Seminar Parametrisierte Algorithmen für $\mathcal{NP}$-schwere Probleme

Reinhard Bauer, Marcus Krug, Ignaz Rutter, Dorothea Wagner

Lehrstuhl für Algorithmik I
Institut für Theoretische Informatik
Universität Karlsruhe (TH)

8. Mai 2009

Übersicht

Themen

- Parametrisierung / FPT
- Charakterisierung von FPT
- Approximierbarkeit und FPT
- Beschränkte FPT
- Parametrisierte Komplexitätstheorie

Literatur

- *Parameterized Complexity Theory*, J. Flum, M. Grohe, 2006
- *An Invitation to Fixed-Parameter Algorithms*, R. Niedermeier, 2006

Disclaimer

- viele Definitionen, Lemmata und Theoreme
- wenige Beweise
- einiges etwas trocken, aber wichtig, die Dinge mal sauber definiert zu haben

- $\mathbb{N} = \{1, 2, \ldots\}$

Parametrisierung

Definition (Parametrisierung / Parametrisiertes Problem)

Sei Σ ein endliches Alphabet.

- Eine Parametrisierung ist eine Abbildung $\kappa: \Sigma^* \to \mathbb{N}$, die in Polynomialzeit berechenbar ist.
- Ein Parametrisiertes Problem (über Σ) ist ein Tupel (Q, κ) mit $Q \subseteq \Sigma^*$ und einer Parametrisierung κ.

Beispiel: Parametrisierung

Problem p-SAT

- $Q := \{\alpha \mid \alpha$ erfüllbare aussagenlogische Formel$\}$
- Parametrisierung: $\alpha \mapsto \kappa(\alpha) :=$ Anzahl der Variablen von α

Problem p-SAT

Instanz:	Aussagenlogische Formel α
Parameter:	Anzahl der Variablen von α
Problem:	Entscheide, ob α erfüllbar ist

Beispiel: Parametrisierung

*Problem p-*COLORABILITY

Instanz: Graph $G = (V, E)$, $k \in \mathbb{N}$

Parameter: k

Problem: Entscheide, ob G k-färbbar ist.

Parametrisierung

- $(G, k) \mapsto k$
- Laufzeit $\mathcal{O}(|(G, k)|)$

Definition ($\mathcal{NP}$-Optimierungsproblem (Wdh.))

Ein $\mathcal{NP}$-Optimierungsproblem über Σ ist ein Tripel $\mathcal{O} = (\mathrm{sol}, \mathrm{cost}, \mathrm{goal})$, so dass

(1) sol ist berechenbare Funktion $\Sigma^* \to 2^{\Sigma^*}$, so dass die Relation

$$\mathcal{R} := \{(x, y) \mid x \in \Sigma^*,\ y \in \mathrm{sol}(x)\}$$

polynomiell balanciert ist

(2) $\mathrm{cost} : \mathcal{R} \to \mathbb{N}$ ist polynomiell berechenbar

(3) $\mathrm{goal} \in \{\max, \min\}$

Optimum:

$$opt_{\mathcal{O}}(x) := \mathrm{goal}\{\mathrm{cost}(x, y) \mid y \in \mathrm{sol}(x)\}$$

Standardparametrisierung

Definition

Sei $\mathcal{O} = (\mathrm{sol}, \mathrm{cost}, \mathrm{goal})$ ein $\mathcal{NP}$-Optimierungsproblem.

Problem p-$\mathcal{O}$

Instanz: $x \in \Sigma^*$, $k \in \mathbb{N}$

Parameter: k

Problem: Entscheide, ob $\mathrm{opt}_{\mathcal{O}}(x) \geq k$, falls $\mathrm{goal} = \max$ oder $\mathrm{opt}_{\mathcal{O}}(x) \leq k$, falls $\mathrm{goal} = \min$

Fixed-Parameter Tractability

Definition (FPT)

Sei Σ endliches Alphabet, κ Parametrisierung

- Ein Algorithmus $\mathcal{A}$ heißt FPT-Algorithmus bzgl. κ, falls $f : \mathbb{N} \to \mathbb{N}$ berechenbar und ein Polynom p existieren, so dass die Laufzeit von $\mathcal{A}$ für alle $x \in \Sigma^*$ beschränkt ist durch

$$f(\kappa(x)) \cdot p(|x|)$$

- Ein Parametrisiertes Problem (Q, κ) ist fixed-parameter tractable (FPT), falls es einen FPT-Algorithmus (bzgl. κ) gibt, der Q entscheidet.

Beispiel: *p*-SAT

Lemma

p-SAT ist FPT bzgl. Parametrisierung "Anzahl der Variablen"

Beweis

- Brute-Force Algorithmus (Suchbaum) hat Laufzeit $\mathcal{O}(2^k \cdot p(|\alpha|))$.

Spezielle Parametrisierungen

Betrachte Parametrisierung

$$\kappa_{size}(x) := \max\{1, |x|\}$$

Beobachtung

- Q entscheidbar $\Rightarrow$ (Q, κ_{size}) ist FPT

Spezielle Parametrisierungen

Betrachte Parametrisierung

$$\kappa_{one}(x) := 1 \text{ für alle } x \in \Sigma^*$$

Beobachtung

- $Q \subseteq \Sigma^* \Rightarrow (Q, \kappa_{one})$ ist FPT genau dann, wenn Q in Polynomialzeit entscheidbar ist

Spezielle Parametrisierungen

Lemma

Sei $g : \mathbb{N} \to \mathbb{N}$ berechenbar, monoton steigend und unbeschränkt, Σ endl. Alphabet, κ Parametrisierung mit

$$\kappa(x) \geq g(|x|) \text{ für alle } x \in \Sigma^*.$$

Sei $Q \subseteq \Sigma^*$ entscheidbar. Dann ist (Q, κ) FPT.

- Jedes parametrisierte Problem, bei dem der Parameter monoton mit der Eingabelänge wächst, ist FPT

Slice

Definition (ℓ-Slice)

Sei (Q, κ) ein parametrisiertes Problem. Die ℓ-te Slice von (Q, κ) ist definiert als

$$(Q, \kappa)_\ell := \{x \in Q \mid \kappa(x) = \ell\} \ .$$

Lemma

Sei (Q, κ) ein parametrisiertes Problem und $\ell \in \mathbb{N}$. Falls (Q, κ) FPT ist, dann ist $(Q, \ell) \in \mathcal{P}$.

Folgerung für p-Colorability

*Problem p-*COLORABILITY

Instanz: Graph $G = (V, E)$, $k \in \mathbb{N}$

Parameter: k

Problem: Entscheide, ob G k-färbbar ist.

Lemma

*Falls $\mathcal{P} \neq \mathcal{NP}$ gilt, ist p-*COLORABILITY *nicht FPT.*

Beweis

- 3-Slice von p-COLORABILITY (3-Color) ist $\mathcal{NP}$-schwer

Kernbildung

Definition (Kernbildung)

Sei (Q, κ) ein parametrisiertes Problem über Σ.

Eine polynomiell berechenbare Funktion $K : \Sigma^* \to \Sigma^*$ heißt Kernbildung, falls es eine berechenbare Funktion $h : \mathbb{N} \to \mathbb{N}$ gibt, so dass

(i) $x \in Q \Leftrightarrow K(x) \in Q$ und

(ii) $|K(x)| \leq h(\kappa(x))$

≫ triviales Beispiel?: $x \mapsto x$ ist Kernbildung für (Q, κ_{size})

Beispiel p-Vertx-Cover

Problem p-Vertex Cover

Gegeben:	Graph $G = (V, E)$, $k \in \mathbb{N}$
Parameter:	k
Problem:	Entscheide, ob es ein Vertex Cover V' mit k Knoten gibt

Reduktion von Buss für p-Vertx-Cover

Beobachtung

- $deg(v) \geq k+1 \Rightarrow v$ muss ins Vertex Cover
- $\max_{v \in V} \deg(v) \leq k$ und G hat Vertex Cover der Größe k
  $\Rightarrow G$ hat höchstens k^2 Kanten

Kernbildung

- $(G,k) \mapsto (G',k') := \textsc{Reduce}(G,k)$

$\Rightarrow |(G',k')| = \mathcal{O}(k^2)$

Reduktion von Buss

Algorithmus 1 : $\textsc{Reduce}(G, k)$

Input : Graph $G = (V, E)$, $k \in \mathbb{N}$

if $k = 0$ **then**
  if $E = \emptyset$ **then** **return** $(G^+, 1)$ **else** **return** $(G^-, 1)$
else
  if $\exists v \in V : \; deg(v) > k$ **then**
    return $\textsc{Reduce}(G - v, k - 1)$
  else
    if $|E| \leq k^2$ **then** **return** (G, k) **else** **return** $(G^-, 1)$

$(G^+, 1)$ ist triviale Ja-Instanz, $(G^-, 1)$ ist triviale Nein-Instanz

FPT und Kernbildung

Theorem

Sei (Q, κ) ein Parametrisiertes Problem. Dann sind äquivalent

(1) $(Q, \kappa) \in FPT$

(2) *Q ist entscheidbar und es existiert eine Kernbildung für (Q, κ)*

Beweis

- Tafel

Alternative FPT-Charakterisierungen

Definition

(Q, κ) ist in $\mathcal{P}$ nach einer Vorberechnung auf dem Parameter, falls es ein Alphabet Π, eine berechenbare Funktion $\pi : \mathbb{N} \rightarrow \Pi^*$ und ein Problem $X \subseteq \Sigma^* \times \Pi^*$ gibt, so dass $X \in \mathcal{P}$ liegt und

$$x \in Q \Leftrightarrow (x, \pi(\kappa(x))) \in X$$

für alle $x \in \Sigma^*$ gilt.

Alternative FPT-Charakterisierungen

Definition

(Q, κ) ist irgendwann in $\mathcal{P}$, falls es eine berechenbare Funktion $h : \mathbb{N} \to \mathbb{N}$ und einen Polynomialzeit-Algorithmus $\mathcal{A}$ gibt, der für alle $x \in \Sigma^*$ mit $|x| \geq h(\kappa(x))$ entscheidet, ob $x \in Q$ gilt.

(Das Verhalten für x mit $|x| \leq h(\kappa(x))$ ist nicht spezifiziert.)

Alternative FPT-Charakterisierungen

Theorem

Sei (Q, κ) ein parametrisiertes Problem. Dann sind äquivalent:

(1) (Q, κ) *ist FPT*

(2) (Q, κ) *ist in* $\mathcal{P}$ *nach einer Vorberechnung auf dem Parameter*

(3) (Q, κ) *ist irgendwann in* $\mathcal{P}$

Approximierbarkeit und FPT

Definition (PTAS (Wdh.))

Sei $\mathcal{O} = (\mathrm{sol}, \mathrm{cost}, \mathrm{goal})$ ein $\mathcal{NP}$-Optimierungsproblem über Σ.

Ein Polynomialzeit-Approximationsschema (PTAS) $\mathcal{A}$ für $\mathcal{O}$ ist ein Algorithmus mit folgenden Eigenschaften:

- Eingabe: $(x, k) \in \Sigma^* \times \mathbb{N}$
- Ausgabe: $\frac{1}{k}$-Approximation $y \in \Sigma^*$ mit

$$\frac{\mathrm{cost}(x,y)}{opt_{\mathcal{O}}(x)} \leq \left(1 + \frac{1}{k}\right) \text{ bzw. } \frac{opt_{\mathcal{O}}(x)}{\mathrm{cost}(x,y)} \leq \left(1 + \frac{1}{k}\right)$$

- für festes k ist Laufzeit polynomiell

Approximierbarkeit und FPT

Definition (EPTAS)

Sei $\mathcal{O} = (\text{sol}, \text{cost}, \text{goal})$ ein $\mathcal{NP}$-Optimierungsproblem über Σ.

Ein effizientes PTAS (EPTAS) für $\mathcal{O}$ ist ein PTAS $\mathcal{A}$, dessen Laufzeit beschränkt ist durch $f(k) \cdot p(|x|)$, wobei $f : \mathbb{N} \to \mathbb{N}$ eine berechenbare Funktion und p ein Polynom sind.

$$\text{FPTAS} \subseteq \text{EPTAS} \subseteq \text{PTAS}$$

Approximierbarkeit und FPT

Theorem

Sei $\mathcal{O} = (\mathrm{sol}, \mathrm{cost}, \mathrm{goal})$ ein $\mathcal{NP}$-Optimierungsproblem, für das es ein EPTAS $\mathcal{A}$ gibt. Dann ist die Standardparametrisierung p-$\mathcal{O}$ FPT.

Beweis

- Tafel

Beschränkte Fixed-Paramter Tractability

Definition ($\mathcal{F}$-FPT)

Sei $\mathcal{F}$ Menge von Funktionen $f : \mathbb{N} \to \mathbb{N}$.

Ein parametrisiertes Problem (Q, κ) ist $\mathcal{F}$-FPT, falls es eine Funktion $f \in \mathcal{F}$, ein Polynom p und einen Algorithmus $\mathcal{A}$ gibt, der in höchstens $f(\kappa(x)) \cdot p(|x|)$ Zeitschritten entscheidet, ob $x \in Q$ liegt.

Beispielklassen

- $2^{k^{\mathcal{O}(1)}}$-FPT= EXPT
- $2^{\mathcal{O}(k)}$-FPT= EPT
- $2^{o^{\text{eff}}(k)}$-FPT= SUBEPT

Parametrisierte Komplexitätstheorie

Definition (FPT-Reduktion)

Seien (Q, κ) und (Q', κ') Parametrisierte Problem über Σ bzw. Σ'.

Eine FPT-Reduktion von (Q, κ) nach (Q', κ') ist eine Abbildung $R : \Sigma^* \to (\Sigma')^*$ mit folgenden Eigenschaften:

- $x \in Q \Leftrightarrow R(x) \in Q'$
- R kann in FPT-Zeit (bzgl. κ) berechnet werden
- es existiert eine berechenbare Funktion $g : \mathbb{N} \to \mathbb{N}$, so dass $\kappa'(R(x)) \leq g(\kappa(x))$ für alle $x \in \Sigma^*$ gilt.

- Notation: $(Q, \kappa) \leq^{\mathsf{FPT}} (Q', \kappa'), (Q, \kappa) \equiv^{\mathsf{FPT}} (Q', \kappa')$

Eigenschaften von FPT-Reduktionen

Lemma

Die Relation $\leq^{FPT}$ ist reflexiv und transitiv.

Lemma

FPT ist abgeschlossen unter FPT-Reduktionen, d.h.
$(Q,\kappa) \leq^{FPT} (Q',\kappa')$ *und* $(Q',\kappa') \in FPT \Rightarrow (Q,\kappa) \in FPT$.

Schwere / Vollständigkeit

Definition ($\mathcal{C}$-schwer / $\mathcal{C}$-vollständig unter FPT-Reduktionen)

Sei $\mathcal{C}$ eine Klasse von parametrisierten Problemen und (Q, κ) ein parametrisiertes Problem.

- (Q, κ) ist $\mathcal{C}$-schwer unter FPT-Reduktionen, falls $(Q', \kappa') \leq^{\text{FPT}} (Q, \kappa)$ für alle $(Q', \kappa') \in C$
- (Q, κ) ist $\mathcal{C}$-vollständig unter FPT-Reduktionen, falls $(Q, \kappa) \in \mathcal{C}$ gilt und (Q, κ) $\mathcal{C}$-schwer ist

Beispiel

- p-$\textsc{Clique} \equiv^{fpt}$ p-$\textsc{Independent-Set}$
- FPT-Reduktion

$$R(x) := \begin{cases} enc(\overline{G}, k), & \text{falls } x = enc(G, k) \\ x, & \text{sonst} \end{cases}$$

- *enc* bildet Graph und Parameter auf Kodierung über Σ^* ab

Para-$\mathcal{NP}$

Definition (FPTpara-$\mathcal{NP}$)

Sei (Q, κ) parametrisiertes Problem über Σ .

(Q, κ) ist in FPTpara-$\mathcal{NP}$, falls

- es eine berechenbare Funktion $f : \mathbb{N} \to \mathbb{N}$, ein Polynom p sowie
- einen nicht-deterministischen Algorithmus $\mathcal{A}$ gibt, der $x \in Q$ für $x \in \Sigma^*$ entscheidet und
- in höchstens $f(\kappa(x)) \cdot p(|x|)$ Schritten terminiert.

Para-$\mathcal{NP}$ Eigenschaften

Proposition

Sei (Q, κ) ein parametrisiertes Problem über Σ. Dann sind äquivalent:

(1) *(Q, κ) ist in para-$\mathcal{NP}$*

(2) *(Q, κ) ist "in $\mathcal{NP}$ nach einer Vorberechnung auf dem Parameter"*

(3) *Q ist entscheidbar und (Q, κ) ist "irgendwann in $\mathcal{NP}$"*

Corollary

$FPT =$ para-$\mathcal{NP}$ genau dann, wenn $\mathcal{P} = \mathcal{NP}$.

Para-$\mathcal{NP}$ Eigenschaften

Theorem

Sei (Q, κ) ein nicht-triviales parametrisiertes Problem in para-$\mathcal{NP}$. Dann sind Äquivalent:

(1) *(Q, κ) ist para-$\mathcal{NP}$-vollständig unter FPT-Reduktionen*

(2) *Die Vereinigung von endlich vielen Slices ist $\mathcal{NP}$-vollständig*

Corollary

p-Colorability *ist para-$\mathcal{NP}$-vollständig unter FPT-Reduktionen.*

Para-$\mathcal{NP}$ Diskussion

Eigenschaften

- FPT $\subseteq$ para-$\mathcal{NP}$ (klar)

Problem

- zu viele Probleme sind para-$\mathcal{NP}$-schwer: eine $\mathcal{NP}$-vollständige Slice genügt
- hauptsächlich negative Resultate

$\mathbf{XP}_{nu}$

Definition (XP_{nu})

Ein parametrisiertes Problem (Q, κ) ist in XP_{nu} (nicht-uniformes XP), wenn die Slices $(Q, \kappa)_\ell$ für alle $\ell \geq 1$ in $\mathcal{P}$ sind.

Probleme

- non-uniform: Algorithmus kann für jedes ℓ unterschiedlich sein
- XP_{nu} enthält unentscheidbare Probleme: (Q, κ_{size}) mit $Q \subseteq \{1\}^*$ unentscheidbar

XP

Definition (XP)

Ein parametrisiertes Problem gehört zur Klasse XP, falls es eine berechenbare Funktion $f : \mathbb{N} \to \mathbb{N}$ und einen Algorithmus $\mathcal{A}$ gibt, die für alle $x \in \Sigma^*$ in höchstens

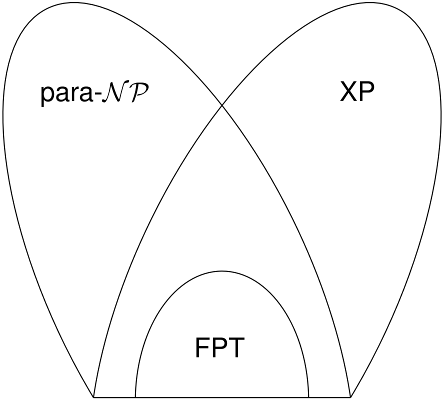
$$|x|^{f(\kappa(x))} + f(\kappa(x))$$

Schritten entscheidet, ob x in Q liegt.

Beobachtung

- Schranke $|x|^{f(\kappa(x))} + f(\kappa(x))$ kann durch berechenbare Funktion $p_{\kappa(x)}(|x|)$ ersetzt werden
- für festes k arbeitet $\mathcal{A}$ in Polynomialzeit

Komplexitätsklassen



Zusammenfassung

- Parametrisierung
  - monoton in $|x|$ wachsende, unbeschr. Parametrisierung $\Rightarrow$ FPT
- Charakterisierung von FPT
  - Kernbildung
  - in $\mathcal{P}$ nach Vorberechnung auf dem Parameter
  - irgendwann in $\mathcal{P}$
  - Zusammenhang mit Approximationsalgorithmen
- Parametrisierte Komplexitätsklassen
  - para-$\mathcal{NP}$
  - XP_{nu}, XP