Contents

1 Introduction

2 Basic Definitions

We will denote the set $\{1, \ldots, n\}$ by $[n]$.

2.1 λ-calculus $\lambda 2$

$FV(\Gamma) = \bigcup \{FV(t) \mid (x : t) \in \Gamma\}$

In the following let $\mathcal{V}_T = \{\alpha, \beta, \ldots\}$ be a countable set (of type-variables) and $\mathcal{V}_V = \{x_1, x_2, \ldots\}$ be a countable set (of value-variables).

Definition 1. The set of all $\lambda 2$ types over $\mathcal{V}_T$, denoted by $\mathrm{T}_{\lambda 2}$, is the smallest set T satisfying the following conditions:

- $\mathcal{V}_T \subseteq \mathrm{T}$,

- if $t_1, t_2 \in \mathrm{T}$ then $t_1 \to t_2 \in \mathrm{T}$, and

- if $t \in \mathrm{T}$ and $\alpha \in \mathcal{V}_T$ then $\forall \alpha . t \in \mathrm{T}$.

Definition 2. The set of all $\lambda 2$ terms over $\mathcal{V}_T$ and $\mathcal{V}_V$, denoted by $\Lambda_{\mathrm{T}_{\lambda 2}}$, is the smallest set Λ_T satisfying the following conditions:

- $\mathcal{V}_V \subseteq \Lambda_T$,

- if $e_1, e_2 \in \Lambda_T$ then $e_1 e_2 \in \Lambda_T$,

- if $x \in \mathcal{V}_V$, $t \in \mathrm{T}_{\lambda 2}$, and $e \in \Lambda_T$ then $\lambda x : t.e \in \Lambda_T$,

- if $\alpha \in \mathcal{V}_T$ and $e \in \Lambda_T$ then $\Lambda \alpha . e \in \Lambda_T$, and

- if $e \in \Lambda_T$ and $t \in \mathrm{T}_{\lambda 2}$ then $e\, t \in \Lambda_T$.

Definition 3. Let $e \in \Lambda_{\mathrm{T}_{\lambda 2}}$. The free variables of e, denoted by $\mathrm{FV}(e)$, are defined inductively as follows:

$$
\mathrm{FV}(e) = \begin{cases}
\{x\} & \text{if } e = x \\
\mathrm{FV}(e_1) \cup \mathrm{FV}(e_2) & \text{if } e = e_1 e_2 \\
\mathrm{FV}(e') \setminus \{x\} & \text{if } e = \lambda x : t.e' \\
\mathrm{FV}(e') & \text{if } e = \Lambda \alpha . e' \\
\mathrm{FV}(e') & \text{if } e = e'\, t
\end{cases}
$$

Or is this definition better?

Definition 4. Let $e \in \Lambda_{T_{\lambda 2}}$. The <u>free variables of e</u>, denoted by $\text{FV}(e)$, are defined inductively as follows:

$$\text{FV}(y) = \{x\}$$
$$\text{FV}(e_1 e_2) = \text{FV}(e_1) \cup \text{FV}(e_2)$$
$$\text{FV}(\lambda x : t.e') = \text{FV}(e') \setminus \{x\}$$
$$\text{FV}(\Lambda \alpha.e') = \text{FV}(e')$$
$$\text{FV}(e'\,t) = \text{FV}(e')$$

Definition 5. A <u>basis</u> is a finite subset of $\mathcal{V}_V \times \Lambda_{T_{\lambda 2}}$

$\lambda 2$ deduction Rules

(Axiom)	$\Gamma, x : t \vdash x : t$
(λ-Introduction)	$\dfrac{\Gamma, x : t_1 \vdash e : t_2}{\Gamma \vdash \lambda x.e : t_1 \to t_2}$
(λ-Elimination)	$\dfrac{\Gamma \vdash e_1 : t_1 \to t_2 \quad \Gamma \vdash e_2 : t_1}{\Gamma \vdash e_1 e_2 : t_2}$
($\forall$-Introduction)	$\dfrac{\Gamma \vdash e : t}{\Gamma \vdash \Lambda \alpha.e : \forall \alpha.t} \qquad \alpha \notin \text{FV}(\Gamma)$
($\forall$-Elimination)	$\dfrac{\Gamma \vdash e : \forall \alpha.t}{\Gamma \vdash e\,t' : t\,[\alpha := t']} \qquad t' \in T_{\lambda 2}$

2.2 first-order logic

Definition 6. A <u>ranked set</u> is a tuple (Σ, rk), where Σ is a countable set and $rk : \Sigma \to \mathbb{N}$ is a function that maps every symbol from Σ to a natural number (its rank).

If the function rk is understood we will just write Σ instead of (Σ, rk). The set of all elements with a certain rank k in Σ, denoted by $\Sigma^{(k)}$, is defined by $\Sigma^{(k)} := rk^{-1}(k)$. In the following we will write $\Sigma = \{P^{(0)}, Q^{(3)}\}$ to say that $\Sigma = \{P, Q\}$, $rk(P) = 0$, and $rk(Q) = 3$.

In the following let $\mathcal{V} = \{x_0, x_1, \dots\}$ be a countable set (of variables), $\mathcal{F}$ a ranked set (of function symbols), and $\mathcal{P}$ a ranked set (of predicate symbols).

Definition 7. The set of <u>terms over $(\mathcal{V}, \mathcal{F})$</u>, denoted by $\mathcal{T}_{(\mathcal{V}, \mathcal{F})}$, is the smallest set $\mathcal{T}$ satisfying the following conditions:

- $\mathcal{V} \subseteq \mathcal{T}$, and

- for every $k \in \mathbb{N}$ if $f \in \mathcal{F}^{(k)}$ and $t_1, t_2, \ldots, t_k \in \mathcal{T}$ then $f(t_1, t_2, \ldots, t_k) \in \mathcal{T}$.

The set of <u>first-order formulas over $(\mathcal{V}, \mathcal{F}, \mathcal{P})$</u>, denoted by $\mathcal{L}_{(\mathcal{V}, \mathcal{F}, \mathcal{P})}$, is the smallest set $\mathcal{L}$ satisfying the following conditions:

- for every $k \in \mathbb{N}$ if $P \in \mathcal{P}^{(k)}$ and $t_1, t_2, \ldots, t_k \in \mathcal{T}_{(\mathcal{V}, \mathcal{F})}$ then $P(t_1, t_2, \ldots, t_k) \in \mathcal{L}$.

- If $\varphi, \psi \in \mathcal{L}$ then $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $\neg \varphi \in \mathcal{L}$, and

- if $x \in \mathcal{V}$ and $\varphi \in \mathcal{L}$ then $\exists x \varphi, \forall x \varphi \in \mathcal{L}$.

We introduce an additional binary operation $\to$ on formulas, where for some $\varphi, \psi \in \mathcal{L}_{(\mathcal{V}, \mathcal{F}, \mathcal{P})}$ the formula $(\varphi \to \psi)$ is defined as $(\neg \varphi \vee \psi)$.

Definition 8. The <u>variables of a term $t \in \mathcal{T}_{(\mathcal{V}, \mathcal{F})}$</u>, denoted by $\mathrm{V}(t)$, are defined by:

$$
\mathrm{V}(t) = \begin{cases} \{x\} & \text{if } t = x \\ \mathrm{V}(t_1) \cup \mathrm{V}(t_2) \cup \cdots \cup \mathrm{V}(t_k) & \text{if } t = f(t_1, t_2, \ldots, t_k) \end{cases}
$$

The <u>free variables of a formula $\varphi \in \mathcal{L}_{(\mathcal{V}, \mathcal{F}, \mathcal{P})}$</u>, denoted by $\mathrm{FV}(\varphi)$, are defined as follows:

$$
\mathrm{FV}(\varphi) = \begin{cases} \mathrm{V}(t_1) \cup \mathrm{V}(t_2) \cup \cdots \cup \mathrm{V}(t_k) & \text{if } \varphi = P(t_1, t_2, \ldots, t_k) \\ \mathrm{FV}(\varphi_1) \cup \mathrm{FV}(\varphi_2) & \text{if } \varphi = \varphi_1 \circ \varphi_2,\ \circ \in \{\wedge, \vee\} \\ \mathrm{FV}(\psi) \setminus \{x\} & \text{if } \varphi = Qx\psi,\ Q \in \{\forall, \exists\} \end{cases}
$$

Definition 9. Let x be in $\mathcal{V}$ and $t, t' \in \mathcal{T}_{(\mathcal{V}, \mathcal{F})}$. The <u>substitution of x by t' in t</u>, denoted by $t\,[x := t']$, is defined as follows:

$$
t\,[x := t'] = \begin{cases} t' & \text{if } t = x \\ y & \text{if } t = y \text{ and } y \neq x \\ f(t_1\,[x := t'], \ldots, t_k\,[x := t']) & \text{if } t = f(t_1, \ldots, t_k) \end{cases}
$$

Now we can lift this definition to formulas, let φ be in $\mathcal{L}_{(\mathcal{V}, \mathcal{F}, \mathcal{P})}$. The <u>substitution of x by t' in φ</u>, denoted by $\varphi\,[x := t']$, is defined as follows:

$$
\varphi\,[x := t'] = \begin{cases} P(t_1\,[x := t'], \ldots, t_k\,[x := t']) & \text{if } \varphi = P(t_1, \ldots, t_k) \\ \psi\,[x := t'] & \text{if } \varphi = \neg \psi \\ \varphi_1\,[x := t'] \circ \varphi_2\,[x := t'] & \text{if } \varphi = (\varphi_1 \circ \varphi_2),\ \circ \in \{\wedge, \vee\} \\ \varphi & \text{if } \varphi = Qx\psi,\ Q \in \{\forall, \exists\} \\ Qy(\psi\,[x := t']) & \text{if } \varphi = Qy\psi,\ Q \in \{\forall, \exists\} \text{ and } y \neq x \end{cases}
$$

Now we come to the semantics of first-order formulas.

Definition 10. An <u>interpretation I over $(\mathcal{V}, \mathcal{F}, \mathcal{P})$</u> is a triple $(\Delta, \cdot^I, \omega)$

where Δ is a nonempty set (which we call domain),

$\cdot^I$ is a function such that

$f^I : \Delta^k \to \Delta$ is a function for every $k \in \mathbb{N}$, $f \in \mathcal{F}^{(k)}$ and

$P^I \subseteq \Delta^k$ is a relation for every $k \in \mathbb{N}$, $f \in \mathcal{P}^{(k)}$

ω is a function from $\mathcal{V}$ to Δ.

Let $I = (\Delta, \cdot^I, \omega)$ be an interpretation, $x \in \mathcal{V}$, and $d \in \Delta$ the interpretation $I\,[x \to d]$ is defined as $(\Delta, \cdot^I, \omega\,[x \to d])$ where

$$(\omega\,[x \to d])(y) = \begin{cases} d & \text{if } y = x \\ \omega(y) & \text{otherwise.} \end{cases}$$

Definition 11. Let $I = (\Delta, \cdot^I, \omega)$ be an interpretation and t a term the <u>interpretation of t under I</u>, denoted by t^I, is defined as follows:

$$t^I = \begin{cases} \omega(x) & \text{if } t = x \\ f^I(t_1^I, \ldots, t_k^I) & \text{if } t = f(t_1, \ldots, t_k) \end{cases}$$

Definition 12. Let $I = (\Delta, \cdot^I, \omega)$ be an interpretation and φ a formula the <u>interpretation of φ under I</u>, denoted by φ^I, is defined recursively as follows:

$$\varphi^I = \begin{cases} \top & \text{if } \varphi = P(t_1, \ldots, t_k) \text{ and } (t_1^I, \ldots, t_k^I) \in P^I \\ \bot & \text{if } \varphi = P(t_1, \ldots, t_k) \text{ and } (t_1^I, \ldots, t_k^I) \notin P^I \\ \text{not } \psi^I & \text{if } \varphi = \neg\psi \\ \varphi_1^I \text{ and } \varphi_2^I & \text{if } \varphi = (\varphi_1 \wedge \varphi_2) \\ \varphi_1^I \text{ or } \varphi_2^I & \text{if } \varphi = (\varphi_1 \vee \varphi_2) \\ \text{exists } d \in \Delta \; \psi^{I[x \to d]} & \text{if } \varphi = \exists x \psi \\ \text{forall } d \in \Delta \; \psi^{I[x \to d]} & \text{if } \varphi = \forall x \psi \end{cases}$$

The interpretation I is a <u>model</u> of φ, denoted by $I \models \varphi$, if $\varphi^I = \top$.

Definition 13. Let Γ be a finite set of first-oder formulas.

We say that an interpretation I is a <u>model</u> of Γ if $I \models \psi$ for every ψ in Γ.

The formula φ is a <u>semantic consequence</u> of Γ, denoted by $\Gamma \vdash \varphi$, if every model of Γ is also a model of φ.

The free variables of Γ, denoted by $\mathrm{FV}(\Gamma)$, are $\bigcup \{\mathrm{FV}(\varphi) \mid \varphi \in \Gamma\}$.

2.3 two-counter automaton

We will use a version of two-counter automaton which only has two types of transitions. First it can increment a register and second it can try to decrement a register and jump of the register is already zero. Formally:

Definition 14. A <u>deterministic two-counter automaton</u> is a 4-tuple $M = (Q, q_0, q_f, R)$,
where $\quad Q \quad$ is a finite set (of states),
$\quad\quad\quad q_0 \quad$ is in Q (the initial state),
$\quad\quad\quad q_f \quad$ is in Q (the final state), and
$\quad\quad\quad R \quad$ is a function from $Q \setminus \{q_f\}$ to $\mathcal{R}_Q$,
$\quad\quad\quad\quad$ where $\mathcal{R}_Q = \{+(i, q') \mid i \in \{0, 1\}, q' \in Q\}$
$\quad\quad\quad\quad\quad\quad\quad \cup \{-(i, q_1, q_2) \mid i \in \{0, 1\}, q_1, q_2 \in Q\}$

An $\underline{\text{ID}}$ of our automaton is a triple $\langle q, m, n \rangle$, where $q \in Q$ and $m, n \in \mathbb{N}$. Let r be in $R(Q \setminus \{q_f\})$, then $\Rightarrow_M^r$ is a binary relation on the ID's of M such that two ID's $\langle q, m, n \rangle$, $\langle q', m', n' \rangle$ of M are in the in the relation if the following conditions hold:

- $q \neq q_f$, $r = R(q)$,

- if $r = +(0, p)$ for some $p \in Q$ then $q' = p$, $m' = m + 1$, and $n' = n$,

- if $r = +(1, p)$ for some $p \in Q$ then $q' = p$, $m' = m$, and $n' = n + 1$,

- if $r = -(0, p_1, p_2)$ for some $p_1, p_2 \in Q$ then

    if $m = 0$ then $q' = p_2$, $m' = 0$, and $n' = n$,

    if $m \geq 1$ then $q' = p_1$, $m' = m - 1$, and $n' = n$,

- if $r = -(1, p_1, p_2)$ for some $p_1, p_2 \in Q$ then

    if $n = 0$ then $q' = p_2$, $m' = m$, and $n' = 0$,

    if $n \geq 1$ then $q' = p_1$, $m' = m$, and $n' = n - 1$.

The $\underline{\text{transition relation of } M}$, denoted by $\Rightarrow_M$, is defined as $\bigcup_{r \in R(Q \setminus \{q_f\})} \Rightarrow_M^r$. We denote the transitive reflexive closure of $\Rightarrow_M$ by $\Rightarrow_M^*$

Let m, n be in $\mathbb{N}$, we say that M terminates on input (m, n) if there exist $m', n' \in \mathbb{N}$ such that $\langle q_0, m, n \rangle \Rightarrow_M^* \langle q_f, m', n' \rangle$.

Definition 15. The halting problem for two-counter automaton, denoted by **HALT**, is defined as follows. Given a two-counter automaton M.

$$\text{Does } M \text{ terminate on input } (0, 0).$$

It is well known that **HALT** is undecidable.

3 System P

3.1 Definitions

In the following let $\mathcal{V}_P = \{\alpha, a, \beta, b, \dots\}$ be a countably infinite set (of variables) and $\mathcal{P}_P = \{\textit{false}^{(0)}, P^{(2)}, Q^{(2)}, \dots\}$ a ranked set (of predicate symbols) such that $\mathcal{P}_P^{(0)} = \{\textit{false}\}$, $\mathcal{P}_P^{(2)} = \{P, Q, \dots\}$ is a countably infinite set, and $\mathcal{P}_P^{(k)} = \emptyset$ for all $k \in \mathbb{N} \setminus \{0, 2\}$. A first-order logic formula φ over $(\mathcal{V}_P, \emptyset, \mathcal{P}_P)$ is an

atomic formula if $\varphi = \textit{false}$ or $\varphi = P(a, b)$ for some $P \in \mathcal{P}_P$ and $a, b \in \mathcal{V}_P$.

universal formula if $\varphi = \forall \overrightarrow{\alpha}(A_1 \rightarrow A_2 \rightarrow \cdots \rightarrow A_n)$ where A_i is an atomic formula for $i \in [n]$, $A_i \neq \textit{false}$ for $i \in [n - 1]$ and for each $\alpha \in \text{FV}(\varphi) \cap \text{FV}(A_n)$ there exists an $i \in [n - 1]$ such that $\alpha \in \text{FV}(A_i)$.

existential formula if there exits $n \geq 0$, atomic formulas $A_i \neq \textit{false}$ for $i \in [n]$ such that $\varphi = \forall \overrightarrow{\alpha}(A_1 \rightarrow A_2 \rightarrow \cdots \rightarrow A_{n-1} \rightarrow \forall \beta(A_n \rightarrow \textit{false}) \rightarrow \textit{false})$.

The set of formulas of System **P** (= set of **P**-formulas) over $(\mathcal{V}_P, \mathcal{P}_P)$ is the set of all first-order formulas over $(\mathcal{V}_P, \emptyset, \mathcal{P}_P)$ that are either an atomic, universal or existential formula.

Deduction Rules

$$
\begin{array}{lll}
\text{(Axiom)} & \Gamma, A \vdash A & \\[2em]
(\rightarrow\text{-Introduction}) & \dfrac{\Gamma, A \vdash B}{\Gamma \vdash A \rightarrow B} & \\[2em]
(\rightarrow\text{-Elimination}) & \dfrac{\Gamma \vdash A \rightarrow B \quad \Gamma \vdash A}{\Gamma \vdash B} & \\[2em]
(\forall\text{-Introduction}) & \dfrac{\Gamma \vdash B}{\Gamma \vdash \forall \alpha B} & \alpha \notin \mathrm{FV}(\Gamma) \\[2em]
(\forall\text{-Elimination}) & \dfrac{\Gamma \vdash \forall \alpha B}{\Gamma \vdash B\,[\alpha := b]} & b \in \mathcal{V}_P
\end{array}
$$

An Interpretation I of a P formula is a tuple $I = (\Delta, \cdot^I)$ where Δ is a set (called domain), $P^I \subseteq \Delta^k$ and $\alpha^I \in \Delta$....

If we interpret *false* with the logical constant false ($\perp$) (denoted by $\vdash_f$) we can add a new deduction rule.

$$
\begin{array}{lll}
(\exists\text{-Introduction}) & \dfrac{\Gamma, A\,[\alpha := a] \vdash_f B}{\Gamma, \forall \alpha(A \rightarrow false) \rightarrow false \vdash_f B} & a \notin FV(\Gamma, A, B)
\end{array}
$$

Proof. Let $I = (\Delta, \cdot^I)$ be a model of $\Gamma, \forall \alpha(A \rightarrow false) \rightarrow false$ with $false^I = \perp$.

$$
\begin{aligned}
I \models \Gamma, \forall \alpha(A \rightarrow false) \rightarrow false \Rightarrow{} & I \models \forall \alpha(A \rightarrow false) \rightarrow false \\
\Rightarrow{} & (\forall \alpha(A \rightarrow false))^I \rightarrow false^I \\
\Rightarrow{} & (\forall \alpha(A \rightarrow false))^I \rightarrow \perp \\
\Rightarrow{} & \neg(\forall \alpha(A \rightarrow false))^I \\
\Rightarrow{} & \neg(\forall d \in \Delta : (A \rightarrow false)^{I[\alpha \mapsto d]}) \\
\Rightarrow{} & \exists d \in \Delta : \neg(A^{I[\alpha \mapsto d]} \rightarrow false^{I[\alpha \mapsto d]}) \\
\Rightarrow{} & \exists d \in \Delta : \neg(A^{I[\alpha \mapsto d]} \rightarrow \perp) \\
\Rightarrow{} & \exists d \in \Delta : \neg(\neg A^{I[\alpha \mapsto d]}) \\
\Rightarrow{} & \exists d \in \Delta : A^{I[\alpha \mapsto d]}
\end{aligned}
$$

Together with $a \notin FV(\Gamma, A)$, it follows that $I[a \mapsto d]$ is a model of $\Gamma, A[\alpha := a]$. Which implies $I[a \mapsto d] \models B$. Since a is not free in B we conclude that I is also a model of B. $\hspace{1cm} \square$

Definition 16. The problem to decide whether a given set of **P**-formulas is consistency, denoted by **CONS**, is defined as follows. Given a set of **P**-formulas Γ.

$$\text{Does } \Gamma \vdash \textit{false} \text{ not hold.}$$

3.2 CONS is undecidable

We will show that **HALT** $\leq$ **CONS** then the undecidability of **CONS** directly follows from the undecidability of **HALT**. Γ_C :

- $Q(a)$

- $R_1(a, a_0), P(a_{i-1}, a_i)$ for $i \in \{1, \ldots, m\}$

- $R_2(a, b_0), P(b_{i-1}, b_i)$ for $i \in \{1, \ldots, n\}$

- $D(a), D(a_i), D(b_j)$ for $i \in \{1, \ldots, m\}$ and $j \in \{1, \ldots, n\}$

- $E(a_m), E(b_n)$

$+(Q, 1, Q') :$

- $\forall \alpha \beta (Q(\alpha) \rightarrow S(\alpha, \beta) \rightarrow Q'(\beta))$
  change of state

- $\forall \alpha \beta \gamma \delta (Q(\alpha) \rightarrow S(\alpha, \beta) \rightarrow R_1(\alpha, \gamma) \rightarrow R_1(\beta, \delta) \rightarrow P(\delta, \gamma))$
  increment register 1

- $\forall \alpha \beta \delta (Q(\alpha) \rightarrow S(\alpha, \beta) \rightarrow R_1(\beta, \delta) \rightarrow D(\delta))$
  prevent zero

- $\forall \alpha \beta \gamma (Q(\alpha) \rightarrow S(\alpha, \beta) \rightarrow R_2(\alpha, \gamma) \rightarrow R_2(\beta, \gamma))$
  do not change register 2

$-(Q, 1, Q_1, Q_2) :$

- $\forall \alpha \beta \gamma (Q(\alpha) \rightarrow S(\alpha, \beta) \rightarrow R_1(\alpha, \gamma) \rightarrow E(\gamma) \rightarrow Q_2(\beta))$
  jump on zero

- $\forall \alpha \beta \gamma (Q(\alpha) \rightarrow S(\alpha, \beta) \rightarrow R_1(\alpha, \gamma) \rightarrow E(\gamma) \rightarrow R_1(\beta, \gamma))$
  register 1 stays zero

- $\forall \alpha \beta \gamma (Q(\alpha) \rightarrow S(\alpha, \beta) \rightarrow R_1(\alpha, \gamma) \rightarrow D(\gamma) \rightarrow Q_1(\beta))$
  change state if register 1 is greater zero

- $\forall \alpha \beta \gamma \delta (Q(\alpha) \rightarrow S(\alpha, \beta) \rightarrow R_1(\alpha, \gamma) \rightarrow D(\gamma) \rightarrow P(\gamma, \delta) \rightarrow R_1(\beta, \delta))$
  decrement register 1

- $\forall \alpha\beta\gamma(Q(\alpha) \to S(\alpha,\beta) \to R_2(\alpha,\gamma) \to R_2(\beta,\gamma))$
  do not change register 2

Lemma 17.

$$M \text{ terminates on input } (0,0) \qquad \text{iff} \qquad \Gamma_M \vdash \text{false } \text{holds in system } P.$$

Claim 18.

$$\Gamma_M \vdash \text{false } \text{holds in system } P \qquad \implies \qquad M \text{ terminates on input } (0,0)$$

Proof. Assume M does not terminate then there is an infinite chain $C_1 \Rightarrow_M C_2 \Rightarrow_M C_3 \Rightarrow_M \ldots$. $(C_i = \langle Q_i, m_i, n_i \rangle)$ Now we construct a model of Γ_M which interprets *false* with $\bot$ this contradicts $\Gamma_M \vdash \text{false}$.

To illustrate the idea we will use a graphical notation for an interpretation I. By $d_1 \overset{\text{R}}{\to} d_2$ we say that $(d_1, d_2) \in R^I$. And we use $\overset{\text{P}}{\underset{d}{}}$ to say that $d \in P^I$ for unary predicate symbols. Now the idea for our model of Γ_M looks like this:
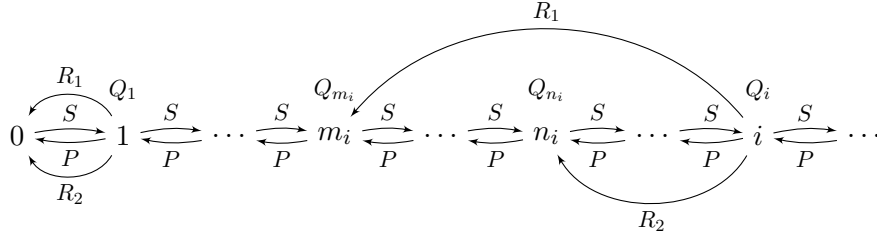
We have $0 \in E^I$ and all other numbers are in D^I.
Here is the more formal definition of our model $I = (\mathbb{N}, \cdot^I)$.

$$P^I = \{(i+1, i) \mid i \in \mathbb{N}\} \qquad R_1^I = \{(i, m_i) \mid i \in \mathbb{N}\} \qquad R_2^I = \{(i, n_i) \mid i \in \mathbb{N}\}$$
$$Q^I = \{i \in \mathbb{N} \setminus \{0\} \mid Q = Q_i\} \qquad D^I = \mathbb{N} \setminus \{0\} \qquad E^I = \{0\}$$
$$S^I = \{(i, i+1) \mid i \in \mathbb{N}\}$$

$$a^I = 1 \qquad\qquad a_0^I = 0 \qquad\qquad b_0^I = 0$$

$\square$

Claim 19. *If a final state is reachable from C then $\Gamma_C \cup \Gamma \vdash \text{false}$.*

Proof. By induction on the length of the computation. For the tableau proofs we will abbreviate *false* by f.
Induction Base trivial ...
Induction Step
$C \Rightarrow_M^r D$

We need to make a case distinction on the rule r.

Case $r = +(Q, 1, Q')$

Basic idea:

$$\frac{\dfrac{IH}{\Gamma_C \cup \Gamma \cup \Gamma_D \vdash f} \qquad \overline{\Gamma_C \cup \Gamma \vdash \Gamma_D}}{\Gamma_C \cup \Gamma \vdash f}$$

Since $I \models \textit{false}$ holds trivially if I interprets $\textit{false}$ with $\top$ we only need to consider models (note that there are none if M terminates which is exactly what we want to proof) of $\Gamma_C \cup \Gamma$ that interpret $\textit{false}$ with $\bot$ (so we can use our new deduction rule).

We will just drop $\Gamma_C \cup \Gamma$ and only write new formulas on the left side.

We first introduce the new variables needed for Γ_D (let $b, d \in \mathcal{V}_P \backslash \mathrm{FV}(\Gamma_C \cup \Gamma)$). Intuitively b will represent the successor state and d will be the anchor for register one.

$$\frac{\dfrac{\dfrac{\vdots}{\dfrac{S(a,b), D(b) \vdash_f f}{S(a,b) \vdash_f D(b) \to f}} \qquad \dfrac{S(a,b) \vdash_f \forall \alpha \beta (S(\alpha, \beta) \to D(\beta))}{\dfrac{S(a,b) \vdash_f S(a,b) \to D(b) \quad S(a,b) \vdash_f S(a,b)}{S(a,b) \vdash_f D(b)}}}{\dfrac{\dfrac{S(a,b) \vdash_f f}{\forall \beta (S(a,\beta) \to f) \to f \vdash_f f}}{\vdash_f (\forall \beta (S(a,\beta) \to f) \to f) \to f} \qquad \dfrac{\vdash_f \forall \alpha (\forall \beta (S(\alpha, \beta) \to f) \to f)}{\vdash_f \forall \beta (S(a, \beta) \to f) \to f}}{\vdash_f f}$$

The formula $R_1(b, d)$ can be acquired in a similar way. Again we will just drop $S(a, b)$ and $D(b)$ on the left side for comprehensibility.

$$\frac{\dfrac{\dfrac{\vdots}{\dfrac{R_1(b,d) \vdash_f f}{\forall \beta (R_1(b, \beta) \to f) \to f \vdash_f f}}}{\vdash_f (\forall \beta (R_1(b, \beta) \to f) \to f) \to f} \qquad \dfrac{\dfrac{\vdash_f \forall \alpha (D(\alpha) \to \forall \beta (R_1(\alpha, \beta) \to f) \to f)}{\vdash_f D(b) \to \forall \beta (R_1(b, \beta) \to f) \to f} \quad \vdash_f D(b)}{\vdash_f \forall \beta (R_1(b, \beta) \to f) \to f}}{\vdash_f f}$$

Now we have all the new free variables we need and we continue by ensuring that these variables fulfill all the formulas in Γ_D.

$$\frac{\dfrac{\vdots}{\dfrac{Q'(b) \vdash_f f}{\vdash_f Q'(b) \to f}} \qquad \dfrac{\dfrac{\vdash_f \forall \alpha \beta (Q(\alpha) \to S(\alpha, \beta) \to Q'(\beta))}{\vdash_f Q(a) \to S(a, b) \to Q'(b)} \quad \vdash_f Q(a)}{\dfrac{\vdash_f S(a, b) \to Q'(b)}{\dfrac{\vdash_f Q'(b)}{}} \quad \vdash_f S(a, b)}}{\vdash_f f}$$

Starting from $Q'(b) \vdash_f \textit{false}$ we can connect d and a_0.

10

$$\dfrac{\begin{array}{c}\vdots\\\hline\dfrac{P(d,a_0)\vdash_f f}{\vdash_f P(d,a_0)\to f}\end{array}\qquad\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\vdash_f \forall\alpha\beta\gamma\delta(Q(\alpha)\to S(\alpha,\beta)\to R_1(\alpha,\gamma)\to R_1(\beta,\delta)\to P(\delta,\gamma))}{\vdash_f Q(a)\to S(a,b)\to R_1(a,a_0)\to R_1(b,d)\to Q'(b)\qquad\vdash_f Q(a)}}{\vdash_f S(a,b)\to R_1(a,a_0)\to R_1(b,d)\to Q'(b)\qquad\vdash_f S(a,b)}}{\vdash_f R_1(a,a_0)\to R_1(b,d)\to Q'(b)\qquad\vdash_f R_1(a,a_0)}}{\vdash_f R_1(b,d)\to Q'(b)\qquad\vdash_f R_1(b,d)}}{\vdash_f P(d,a_0)}}{\vdash_f f}$$

For register one we still need $D(d)$.

$$\dfrac{\begin{array}{c}\vdots\\\hline\dfrac{D(d)\vdash_f f}{\vdash_f D(d)\to f}\end{array}\qquad\dfrac{\dfrac{\dfrac{\dfrac{\vdash_f \forall\alpha\beta\delta(Q(\alpha)\to S(\alpha,\beta)\to R_1(\beta,\delta)\to D(\delta))}{\vdash_f Q(a)\to S(a,b)\to R_1(b,d)\to D(d)\qquad\vdash_f Q(a)}}{\vdash_f S(a,b)\to R_1(b,d)\to D(d)\qquad\vdash_f S(a,b)}}{\vdash_f R_1(b,d)\to D(d)\qquad\vdash_f R_1(b,d)}}{\vdash_f D(d)}}{\vdash_f f}$$

Since register two should not change we only need $R_2(b,b_0)$.

$$\dfrac{\begin{array}{c}\vdots\\\hline\dfrac{R_2(b,b_0)\vdash_f f}{\vdash_f R_2(b,b_0)\to f}\end{array}\qquad\dfrac{\dfrac{\dfrac{\dfrac{\vdash_f \forall\alpha\beta\gamma(Q(\alpha)\to S(\alpha,\beta)\to R_2(\alpha,\gamma)\to R_2(\beta,\gamma))}{\vdash_f Q(a)\to S(a,b)\to R_2(a,b_0)\to R_2(b,b_0)\qquad\vdash_f Q(a)}}{\vdash_f S(a,b)\to R_2(a,b_0)\to R_2(b,b_0)\qquad\vdash_f S(a,b)}}{\vdash_f R_2(a,b_0)\to R_2(b,b_0)\qquad\vdash_f R_2(a,b_0)}}{\vdash_f R_2(b,b_0)}}{\vdash_f f}$$

Now we have Γ_C (Since $P(a_{i-1},a_i)$ is already in Γ_D) and can deduce *false* by induction hypothesis.

Case $r = -(Q,1,Q_1,Q_2)$

$r_1 = 0$

$$\dfrac{\begin{array}{c}\vdots\\\hline\dfrac{Q_2(b)\vdash_f f}{\vdash_f Q_2(b)\to f}\end{array}\qquad\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\vdash_f \forall\alpha\beta\gamma(Q(\alpha)\to S(\alpha,\beta)\to R_1(\alpha,\gamma)\to E(\gamma)\to Q_2(\beta))}{\vdash_f Q(a)\to S(a,b)\to R_1(a,a_0)\to E(a_0)\to Q_2(b)\qquad\vdash_f Q(a)}}{\vdash_f S(a,b)\to R_1(a,a_0)\to E(a_0)\to Q_2(b)\qquad\vdash_f S(a,b)}}{\vdash_f R_1(a,a_0)\to E(a_0)\to Q_2(b)\qquad\vdash_f R_1(a,a_0)}}{\vdash_f E(a_0)\to Q_2(b)\qquad\vdash_f E(a_0)}}{\vdash_f Q_2(b)}}{\vdash_f f}$$

r_1 stays zero

$$\cfrac{\cfrac{\vdash_f \forall\alpha\beta\gamma(Q(\alpha) \to S(\alpha,\beta) \to R_1(\alpha,\gamma) \to E(\gamma) \to R_1(\beta,\gamma))}{\cfrac{\vdash_f Q(a) \to S(a,b) \to R_1(a,a_0) \to E(a_0) \to R_1(b,a_0) \qquad \vdash_f Q(a)}{\cfrac{\vdash_f S(a,b) \to R_1(a,a_0) \to E(a_0) \to R_1(b,a_0) \qquad \vdash_f S(a,b)}{\cfrac{\vdash_f R_1(a,a_0) \to E(a_0) \to R_1(b,a_0) \qquad \vdash_f R_1(a,a_0)}{\cfrac{\vdash_f E(a_0) \to R_1(b,a_0) \qquad \vdash_f E(a_0)}{\vdash_f R_1(b,a_0)}}}}} \qquad \cfrac{\cfrac{\vdots}{\cfrac{R_1(b,a_0) \vdash_f f}{\vdash_f R_1(b,a_0) \to f}}}{} }{\vdash_f f}$$

$\underline{r_1 \geq 1}$
new state Q_1

$$\cfrac{\cfrac{\vdash_f \forall\alpha\beta\gamma(Q(\alpha) \to S(\alpha,\beta) \to R_1(\alpha,\gamma) \to D(\gamma) \to Q_1(\beta))}{\cfrac{\vdash_f Q(a) \to S(a,b) \to R_1(a,a_0) \to D(a_0) \to Q_1(b) \qquad \vdash_f Q(a)}{\cfrac{\vdash_f S(a,b) \to R_1(a,a_0) \to D(a_0) \to Q_1(b) \qquad \vdash_f S(a,b)}{\cfrac{\vdash_f R_1(a,a_0) \to D(a_0) \to Q_1(b) \qquad \vdash_f R_1(a,a_0)}{\cfrac{\vdash_f D(a_0) \to Q_1(b) \qquad \vdash_f D(a_0)}{\vdash_f Q_1(b)}}}} \qquad \cfrac{\cfrac{\vdots}{\cfrac{Q_1(b) \vdash_f f}{\vdash_f Q_1(b) \to f}}}{}}{\vdash_f f}$$

decrement r_1

$$\cfrac{\cfrac{\vdash_f \forall\alpha\beta\gamma\delta(Q(\alpha) \to S(\alpha,\beta) \to R_1(\alpha,\gamma) \to D(\gamma) \to P(\gamma,\delta) \to R_1(\beta,\delta))}{\cfrac{\vdash_f Q(a) \to S(a,b) \to R_1(a,a_0) \to D(a_0) \to P(a_0,a_1) \to R_1(b,a_1) \quad \vdash_f Q(a)}{\cfrac{\vdash_f S(a,b) \to R_1(a,a_0) \to D(a_0) \to P(a_0,a_1) \to R_1(b,a_1) \quad \vdash_f S(a,b)}{\cfrac{\vdash_f R_1(a,a_0) \to D(a_0) \to P(a_0,a_1) \to R_1(b,a_1) \quad \vdash_f R_1(a,a_0)}{\cfrac{\vdash_f D(a_0) \to P(a_0,a_1) \to R_1(b,a_1) \quad \vdash_f D(a_0)}{\cfrac{\vdash_f P(a_0,a_1) \to R_1(b,a_1) \quad \vdash_f P(a_0,a_1)}{\vdash_f R_1(b,a_1)}}}}} \qquad \cfrac{\cfrac{\vdots}{\cfrac{R_1(b,a_1) \vdash_f f}{\vdash_f R_1(b,a_1) \to f}}}{}}{\vdash_f f}$$

$\square$