Insgesamt erhalten wir somit, dass die Datenkomplexität von posDTC in LOGSPACE liegt.

Auf ähnliche Weise kann man auch zeigen, dass die Datenkomplexität von posTC in NLOGSPACE liegt — zum Auswerten einer Formel der Form $[\mathbf{tc}_{\vec{x},\vec{y}}\,\varphi](\vec{s},\vec{t})$ “rät” eine nicht-deterministische Logspace-beschränkte Turing-Maschine an Stelle der Schleife in den Zeilen 4–7 dabei einfach einen Knoten $\vec{u}$, für den sie dann testet, ob $\mathfrak{A} \models \varphi[\vec{v},\vec{u}]$ gilt.

“$\supseteq$:” Sei nun C eine Klasse endlicher geordneter σ-Strukturen, so dass $L_{\mathsf{C}} \in$ LOGSPACE. D.h. es gibt eine deterministische Turing-Maschine $M = (Q,\Sigma,\Gamma,\Delta,q_0,F)$ (die ein separates Eingabeband besitzt) und eine Konstante k, so dass M bei Eingabe der Kodierung einer endlichen geordneten σ-Struktur $\mathfrak{A}$ mit $n := |A|$ auf Platz $k \cdot \lfloor \log n \rfloor$ entscheidet, ob $\mathfrak{A} \in \mathsf{C}$. Wir können o.B.d.A. annehmen, dass $|enc(\mathfrak{A})| \leqslant n^k$.

Idee: Jede Konfiguration von M bei Eingabe $enc(\mathfrak{A})$ ist durch ein Tupel $(q, p_{\text{Eingabe}}, p, w)$ eindeutig bestimmt, wobei $q \in Q$ den aktuellen Zustand bezeichnet, $p_{\text{Eingabe}} \in \{0,..,n^k\}$ die Kopfposition auf dem Eingabeband, $p \in \{0,..,k\cdot \lfloor \log n\rfloor\}$ die Kopfposition auf dem Arbeitsband und $w \in \Gamma^{k\cdot\lfloor \log n\rfloor}$ die komplette Beschriftung des Arbeitsbandes. Daher gibt es eine Konstante $d \in \mathbb{N}$, so dass M bei Eingabe einer Struktur $\mathfrak{A}$ mit $n := |A|$ höchstens $2^{d\cdot\log n} = n^d$ Konfigurationen durchlaufen kann. Jede einzelne solche Konfiguration kann man durch ein d-Tupel $\vec{v} \in A^d$ repräsentieren.
Das Problem C wird dann durch eine posDTC$[\sigma]$-Formel der Form

$$\psi \quad := \quad \exists\vec{s}\ \exists\vec{t}\ \Big(\varphi_{\mathit{Start}}(\vec{s}\,) \ \wedge\ \varphi_{\mathit{Akzeptiere}}(\vec{t}\,) \ \wedge\ \big[\ \mathbf{dtc}_{\vec{x},\vec{y}}\ \varphi_{\mathit{Schritt}}(\vec{x},\vec{y})\,\big]\ (\vec{s},\vec{t}\,)\Big)$$

beschrieben, wobei

- $\varphi_{\mathit{Start}}(\vec{s}\,)$ besagt, dass $\vec{s}$ die Startkonfiguration von M bei Eingabe $enc(\mathfrak{A})$ ist,
- $\varphi_{\mathit{Akzeptiere}}(\vec{t}\,)$ besagt, dass $\vec{t}$ eine akzeptierende Endkonfiguration ist,
- $\varphi_{\mathit{Schritt}}(\vec{x},\vec{y})$ besagt, dass $\vec{y}$ eine Nachfolgekonfiguration von $\vec{x}$ ist.

Genauer: Jede Konfiguration $(q, p_{\text{Eingabe}}, p, w)$ wird durch ein Tupel der Länge

$$d \ := \ 1 + k + k + |\Gamma| \cdot k$$

der Form

$$\Big(q,\ \vec{p}_{\text{Eingabe}},\ \vec{p},\ \big(\vec{x}_\gamma\big)_{\gamma\in\Gamma}\Big)$$

repräsentiert, wobei q, $\vec{p}_{\text{Eingabe}}$ und $\vec{p}$ Zustand sowie Bandpositionen analog zum Beweis von Theorem 2.46 kodieren. Die Tupel $\vec{x}_\gamma = (x_{\gamma,0},..,x_{\gamma,k-1})$, für $\gamma \in \Gamma$, beschreiben die Beschriftung des Arbeitsbandes folgendermaßen. Für alle $j \in \{0,..,k{-}1\}$ und $i \in \{0,..,(\log n){-}1\}$ gilt:

$$\begin{array}{lcl} \text{an Position } j\cdot\log n + i \text{ steht das Symbol } \gamma & \iff & \text{das } i\text{-te Bit der Zahl } x_{\gamma,j} \text{ ist 1,} \\ & & \text{d.h. } \left\lfloor \frac{x_{\gamma,j}}{2^i} \right\rfloor \text{ ist ungerade.} \end{array}$$

Man kann zeigen (*Übung*), dass es posDTC[<]-Formeln $\varphi_{Bit=1}(x,y)$ und $\varphi_{Bit=0}(x,y)$ gibt, die besagen, dass das y-te Bit der Zahl x 1 bzw. 0 ist. Diese Formeln kann man nutzen, um posDTC[σ]-Formeln φ_{Start}, $\varphi_{Akzeptiere}$ und $\varphi_{Schritt}$ mit den gewünschten Eigenschaften zu konstruieren. Rest: *Übung*.

Mit derselben Vorgehensweise kann man auch für jede Klasse C mit $L_{\mathsf{C}} \in$ NLOGSPACE eine posTC-Formel konstruieren, die von genau den Strukturen erfüllt wird, die in C liegen. □

Aus Theorem 2.64 (a) und Proposition 2.62 folgt direkt:

2.65 Theorem. DTC *beschreibt* LOGSPACE *auf* FinOrd.

Um die analoge Aussage auch für TC und NLOGSPACE zu beweisen, muss man zeigen, dass auch *Negationen* von TC-Formeln durch nichtdeterministische Logspace-beschränkte Turing-Maschinen ausgewertet werden können. Dies wird durch folgenden Satz gewährleistet:

2.66 Theorem (Immerman, 1987). *Auf endlichen geordneten Strukturen ist* TC *äquivalent zu* posTC *(kurz:* posTC = TC *auf* FinOrd*).*

Beweis: Per Induktion nach dem Formelaufbau. Der Hauptschritt besteht darin, eine TC[σ]-Formel der Form

$$\psi \ := \ \neg \left[\ \mathbf{tc}_{\vec{x},\vec{y}} \ \varphi(\vec{x},\vec{y}) \right] (\vec{s},\vec{t})$$

in eine posTC[σ]-Formel zu übersetzen, die auf allen *geordneten* endlichen σ-Strukturen äquivalent zu ψ ist. Gemäß Induktionsannahme können wir davon ausgehen, dass sowohl φ als auch $\neg\varphi$ auf geordneten endlichen Strukturen äquivalent zu einer posTC-Formel sind. Sei $k \geqslant 1$ die Länge der Tupel $\vec{x}$ und $\vec{y}$, d.h. $\vec{x} = x_1,..,x_k$ und $\vec{y} = y_1,..,y_k$. Sei $\sigma' := \sigma \dot\cup \mathit{frei}(\psi)$.

Zu einer endlichen geordneten σ'-Struktur $\mathfrak{A}$ und k-Tupeln $\vec{a},\vec{b} \in A^k$ und jeder posTC[σ']-Formel $\chi(\vec{x},\vec{y})$ definieren wir

$$\mathrm{Dist}_{\chi}^{\mathfrak{A}}(\vec{a},\vec{b}) \ := \ \begin{cases} \infty\ , & \text{falls es keinen Pfad der Länge} \geqslant 1 \text{ in } G_{\chi,\mathfrak{A}} \text{ von } \vec{a} \text{ nach } \vec{b} \text{ gibt,} \\ \min\left\{ \ell \geqslant 1 \ \middle|\ \begin{array}{l} \text{es gibt in } G_{\chi,\mathfrak{A}} \text{ einen Pfad der} \\ \text{Länge } \ell \text{ von } \vec{a} \text{ nach } \vec{b} \end{array} \right\}, & \text{sonst.} \end{cases}$$

Es gilt:

$$\begin{array}{ll} & \mathfrak{A} \models \neg[\mathbf{tc}_{\vec{x},\vec{y}}\,\varphi](\vec{s},\vec{t}) \\ \iff & \left|\{\vec{b} \in A^k \ : \ \mathrm{Dist}_{\varphi}^{\mathfrak{A}}(\vec{s},\vec{b}) < \infty\}\right| \ = \ \left|\{\vec{b} \in A^k \ : \ \mathrm{Dist}_{\varphi(\vec{x},\vec{y})\,\wedge\,\neg\vec{y}=\vec{t}}^{\mathfrak{A}}(\vec{s},\vec{b}) < \infty\}\right| \qquad (*) \end{array}$$

d.h. die Anzahl der Elemente, die in $G_{\varphi,\mathfrak{A}}$ von Knoten $\vec{s}$ aus erreichbar sind, ist gleich der Anzahl der Elemente, die von $\vec{s}$ aus über einen Pfad erreichbar sind, der den Knoten $\vec{t}$ *nicht* durchläuft.

Ziel für den Rest des Beweises: Drücke die Gleichung $(*)$ durch eine posTC-Formel aus.

Schritt 1: Es gilt für alle Formeln $\chi(\vec{x},\vec{y})$ und alle Tupel $\vec{s},\vec{b}\in A^k$ und für $n:=|A|$:

$$\mathrm{Dist}_\chi^{\mathfrak{A}}(\vec{s},\vec{b})<\infty \iff \mathrm{Dist}_\chi^{\mathfrak{A}}(\vec{s},\vec{b})\leqslant n^k,$$

denn der Graph $G_{\chi,\mathfrak{A}}$ hat nur $|A|^k$ Knoten.

Schritt 2: Wie beim Beweis des Satzes von Immerman und Vardi (Theorem 2.46) können wir für $n:=|A|$ mittels der linearen Ordnung $<^{\mathfrak{A}}$ das Universum A mit der Menge $\{0,..,n{-}1\}$ identifizieren. Unter Verwendung der lexikographischen Ordnung $<_{lex}^{\mathfrak{A}}$ identifizieren wir $k{+}1$-Tupel $\vec{c}\in A^{k+1}$ mit Zahlen in $\{0,..,n^{k+1}{-}1\}$, indem wir dem Tupel $\vec{c}$ die Zahl

$$\big[\vec{c}\big]^{\mathfrak{A}} \;:=\; rg_{<_{lex}^{\mathfrak{A}}}(\vec{c}) \;\in\; \{0,..,n^{k+1}{-}1\}$$

zuordnen. Insbesondere gilt

$$|A|^k \;=\; n^k \;=\; \big[\,(1,\underbrace{0,0,\cdots,0}_{k})\,\big]^{\mathfrak{A}},$$

wobei 0 das kleinste und 1 das zweitkleinste Element in A bzgl. $<^{\mathfrak{A}}$ bezeichnen.
Wir nehmen im Folgenden o.B.d.A. an, dass die Signatur σ zwei Konstantensymbole 0 und 1 enthält, die in geordneten σ-Strukturen stets mit dem kleinsten bzw. dem zweitkleinsten Element interpretiert werden.

Schritt 3: Für jede posTC$[\sigma']$-Formel $\chi(\vec{x},\vec{y})$ konstruieren wir eine posTC-Formel $dist_\chi(\vec{x},\vec{y},\vec{u})$ (wobei $\vec{x}$ und $\vec{y}$ jeweils k-Tupel und $\vec{u}$ ein $k{+}1$-Tupel von Variablen erster Stufe sind), so dass für alle endlichen geordneten σ'-Strukturen $\mathfrak{A}$ und alle $\vec{a},\vec{b}\in A^k$ und $\vec{d}\in A^{k+1}$ gilt:

$$\mathfrak{A}\models dist_\chi[\vec{a},\vec{b},\vec{d}] \quad\iff\quad \mathrm{Dist}_\chi^{\mathfrak{A}}(\vec{a},\vec{b}) \;\leqslant\; \big[\vec{d}\,\big]^{\mathfrak{A}}.$$

Dazu wählen wir

$$dist_\chi(\vec{x},\vec{y},\vec{u}) \;:=\; \big[\;\mathbf{tc}_{\vec{x},\vec{u},\vec{x}',\vec{u}'}\;\chi(\vec{x},\vec{x}\,')\,\wedge\, succ_{<_{lex}}(\vec{u},\vec{u}\,')\,\big]\,(\vec{x},\vec{0},\vec{y},\vec{u}).$$

(Idee dabei: In dem Tupel $\vec{u}$ wird die Länge eines Pfads von $\vec{x}$ nach $\vec{y}$ gezählt.)

Schritt 4: Für jede TC$[\sigma']$-Formel $\chi(\vec{x},\vec{y})$, so dass sowohl χ als auch $\neg\chi$ äquivalent zu einer posTC-Formel sind, konstruieren wir eine posTC-Formel $anzahl_\chi(\vec{x},\vec{z})$, so dass für alle endlichen geordneten σ'-Strukturen $\mathfrak{A}$, alle $\vec{s}\in A^k$ und alle $\vec{c}\in A^{k+1}$ gilt:

$$\mathfrak{A}\models anzahl_\chi\,[\vec{s},\vec{c}] \quad\iff\quad \big[\vec{c}\big]^{\mathfrak{A}} \;=\; \big|\,\{\vec{b}\,:\mathrm{Dist}_\chi^{\mathfrak{A}}\,(\vec{s},\vec{b})<\infty\,\}\,\big|\,.$$

Dazu betrachten wir zunächst die Funktion

$$\mathrm{Anz}_\chi^{\mathfrak{A}}\;:\;A^k\times\{0,..,n^k\}\;\to\;\{0,..,n^k\}$$

mit

$$\mathrm{Anz}_\chi^{\mathfrak{A}}(\vec{s},d) \;\; := \;\; \big|\,\{\vec{b}\in A^k \;:\; \mathrm{Dist}_\chi^{\mathfrak{A}}(\vec{s},\vec{b}) \leqslant d\}\,\big|\,.$$

Für jedes feste $\vec{s}\in A^k$ können wir $\mathrm{Anz}_\chi^{\mathfrak{A}}(\vec{s},d)$ induktiv für $d=0,1,2,\ldots$ folgendermaßen bestimmen:

Induktionsanfang $d=0$:

$\mathrm{Anz}_\chi^{\mathfrak{A}}(\vec{s},0)=0$, denn für alle $\vec{b}\in A^k$ gilt gemäß Definition, dass $\mathrm{Dist}_\chi^{\mathfrak{A}}(\vec{s},\vec{b})\geqslant 1$.

Induktionssschritt $d\to d{+}1$:

Sei $c := \mathrm{Anz}_\chi^{\mathfrak{A}}(\vec{s},d)$. Gesucht ist die Zahl c', so dass $\mathrm{Anz}_\chi^{\mathfrak{A}}(\vec{s},d{+}1) = c'$. Algorithmisch können wir dies ermitteln, indem wir mit einem Variablentupel $\vec{y}\,'$ nach und nach ganz A^k gemäß der lexikographischen Ordnung durchlaufen und dabei den Wert einer Variablen v jedesmal dann um 1 hochzählen, wenn für das gerade betrachtete Tupel $\vec{y}\,'$ gilt: $\mathrm{Dist}_\chi^{\mathfrak{A}}(\vec{s},\vec{y}\,')\leqslant d{+}1$.

Unser nächstes Ziel ist nun, eine posTC-Formel ζ_χ zu konstruieren, die auf dieser Methode basiert. Sie benutzt $k{+}1$-Tupel $\vec{v}$, $\vec{u}$ und $\vec{u}\,'$, um die Anzahl der gefundenen k-Tupel bzw. die Distanzen d und $d{+}1$ zu repräsentieren. Die Formel

$$\zeta_\chi(\vec{u},\vec{z},\vec{u}\,',\vec{z}\,',\vec{s})$$

soll für $\vec{u},\vec{u}\,'$ mit $\big[\vec{u}\,'\big]^{\mathfrak{A}} = \big[\vec{u}\big]^{\mathfrak{A}}{+}1$ besagen:

$$\text{Falls} \quad \big[\vec{z}\big]^{\mathfrak{A}} = \mathrm{Anz}_\chi^{\mathfrak{A}}\big(\vec{s},\big[\vec{u}\big]^{\mathfrak{A}}\big), \quad \text{so} \quad \big[\vec{z}\,'\big]^{\mathfrak{A}} = \mathrm{Anz}_\chi^{\mathfrak{A}}\big(\vec{s},\big[\vec{u}\,'\big]^{\mathfrak{A}}\big).$$

Wir wählen dazu

$$\zeta_\chi(\vec{u},\vec{z},\vec{u}\,',\vec{z}\,',\vec{s}) \;\; := \;\; \big[\,\mathbf{tc}_{\vec{y},\vec{v},\vec{y}',\vec{v}'}\;\delta_\chi\,\big]\,(\vec{0},\vec{0},\vec{max},\vec{z}\,'),$$

wobei

$$\begin{aligned}\delta_\chi(\vec{y},\vec{v},\vec{y}\,',\vec{v}\,',\vec{s},\vec{u},\vec{u}\,') \;\; := \;\; \Big(& \big(\, \mathit{succ}_{<_{lex}}(\vec{y},\vec{y}\,') \;\wedge\; \mathit{dist}_\chi(\vec{s},\vec{y}\,',\vec{u}\,') \;\wedge\; \mathit{succ}_{<_{lex}}(\vec{v},\vec{v}\,')\,\big) \\ & \vee\; \big(\, \mathit{succ}_{<_{lex}}(\vec{y},\vec{y}\,') \;\wedge\; \text{“}\neg\,\mathit{dist}_\chi(\vec{s},\vec{y}\,',\vec{u}\,')\text{”} \;\wedge\; \vec{v}=\vec{v}\,' \,\big) \Big).\end{aligned}$$

Um sicherzustellen, dass die Formel ζ_χ in posTC liegt, müssen wir noch eine posTC-Formel für “$\neg\,\mathit{dist}_\chi(\vec{s},\vec{y}\,',\vec{u}\,')$” finden, d.h. wir müssen eine posTC-Formel finden, die besagt

$$\mathrm{Dist}_\chi^{\mathfrak{A}}\big(\vec{s},\vec{y}\,'\big) \;\; > \;\; \big[\vec{u}\,'\big]^{\mathfrak{A}}.$$

Falls $\big[\vec{u}\,'\big]^{\mathfrak{A}} = \big[\vec{u}\big]^{\mathfrak{A}}{+}1$ und $\big[\vec{z}\big]^{\mathfrak{A}} = \mathrm{Anz}_\chi^{\mathfrak{A}}\big(\vec{s},\big[\vec{u}\big]^{\mathfrak{A}}\big)$, so gilt:

- Ist $\big[\vec{u}\big]^{\mathfrak{A}}=0$, so $\mathrm{Dist}_\chi^{\mathfrak{A}}(\vec{s},\vec{y}\,') > \big[\vec{u}\,'\big]^{\mathfrak{A}} \iff \mathfrak{A}\not\models\chi(\vec{s},\vec{y}\,')$.
- Ist $\big[\vec{u}\big]^{\mathfrak{A}}>0$, so $\mathrm{Dist}_\chi^{\mathfrak{A}}(\vec{s},\vec{y}\,') > \big[\vec{u}\,'\big]^{\mathfrak{A}} \iff$ es gibt $\vec{z}$ Elemente $\vec{w}\,'\neq\vec{y}\,'$, so dass $\mathrm{Dist}_\chi^{\mathfrak{A}}(\vec{s},\vec{w}\,')\leqslant\big[\vec{u}\big]^{\mathfrak{A}}$ und $\mathfrak{A}\not\models\chi(\vec{w}\,',\vec{y}\,')$.

Genau dies wird durch die folgende posTC-Formel bewerkstelligt, die in der Formel δ_χ an die Stelle von "$\neg dist_\chi(\vec{s}, \vec{y}\,', \vec{u}\,')$" gesetzt werden muss:

$$\Big(\vec{u} = \vec{0} \;\wedge\; \neg\,\chi(\vec{s}, \vec{y}\,')\Big) \vee$$
$$\Big(\vec{u} \neq \vec{0} \;\wedge\; \left[\mathbf{tc}_{\vec{w},\vec{q},\vec{w}',\vec{q}'} \begin{pmatrix} \big(\mathit{succ}_{<_{lex}}(\vec{w}, \vec{w}\,') \;\wedge\; \vec{q} = \vec{q}\,'\big) \vee \\ \big(\mathit{succ}_{<_{lex}}(\vec{w}, \vec{w}\,') \;\wedge\; \mathit{succ}_{<_{lex}}(\vec{q}, \vec{q}\,') \;\wedge \\ \vec{w}\,' \neq \vec{y}\,' \;\wedge\; \mathit{dist}_\chi(\vec{s}, \vec{w}\,', \vec{u}) \;\wedge\; \neg\,\chi(\vec{w}\,', \vec{y}\,')\big) \end{pmatrix}\right] (\vec{0}, \vec{0}, m\vec{a}x, \vec{z})\Big).$$

(Idee dabei: Durchlaufe A^k mit Variablen $\vec{w}\,'$ gemäß der lexikographischen Ordnung und zähle den Wert der Variablen $\vec{q}$ immer dann um 1 hoch, wenn $\vec{w}\,' \neq \vec{y}\,'$ und $\mathrm{Dist}_\chi^{\mathfrak{A}}(\vec{s}, \vec{w}\,') \leqslant [\vec{u}]^{\mathfrak{A}}$ und $\mathfrak{A} \not\models \chi(\vec{w}, \vec{y}\,')$.)

Schließlich wählen wir

$$\mathit{anzahl}_\chi(\vec{s}, \vec{z}) \;:=\; \big[\,\mathbf{tc}_{\vec{u},\vec{z},\vec{u}',\vec{z}'}\; \mathit{succ}_{<_{lex}}(\vec{u}, \vec{u}\,') \;\wedge\; \zeta_\chi(\vec{u}, \vec{z}, \vec{u}\,', \vec{z}\,', \vec{s})\,\big]\,(\underbrace{0\vec{0}}_{\triangleq 0}, \vec{0}, \underbrace{1\vec{0}}_{\triangleq n^k}, \vec{z})\,.$$

Dies ist eine posTC-Formel, die besagt, dass $[\vec{z}]^{\mathfrak{A}}$ die Anzahl aller Knoten von $G_{\chi,\mathfrak{A}}$ ist, die über einen Pfad der Länge $\leqslant n^k$ von Knoten $\vec{s}$ aus erreichbar sind. Gemäß Schritt 1 besagt die Formel $\mathit{anzahl}_\chi(\vec{s}, \vec{z})$, dass $[\vec{z}]^{\mathfrak{A}}$ die Anzahl der von $\vec{s}$ aus erreichbaren Knoten in $G_{\chi,\mathfrak{A}}$ ist. Somit sind wir fertig mit Schritt 4.

Schritt 5: Unter Benutzung der Gleichung $(*)$ und der posTC-Formel aus Schritt 4 erhalten wir, dass die Formel

$$\neg\,\big[\,\mathbf{tc}_{\vec{x},\vec{y}}\;\varphi(\vec{x}, \vec{y})\,\big]\,(\vec{s}, \vec{t})$$

äquivalent zur posTC-Formel

$$\exists \vec{z}\,\Big(\mathit{anzahl}_\varphi(\vec{s}, \vec{z}) \;\wedge\; \mathit{anzahl}_{\varphi(\vec{x},\vec{y})\wedge\neg\vec{y}=\vec{t}}(\vec{s}, \vec{z})\Big)$$

ist. Dies beendet den Beweis von Theorem 2.66. □

Aus Theorem 2.64 (b) und Theorem 2.66 folgt direkt:

2.67 Theorem. TC *beschreibt* NLOGSPACE *auf* FinOrd.

Mit Hilfe dieser logischen Beschreibung von NLOGSPACE kann man leicht beweisen, dass NLOGSPACE unter Komplementbildung abgeschlossen ist:

2.68 Korollar (Satz von Immerman und Szelepcsényi). NLOGSPACE = co-NLOGSPACE.

Beweis: "$\supseteq$:" Wir müssen zeigen, dass für jede Sprache $L \in$ NLOGSPACE gilt, dass auch das Komplement $\overline{L}$ in NLOGSPACE liegt. Sei also L ein Problem in NLOGSPACE. Gemäß Theorem 2.67 gibt es einen TC-Satz ψ, der L beschreibt, d.h. für alle Strukturen $\mathfrak{A}$ gilt: $\mathfrak{A} \in L \iff \mathfrak{A} \models \psi$. Da die Logik TC abgeschlossen ist unter Negation, ist $\psi' := \neg\psi$ ein TC-Satz, und es gilt für alle Strukturen $\mathfrak{A}$: $\mathfrak{A} \models \psi' \iff \mathfrak{A} \notin L \iff \mathfrak{A} \in \overline{L}$. Somit ist ψ' ein TC-Satz, der das Problem $\overline{L}$ beschreibt. Gemäß Theorem 2.67 folgt $\overline{L} \in$ NLOGSPACE.
"$\subseteq$:" analog. □

Zusammenfassung der logischen Charakterisierungen:

In Kapitel 2 wurden die in Abbildung 2.1 dargestellten logischen Charakterisierungen von Komplexitätsklassen behandelt.

PFP	$\longrightarrow$ beschreibt auf FinOrd $\longrightarrow$	PSPACE
		$\cup\!\mid$
SO	$\longrightarrow$ beschreibt auf Fin $\longrightarrow$	PH
		$\cup\!\mid$
ESO	$\longrightarrow$ beschreibt auf Fin $\longrightarrow$	NP
		$\cup\!\mid$
LFP, IFP	$\longrightarrow$ beschreibt auf FinOrd $\longrightarrow$	PTIME
		$\cup\!\mid$
TC, posTC	$\longrightarrow$ beschreibt auf FinOrd $\longrightarrow$	NLOGSPACE
		$\cup\!\mid$
DTC, posDTC	$\longrightarrow$ beschreibt auf FinOrd $\longrightarrow$	LOGSPACE

Abbildung 2.1: Logische Beschreibungen von Komplexitätsklassen.

2.4 Interpretationen und Logische Reduktionen

2.69 Definition. Sei $\mathscr{L}$ eine Logik, S eine Klasse von Strukturen, σ eine Signatur und $\mathsf{C} \subseteq \mathsf{S}$ eine Klasse von σ-Strukturen.
C heißt *$\mathscr{L}$-definierbar in* S (auch: $\mathscr{L}$-axiomatisierbar), falls es einen $\mathscr{L}[\sigma]$-Satz φ gibt, so dass $\mathsf{C} = \mathrm{Mod}_{\mathsf{S}}(\varphi)$.
(D.h. für alle σ-Strukturen $\mathfrak{A} \in \mathsf{S}$ gilt: $\mathfrak{A} \in \mathsf{C} \iff \mathfrak{A} \models \varphi$.)

2.70 Definition. Sei σ eine Signatur. Mit σ-$\mathsf{STRUKTUREN}$ bezeichnen wir die Klasse aller σ-Strukturen.

Ziel in diesem Abschnitt:

Begriff für "logische Reduktionen", analog zum Begriff der Polynomialzeit-Reduktionen. Statt Problemen $A \subseteq \Sigma_1^*$, $B \subseteq \Sigma_2^*$ und einer Polynomialzeit-berechenbaren Reduktion $f : \Sigma_1^* \to \Sigma_2^*$ mit $w \in A \iff f(w) \in B$ jetzt: