依存型意味論のための型チェックの実装に向けて

佐藤 未歩 [1]　　戸次 大介 [1,2,3]

[1] お茶の水女子大学大学院人間文化創成科学研究科

[2] 科学技術振興機構, CREST　[3] 産業技術総合研究所人工知能センター

{satoh.miho, bekki}@is.ocha.ac.jp

1 依存型意味論 (DTS)

依存型意味論 (Dependent Type Semantics, 以下DTS)[1] は、依存型理論 (Dependent Type Theory)[5] に基づく自然言語の証明論的意味論である。文の意味表示に依存型理論の型を対応づけており、$A \to B$ の一般化である Π 型 $(x{:}A) \to B$ と、$A \times B$ の一般化である Σ 型 $\begin{bmatrix} x{:}A \\ B \end{bmatrix}$ を用いることで文脈に依存した文の意味を表示することができる。

DTS において照応・前提現象は未指定項 (underspecified term) を用いることによって記述する。たとえば、次の (1) の二文の意味表示は、図 1 のように与えられる。

(1) a. A man entered.
　　b. He whistled.

$$\begin{bmatrix} v{:}\begin{bmatrix} u{:}\begin{bmatrix} x{:}\mathbf{entity} \\ \mathbf{man}(x) \end{bmatrix} \\ \mathbf{enter}(\pi_1(u)) \end{bmatrix} \\ \mathbf{whistle}\left(\pi_1(@ : \begin{bmatrix} x{:}\mathbf{entity} \\ \mathbf{man}(x) \end{bmatrix})\right) \end{bmatrix}$$

図 1: (1) の意味表示

図 1 中の @ が未指定項であり、照応代名詞 *He* の意味表示に含まれている。DTS において照応解決・前提束縛はこの @ を具体的な項に置き換える操作に相当する。(1) の例では、@ の型は $\begin{bmatrix} x{:}\mathbf{entity} \\ \mathbf{man}(x) \end{bmatrix}$ であるが、その型を持つ項として $\pi_1(v)$ が存在するため、図 1 中の @ を $\pi_1(v)$ に置き換え、図 2 のような意味表示を得ることで照応解決となる。同様に前提現象についても、前提現象を引き起こす前提トリガーの意味を @ を用いて表示する。このように、DTS において照応・前提現象は統一的に定式化され、照応解決・前提束縛の問題は @ に対する証明探索の問題に帰着する。よって、DTS を用いて照応・前提現象を扱う際には DTS のための型推論・証明探索を定式化することが必要不可欠である。

$$\begin{bmatrix} v{:}\begin{bmatrix} u{:}\begin{bmatrix} x{:}\mathbf{entity} \\ \mathbf{man}(x) \end{bmatrix} \\ \mathbf{enter}(\pi_1(u)) \end{bmatrix} \\ \mathbf{whistle}\,(\pi_1(\pi_1(v))) \end{bmatrix}$$

図 2: (1) の意味表示

2 DTS のための型チェック・型推論

2.1 Bekki and Sato (2015)

DTS のための型チェック・型推論アルゴリズムの定式化については、Bekki and Sato (2015)[3] において一定の成果を納めている。依存型理論において型推論は一般に undecidable であるため、[3] では Löh による Agda[4] の型推論の体系を参考に、構文的に制限された依存型の部分体系を用いることによって型推論を実現している。具体的には、アノテーション構文を用いることで型推論の及ばない部分式の型をあらかじめ指定し、型推論と型チェックを相互再帰的に呼び出すことで @ の型推論を可能にしている。

[3] は DTS のための型チェック・型推論に一定の成果をあげた一方で、いくつかの問題点を含んでいる。そのうちの 1 つが、文（または談話）の意味表示中に複数の @ が含まれていた場合、どの @ から証明探索を行うかが明示的でないという問題である。次の例文 (2) について考える。

(2) Switzerland cherishes its king.

(2) は、照応代名詞 *its* による照応現象と、所有格 *its* が前提とする *king* の存在という前提現象が同時に起こっている例である。そのため、(2) の意味表示（図 4）には 2 つの @ が含まれる。意味表示中の $@_1$ が *its*

$$
\frac{\Gamma \vdash M :_{\uparrow} A}{\Gamma \vdash M :_{\downarrow} A}\,(\mathit{CHK}) \qquad
\frac{(c:A) \in \sigma}{\Gamma \vdash c :_{\uparrow} A}\,(\mathit{CON}) \qquad
\frac{(x:M) \in \Gamma}{\Gamma \vdash x :_{\uparrow} M}\,(\mathit{VAR}) \qquad
\frac{}{\Gamma \vdash \mathsf{type} :_{\uparrow} \mathsf{kind}}\,(\mathit{typeF}) \qquad
\frac{\Gamma \vdash A :_{\downarrow} \mathsf{s} \quad [\![\mathcal{D}]\!]^{\mathsf{tm}} \Downarrow A' \quad \Gamma \vdash A' \mathit{true}}{\Gamma \vdash (@:A) :_{\uparrow} A}\,(@)
$$

$$
\frac{\overset{\mathcal{D}}{\Gamma \vdash A :_{\downarrow} \mathsf{s}_1} \quad [\![\mathcal{D}]\!]^{\mathsf{tm}} \Downarrow A' \quad \Gamma, x:A' \vdash B :_{\downarrow} \mathsf{s}_2}{\Gamma \vdash \begin{bmatrix} x{:}A \\ B \end{bmatrix} :_{\uparrow} \mathsf{s}_2}\,(\mathit{\Sigma F}) \qquad
\frac{\overset{\mathcal{D}}{\Gamma \vdash M :_{\downarrow} A} \quad [\![\mathcal{D}]\!]^{\mathsf{tm}} \Downarrow M' \quad B[M'/x] \Downarrow B' \quad \Gamma \vdash N :_{\downarrow} B'}{\Gamma \vdash (M,N) :_{\downarrow} \begin{bmatrix} x{:}A \\ B \end{bmatrix}}\,(\mathit{\Sigma I})
$$

$$
\frac{\Gamma \vdash M :_{\uparrow} \begin{bmatrix} x{:}A \\ B \end{bmatrix}}{\Gamma \vdash \pi_1(M) :_{\uparrow} A}\,(\mathit{\Sigma E}) \qquad
\frac{\Gamma \vdash M :_{\uparrow} \begin{bmatrix} x{:}A \\ B \end{bmatrix} \quad B[\pi_1(M)/x] \Downarrow B'}{\Gamma \vdash \pi_2(M) :_{\uparrow} B'}\,(\mathit{\Sigma E}) \qquad
\frac{\overset{\mathcal{D}}{\Gamma \vdash A :_{\downarrow} \mathsf{s}_1} \quad [\![\mathcal{D}]\!]^{\mathsf{tm}} \Downarrow A' \quad \Gamma, x:A' \vdash B :_{\downarrow} \mathsf{s}_2}{\Gamma \vdash (x{:}A) \to B :_{\uparrow} \mathsf{s}_2}\,(\mathit{\Pi F})
$$

$$
\frac{\overset{\mathcal{D}}{\Gamma \vdash A :_{\downarrow} \mathsf{s}} \quad [\![\mathcal{D}]\!]^{\mathsf{tm}} \Downarrow A' \quad \Gamma, x:A' \vdash M :_{\downarrow} B}{\Gamma \vdash \lambda x.M :_{\downarrow} (x{:}A) \to B}\,(\mathit{\Pi I}) \qquad
\frac{\Gamma \vdash M :_{\uparrow} (x{:}A) \to B \quad \Gamma \vdash N :_{\downarrow} A \quad B[N/x] \Downarrow B'}{\Gamma \vdash MN :_{\uparrow} B'}\,(\mathit{\Pi E})
$$

ただし、$\mathsf{s}, \mathsf{s}_1, \mathsf{s}_2 \in \{\mathsf{type}, \mathsf{kind}\}$

図 3: UDTT の型規則

$$
\lambda c.\mathbf{cherish}\left(s, \pi_1(@_1(c) : \begin{bmatrix} x{:}\mathbf{entity} \\ \mathbf{kingOf}(x, @_2(c) : \mathbf{entity}) \end{bmatrix})\right)
$$

図 4: (2) の意味表示

king による前提現象に、$@_2$ が *its* による照応現象にそれぞれ対応している。図 4 の意味表示に対する型推論によって、$@_1$ と $@_2$ はそれぞれ以下のような型を持つことがわかる。

$$
@_1 : d \to \begin{bmatrix} x{:}\mathbf{entity} \\ \mathbf{kingOf}(x, @_2(c) : \mathbf{entity}) \end{bmatrix}
$$

$$
@_2 : d \to \mathbf{entity}
$$

ここで、$@_1$ の型には $@_2$ が含まれている。言い換えれば、$@_1$ の型は $@_2$ に依存している。この 2 つの型に対して証明探索を行う場合、$@_2$ の証明探索を終え、その証明項を $@_1$ の型の中に現れる $@_2$ に代入した上で $@_1$ の証明探索を行わなければならない。すなわち、@ の型に対する証明探索は他の @ に依存していない @ から順に行っていかねばならず、正しい順序で証明探索を行わなければ照応解決はできないのである。しかし、[3] では証明探索の順序については何の制約も設けられていなかった。

2.2 Underspecified Dependent Type Theory (UDTT)

Bekki (2015)[2] において再定義された DTS の体系では、照応代名詞や前提トリガーの意味を未指定項 @ を用いて表示するのは変わらないが、その扱い方を変更している。最も大きな変更点は、依存型理論の体系と @ を加えた依存型理論の体系を区別して考える点である。以下、依存型理論の体系を DTT (Dependent Type Theory)、@ を加えた依存型理論の体系を UDTT (Underspecified Dependent Type Theory) と呼ぶ。以下の図 5 に UDTT の項の定義を示す。（DTT の項の定義は、UDTT の定義から @ に関する項を削除したものとほぼ同じである。）

$$
\begin{array}{rcl}
\Lambda_{\uparrow} & ::= & x \mid c \mid \mathsf{type} \mid (x{:}\Lambda_{\downarrow}) \to \Lambda_{\downarrow} \mid \Lambda_{\uparrow}\Lambda_{\downarrow} \mid \begin{bmatrix} x{:}\Lambda_{\downarrow} \\ \Lambda_{\downarrow} \end{bmatrix} \mid \pi_i(\Lambda_{\uparrow}) \mid @ : \Lambda_{\uparrow} \\
\Lambda_{\downarrow} & ::= & \Lambda_{\uparrow} \mid \lambda x.\Lambda_{\downarrow} \mid (\Lambda_{\downarrow}, \Lambda_{\downarrow}) \\
v & ::= & n \mid \mathsf{type} \mid \mathsf{kind} \mid (x{:}v) \to v' \mid \lambda x.v \mid \begin{bmatrix} x{:}v \\ v' \end{bmatrix} \mid (v, v') \\
n & ::= & x \mid c \mid nv \mid \pi_i(n) \mid @ : v
\end{array}
$$

図 5: UDTT の項の定義

前節でも述べたように、[3] には文（または談話）の意味表示の中に @ が複数存在した場合、どの @ から証明探索し、@ を解消していくのか、その順番が明示的でないという問題が存在した。これに対して [2] では、文の意味表示に対する型チェックを行う段階で @ の型に対する証明探索も同時に行うことでこの問題を解決している。

具体的には、[2] で再定義された DTS の体系において照応・前提現象を扱う際には、まず照応代名詞や前提トリガーの意味を @ を用いて表示し、@ を含む文の意味表示を得る。この意味表示は @ を含むので、UDTT の体系の意味表示である。次に、UDTT の体系の下で文の意味表示に対する型チェックを行い、型チェックの最中に @ の型が判明したらその時点で @ の型に対する証明探索を行う。ここで、文の意味表示中に複数の @ が存在する場合、他の @ に依存していないものから順に証明探索を行う。これにより、型チェックを行うことで全ての @ に対する証明項を得ることができる。文の意味表示に対する型チェックが終了したら、文の意味表示中の @ を証明探索によって得られた具体的な証明項によって置き換え（この操作を @-Elimination

$$\left[\!\!\left[\dfrac{\overset{\mathcal{D}_1}{\Gamma\vdash A:_{\downarrow}\mathsf{s}}\quad [\![\mathcal{D}]\!]^{\mathsf{tm}}\Downarrow A'\quad \overset{\mathcal{D}_2}{\Gamma\vdash M:A'}}{\Gamma\vdash(@:A):_{\uparrow}A}\,(@)\right]\!\!\right]=\overset{[\![\mathcal{D}_2]\!]}{\Gamma\vdash M:A'}\qquad \left[\!\!\left[\dfrac{\overset{\mathcal{D}_1}{\Gamma\vdash A:\mathsf{s}_1}\quad [\![\mathcal{D}_1]\!]^{\mathsf{tm}}\Downarrow A'\quad \overset{\mathcal{D}_2}{\Gamma,x:A'\vdash B:\mathsf{s}_2}}{\Gamma\vdash(x{:}A)\to B:\mathsf{s}_2}\,(\underline{\mathit{\Pi F}})\right]\!\!\right]=\dfrac{\overset{[\![\mathcal{D}_1]\!]}{\Gamma\vdash A':\mathsf{s}_1}\quad A'\Downarrow A''\quad \overset{[\![\mathcal{D}_2]\!]}{\Gamma,x:A''\vdash B':\mathsf{s}_2}}{\Gamma\vdash(x{:}A')\to B':\mathsf{s}_2}\,(\mathit{\Pi F})$$

$$\left[\!\!\left[\dfrac{\overset{\mathcal{D}_1}{\Gamma\vdash A:\mathsf{s}}\quad [\![\mathcal{D}_1]\!]^{\mathsf{tm}}\Downarrow A'\quad \overset{\mathcal{D}_2}{\Gamma,x:A'\vdash M:B}}{\Gamma\vdash\lambda x.M:(x{:}A)\to B}\,(\underline{\mathit{\Pi I}})\right]\!\!\right]=\dfrac{\overset{[\![\mathcal{D}_1]\!]}{\Gamma\vdash A':\mathsf{s}}\quad A'\Downarrow A''\quad \overset{[\![\mathcal{D}_2]\!]}{\Gamma,x:A''\vdash M':B'}}{\Gamma\vdash\lambda x.M':(x{:}A')\to B'}\,(\mathit{\Pi I})$$

$$\left[\!\!\left[\dfrac{\overset{\mathcal{D}_1}{\Gamma\vdash M:(x{:}A)\to B}\quad \overset{\mathcal{D}_2}{\Gamma\vdash N:A}\quad B[N/x]\Downarrow B'}{\Gamma\vdash MN:B'}\,(\underline{\mathit{\Pi E}})\right]\!\!\right]=\dfrac{\overset{[\![\mathcal{D}_1]\!]}{\Gamma\vdash M':(x{:}A')\to B'}\quad \overset{[\![\mathcal{D}_2]\!]}{\Gamma\vdash N':A'}\quad B'[N'/x]\Downarrow B''}{\Gamma\vdash M'N':B''}\,(\mathit{\Pi E})$$

$$\left[\!\!\left[\dfrac{\overset{\mathcal{D}_1}{\Gamma\vdash A:\mathsf{s}_1}\quad [\![\mathcal{D}_1]\!]^{\mathsf{tm}}\Downarrow A'\quad \overset{\mathcal{D}_2}{\Gamma,x:A'\vdash B:\mathsf{s}_2}}{\Gamma\vdash\begin{bmatrix}x{:}A\\ B\end{bmatrix}:\mathsf{s}_2}\,(\underline{\mathit{\Sigma F}})\right]\!\!\right]=\dfrac{\overset{[\![\mathcal{D}_1]\!]}{\Gamma\vdash A':\mathsf{s}_1}\quad A'\Downarrow A''\quad \overset{[\![\mathcal{D}_2]\!]}{\Gamma,x:A''\vdash B':\mathsf{s}_2}}{\Gamma\vdash\begin{bmatrix}x{:}A'\\ B'\end{bmatrix}:\mathsf{s}_2}\,(\mathit{\Sigma F})$$

$$\left[\!\!\left[\dfrac{\overset{\mathcal{D}_1}{\Gamma\vdash M:A}\quad [\![\mathcal{D}_1]\!]^{\mathsf{tm}}\Downarrow M'\quad B[M'/x]\Downarrow B'\quad \overset{\mathcal{D}_2}{\Gamma\vdash N:B'}}{\Gamma\vdash(M,N):\begin{bmatrix}x{:}A\\ B\end{bmatrix}}\,(\underline{\mathit{\Sigma I}})\right]\!\!\right]=\dfrac{\overset{[\![\mathcal{D}_1]\!]}{\Gamma\vdash M':A'}\quad B'[M'/x]\Downarrow B''\quad \overset{[\![\mathcal{D}_2]\!]}{\Gamma\vdash N:B''}}{\Gamma\vdash(M',N'):\begin{bmatrix}x{:}A'\\ B'\end{bmatrix}}\,(\mathit{\Sigma I})\qquad \text{ただし、}\mathsf{s},\mathsf{s}_1,\mathsf{s}_2\in\{\mathsf{type},\mathsf{kind}\}$$

$$\left[\!\!\left[\dfrac{\overset{\mathcal{D}}{\Gamma\vdash M:\begin{bmatrix}x{:}A\\ B\end{bmatrix}}}{\Gamma\vdash\pi_1(M):A}\,(\underline{\mathit{\Sigma E}})\right]\!\!\right]=\dfrac{\overset{[\![\mathcal{D}]\!]}{\Gamma\vdash M':\begin{bmatrix}x{:}A'\\ B'\end{bmatrix}}}{\Gamma\vdash\pi_1(M'):A'}\,(\mathit{\Sigma E})\qquad \left[\!\!\left[\dfrac{\overset{\mathcal{D}}{\Gamma\vdash M:\begin{bmatrix}x{:}A\\ B\end{bmatrix}}\quad B[\pi_1(M)/x]\Downarrow B'}{\Gamma\vdash\pi_2(M):B'}\,(\underline{\mathit{\Sigma E}})\right]\!\!\right]=\dfrac{\overset{[\![\mathcal{D}]\!]}{\Gamma\vdash M':\begin{bmatrix}x{:}A'\\ B'\end{bmatrix}}\quad B'[\pi_1(M')/x]\Downarrow B''}{\Gamma\vdash\pi_2(M):B''}\,(\mathit{\Sigma E})$$

図 6: @-Elinination の定義（抜粋）

という）、@ の含まれていない文の意味表示を得ることで照応解決となる。

2.3 UDTT での型チェック

UDTT における型規則の定義を図 3 に示す。UDTT の型チェック・型推論はその規則中で @-Elimination を呼び出しており、各規則に現れる $[\![\mathcal{D}]\!]^{\mathsf{tm}}$ が @-Elimination を行っている箇所である。@-Elimination は UDTT の証明木に対する操作であり、UDTT の証明木を受け取ったら、その中の @ を解消し、具体的な証明項で置き換えた DTT の証明木を返す。@-Elimination の定義（抜粋）は図 6 に示す。

前節において解説した、型チェックの最中に証明探索を行う操作を定式化しているのが (@) 規則である。(@) 規則では、アノテーション構文により @ の型として指定された型 A に対して型チェックを行った後、その型に対して証明探索を行っている。@ の型 A が他の @ に依存している場合には、A に対する型チェックの最中に A が依存している @ の証明探索を先に行う。その後 @-Elimination によって A から @ を解消した型 A' に対して証明探索を行う。

例として、文 (2) に対する型チェックの過程を考える。Bekki (2015)[2] で定義した体系での (2) の意味表示は図 7 のようになる。

$$\mathbf{cherish}\left(s,\pi_1(@_1:\begin{bmatrix}x{:}\mathbf{entity}\\ \mathbf{kingOf}(x,@_2:\mathbf{entity})\end{bmatrix})\right)$$

図 7: (2) の意味表示

この意味表示が型 type を持つかどうか型チェックを行うと、図 8（次頁）のような証明木が得られる。図 8 の証明木において、初めて $@_1$ に対する型推論が呼び出されるのが以下の部分木である。

$$\dfrac{\overset{\mathcal{D}}{\vdash\begin{bmatrix}x{:}\mathbf{entity}\\ \mathbf{kingOf}(x,@_2:\mathbf{entity})\end{bmatrix}:_{\downarrow}\mathsf{type}}\qquad \dfrac{(k,\begin{bmatrix}x{:}\mathbf{entity}\\ \mathbf{kingOf}(x,s)\end{bmatrix})\in\sigma}{\vdash k:\begin{bmatrix}x{:}\mathbf{entity}\\ \mathbf{kingOf}(x,s)\end{bmatrix}}\,(\mathit{CON})}{\vdash(@_1:\begin{bmatrix}x{:}\mathbf{entity}\\ \mathbf{kingOf}(x,@_2:\mathbf{entity})\end{bmatrix}):_{\uparrow}\begin{bmatrix}x{:}\mathbf{entity}\\ \mathbf{kingOf}(x,@_2:\mathbf{entity})\end{bmatrix}}\,(@)$$

$@_1$ に対する型推論が初めて呼び出された時点では、$@_1$ に対してアノテーション構文で指定された型 $\begin{bmatrix}x{:}\mathbf{entity}\\ \mathbf{kingOf}(x,@_2:\mathbf{entity})\end{bmatrix}$ に $@_2$ が含まれているため、証明探索を行うことができない。しかし (@) 規則の定義によって、先にアノテーション部分である $\begin{bmatrix}x{:}\mathbf{entity}\\ \mathbf{kingOf}(x,@_2:\mathbf{entity})\end{bmatrix}$ に対して型チェックが行われる。アノテーション部分に対する型チェックを行った結果の証明木が $\mathcal{D}$ である。$\mathcal{D}$ において $@_2$ に対する型推論と証明探索が呼び出されるのは次頁の部分木である。

$$
\dfrac{\dfrac{\dfrac{(\mathbf{cherish}, \begin{bmatrix} \mathbf{entity} \\ \mathbf{entity} \end{bmatrix} \to \mathsf{type}) \in \sigma}{\vdash \mathbf{cherish} :_{\uparrow} \begin{bmatrix} \mathbf{entity} \\ \mathbf{entity} \end{bmatrix} \to \mathsf{type}}\,(\underline{CON}) \quad \dfrac{\dfrac{\dfrac{(s, \mathbf{entity}) \in \sigma}{\vdash s :_{\uparrow} \mathbf{entity}}\,(\underline{CON})}{\vdash s :_{\downarrow} \mathbf{entity}}\,(\underline{CHK}) \quad \dfrac{\dfrac{\dfrac{\begin{matrix} \mathcal{D} \\ \vdash \begin{bmatrix} x{:}\mathbf{entity} \\ \mathbf{kingOf}(x, @_2 : \mathbf{entity}) \end{bmatrix} :_{\downarrow} \mathsf{type} \end{matrix} \quad \dfrac{(k, \begin{bmatrix} x{:}\mathbf{entity} \\ \mathbf{kingOf}(x, s) \end{bmatrix}) \in \sigma}{\vdash k : \begin{bmatrix} x{:}\mathbf{entity} \\ \mathbf{kingOf}(x, s) \end{bmatrix}}\,(\underline{CON})}{\vdash (@_1 : \begin{bmatrix} x{:}\mathbf{entity} \\ \mathbf{kingOf}(x, @_2 : \mathbf{entity}) \end{bmatrix}) :_{\uparrow} \begin{bmatrix} x{:}\mathbf{entity} \\ \mathbf{kingOf}(x, @_2 : \mathbf{entity}) \end{bmatrix}}\,(\underline{@})}{\vdash \pi_1(@_1 : \begin{bmatrix} x{:}\mathbf{entity} \\ \mathbf{kingOf}(x, @_2 : \mathbf{entity}) \end{bmatrix}) :_{\uparrow} \mathbf{entity}}\,(\underline{\Sigma E})}{\vdash \pi_1(@_1 : \begin{bmatrix} x{:}\mathbf{entity} \\ \mathbf{kingOf}(x, @_2 : \mathbf{entity}) \end{bmatrix}) :_{\downarrow} \mathbf{entity}}\,(\underline{CHK})}{\vdash (s, \mathbf{cherish}\left(s, \pi_1(@_1 : \begin{bmatrix} x{:}\mathbf{entity} \\ \mathbf{kingOf}(x, @_2 : \mathbf{entity}) \end{bmatrix})\right)) :_{\downarrow} \begin{bmatrix} \mathbf{entity} \\ \mathbf{entity} \end{bmatrix}}\,(\underline{\Sigma I})}{\vdash \mathbf{cherish}\left(s, \pi_1(@_1 : \begin{bmatrix} x{:}\mathbf{entity} \\ \mathbf{kingOf}(x, @_2 : \mathbf{entity}) \end{bmatrix})\right) :_{\uparrow} \mathsf{type}}\,(\underline{\Pi E})}{\vdash \mathbf{cherish}\left(s, \pi_1(@_1 : \begin{bmatrix} x{:}\mathbf{entity} \\ \mathbf{kingOf}(x, @_2 : \mathbf{entity}) \end{bmatrix})\right) :_{\downarrow} \mathsf{type}}\,(\underline{CHK})
$$

$$
\mathcal{D} \equiv \dfrac{\dfrac{\dfrac{\dfrac{(\mathbf{entity}, \mathsf{type}) \in \sigma}{\vdash \mathbf{entity} :_{\uparrow} \mathsf{type}}\,(\underline{CON})}{\vdash \mathbf{entity} :_{\downarrow} \mathsf{type}}\,(\underline{CHK}) \quad \dfrac{\dfrac{\dfrac{(\mathbf{kingOf}, \begin{bmatrix} \mathbf{entity} \\ \mathbf{entity} \end{bmatrix} \to \mathsf{type}) \in \sigma}{x : \mathbf{entity} \vdash \mathbf{kingOf} :_{\uparrow} \begin{bmatrix} \mathbf{entity} \\ \mathbf{entity} \end{bmatrix} \to \mathsf{type}}\,(\underline{CON}) \quad \dfrac{\dfrac{(x, \mathbf{entity}) \in \Gamma}{x : \mathbf{entity} \vdash x :_{\downarrow} \mathbf{entity}}\,(\underline{VAR}) \quad \dfrac{\dfrac{\dfrac{\dfrac{(\mathbf{entity}, \mathsf{type}) \in \sigma}{x : \mathbf{entity} \vdash \mathbf{entity} :_{\uparrow} \mathsf{type}}\,(\underline{CON})}{x : \mathbf{entity} \vdash \mathbf{entity} :_{\downarrow} \mathsf{type}}\,(\underline{CHK}) \quad \dfrac{(s, \mathbf{entity}) \in \sigma}{x : \mathbf{entity} \vdash s : \mathbf{entity}}\,(\underline{CON})}{x : \mathbf{entity} \vdash (@_2 : \mathbf{entity}) :_{\uparrow} \mathbf{entity}}\,(\underline{@})}{x : \mathbf{entity} \vdash (@_2 : \mathbf{entity}) :_{\downarrow} \mathbf{entity}}\,(\underline{CHK})}{x : \mathbf{entity} \vdash (x, @_2 : \mathbf{entity}) :_{\downarrow} \begin{bmatrix} \mathbf{entity} \\ \mathbf{entity} \end{bmatrix}}\,(\underline{\Sigma I})}{x : \mathbf{entity} \vdash \mathbf{kingOf}(x, @_2 : \mathbf{entity}) :_{\uparrow} \mathsf{type}}\,(\underline{\Pi E})}{x : \mathbf{entity} \vdash \mathbf{kingOf}(x, @_2 : \mathbf{entity}) :_{\downarrow} \mathsf{type}}\,(\underline{CHK})}{\vdash \begin{bmatrix} x{:}\mathbf{entity} \\ \mathbf{kingOf}(x, @_2 : \mathbf{entity}) \end{bmatrix} :_{\uparrow} \mathsf{type}}\,(\underline{\Sigma F})}{\vdash \begin{bmatrix} x{:}\mathbf{entity} \\ \mathbf{kingOf}(x, @_2 : \mathbf{entity}) \end{bmatrix} :_{\downarrow} \mathsf{type}}\,(\underline{CHK})
$$

図 8: (2) の意味表示に対する型チェック

$$
\dfrac{\dfrac{\dfrac{(\mathbf{entity}, \mathsf{type}) \in \sigma}{x : \mathbf{entity} \vdash \mathbf{entity} :_{\uparrow} \mathsf{type}}\,(\underline{CON})}{x : \mathbf{entity} \vdash \mathbf{entity} :_{\downarrow} \mathsf{type}}\,(\underline{CHK}) \quad \dfrac{(s, \mathbf{entity}) \in \sigma}{x : \mathbf{entity} \vdash s : \mathbf{entity}}\,(\underline{CON})}{x : \mathbf{entity} \vdash (@_2 : \mathbf{entity}) :_{\uparrow} \mathbf{entity}}\,(\underline{@})
$$

この部分木で $@_2$ の型 **entity** に対する証明探索を行うことで、$@_2$ に対する証明項が s (= Switzerland) であることが判明する。これにより、$@_1$ に対する証明探索では $@_1$ の型の中に現れる $@_2$ を s で置き換えた $\begin{bmatrix} x{:}\mathbf{entity} \\ \mathbf{kingOf}(x, s) \end{bmatrix}$ に対して証明探索を行えばよいことが分かり、その証明項として k (= The king of switzerland) が存在することが判明する。

このように、DTT と UDTT を区別して考え、UDTT の型チェックの際に証明探索も同時に行うよう定式化することで他の @ に依存していない @ から順に証明探索を行っていくことが可能となっている。

3 おわりに

本研究では Bekki and Sato (2015)[3] において問題となっていた、意味表示中に @ が複数存在する場合の意味処理に対応するための新たな DTS の体系 [2] を紹介し、その型チェック・型推論の体系を定式化した。

今後の課題としては、これらの体系を実装することがあげられる。[2] における DTS の体系では DTT と UDTT を区別して考えたため、その実装においては DTT と UDTT の両方を実装する必要がある。DTT の実装については [3] における実装を参考にすることで実現できると考えられる。UDTT の実装については、型チェックの際に証明探索や @-Elimination を同時に行わなくてはならず、さらに依存型の証明探索は一般に undecidable であるという難しさが存在する。しかし、[3] において依存型の構文的に制限された体系を用いることで DTS のための型推論の定式化・実装を行ったように、証明探索についても証明探索を行うことのできる制限された体系を用いることで実現できるのではないかと考えている。

参考文献

[1] Daisuke Bekki. Representing anaphora with dependent types. In *Logical Aspects of Computational Linguistics*, pages 14–29. Springer, 2014.

[2] Daisuke Bekki. Anaphora and presuppositions in dependent type semantics, 2015. A talk at Dynamic Semantics: Modern Type Theoretic and Category Theoretic Approaches, Ohio State University, United States, 24-25 October.

[3] Daisuke Bekki and Miho Satoh. Calculating projections via type checking. *Proceedings of TYTLES*, 2015.

[4] Andres Löh, Conor McBride, and Wouter Swierstra. A tutorial implementation of a dependently typed lambda calculus. *Fundamenta informaticae*, 102(2):177–207, 2010.

[5] Per Martin-Löf. *Intuitionistic Type Theory*, volume 17. Italy: Bibliopolis, Naples, 1984. Sambin, Giovanni (ed.).