

第 2 回

Rei Tomori

4/21/2025.

Overview

今回は以下について扱う：

- ◆ Section3.5: 評価
- ◆ Chapter5: 型無し λ 計算
 - ▶ Section5.1. 基礎
 - ▶ (以下，時間が許せば)
 - ▶ Section5.2. λ 計算でのプログラミング
 - ▶ Section5.3. 形式的議論

3.5. 評価

3.2. 評価

3.5. ブール式の操作的意味論

定義 (3.1: 型無しブール式の構文と評価規則)

ブール式の項と値の構文，および評価関係を次に定める：



(項の構文) $t ::= \text{true} \mid \text{false} \mid \text{if } t \text{ then } t \text{ else } t$



(値の構文) $v ::= \text{true} \mid \text{false}$



(E-IFTRUE) $\text{if true then } t_1 \text{ else } t_2 \rightarrow t_1$

(E-IFFALSE) $\text{if false then } t_1 \text{ else } t_2 \rightarrow t_2$

(E-IF)
$$\frac{t_1 \rightarrow t'_1}{\text{if } t_1 \text{ then } t_2 \text{ else } t_3 \rightarrow \text{if } t'_1 \text{ then } t_2 \text{ else } t_3}$$

3.5 評価規則

評価関係 $t \rightarrow t'$ は, "項 t が 1 ステップで t' に簡約される"ことを表わす.

評価規則は**計算規則**と**合同規則**に分けられる.

- ◆ 計算規則は状態遷移を定める.

- ▶ 規則 E-IFTRUE は状態 `if true then t_1 else t_2` を t_1 に遷移させる.

- ◆ 合同規則は項が条件部から評価されることを定める.

- ▶ つまり, 項 t が別の項 t' に 1step で簡約されるとき, 条件部として t を含む項は条件部として t' を含む項に 1step で評価される.
 - ▶ これ以外の合同規則はないので, 例えば項が `then` 節から評価されることはない.

3.5. 評価規則 (cont'd)

定義 (3.5.1: 規則のインスタンス)

規則の**インスタンス**は、各メタ変数を一貫してある項で置き換えたものである。

たとえば, `if true true else false → true` は規則 E-IFTRUE のインスタンス。

各評価規則は `sequent` とみなせるので、次の定義を得る。

定義 (3.5.2: 規則がある関係によって満たされること)

規則がある関係によって**満たされる**とは、規則の各インスタンスに対し、結論が関係を満たす、またはある前提が仮定を満たさないことである。

3.5: 規則上の関係の例 : 1 ステップの評価関係

定義 (3.5.3: 1 ステップの評価関係)

1 ステップの評価関係 \rightarrow は、定義 1 の 3 つの規則を満たす最小の二項関係である。組 (t, t') がこの評価関係の元であるとき、評価文 $t \rightarrow t'$ が**導出可能**であるという。

- ◆ "規則を満たす最小の二項関係"とすることで、評価関係式 $t \rightarrow t'$ が導出可能であることが組 (t, t') が評価関係を満たすことの必要十分条件であることが保証される。
- ▶ \Leftrightarrow 評価関係式 $t \rightarrow t'$ が導出可能であることを示すには、葉が計算規則のインスタンスで内部ノードが評価規則のインスタンスである導出木を構成すればよい。

3.5: 1 ステップ評価の決定性

定理 (3.5.4: 1 ステップ評価の決定性)

$t \rightarrow t'$ かつ $t \rightarrow t''$ ならば, $t' = t''$.

Proof. $t \rightarrow t'$ の導出に関する induction で示す. 以下 $t \rightarrow t'$ に至る導出で最後に使われた規則で分ける.

◆ E-IFTRUE が使われた場合.

- ▶ $t \rightarrow t'$ より小さな全ての導出で評価の決定性を仮定.
- ▶ いま, 項 t の形は $t = \text{if } t_1 \text{ then } t_2 \text{ else } t_3$, ただし $t_1 = \text{true}$.
- ▶ 項 t に他の規則は適用できない. たとえば E-IFFALSE が適用できるとすると, $t_1 = \text{false}$ となり矛盾. E-IF も同様.

3.5: 1 ステップ評価の決定性 (cont'd)

$t \rightarrow t'$ に至る導出で最後に適用された規則が E-IFFALSE の場合も全く同様.

◆ E-IF の場合.

- ▶ $t \rightarrow t'$ より小さな全ての導出が一意的であることを仮定する.
- ▶ いま, $t_1 \rightarrow t'_1$ かつ $t = \text{if } t_1 \text{ then } t_2 \text{ else } t_3$ かつ $t' = \text{if } t'_1 \text{ then } t_2 \text{ else } t_3$ なる項 t_1, t'_1, t_2, t_3 を取れる.
- ▶ t に対して他の規則を適用できるとすると t_1 は正規形でなければならない.
 $t_1 \rightarrow t'_1$ なる項 t'_1 の存在に反するので t に適用できる規則は E-IF に限られる.
- ▶ 更に, t を 1 ステップで簡約した t' は仮定により決定的.

以上より induction が完結し, 題意が示された. ■^{*1}

¹ 以上の形の帰納法は証明図の深さに関する帰納法と全く同一である.

3.5: 正規形

われわれは特に項の最終的な評価結果に興味があるので, "評価規則をこれ以上適用できない項"を定式化しよう.

定義 (3.5.6: 正規形)

いかなる評価規則も適用できない項を**正規形**という. すなわち, t が正規形であるとは, $\forall t', \neg(t \rightarrow t')$ なることである.

定理 (3.5.7)

任意の値は正規形である.

Proof. 評価関係より, 値に適用できる規則が存在しないことから直ちに従う. ■

3.5: 正規形 (cont'd)

今回の体系においては Thm.3.5.7 の逆も成立することに注意する． すなわち，以下が成り立つ．

定理 (3.5.8: Thm.3.5.7 の逆)

t が正規形ならば t は値である．

Proof. 対偶を示す． t が値でないことを仮定し， t に関する構造的帰納法により t が正規形でないことを示す． 仮定より， $t = \text{if } t_1 \text{ then } t_2 \text{ else } t_3$ なる項 t_1, t_2, t_3 が存在する． 以下， t_1 について場合を分ける．

いま， t の任意の部分項 t' に対して， t' が値でないならば t' が正規形でないことを仮定する．

3.5: 正規形 (cont'd)

- ◆ t_1 が値のとき. t に対して計算規則を適用できるので t は正規形でない.
- ◆ それ以外のとき. 帰納法の仮定から, 項 t'_1 が存在して $t_1 \rightarrow t'_1$. よって, 合同規則 E-IF から $t \rightarrow \text{if } t'_1 \text{ then } t_2 \text{ else } t_3$ が導出でき, t は正規形でないことが従う. ■

3.5: 多ステップ評価の関係

定義 (3.5.9: 多ステップ評価関係)

多ステップ評価関係 (\rightarrow^*) は, 1 ステップの評価関係 (\rightarrow) の反射推移閉包である.

すなわち, (\rightarrow^*) とは, (1) $\forall t, \forall t', t \rightarrow t' \Rightarrow t \rightarrow^* t'$, (2) $\forall t, t \rightarrow^* t$, および (3) $\forall t \forall t' \forall t'', t \rightarrow^* t' \wedge t' \rightarrow^* t'' \Rightarrow t \rightarrow^* t''$ を満たす最小の二項関係である. これは次の推論規則と等価である:

$$\frac{t \rightarrow t'}{t \rightarrow^* t'} \quad \frac{}{t \rightarrow^* t} \quad \frac{t \rightarrow t' \quad t' \rightarrow^* t''}{t \rightarrow^* t''}$$

3.5: 多ステップの評価規則 (cont'd)

次の主張は 1 ステップの評価関係が決定的であることから従う：

定理 (3.5.11. 正規形の一意性)

項 u, u' が正規形であり, $t \rightarrow^* u$ かつ $t \rightarrow^* u'$ ならば, $u = u'$.

Proof. 項 t, u, u' に対し, u, u' が正規形かつ $t \rightarrow^* u$ かつ $t \rightarrow^* u'$ なることを仮定する. 関係 $t \rightarrow^* u$ に関する induction で示す.

- ◆ $t \rightarrow u$ するとき. 仮定と Thm.3.5.4 より $u = u'$ が従う.
- ◆ $t = u$, すなわち t が値のとき. t に適用可能な規則はないので $u = u'$.
- ◆ $t \rightarrow^* t' \wedge t' \rightarrow^* u$ なる t' が存在するとき. Thm.3.5.4 を繰返し適用することで t' は特に一意的である. 更に $t' \rightarrow u$ に対して帰納法の仮定を適用すれば, 正規形 u が一意的であることが従う. ■

3.5: 評価の停止性

最後に、 \mathbb{B} の全ての項が正規形に評価できる，すなわち評価は有限ステップで停止することを示そう．証明の方針としては次のようになる．*2

- ◆ 整礎集合 $S = (S, <_S)$ と，項の集合から S への関数 f を用意する．
- ◆ 抽象機械の状態が t のとき， t' に遷移できるならば $f(t') <_S f(t)$ なることを示す．
- ◆ $f(t)$ を初項とする無限降下列が存在したとしよう．このとき， S の well-foundedness からこのような無限降下列は存在しえず矛盾．ここから任意の評価列が有限であることが従う．

関数 f は評価関係の**停止尺度**と呼ばれる．

2 この性質を満たさない言語も存在する．例えば，評価戦略として値呼びを取った場合の型無し λ 計算では，後に見るように Ω コンビネータ $(\lambda x.x x)(\lambda x.x x)$ は正規形に簡約されえない．

3.5: 評価の停止性 (cont'd)

定理 (3.5.12. 評価の停止性)

言語 \mathbb{B} の任意の項 t に対して，正規形 t' が存在して $t \rightarrow t'$.

Proof. 停止尺度を項のサイズとし，その定義域を \mathbb{N} に制限する .

いま，任意の項 t に対し，評価規則を適用するごとに項のサイズは減少することに注意する . \mathbb{N} は整礎集合であり，任意の評価列は有限なので題意が従う . ■

演習 3.5.15: \mathbb{B}

問題 (演習 3.5.13)

1. 規則 $\text{if true then } t_2 \text{ else } t_3 \rightarrow t_3$ を追加したとき, Thm.3.5.4, 3.5.7, 3.5.8, 3.5.11, 3.5.12 のうち成立するもの.
2. 規則 $t_2 \rightarrow t'_2 \vdash \text{if } t_1 \text{ then } t_2 \text{ else } t_3 \rightarrow \text{if } t_1 \text{ then } t'_2 \text{ else } t_3$ を追加したとき, 以上のうち成立するもの.

Proof. 1. 条件部が `true` のとき, とくに t_2, t_3 の評価先が一致しなければ評価が非決定的になることに注意. 定理 3.5.7, 3.5.8, 3.5.12 のみが成立.

2. Thm.3.5.4 以外は成り立つが³Thm.3.5.8, 3.5.11 は証明の修正が必要.

³ 条件節と `then` 節が正規形で, それらの評価先が一致しない場合を考えればよい.

演習 3.5.15(cont'd)

以下の補題を示そう：

補題 (ダイヤモンド性)

項 $r, s, t (s \neq t)$ に対し, $r \rightarrow s$ かつ $r \rightarrow t$ ならば, 項 u が存在して $s \rightarrow u$ かつ $t \rightarrow u$

Proof. $r = \text{if true then } t_2 \text{ else } t_3$ の形で, $t_2(t_3)$ が正規形でない場合のみ示す*4. 仮定より, $t_2 \rightarrow t'_2$ なる項 t'_2 が取れる. このとき, r は

◆ E-IFTRUE の適用後 t_2 を評価

◆ E-FUNNY₂ の適用後 E-IFTRUE を適用

の二通りの評価ができ, 評価先は共に t'_2 となることから従う.

4 条件節が他の形の場合も, E-FALSE や E-IF などでも評価すれば同様

演習 3.5.15(cont'd)

補題の内容を踏まえると, Thm.3.5.8~3.5.12 の証明はそれぞれ次のように修正される:

- ◆ Thm.3.5.8: 帰納部で, t_2 が正規形でない場合を足せばよい.
- ◆ Thm.3.5.11(概略): $r \rightarrow^* s \wedge r \rightarrow^* t$ としよう. このとき, 補題を繰返し適用することで $s \rightarrow^* u, t \rightarrow^* u$, ひいては $r \rightarrow^* u, s \rightarrow^* u$ なる正規形 u の存在が示せ, しかもそれは一意的.

3.5: 算術式への拡張

値の定義に数値を表わす構文要素を足し，評価の定義を算術式に拡張しよう．ブール式と共通した構文要素は (...) と略記する．

定義 (3.2: 型無し算術式の構文)

◆ (項の構文)

$$t ::= \dots \mid \mathbf{0} \mid \mathbf{succ} \ t \mid \mathbf{pred} \ t \mid \mathbf{iszero} \ t$$

◆ (値の構文)

$$\begin{aligned} v &::= \dots nv \\ nv &::= \mathbf{0} \mid \mathbf{succ} \ nv \end{aligned}$$

3.5: 算術式への拡張 (cont'd)

定義 (3.2: 型無し算術式の評価規則)

- ◆ 型無し算術式の評価規則は、以下を満たす最小の二項関係である。

(E-SUCC) $t_1 \rightarrow t'_1 \vdash \text{succ } t_1 \rightarrow \text{succ } t'_1$

(E-PREDZERO) $\text{pred } 0 \vdash 0$

(E-PREDSUCC) $\text{pred}(\text{succ } nv_1) \vdash nv_1$

(E-PRED) $t_1 \rightarrow t'_1 \vdash \text{pred } t_1 \rightarrow \text{pred } t'_1$

(E-ISZEROZERO) $\text{iszero } 0 \vdash \text{true}$

(E-ISZEROSUCC) $\text{iszero } (\text{succ } nv_1) \vdash \text{false}$

(E-ISZERO) $t_1 \rightarrow t'_1 \vdash \text{iszero } t_1 \rightarrow \text{iszero } t'_1$

3.5: 行き詰まり状態とその扱い

p.19 の例のように，この言語には評価規則が適用できず，値でもない項が存在する．これらは実行時エラーの特徴付けに用いられる．

定義 (3.5.15: 行き詰まり状態)

項が正規形であるが値ではないとき，その項は**行き詰まり状態**であるという．

実行時エラーを特徴付ける方法はいまだではない．例えば Def.3.2 で定めた構文を拡張し，全ての行き詰まり先の評価先となる項 `wrong` を導入することを考える．(次頁)

3.5: 行き詰まり状態とその扱い (cont'd)

問題 (Def.3.2 の拡張)

◆ 構文要素

- ▶ $\text{badnat} ::= \text{wrong} \mid \text{true} \mid \text{false}$
- ▶ $\text{badbool} ::= \text{wrong} \mid \text{nv}$

◆ 評価関係

- ▶ $\text{if badbool then } t_1 \text{ else } t_2 \rightarrow \text{wrong}$
- ▶ $\text{succ badnat} \rightarrow \text{wrong}$
- ▶ $\text{pred badnat} \rightarrow \text{wrong}$
- ▶ $\text{iszero badnat} \rightarrow \text{wrong}$

3.5: 行き詰まり状態とその扱い (cont'd)

問題 (演習 3.5.16)

実行時エラーの 2 つの取り扱いが一致することを示せ。

Proof. Def.3.2 の言語を \mathcal{L} , \mathcal{L} を **wrong** 項で拡張した言語を \mathcal{L}_w とかき, \mathcal{L} 上の 1 ステップ評価関係を (\xrightarrow{o}) , \mathcal{L}_w 上の評価関係を (\xrightarrow{w}) で定める。

項 g が行き詰まり状態に評価されることと, g が \mathcal{L}_w 上で **wrong** に評価されることが同値であることを示す:

$$\forall g, (g \xrightarrow{o^*} g' \wedge \forall w, \neg(g' \xrightarrow{o} w)) \Leftrightarrow g \xrightarrow{w^*} \mathbf{wrong}$$

演習 3.5.16(cont'd)

まず、 \mathcal{L} 上で行き詰まる項は \mathcal{L}_w で正規形に評価されない事を確認する． 実際、評価規則の形より (\xrightarrow{w}) が決定的であることから従う． あとは必要性和十分性に分けて示せばよい．

補題 (\Rightarrow)

任意の \mathcal{L} の項 g について、 g が行き詰まり状態 $\Rightarrow g \xrightarrow{w^*} \text{wrong}$.

略証． g の構造帰納法による．

- ◆ g がブール値または 0 のとき． g は行き詰まりでなく $g \xrightarrow{w^*} \text{wrong}$ が従う．
- ◆ $g = \text{if } g_1 \text{ then } g_2 \text{ else } t_3$ のとき． g は行き詰まるので、 g_1 は行き詰まるか数値に評価される．

演習 3.5.16(cont'd)

◆ $g = \text{if } g_1 \text{ then } g_2 \text{ else } g_3$ のとき (cont'd)

- ▶ g_1 が行き詰まるとき. 帰納法の仮定より $g_1 \xrightarrow{w^*} \text{wrong}$, ひいては評価関係から $g \xrightarrow{w^*} \text{wrong}$.
- ▶ g_1 が数値のとき. g_1 は定義より `badbool` に属するから $g \rightarrow^* \text{wrong}$.

◆ $g = \text{succ } g_1$ のとき. g は行き詰まるので g_1 は行き詰まる. 帰納法の仮定より $g_1 \xrightarrow{*} \text{wrong}$. ゆえに評価関係より $g' \xrightarrow{w^*} \text{succ wrong} \xrightarrow{w} \text{wrong}$.

- ▶ $g = \text{pred } g_1, g = \text{iszero } g_1$ が行き詰まる場合も同様.

以上より induction が完結し, \mathcal{L}_w 上の項の評価の決定性と合わせることで主張が示される.

演習 3.5.16(cont'd)

逆を示す.

補題 (\Leftarrow)

\mathcal{L}_w において $g \xrightarrow{w^*} t$ かつ t が **wrong** を部分項として含むならば, g は \mathcal{L} において行き詰まり状態である.

Proof. 導出に関する帰納法に依る. t を部分項に **wrong** を含む項としよう. このとき, $g \xrightarrow{w^*} t$ に至る導出で最後に使われた規則で分ける. このような t を導出する評価規則は E-SUCC, E-PRED, E-ISZERO, および \mathcal{L}_w で追加された評価規則に限られるから, これらについてのみ示せばよい.

規則 $\text{iszero badnat} \xrightarrow{w^*} \text{wrong}$ の場合のみ示すことにする.

演習 3.5.16(cont'd)

$g \xrightarrow{w^*} t$ より小さな任意の導出 $\frac{\vdots}{g' \xrightarrow{w^*} t'}$ (但し t' は subterm に **wrong** を含むとする) おいて題意が成立することを仮定し, $g = \text{iszero } u$ (u は $u \xrightarrow{w^*} \text{badnat}$ を満たす) とする. このとき, u' を $u \xrightarrow{w^*} u' \xrightarrow{w^*} \text{badnat}$ なる項として, 導出図は次の形に限られる:

$$\begin{array}{c}
 \frac{\frac{\frac{\vdots}{u' \xrightarrow{w^*} u'}}{\text{iszero } u \xrightarrow{w^*} \text{iszero } u'}}{\text{iszero } u \xrightarrow{w^*} \text{iszero } u'} \quad \frac{\frac{\frac{\vdots}{\text{iszero } u' \xrightarrow{w^*} \text{iszero } \text{badnat}}}{\text{iszero } u' \xrightarrow{w^*} \text{iszero } \text{badnat}} \quad \frac{\text{iszero } \text{badnat} \xrightarrow{w} \text{wrong}}{\text{iszero } \text{badnat} \xrightarrow{w^*} \text{wrong}}}{\text{iszero } u' \xrightarrow{w^*} \text{wrong}} \\
 \hline
 \text{iszero } u \xrightarrow{w^*} \text{wrong}
 \end{array}$$

.

演習 3.5.16(cont'd)

あとは `badnat` が `wrong` か否かについて分ける.

1. `badnat` が `wrong` のとき. 帰納法の仮定を導出図の右から 2 番目の孫に適用し, `iszero u'` は \mathcal{L} において行き詰まり状態である. ひいては, g が行き詰まり状態であることが従う.
2. それ以外のとき. `badnat` の定義より u' は `true` か `false` である. 帰納法の仮定を導出図の右側の子に適用し, `iszero u'` は \mathcal{L} において行き詰まり状態であるから, 1 同様にして g が行き詰まり状態であることが従う.

いずれの場合も g は行き詰まるから, この規則に対しては帰納法が完結する. 他の場合も同様である.

演習 3.5.17: 自然意味論と多ステップ評価関係の等価性

問題 (演習 3.5.17)

$t \Downarrow v \Leftrightarrow t \rightarrow^* v$. ただし, 項 t と値 v に対し, $t \Downarrow v$ は p.32 の規則に従い, t が v に最終的に評価されることを表わす.

Proof. まず, $t \Downarrow v \Rightarrow t \rightarrow^* v$ を, $t \Downarrow v$ の導出に関する induction で示す. 最後に使われる規則について分ければよい:

- ◆ B-VALUE の場合. このとき $t = v$ で, (\rightarrow^*) の反射性から従う.
- ◆ t の任意の部分式 t' に対し $t' \Downarrow v' \Rightarrow t' \rightarrow^* v'$ を仮定し, t について場合分けする.
 - ▶ B-IFTRUE の場合. $t_1 \Downarrow \text{true}$ かつ $t_2 \Downarrow v_2$ とする. 帰納法の仮定から, $t_1 \rightarrow^* \text{false}$ かつ $t_2 \rightarrow^* v_2$.

演習 3.5.17(cont'd)

◆ inductive case

▶ B-IFTRUE, B-IFFALSE の場合. (cont'd)

- $\text{if } t_1 \text{ then } t_2 \text{ else } t_3 \Downarrow v_2$ を仮定. 推移性と合同規則から,

$$\text{if } t_1 \text{ then } t_2 \text{ else } t_3 \rightarrow^* \text{if true then } t_2 \text{ else } t_3 \rightarrow t_2 \rightarrow^* v_2$$

ゆえに $\text{if } t_1 \text{ then } t_2 \text{ else } t_3 \Downarrow v_2 \Rightarrow \text{if } t_1 \text{ then } t_2 \text{ else } t_3 \rightarrow^* v_2$. B-IFFALSE も全く同様.

- ▶ B-SUCC の場合. $t_1 \Downarrow nv_1$ とすると, 帰納法の仮定から $t_1 \rightarrow^* nv_1$. いま, $\text{succ } t_1 \Downarrow \text{succ } nv_1$ なることを仮定して $\text{succ } t_1 \rightarrow^* \text{succ } nv_1$ を示そう. 規則 E-SUCC を $t_1 \rightarrow^* nv_1$ の各ステップに適用し, $\text{succ } t_1 \rightarrow^* \text{succ } nv_1$. ゆえに $\text{succ } t_1 \Downarrow \text{succ } nv_1 \Rightarrow \text{succ } t_1 \rightarrow^* \text{succ } nv_1$ が従う.

演習 3.5.17(cont'd)

◆ inductive case

- ▶ B-PREDZERO, B-PREDSUCC の場合. B-PREDSUCC についてのみ示す.
 - $t_1 \Downarrow \text{succ } nv_1$ を仮定すると, 帰納法の仮定から $t_1 \rightarrow^* \text{succ } nv_1$. $\text{pred } t_1 \Downarrow nv_1$ とすると, 仮定, E-PRED と E-PREDSUCC より $\text{pred } t_1 \rightarrow^* \text{pred}(\text{succ } nv_1) \rightarrow nv_1$.
- ▶ B-ISZEROZERO, B-ISZEROSUCC の場合. B-ISZEROSUCC についてのみ示す.
 - $t_1 \Downarrow \text{succ } nv_1$ を仮定する. 帰納法の仮定より $t_1 \rightarrow^* \text{succ } nv_1$.
 - $\text{iszero } t_1 \Downarrow \text{false}$ とする. 仮定, E-ISZERO および E-ISZEROSUCC より

$$\text{iszero } t_1 \rightarrow^* \text{iszero}(\text{succ } nv_1) \rightarrow \text{false}$$

となるから, $\text{iszero } t_1 \Downarrow \text{false} \Rightarrow \text{iszero } t_1 \rightarrow^* \text{false}$ が従う.

演習 3.5.17(cont'd)

逆, すなわち $t \rightarrow^* v \Rightarrow t \Downarrow v$ を示そう.

$t \rightarrow^* v$ の導出における small-step の評価のステップ数 n に関する induction で示す.

◆ $n = 0$ の場合. $t = v$ であり, B-VALUE から従う.

◆ $t \rightarrow^* v$ が n ステップの評価で得られたとする. n 未満の評価列に於いて主張を仮定する. t の形について場合を分けて示す.

▶ $t = \text{if } t_1 \text{ then } t_2 \text{ else } t_3$ の場合. t_1 の最終的な評価先で分ける.

- $t_1 \rightarrow^* \text{true}$ のとき. $t \rightarrow^* v$ より $t_2 \rightarrow^* v$ で, これらを得る評価列の長さは n 未満. 帰納法の仮定から $t_1 \Downarrow \text{true} \wedge t_2 \Downarrow v$. 故に B-IFTRUE から $t \Downarrow v$.

◆ B-IFFALSE や他の規則でも同様. ゆえに帰納法が完結する.

演習 3.5.18: 評価規則の変更

問題 (演習 3.5.18)

評価戦略を変えて, if 式の then 節と else 節が (この順序で) 条件部より先に評価されるようにしたいとする. この効果を達成するために評価規則をどのように変えればよいか.

(解) big-step semantics を与えれば十分である. then 節の後に else 節を値に評価し, 最後に条件節を評価して式全体を値に評価すればよい.

そこで, E-IFTRUE, E-IFFALSE を削除し, 以下の規則を追加する (次ページ):

演習 3.5.18: 評価規則の変更

1.
$$\frac{t_2 \Downarrow v_2}{\text{if } t_1 \text{ then } t_2 \text{ else } t_3 \Downarrow \text{if } t_1 \text{ then } v_2 \text{ else } t_3}$$
2.
$$\frac{\text{if } t_1 \text{ then } t_2 \text{ else } t_3 \Downarrow \text{if } t_1 \text{ then } v_2 \text{ else } t_3 \quad t_3 \Downarrow v_3}{\text{if } t_1 \text{ then } t_2 \text{ else } t_3 \Downarrow \text{if } t_1 \text{ then } v_2 \text{ else } v_3}$$
3.
$$\frac{\text{if } t_1 \text{ then } t_2 \text{ else } t_3 \Downarrow \text{if } t_1 \text{ then } v_2 \text{ else } v_3 \quad t_1 \Downarrow \text{true}}{\text{if } t_1 \text{ then } t_2 \text{ else } t_3 \Downarrow v_2}$$
4.
$$\frac{\text{if } t_1 \text{ then } t_2 \text{ else } t_3 \Downarrow \text{if } t_1 \text{ then } v_2 \text{ else } v_3 \quad t_1 \Downarrow \text{false}}{\text{if } t_1 \text{ then } t_2 \text{ else } t_3 \Downarrow v_3}$$