Lemma:

Die Transitionsrelation ist deterministisch,
d.h. für alle Konfigurationen $(c, \sigma) \in Prog \times State$
und alle $k_1, k_2 \in (Prog \times State) \cup State$ gilt:

$$(c, \sigma) \to k_1 \text{ und } (c, \sigma) \to k_2 \text{ impliziert } k_1 = k_2.$$

- Der Zustandsraum eines Programms $State = \mathbb{Z}^{Vars}$ ist kein vollständiger Verband.
- Um Galois-Verbindungen zur Abstraktion von Zustandsmengen nutzen zu können, definieren für alle $c, c' \in Prog$:

$$post_{c,c'}, \; post_c : \mathbb{P}(State) \to \mathbb{P}(State)$$

mit

$$post_{c,c'}(State') := \{ \sigma' \in State \mid \exists \sigma \in State' : (c, \sigma) \to (c', \sigma') \}$$
$$post_c(State') := \{ \sigma' \in State \mid \exists \sigma \in State' : (c, \sigma) \to \sigma' \}.$$

- Jeder Befehl wird also als Transformer von Zustandsmengen aufgefasst (à la Dijkstra).

4.4 Abstrakte Semantik:

Ziel: Imitiere die konkrete Semantik, genauer $post_{c,c'}$ und $post_c$, auf einer abstrakten Datendomäne.

Definition (Sichere Approximation von Funktionen):
Sei (α, γ) eine Galois-Verbindung mit $L \underset{\gamma}{\overset{\alpha}{\rightleftarrows}} M$.
Sei ferner $f : L \to L$ eine Funktion.

- Dann heißt $f^\# : M \to M$ sichere Approximation von f, falls $\alpha \circ f \circ \gamma \leq_M f^\#$, d.h. $\alpha(f(\gamma(m))) \leq_M f^\#(m)$ für alle $m \in M$.
- Funktion $f^\#$ heißt genaueste sichere Approximation von f, falls $\alpha \circ f \circ \gamma = f^\#$.

Illustration:

$$\begin{tikzcd}
\gamma(m) \arrow[r, "f"] & f(\gamma(m)) \arrow[d, "\alpha"] \\
 & \alpha(f(\gamma(m))) \arrow[d, phantom, "\leq_M"] \\
m \arrow[uu, "\gamma"] \arrow[r, dashed, "f^\#"] & f^\#(m)
\end{tikzcd}$$

Kommentar:
Oft sind f und $f^\#$ monoton.

Lemma:

Falls f und $f^\#$ monoton sind, gilt

$$\alpha \circ f \circ \gamma \leq_M f^\# \quad \text{gdw.} \quad \alpha \circ f \leq_M f^\# \circ \alpha.$$

Beispiel (Sichere Approximation):

- Betrachte $\mathbb{P}(\mathbb{Z}) \underset{\gamma_{sign}}{\overset{\alpha_{sign}}{\rightleftarrows}} \mathbb{P}(\{-,0,+\})$, die Vorzeichenabstraktion.
- Sei f_{-2} die Subtraktion von 2 auf dem Potenzmengenverband $\mathbb{P}(\mathbb{Z})$.

$$f_{-2} : \mathbb{P}(\mathbb{Z}) \to \mathbb{P}(\mathbb{Z})$$
$$f_{-2}(Z) := \{z-2 \mid z \in Z\}.$$

- Definiere eine sichere Approximation von f_{-2} mittels

$$f_{-2}^\# : \mathbb{P}(\{-,0,+\}) \to \mathbb{P}(\{-,0,+\})$$
$$f_{-2}^\#(A) := \{-\}, \text{ falls } A \neq \emptyset$$
$$\cup \{0,+\}, \text{ falls } + \in A.$$

- Es ist zu zeigen, dass für alle $A \subseteq \{-,0,+\}$ gilt

$$\alpha(f_{-2}(\gamma(A))) \subseteq f_{-2}^\#(A).$$

Zum Beispiel:

$$\alpha(f_{-2}(\gamma(\{0,+\}))) = \alpha(f_{-2}(\{0,1,2,3,\dots\}))$$
$$= \alpha(\{-2,-1,0,1,\dots\}) = \{-,0,+\} = f_{-2}^\#(\{0,+\}).$$

- Definiere nun die operationelle Semantik von while-Programmen auf einer abstrakten Datendomäne.
- Beachte, dass die Transitionsrelation nicht-deterministisch wird:

$$(\underline{\text{if}}\ x=0\ \underline{\text{then}}\ c_1\ \underline{\text{else}}\ c_2\ \underline{\text{fi}},\ \{(x=\text{even})\}).$$

Die Bedingung kann wahr oder falsch sein.

- Betrachte eine Galois-Verbindung $\mathbb{P}(State) \underset{\gamma}{\overset{\alpha}{\rightleftarrows}} M$.

Dabei ist $\mathbb{P}(State)$ der vollständige Verband der Mengen von Variablenbelegungen.
Ferner ist M ein vollständiger Verband abstrakter Werte.

- Die abstrakte Semantik soll eine sichere Approximation der konkreten sein.

Definition (Abstrakte Semantik):

Betrachte die Galois-Verbindung $\mathbb{P}(State) \underset{\gamma}{\overset{\alpha}{\rightleftarrows}} M$.

- Eine <u>abstrakte Semantik</u> ist gegeben durch eine Familie von Funktionen
$$post^{\#}_{c,c'}, \; post^{\#}_{c} : M \to M$$
mit
$$\alpha \circ post_{c(c')} \circ \gamma \leq_M post^{\#}_{c(c')}.$$
- Sind alle $post^{\#}_{c(c')}$ genaueste sichere Approximationen der $post_{c(c')}$, spricht man von der <u>genauesten</u> abstrakten Semantik. (Unpräzise Semantiken auch zulässig).
- Die abstrakte Semantik induziert die <u>abstrakte Transitionsrelation</u>
$$\Rightarrow \subseteq (Prog \times M) \times ((Prog \times M) \cup M)$$
zwischen <u>abstrakten Konfigurationen</u> $(c, m) \in Prog \times M$ mittels
$$(c, m) \Rightarrow (c', post_{c,c'}(m))$$
$$(c, m) \Rightarrow post_c(m).$$

Beispiel (Genaueste abstrakte Semantik):

Betrachte $\mathbb{P}(\mathbb{Z}^{\{n\}}) \underset{\gamma_{parity}}{\overset{\alpha_{parity}}{\rightleftarrows}} \mathbb{P}(\{odd, even\}^{\{n\}})$, es gibt also nur eine Variable:

$$([n := 3n+1]^l, \{(n = odd)\}) \Rightarrow \{(n = even)\}$$
$$([n := 3n+1]^l, \{(n = even)\}) \Rightarrow \{(n = odd)\}$$
$$([n := 3n+1]^l, \{(n = even), (n = odd)\}) \Rightarrow \{(n = even), (n = odd)\}$$
$$(\underline{while}\ [n \neq 1]^l\ \underline{do}\ c\ \underline{od}, \{(n = odd)\}) \Rightarrow \{(n = odd)\}$$
$$(\underline{while}\ [n \neq 1]^l\ \underline{do}\ c\ \underline{od}, \{(n = odd)\}) \Rightarrow (c; \underline{while}\ [n \neq 1]^l\ \underline{do}\ c\ \underline{od}, \{(n = odd)\})$$
$$(\underline{while}\ [n \neq 1]^l\ \underline{do}\ c\ \underline{od}, \{(n = even)\}) \not\Rightarrow \{(n = even)\}$$
$$(\underline{while}\ [n \neq 1]^l\ \underline{do}\ c\ \underline{od}, \{(n = even)\}) \Rightarrow (c; \underline{while}\ [n \neq 1]^l\ \underline{do}\ c\ \underline{od}, \{(n = even)\}).$$

Lemma:

Die genaueste abstrakte Semantik ist im Allgemeinen <u>nicht</u> berechenbar.

Warum?

- Betrachte die Galois-Verbindung $\mathbb{P}(State) \xrightleftharpoons[\gamma_{sign}]{\alpha_{sign}} \mathbb{P}(\{-,0,+\})$.
- Sei die abstrakte Konfiguration

$$(\text{if } [n>2 \wedge x^n+y^n=z^n]^{l} \text{ then } [n:=1]^{l_1} \text{ else } [n:=-1]^{l_2} \text{ fi},$$

$$\{(n=+,\ x=+,\ y=+,\ z=+)\}) \text{ gegeben.}$$

- Um zu entscheiden, ob n auf $+$ oder $-$ gesetzt wird, muss man entscheiden, ob es Belegungen von x, y, z, n in $\mathbb{N}\setminus\{0\}$ gibt, die die Bedingung erfüllen.
- Letzter Satz von Fermat bewiesen: nein.

Allgemein:

Es ist unentscheidbar, ob eine Diophantische Gleichung

$$p(x_1, \ldots, x_n) = 0$$

mit p einem Polynom mit Koeffizienten in $\mathbb{Z}$ eine Lösung in $\mathbb{Z}$ hat.

(Hilberts 10. Problem von 1900,
Unentscheidbarkeit nach Matiyasevich 1970)

Ungenauere abstrakte Semantiken lassen sich immer herleiten.

5.5 Herleitung einer abstrakten Semantik:

Ziel: Berechne die abstrakte Semantik für Galois-Verbindungen $(\alpha_\beta, \gamma_\beta)$, die durch Liften einer Extraktionsfunktion $\beta: \mathbb{Z} \to D$ auf State entstanden sind.

Also: $$\mathbb{P}(State) = \mathbb{P}(\mathbb{Z}^{Var}) \xrightleftharpoons[\gamma_\beta]{\alpha_\beta} \mathbb{P}(D^{Var}).$$

Problem:
- Werte Boolesche Ausdrücke auf der abstrakten Domäne $\mathbb{P}(D)$ aus. (schon auf D problematisch)
- Benötige sichere Approximation von Prädikaten.

Lösung:
- Werte die Approximation in 3-wertiger Logik aus:

$(\mathbb{P}(\mathbb{B})\setminus\{\emptyset\}, \wedge_3, \vee_3, \neg_3)$ mit

$\wedge_3$	0	1	1/2
0	0	0	0
1	0	1	1/2
1/2	0	1/2	1/2

$\vee_3$	0	1	1/2
0	0	1	1/2
1	1	1	1
1/2	1/2	1	1/2

$\neg_3$	
0	1
1	0
1/2	1/2

Definition (Sichere Approximation von Prädikaten):

Sei $p: \mathbb{Z} \to \mathbb{B}$ ein Prädikat

das auch auf Mengen verstanden werden kann:

$$p: \mathbb{P}(\mathbb{Z}) \longrightarrow \mathbb{P}(\mathbb{B}).$$

Dann heißt $p^{\#}: \mathbb{P}(D) \longrightarrow \mathbb{P}(\mathbb{B})$

sichere Approximation von p, falls

$$p \circ \gamma_\beta \leq p^{\#}$$

Definition (Abstrakte Sig-Struktur):

Sei $\mathcal{S} = (\mathbb{Z}, I)$ und $(\alpha_\beta, \gamma_\beta)$ die Galois-Verbindung $\mathbb{P}(\mathbb{Z}) \underset{\gamma_\beta}{\overset{\alpha_\beta}{\rightleftarrows}} \mathbb{P}(\mathbb{D}$

Dann heißt $\mathcal{S}_{Abs} = (\mathbb{P}(D), I^{\#})$ **abstrakte Sig-Struktur**,

falls

$f^{\#}_I : \mathbb{P}(D)^n \to \mathbb{P}(D)$ ist sichere Approximation von $f_I : \mathbb{Z}^n \to \mathbb{Z}$

$p^{\#}_I : \mathbb{P}(D)^n \to \mathbb{P}(\mathbb{B})$ ist sichere Approximation von $p_I : \mathbb{Z}^n \to \mathbb{B}$

Die Semantik Boolescher Ausdrücke

in 3-wertiger Logik ist dabei $(\sigma: Vars \to \mathbb{P}(D))$:

$$\mathcal{S}_{Abs}[\![p(a_1, \dots, a_n)]\!](\sigma) := p^{\#}_I(\mathcal{S}_{Abs}[\![a_1]\!](\sigma), \dots, \mathcal{S}_{Abs}[\![a_n]\!](\sigma))$$

$$\mathcal{S}_{Abs}[\![b_1 \underset{\substack{\lor \\ \neg}}{\land} b_2]\!](\sigma) := \mathcal{S}_{Abs}[\![b_1]\!](\sigma) \underset{\substack{\lor_3 \\ \neg_3}}{\land_3} \mathcal{S}_{Abs}[\![b_2]\!](\sigma)$$

Lemma:

Es gilt $\beta(\mathcal{S}[\![a]\!](\sigma)) \in \mathcal{S}_{Abs}[\![a]\!](\sigma')$

$\mathcal{S}[\![b]\!](\sigma) \in \mathcal{S}_{Abs}[\![b]\!](\sigma')$ mit $\sigma'(x) := \{\beta(\sigma(x))\}$

Mögliche Definitionen für $f^{\#}_I$ und $p^{\#}_I$:

$$f^{\#}_I(D_1, \dots, D_n) := \beta(f(\beta^{-1}(D_1), \dots, \beta^{-1}(D_n)))$$

$$p^{\#}_I(D_1, \dots, D_n) := \underbrace{p(\beta^{-1}(D_1), \dots, \beta^{-1}(D_n))}_{\substack{0, \text{ falls } p(z_1, \dots, z_n) = 0 \text{ für alle } z_1 \in \beta^{-1}(D_1), \dots, z_n \in \beta^{-1}(D_n) \\ 1, \text{ falls } p(z_1, \dots, z_n) = 1 \text{ für alle } —"— \\ 1/2, \text{ sonst.}}}$$

Ist eine abstrakte Sig-Struktur gegeben,
erhält man die <u>abstrakte Transitionsrelation</u>

$$\Rightarrow \;\subseteq\; (Prog \times \mathbb{P}(D^{Vars})) \times ((Prog \times \mathbb{P}(D^{Vars})) \cup \mathbb{P}(D^{Vars}))$$

mit folgender Definition:

$$\frac{}{([x:=a]^l, Abs) \Rightarrow \{\sigma[x:=d] \mid \sigma \in Abs,\ d \in S_{Abs}[\![a]\!](\sigma') \text{ mit } \sigma'(x) := \{\sigma(x)\}\}}$$

$$\frac{}{(\underline{skip}, Abs) \Rightarrow Abs}$$

$$\frac{(c_1, Abs) \Rightarrow (c_1', Abs')}{(c_1; c_2, Abs) \Rightarrow (c_1'; c_2, Abs')} \qquad \frac{(c_1, Abs) \Rightarrow Abs'}{(c_1; c_2, Abs) \Rightarrow (c_2, Abs')}$$

$$\frac{}{(\text{if } b \ \underline{then}\ c_1 \ \underline{else}\ c_2 \ \underline{fi}, Abs) \Rightarrow (c_1, Abs \setminus \{\sigma \mid S_{Abs}[\![b]\!](\sigma') = \{0\}\})}$$

$$\frac{}{(\text{if } b \ \underline{then}\ c_1 \ \underline{else}\ c_2 \ \underline{fi}, Abs) \Rightarrow (c_2, Abs \setminus \{\sigma \mid S_{Abs}[\![b]\!](\sigma') = \{1\}\})}$$

$$\frac{}{(\underline{while}\ b\ \underline{do}\ c\ \underline{od}, Abs) \Rightarrow (c; \underline{while}\ b\ \underline{do}\ c\ \underline{od}, Abs \setminus \{\sigma \mid S_{Abs}[\![b]\!](\sigma') = \{0\}\})}$$

$$\frac{}{(\underline{while}\ b\ \underline{do}\ c\ \underline{od}, Abs) \Rightarrow Abs \setminus \{\sigma \mid S_{Abs}[\![b]\!](\sigma') = \{1\}\}}$$

Beachte: Bei bedingten Anweisungen werden die abstrakten Zustände entfernt, die auf jeden Fall die andere Verzweigung ausgeführt hätten.

<u>Definition:</u>

$$post^{\#}_{c,c'}(Abs) := \begin{cases} Abs', & \text{falls } (c, Abs) \Rightarrow (c', Abs') \\ \emptyset, & \text{sonst.} \end{cases}$$

$$post^{\#}_{c}(Abs) := \begin{cases} Abs', & \text{falls } (c, Abs) \Rightarrow Abs' \\ \emptyset, & \text{sonst} \end{cases}$$

<u>Satz:</u>
Die Familie von Funktionen $post^{\#}_{c(c')}$ ist eine abstrakte Semantik,
also

$$\alpha \circ post_{c(c')} \circ \gamma \leq post^{\#}_{c(c')}.$$