)$ を満たす全ての $b_1, \dots, b_n$ に関し、

$$((Z_{\Psi_{(n)}}, \lambda_Z.Zb_1..b_n) \ \mathbf{realize}_L \ Ib_1..b_n)$$

とすると、 $\exists xG(x)$, C の

$$< \exists > x((x_1, \dots x_n), H(x_1, \dots x_n), \Psi_{(n)}, p_{(n)}, F(x))$$

の形の subformula を最外から順に $< \forall > xF(x)$ に変えた Formula F', F'' に対し、

$$(e_a \ \mathbf{realize}_L \ F')$$

$$(e_b \ \mathbf{realize}_L \ F'')$$

であるので,

$$(e_c \ \mathbf{realize}_L \ \forall G(x))$$

OK.

- ⁝

⁝ と同様. OK.

(c)
$I = \mu_{\Phi_{(n)}x_1..x_n}.G(x_1,..,x_n,\Psi a_{11}..a_{1n},..,\Psi a_{m1}..a_{mn})$
$\varsigma = \Gamma, H(y_1,...y_n) \vdash \varpi((x_1,..,x_n), H(x_1,...x_n), \Psi_{(n)}, I, K(y_1,..,y_n))$
$\xi = H(y_1,...y_n)$
$\nu = \varpi((x_1,...x_n), H(x_1,..,x_n), \Psi_{(n)}, I, G(y_1,..,y_n, Ga_{11}..a_{1n},..,Ga_{m1}..a_{mn}))$
$\ell = \Gamma \vdash Ib_1..b_n$
とし，Γ を $F_1,..,F_t$，$\Gamma \vdash Ib_1..b_n$ に含まれる object 自由変数を $y_1,...y_n, z_1,..,z_v$ とする．
仮定より，

$$\forall y_1,..,y_n, z_1,..,z_v(Real(t, F_i, e_i) \Rightarrow$$
$$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.((\lambda_{Rv(\xi)}.Ext(\varsigma))nop))e_t...)e_2)e_1 \ \mathbf{realize}_L\ \nu) \qquad ......(1)$$

realizer extraction rule より，

$$Ext(\ell) = ((rec, \lambda_{Z_{\Phi_{(n)}}, y_1..y_n}.(\lambda_{Rv(\xi)}.Ext(\varsigma))nop), (\lambda_Z.Za_1..a_n))$$

$y_1,...y_n, z_1,...,z_v$ を値を持つ任意の object とし，$Real(t, F_i, e_i)$ を仮定する．
このとき，

$$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.Ext(\ell))e_t...)e_2)e_1$$
$$= ((rec, \lambda_{Z_{\Phi_{(n)}}, y_1...y_n}.(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_t)}.((\lambda_{Rv(\xi)}.Ext(\varsigma))nop))e_t..)e_2)e_1), (\lambda_Z.Za_1..a_n))$$

よって，

$$(\lambda_{Rv(F_1)}.(...(\lambda_{Rv(F_n)}.Ext(\ell))e_n...)e_2)e_1$$
$$= ((rec, \lambda_{Z_{\Phi_{(n)}}, y_1...y_n}.e'), (\lambda_Z.Za_1..a_n)) \wedge (e' \ \mathbf{realize}_L\ \nu)$$

よって，$TDC(((rec, \lambda_{Z_{\Phi_{(n)}}, y_1...y_n}.e'), (\lambda_Z.Za_1..a_n)))$ を証明すればよい．
$\sigma = ((rec, \lambda_{Z_{\Phi_{(n)}}, y_1...y_n}.e'), (\lambda_Z.Za_1..a_n))$ とおこう．ν には，ϖ の定義より，ξ，ς の内側以外に現れる自由述語変数は存在しない．よって，仮定より e' の $Z_{\Phi_{(n)}}$ に SNN であるどのような Program Term を代入しても（代入したものを e'' とおく），$TIC(e'')$．よって $tic(\sigma) \leq 0$．これは $H(a_1,...,a_n)$ を満たす任意の $a_1,...,a_n$ で成り立つ．また，

$$tic(\sigma) = tic((\lambda_{Z_{\Phi_{(n)}}, y_1...y_n}.e')(rec, \lambda_{Z_{\Phi_{(n)}}, y_1...y_n}.e')a_1..a_n)$$

であるから，仮定により，$tic(\sigma) \leq 1$．これを繰り返し適用することにより，$tic(\sigma) = \omega$．
よって，$TDC(\sigma)$．OK． □

5 おわりに

本稿では，証明から外界とコミュニケーションを行うプロセス生成を目的とし，

- 体系 μ の提案．
- 一般的なプロセスを記述するためのプログラム言語 Program Term を提案した．
- プロセスの重要な性質として，良停止性を示した．
- 体系 μ に対し，Program Term を対象とした realizability interpretation を示し，実際に証明から realizability interpretation を満たすような Program の生成方法を示した．

現在プログラム生成を目的としたシステム PAPYRUS 上に，体系 μ 及び realizer extraction rule をインプリメントしている．今後の課題としては，

- システムで実際に様々なプログラムの生成実験を行う．

- 良停止性に関する性質をさらに研究する.
- if then を導入するなど記述しやすさを高める.
- GHC プログラムの生成，もしくは Program Term の GHC への変換方法の検討.
- R.Milner の Definitional Equal をもとにしたプログラム変換規則の研究

が上げられる．また，本稿での τ 遷移は非決定性を表現していたが，その他に，入力が行われるか行われないかによって遷移が決定するようなものも表現できるように改良を研究中である.

参考文献

[1] Howard, W.A., "The formulae as types notion of construction", in *Essays on Combinatory Logic, Lambda Calculus and Formalism*, Academic Press, pp.479-490, 1980

[2] Martin-Löf, P., "Constructive mathematics and computer programming", in *Logic, Methodology, and Philosophy of Science VI*, Cohen, L.J. et al. eds., North-Holland, pp.153-179, 1982

[3] Sato, M., "QJ: A Constructive Logical System with Types", France-Japan Artificial Intelligence and Computer Science Symposium 86, Tokyo, 1986

[4] Hayashi, S. and Nakano, H., "*PX: a computational logic*", RIMS-573, RIMS, Kyoto University, 1987

[5] Takayama,Y., "Extended Projection - a new technique to extract efficient programs from constructive proofs", Proceedings of 1989 Conference on Functional Programming Languages and Computer Architecture, ACM, 1989

[6] Milner, R."*Communication and Concurrency*", Prentice-Hall International, 1989.

[7] Hoare, C. "*Communicating Sequential Processes*", Prentice Hall International, 1985.