

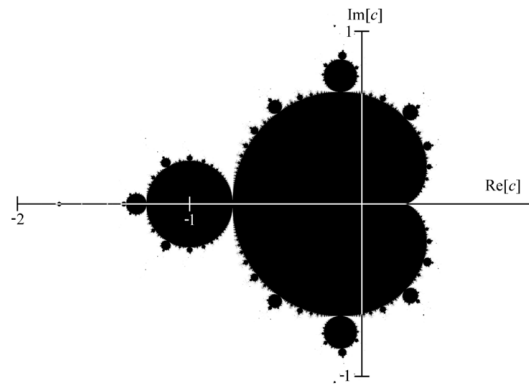
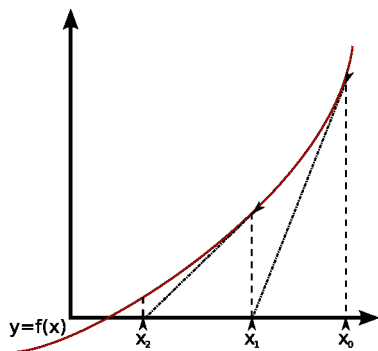
BSS-Rechenmodell Einführung

Seminar Reelle Komplexität

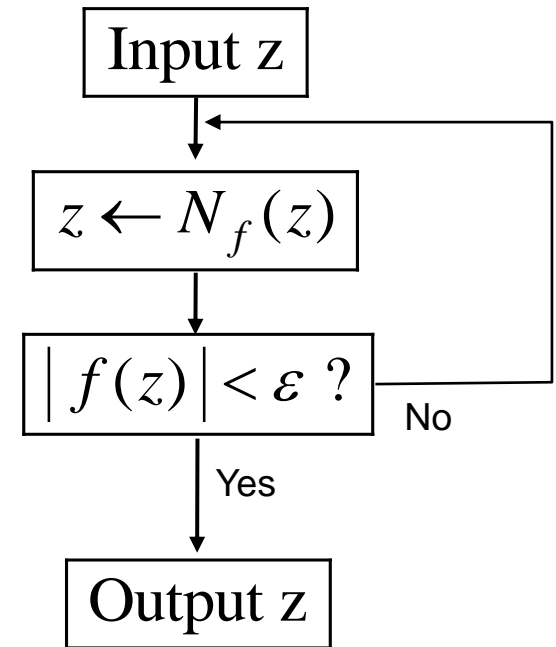
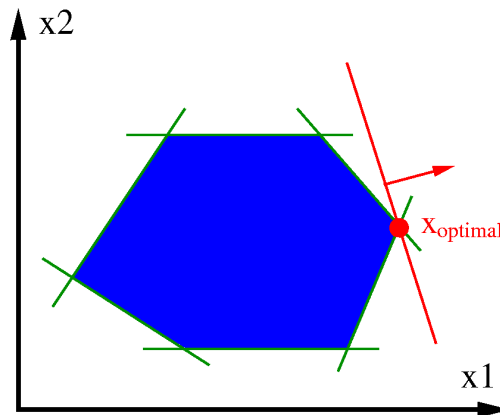


TECHNISCHE
UNIVERSITÄT
DARMSTADT

Yevgen Chebotar



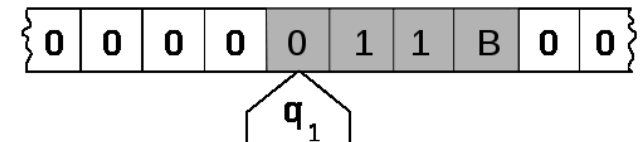
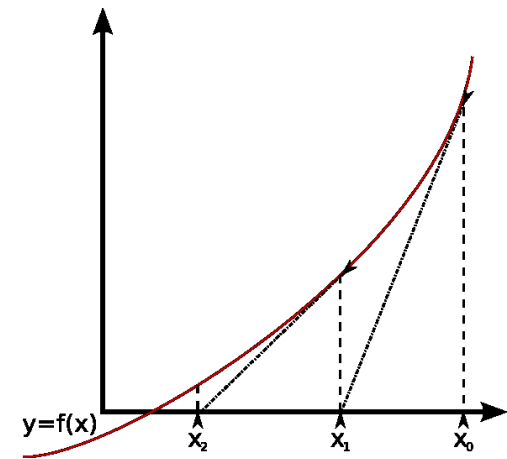
$$\sum b_i x_i = 1$$



1. Motivation
2. Beispielprobleme
 - 2.1 Mandelbrot-Menge
 - 2.2 Newton-Verfahren
 - 2.3 Rucksackproblem
 - 2.4 Hilbert-Nullstellensatz
 - 2.5 Lineare Programmierung
3. Komplexitätstheorie in der Numerik
4. Endlich-dimensionales Modell

1. Motivation

- Wissenschaft: Reelle Algorithmen
 - Newton
 - Euler
 - Gauss etc.
- Turing Maschine
 - Unzulänglich für reelle Berechnungen
 - *Beispiel:*
Gleichheit zwei reellen Zahlen
 $2 == 2,000\dots001$??
→ notwendig für viele Probleme



1. Motivation

- Formale Logik
 - 1/0 Alles-oder-nichts Konzept → starr, schwierig im Umgang
 - Kombinatorik
 - Analysis
 - Sehr erfolgreich in der Mathematik
 - Weitentwickelt
- Neue Theorie für reelle Berechnungen
- Blum-Shub-Smale Maschine
 - Turingtheorie: spezieller Fall

2.1. Mandelbrot-Menge

Betrachten wir:

$$p_c(z) = z^2 + c, \quad c \in \mathbb{C}$$

n-mal angewendet:

$$p_c^n(z) = p_c(\dots p_c(p_c(p_c(z))))$$

Für $z = 0$:

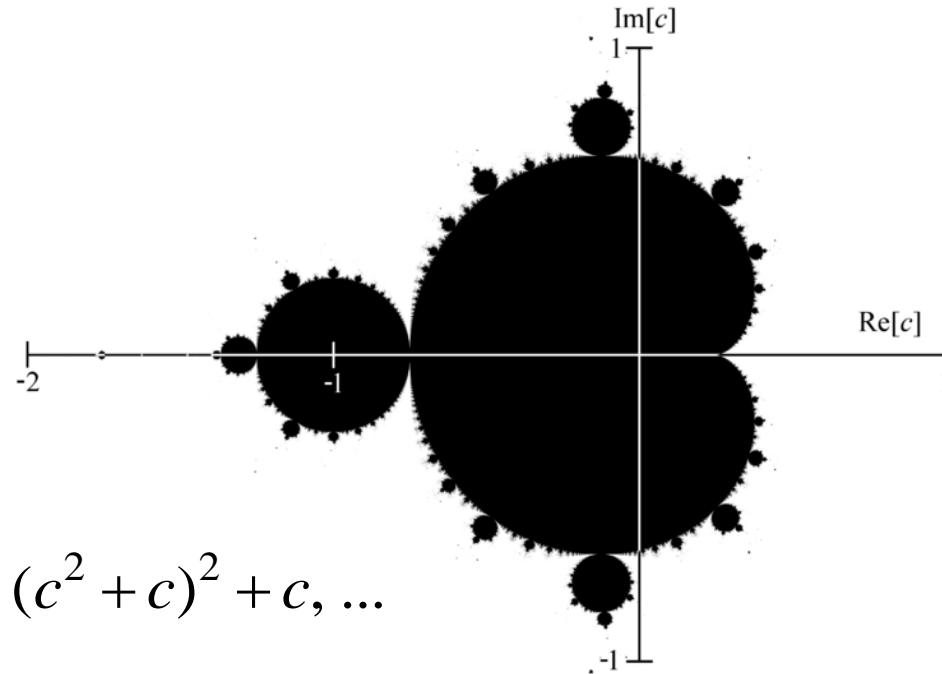
$$p_c(0) = c, \quad p_c^2(0) = c^2 + c, \quad p_c^3(0) = (c^2 + c)^2 + c, \dots$$

Mandelbrot-Menge: $c, c^2 + c, (c^2 + c)^2 + c, \dots$ **beschränkt**

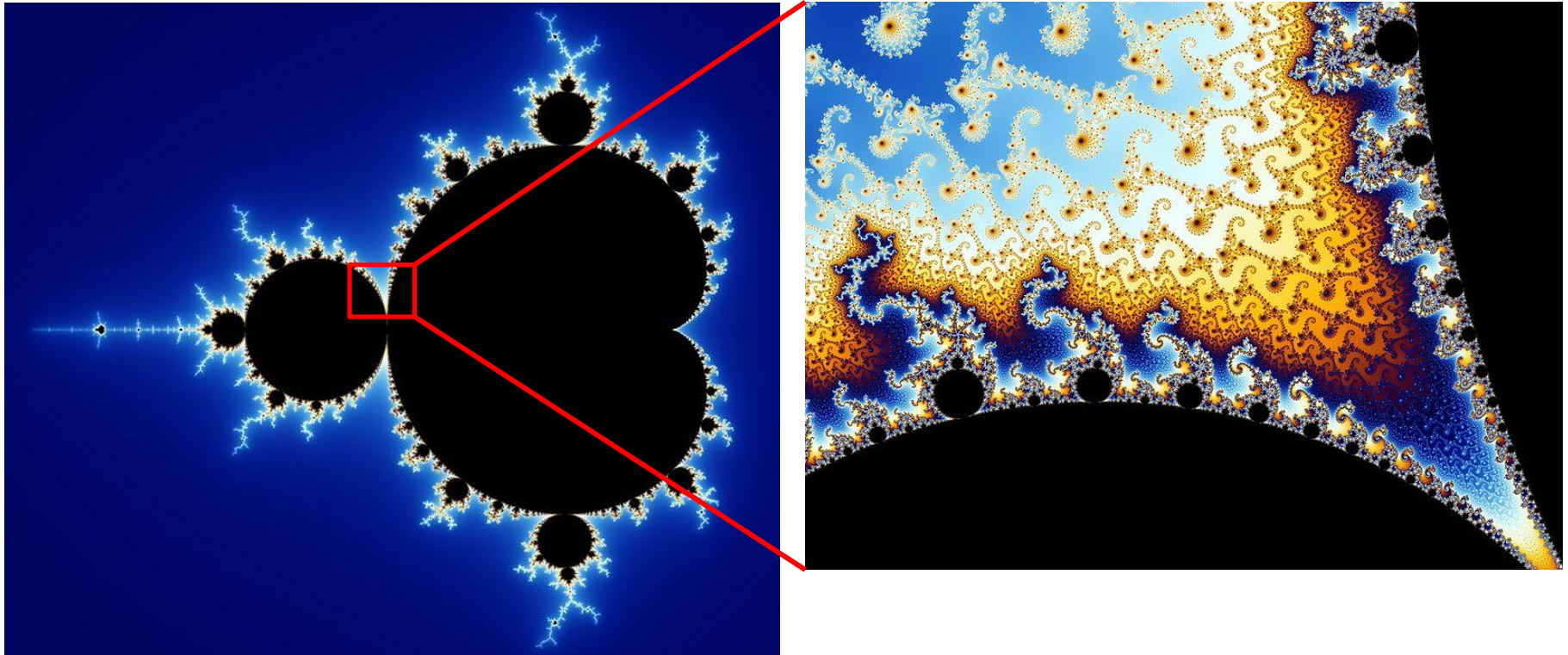
$$\mathbf{M} = \{c \in \mathbb{C} \mid \exists s \in \mathbb{R} \forall n \in \mathbb{N} \mid |p_c^n(0)| \leq s\}$$

Komplement:

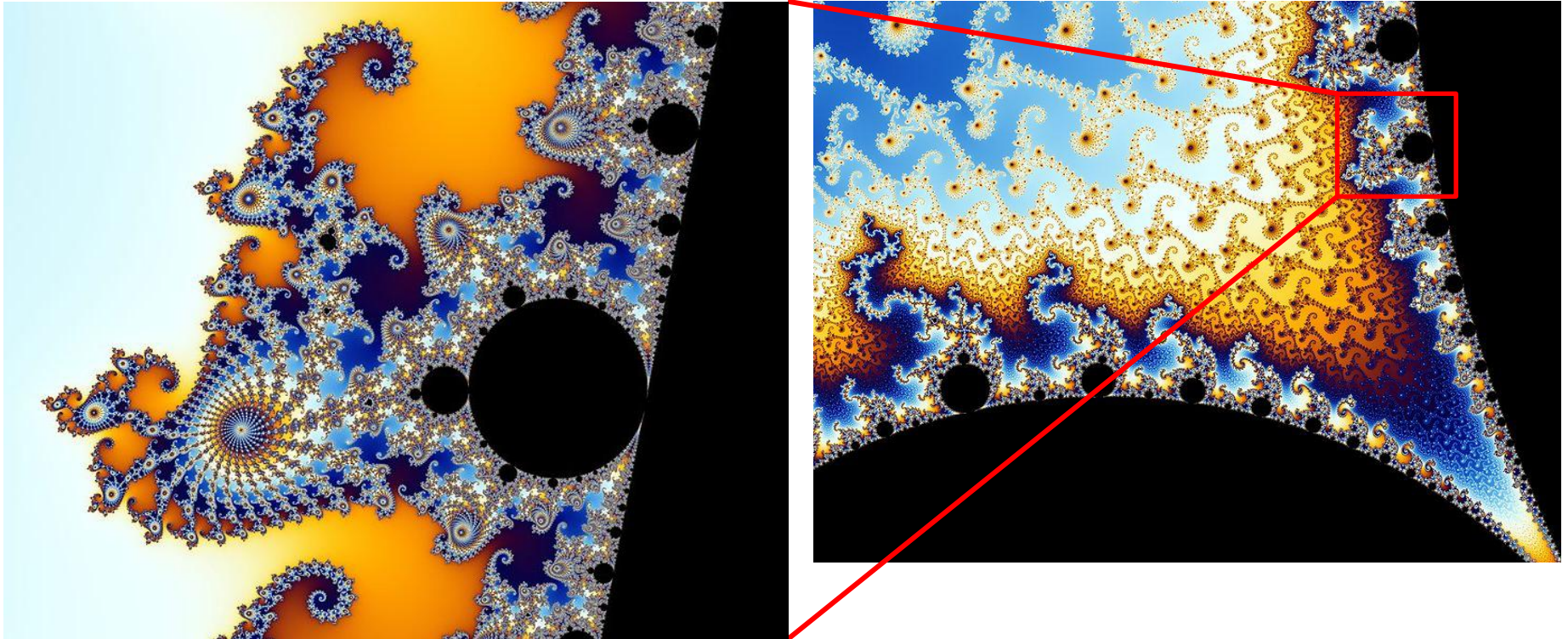
$$\mathbf{M}' = \{c \in \mathbb{C} \mid |p_c^n(0)| \rightarrow \infty \text{ für } n \rightarrow \infty\}$$



2.1. Mandelbrot-Menge



2.1. Mandelbrot-Menge



2.1. Mandelbrot-Menge

Roger Penrose: Ist Mandelbrot-Set entscheidbar?

Probleme:

Klassische Theorie:

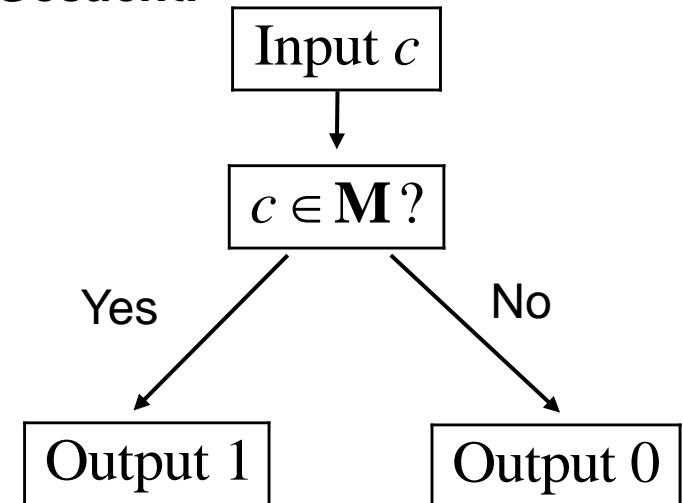
Menge nicht abzählbar

Recursive Analysis:

Gleichheit/Vergleich von reellen Zahlen

Neues Modell: Algebraische Maschine

Gesucht:

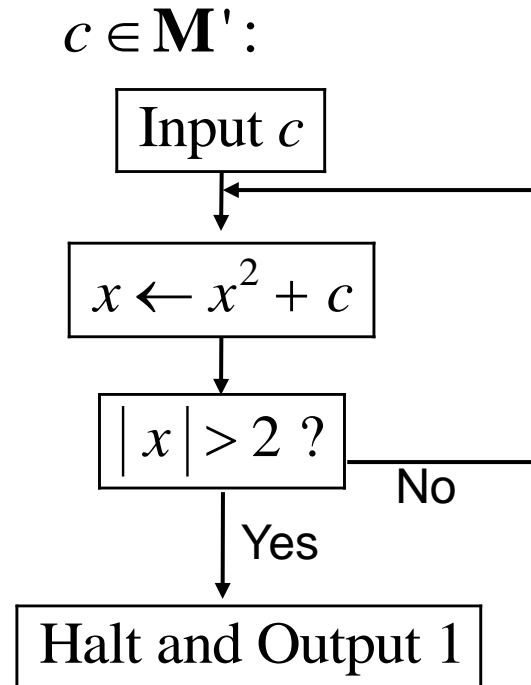


2.1. Mandelbrot-Menge

Ist Mandelbrot-Set entscheidbar?

Eigenschaft: $\exists m \in \mathbb{N} \mid p_c^m(0) > 2$, dann $p_c^n(0) \rightarrow \infty$ für $n \rightarrow \infty$

- \mathbf{M}' semi-entscheidbar
- \mathbf{M} auch?

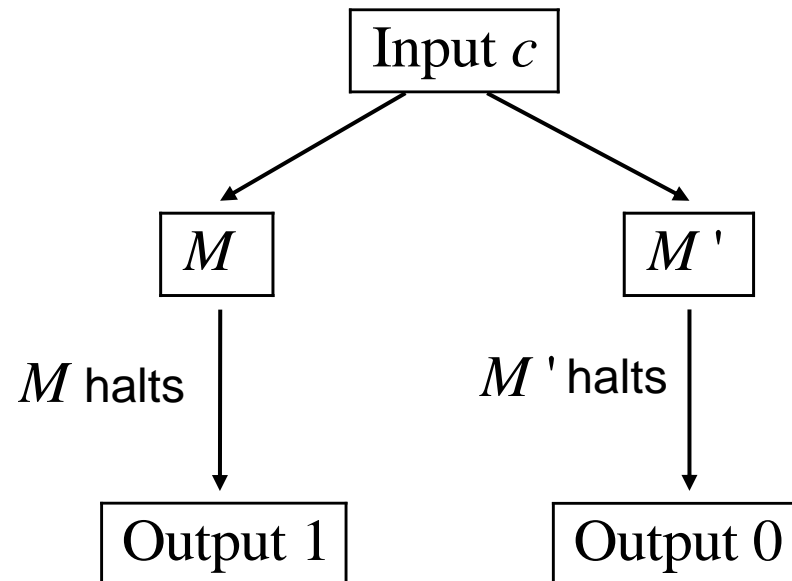


2.1. Mandelbrot-Menge

Ist Mandelbrot-Set entscheidbar?

Eigenschaft: $\exists m \in \mathbb{N} \mid |p_c^m(0)| > 2$, dann $p_c^n(0) \rightarrow \infty$ für $n \rightarrow \infty$

- M' semi-entscheidbar
- M auch?



2.2. Newton-Verfahren

Nullstellen-Suchalgorithmus

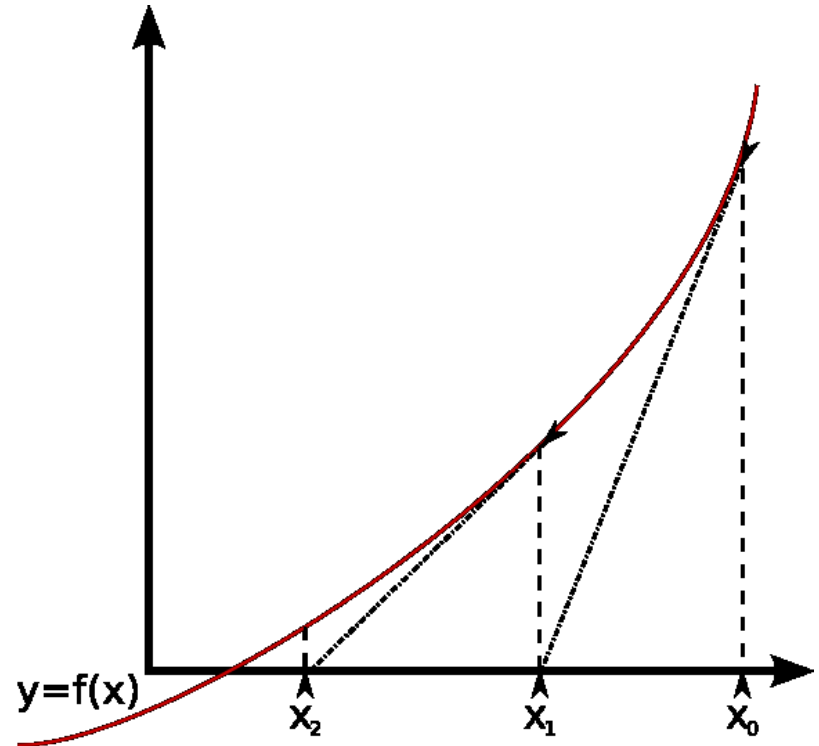
$$N_f : \mathbb{C} \rightarrow \mathbb{C}$$

$$N_f(z) = z - \frac{f(z)}{f'(z)}, \quad f'(z) \neq 0$$

Auswahl von Startpunkt: $z_0 \in \mathbb{C}$

$$z_{k+1} = N_f(z_k) = N_f^{k+1}(z_0)$$

Abbruchregel: $|f(z_k)| < \varepsilon$



2.2. Newton-Verfahren

Nullstellen-Suchalgorithmus

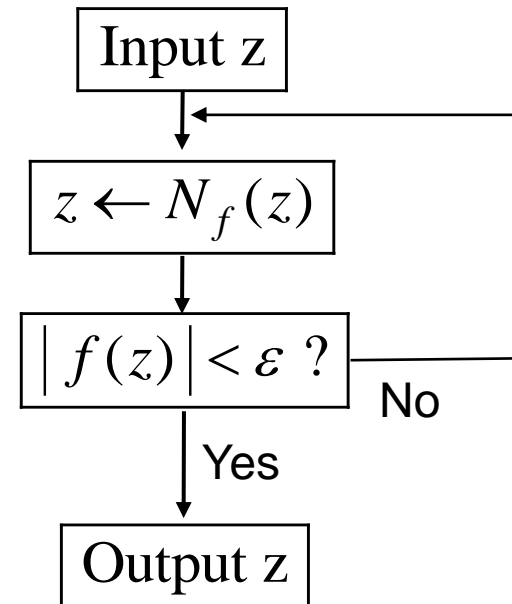
$$N_f : \mathbb{C} \rightarrow \mathbb{C}$$

$$N_f(z) = z - \frac{f(z)}{f'(z)}, \quad f'(z) \neq 0$$

Auswahl von Startpunkt: $z_0 \in \mathbb{C}$

$$z_{k+1} = N_f(z_k) = N_f^{k+1}(z_0)$$

Abbruchregel: $|f(z_k)| < \varepsilon$



2.2. Newton-Verfahren

$$N_f(z) = z - \frac{f(z)}{f'(z)}$$

1) $f(\zeta) = 0$ gdw. $N_f(\zeta) = \zeta$

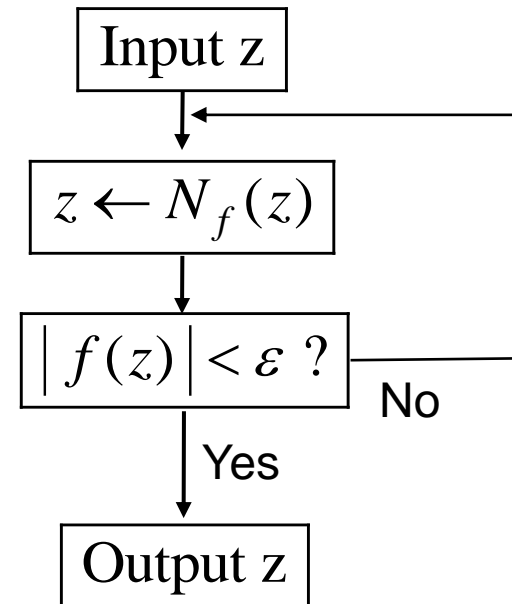
→ Fixpunkt

2) $N_f(\zeta) = \zeta \rightarrow |N'_f(\zeta)| < 1$

→ Fixpunkt ist anziehend

ζ einfache Nullstelle von $f \rightarrow N'_f(\zeta) = 0 \rightarrow \zeta$ super-anziehend

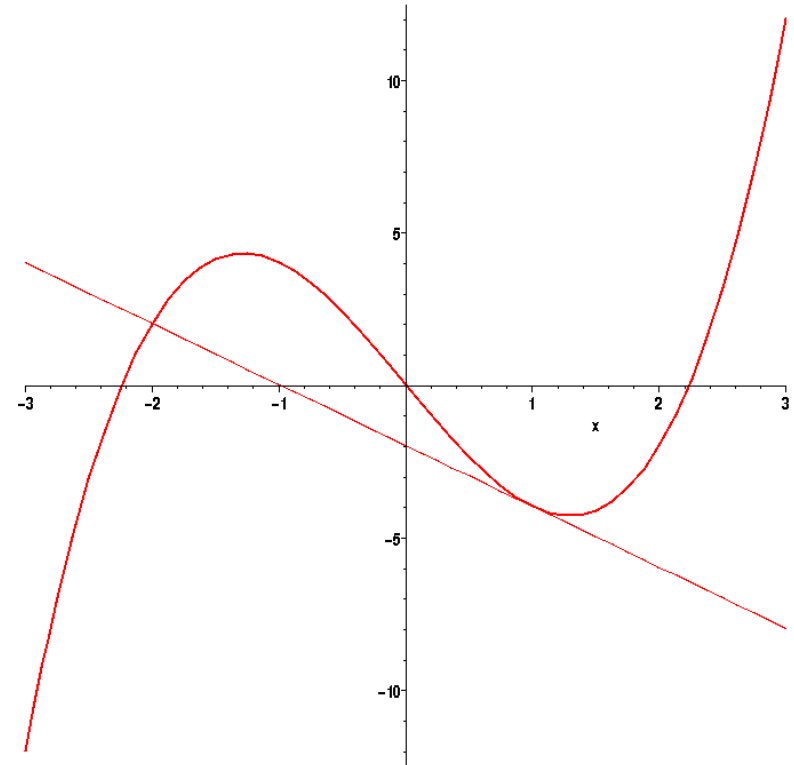
→ **Quadratische Konvergenz**



2.2. Newton-Verfahren

Nicht immer konvergent

- Anziehende periodische Punkte
- Guter Startpunkt:
 N_f konvergiert zur Nullstelle



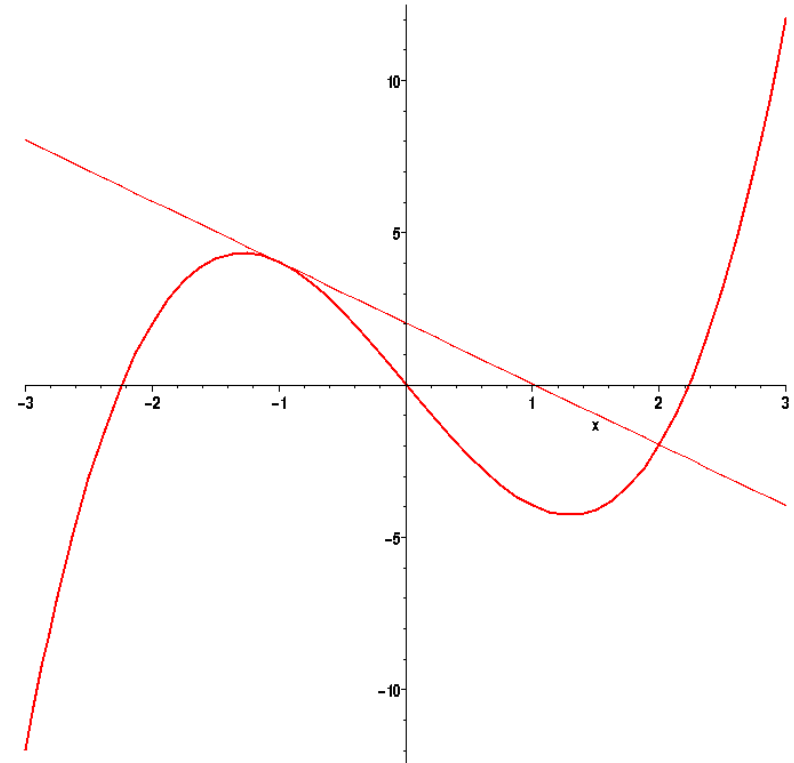
Entscheidungsproblem:

Menge von guten Startpunkten entscheidbar?

2.2. Newton-Verfahren

Nicht immer konvergent

- Anziehende periodische Punkte
- Guter Startpunkt:
 N_f konvergiert zur Nullstelle



Entscheidungsproblem:

Menge von guten Startpunkten entscheidbar?

2.3. Rucksackproblem

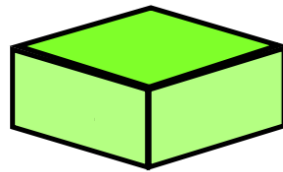
R - Kommutativer Ring mit Einheit (\mathbb{Z} , \mathbb{Q} , \mathbb{R} , \mathbb{C})

$$K_n = \{ x \in R^n \mid \exists b \in \{0, 1\}^n \text{ so dass } \sum b_i x_i = 1 \}$$

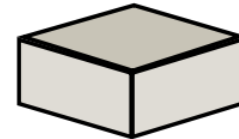
$b = 1101$ - Gültige Belegung $\rightarrow x = (x_0, x_1, x_2, x_3) \in K_n$



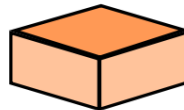
$$\sum b_i x_i = 1$$



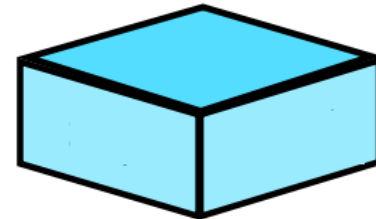
$$x_0 = 0,3$$



$$x_1 = 0,2$$



$$x_2 = 0,15$$



$$x_3 = 0,5$$

2.3. Rucksackproblem

$$K_n = \{ x \in R^n \mid \exists b \in \{0, 1\}^n \text{ so dass } \sum b_i x_i = 1 \}$$

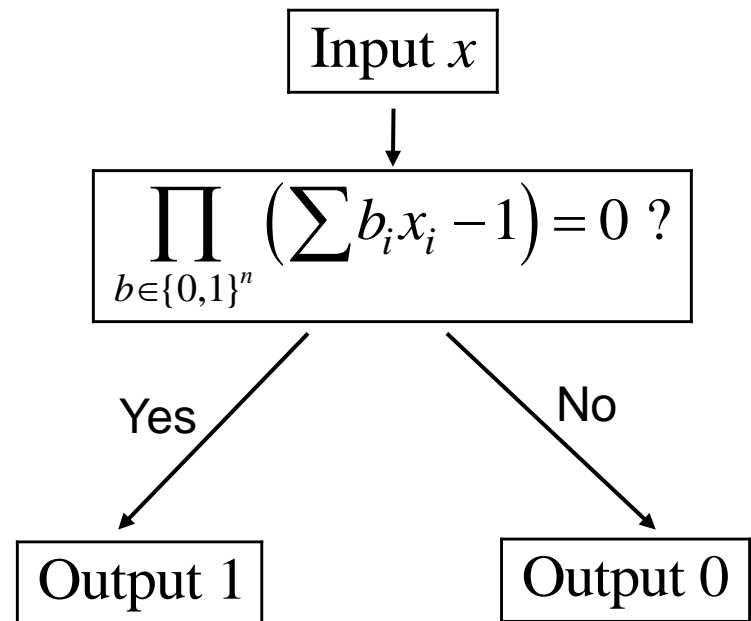
- Durch Aufzählung lösbar (max. $2^n - 1$ Schritte)
- Entscheidbar

Algebraischer Ansatz

$$k_n(x) = \prod_{b \in \{0,1\}^n} (\sum b_i x_i - 1)$$

$$V_{k_n} = \{ x \in R^n \mid k_n(x) = 0 \}$$

$$\rightarrow V_{k_n} = K_n$$



2.4. Hilbert Nullstellensatz

Polynome f_1, \dots, f_k über \mathbb{C} mit n Variablen

Entscheidungsproblem HN/\mathbb{C} :

Gibt es eine gemeinsame Nullstelle?

Gesucht: Algebraischer Algorithmus

- \mathbb{C} nicht geordnet \rightarrow Vergleich auf Gleichheit

2.4. Hilbert Nullstellensatz

Komplexitätsbetrachtung

Eingabe: Koeffizienten von f_i in \mathbb{C}^N

Eingabegröße $S(f)$: Anzahl aller Koeffizienten

$$N = \sum_{i=1}^k \binom{n + d_i}{n}, \quad d = \deg f_i, \quad i = 1, \dots, k$$

Komplexitätsmaß: Anzahl von arithmetischen Operationen $A(f)$

2.4. Hilbert Nullstellensatz

Eigenschaft

Es existieren Polynome g_1, \dots, g_k so dass $\sum_{i=1}^k g_i f_i = 1$

→ f_1, \dots, f_k keine gemeinsame Nullstelle

Suche nach Koeffizienten für g_i : Gauß-Eliminationsverfahren

→ **Exponentiell** in $S(f)$

Vermutung: Es gibt keinen Algorithmus mit $A(f) \leq S(f)^c$

Aber: $\zeta \in \mathbb{C}^n$ gegeben, wir können testen $f_i(\zeta) = 0$ für $i = 1, \dots, k$

→ Testen in polynomieller Zeit

HN/ \mathbb{C} : **NP**-vollständig in \mathbb{C}

Ist **P** \neq **NP** in \mathbb{C} ?

→ Klassische Komplexitätsprobleme mit neuen Werkzeugen

2.5. Lineare Programmierung

Lineare Programmierung:

m Ungleichungen

$$A_i x \geq b_i, \quad i = 1, \dots, m$$

$$A_i x = \sum_{j=1}^n a_{ij} x_j, \quad a_{ij} \in \mathbb{R}, \quad b_i \in \mathbb{R}$$

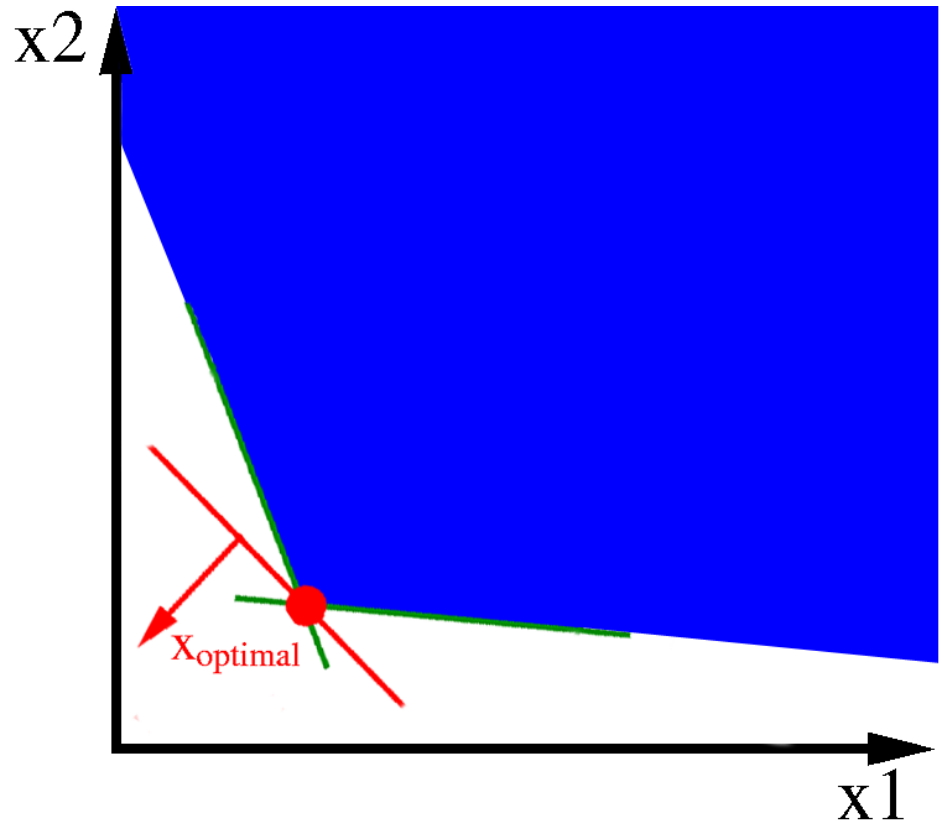
Gibt es einen passenden $x \in \mathbb{R}^n$?

Optimisierung

Minimisiere $c \cdot x$, $c \in \mathbb{R}$

unter Beachtung von $A_i x \geq b_i$

Alternativ: Maximisierung



2.5. Lineare Programmierung

Lineare Programmierung:

m Ungleichungen

$$A_i x \geq b_i, \quad i = 1, \dots, m$$

$$A_i x = \sum_{j=1}^n a_{ij} x_j, \quad a_{ij} \in \mathbb{R}, \quad b_i \in \mathbb{R}$$

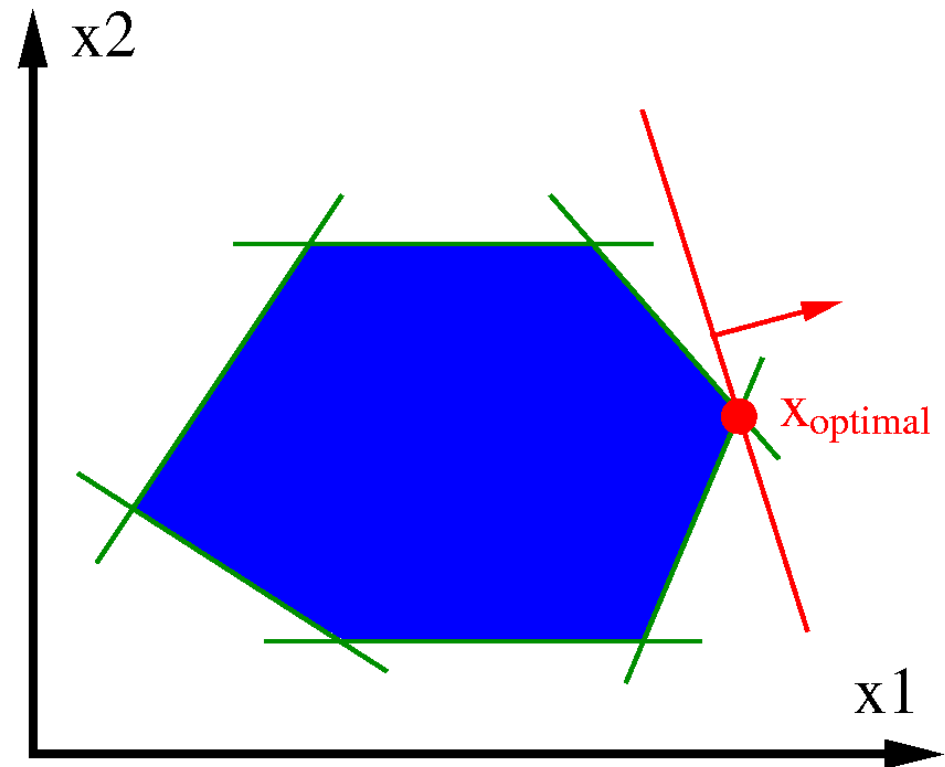
Gibt es einen passenden $x \in \mathbb{R}^n$?

Optimisierung

Minimisiere $c \cdot x$, $c \in \mathbb{R}$

unter Beachtung von $A_i x \geq b_i$

Alternativ: Maximisierung



2.5. Lineare Programmierung

Integer Programmierung:

m Ungleichungen

$$A_i x \geq b_i, \quad i = 1, \dots, m$$

$$A_i x = \sum_{j=1}^n a_{ij} x_j, \quad a_{ij} \in \mathbb{Z}, \quad b_i \in \mathbb{Z}$$

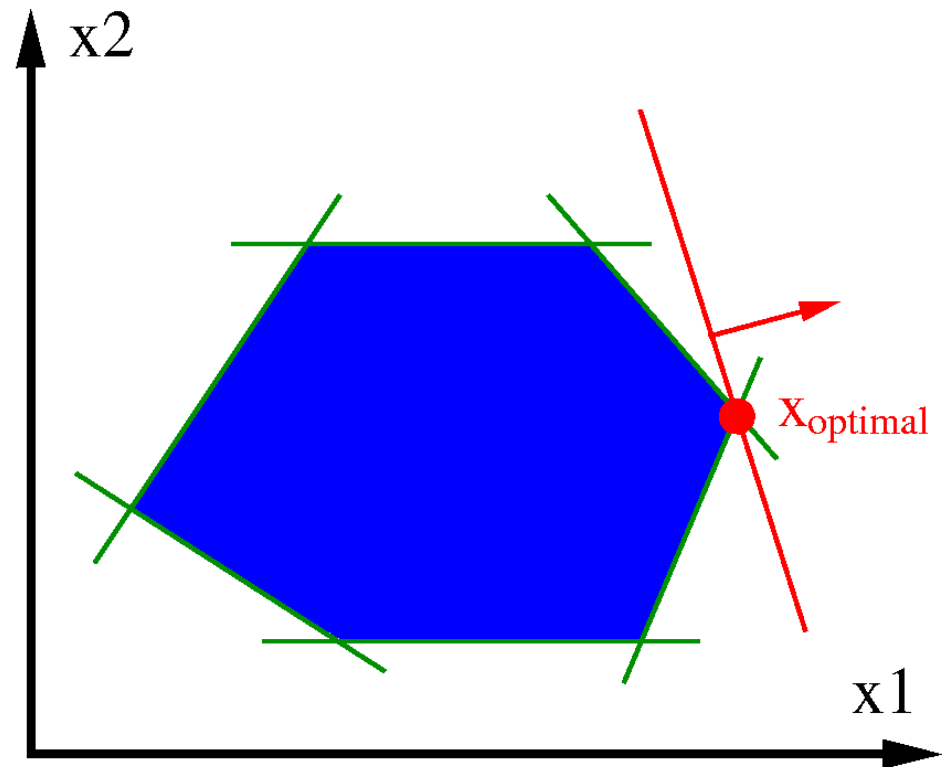
Gibt es einen passenden $x \in \mathbb{Z}^n$?

Optimisierung

Minimisiere $c \cdot x$, $c \in \mathbb{Z}$

unter Beachtung von $A_i x \geq b_i$

Alternativ: Maximisierung



2.5. Lineare Programmierung

Komplexitätsbetrachtung

Eingabe: $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ **Optimisierung:** $c \in \mathbb{R}^n$

Eingabegröße: $S(A, b) = mn + m$ **Optimisierung:** $S(A, b, c) = mn + m + n$

Kosten: Anzahl der algebraischen Operationen $A(A, b)$ bzw. $A(A, b, c)$

Kein Algorithmus bekannt mit

$$A(A, b) \leq S(A, b)^d$$

$$A(A, b, c) \leq S(A, b, c)^d$$

2.5. Lineare Programmierung

Komplexitätsbetrachtung – Integer Programmierung

Eingabe: $A \in \mathbb{Z}^{m \times n}$, $b \in \mathbb{Z}^m$

Binäre Länge von Integer(height): $ht(x) = \lceil \log(|x| + 1) \rceil$

Eingabegröße $S_{ht}(A, b)$: $S(A, b)$ mal maximale $ht(x)$ in A und b

Kosten $C_{ht}(A, b)$: $A(A, b)$ mal maximal in Berechnung auftretende $ht(x)$

Klassisches Modell: **NP** - vollständig

3. Komplexitätstheorie in der Numerik

Effizient lösbar in klassischer Theorie: $T(x) \leq c \cdot (\text{size}(x))^q$

Viele numerische Probleme:

- Begrenzte **Genauigkeit** ε

Effizient lösbar für $\varepsilon < 1$: $T(\varepsilon, x) \leq (|\log \varepsilon| + \text{size}(x))^q$

→ Je kleiner ε desto länger kann die Berechnung dauern

Konditionszahl $\mu(x)$:

Abhängigkeit der Lösung von der Störung der Eingabe

Effizient lösbar: $T(\varepsilon, x) \leq (|\log \varepsilon| + \log \mu(x) + \text{size}(x))^q$

→ Je größer $\mu(x)$, desto länger kann die Berechnung dauern

4. Endlich-dimensionales Modell

Newton Maschine

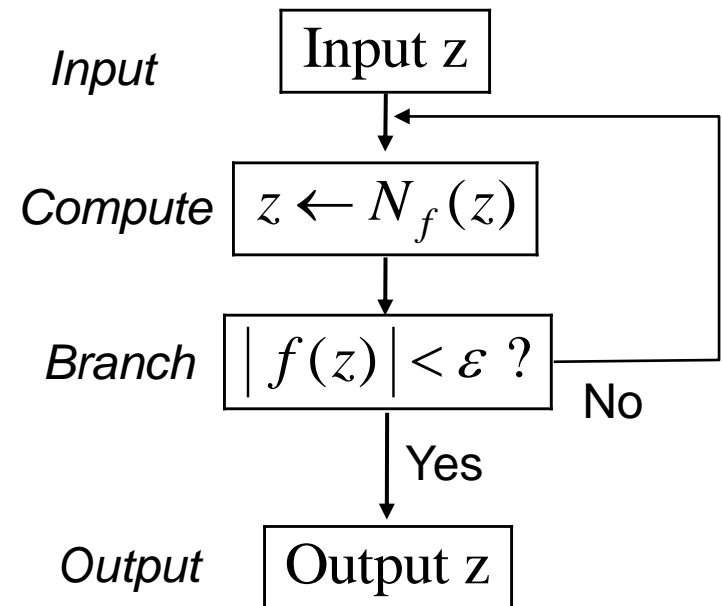
time-T Haltemenge: $\Omega_T = \{z \in \mathbb{C} \mid |f(N_f^{T'}(z))| < \varepsilon \text{ für } T', 0 \leq T' \leq T\}$

Haltemenge: $\Omega = \bigcup_{0 < T < \infty} \Omega_T$

Input-Output Abbildung $\Phi : \Omega \rightarrow \mathbb{C}$

Für einen Startpunkt $z \in \Omega_T$:

- $\Phi(z) = N_f^{T'}(z)$
- T' - kleinste $T > 0$ so dass $z \in \Omega_T$



4. Endlich-dimensionales Modell



Newton Maschine

$$N_f : \mathbb{C} \rightarrow \mathbb{C} \quad N_f(z) = z - \frac{f(z)}{f'(z)}, \quad f'(z) \neq 0$$

Betrachten wir \mathbb{C} als \mathbb{R}^2 :

$$g = (g_1, g_2) : \mathbb{R}^2 \rightarrow \mathbb{R}^2$$

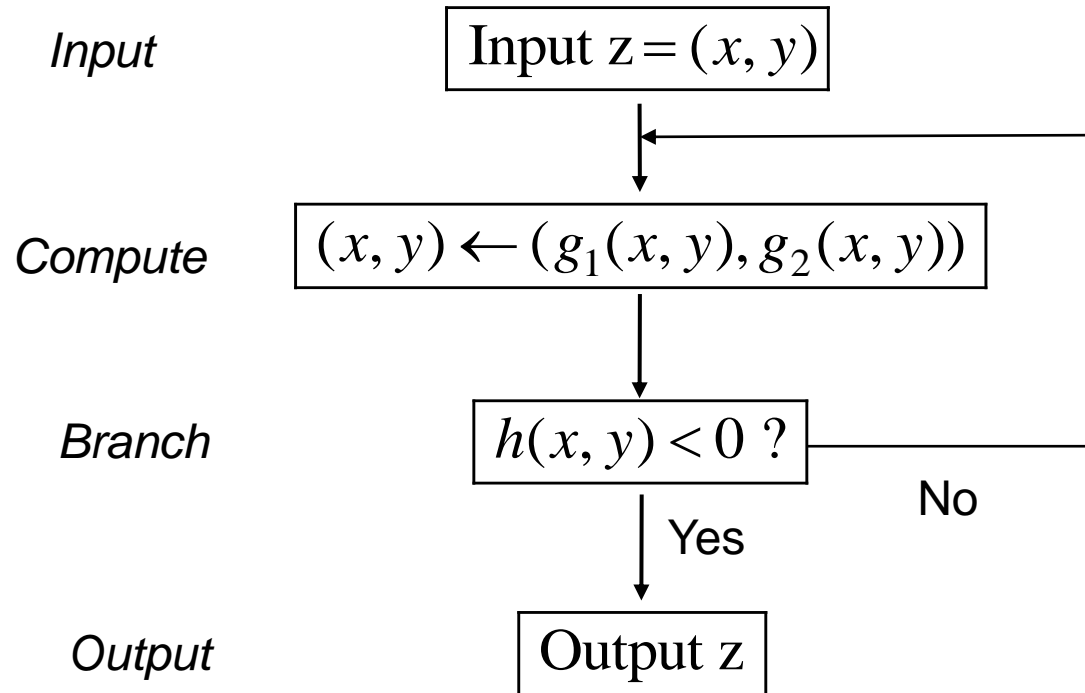
$$g_1(x, y) = \operatorname{Re} N_f(x + iy), \quad g_2(x, y) = \operatorname{Im} N_f(x + iy)$$

Abbruchregel: $|f(z)|^2 < \varepsilon^2$ oder $h(x, y) < 0$

$$h = (\operatorname{Re} f)^2 + (\operatorname{Im} f)^2 - \varepsilon^2 : \mathbb{R}^2 \rightarrow \mathbb{R}$$

4. Endlich-dimensionales Modell

Endlich-dimensionale Maschine über Ring



4. Endlich-dimensionales Modell



Definition

Sei R ein geordneter kommutativer Ring (oder Körper)

Endlich-dimensionale Maschine M über R ist ein gerichteter Graph

\mathcal{I}_M : Eingaberaum von der Form R^n

\mathcal{S}_M : Zustandsraum von der Form R^m

\mathcal{O}_M : Ausgaberaum von der Form R^l

4 Arten von Knoten

4. Endlich-dimensionales Modell



Definition

\mathcal{I}_M : Eingaberaum (R^n), \mathcal{S}_M : Zustandsraum (R^m), \mathcal{O}_M : Ausgaberaum (R^l)

Knoten:

- 1) *Input*: Lineare Abbildung $I : \mathcal{I}_M \rightarrow \mathcal{S}_M$ Nächster Knoten: β_1
- 2) *Computation*: Polynomiale(Rationale) Abbildung $g_\eta : \mathcal{S}_M \rightarrow \mathcal{S}_M$
Nächster Knoten: β_η
- 3) *Branch*: Polynomiale Funktion $h_\eta : \mathcal{S}_M \rightarrow R$ Nächster Knoten:
 - $h_\eta(z) \geq 0$ (geordnet), $h_\eta(z) = 0$ (nicht geordnet) \rightarrow Yes-Kante $\rightarrow \beta_\eta^+$
 - $h_\eta(z) < 0$ (geordnet), $h_\eta(z) \neq 0$ (nicht geordnet) \rightarrow No-Kante $\rightarrow \beta_\eta^-$
- 4) *Output*: Lineare Abbildung $O_\eta : \mathcal{S}_M \rightarrow \mathcal{O}_M$ Nächster Knoten: kein

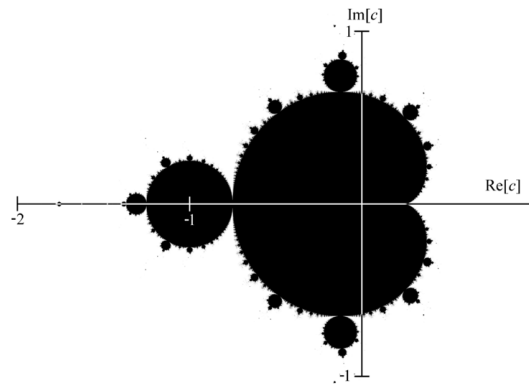
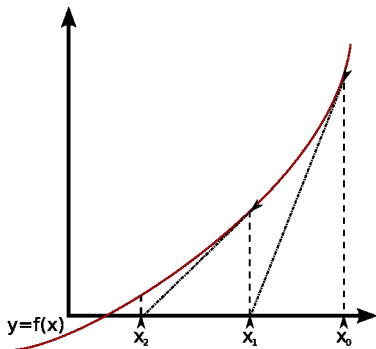
BSS-Rechenmodell Einführung

Seminar Reelle Komplexität

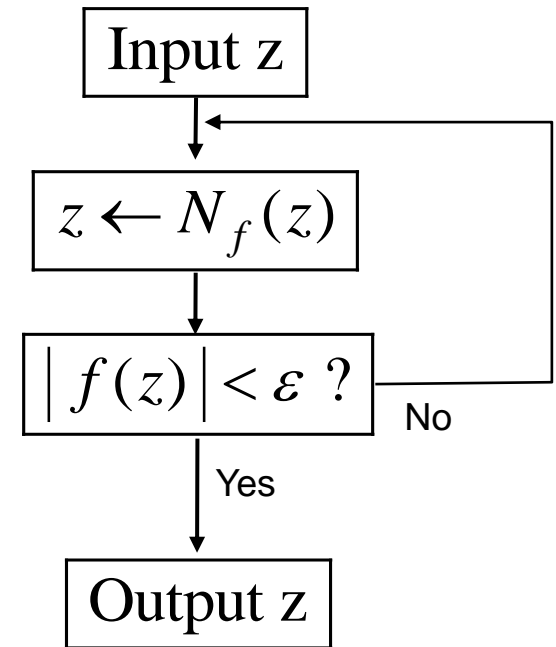
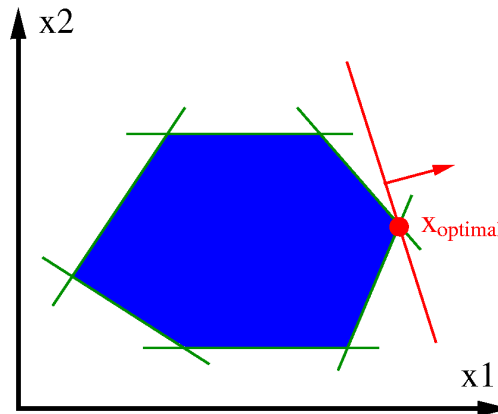


TECHNISCHE
UNIVERSITÄT
DARMSTADT

Yevgen Chebotar



$$\sum b_i x_i = 1$$



Bildquellen

http://sl.wikipedia.org/wiki/Slika:Newton-Raphson_method.png

http://en.wikipedia.org/wiki/File:Turing_machine_2b.svg

http://en.wikipedia.org/wiki/File:Mandelset_hires.png

http://en.wikipedia.org/wiki/File:Mandel_zoom_00_mandelbrot_set.jpg

http://en.wikipedia.org/wiki/File:Mandel_zoom_09_satellite_head_and_shoulder.jpg

http://en.wikipedia.org/wiki/File:Mandel_zoom_03_seehorse.jpg

<http://en.wikipedia.org/wiki/File:Knapsack.svg>

<http://upload.wikimedia.org/wikipedia/commons/f/fc/Linopt-feasible-region2.png>