

Lösung positiver polynomieller Gleichungssysteme

Stefan Kiefer

Technische Universität München

19. Oktober 2009

Positive Polynomielle Gleichungssysteme

Wir betrachten Gleichungssysteme der Form

$$\mathbf{x} = \mathbf{f}(\mathbf{x}),$$

dabei ist

- \mathbf{x} ein Vektor aus n Variablen,
- $\mathbf{f}(\mathbf{x})$ ein Vektor aus Polynomen mit **positiven Koeffizienten**.

Beispiel ($n = 2$)

$$\begin{pmatrix} X \\ Y \end{pmatrix} = \begin{pmatrix} 1.2 \cdot X \cdot Y + 0.3 \cdot Y^2 \\ 0.7 \cdot X^2 + 0.2 \end{pmatrix}$$

Wir nennen solche Gleichungssysteme
positive polynomielle Gleichungssysteme.

Im Vortrag angenommen: es gibt eine nichtnegative Lösung.

Dann gibt es eine kleinste, genannt $\mu\mathbf{f}$.

Wir möchten $\mu\mathbf{f}$ berechnen oder approximieren.

Probabilistische Rekursive Programme

Einfaches Modell für probabilistische rekursive Programme:

$$X \xrightarrow{0.4} \text{call } Y$$

$$Y \xrightarrow{0.3} \text{call } X$$

$$Z \xrightarrow{0.3} \text{call } X$$

$$X \xrightarrow{0.6} \text{return}$$

$$Y \xrightarrow{0.4} \text{call } Z$$

$$Z \xrightarrow{0.7} \text{return}$$

$$Y \xrightarrow{0.3} \text{return}$$



Probabilistische Rekursive Programme

Einfaches Modell für probabilistische rekursive Programme:

$X \xrightarrow{0.4} \text{call } Y$

$X \xrightarrow{0.6} \text{return}$

$Y \xrightarrow{0.3} \text{call } X$

$Y \xrightarrow{0.4} \text{call } Z$

$Y \xrightarrow{0.3} \text{return}$

$Z \xrightarrow{0.3} \text{call } X$

$Z \xrightarrow{0.7} \text{return}$



Probabilistische Rekursive Programme

Einfaches Modell für probabilistische rekursive Programme:

$X \xrightarrow{0.4} \text{call } Y$

$Y \xrightarrow{0.3} \text{call } X$

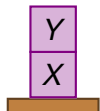
$Z \xrightarrow{0.3} \text{call } X$

$X \xrightarrow{0.6} \text{return}$

$Y \xrightarrow{0.4} \text{call } Z$

$Z \xrightarrow{0.7} \text{return}$

$Y \xrightarrow{0.3} \text{return}$



Probabilistische Rekursive Programme

Einfaches Modell für probabilistische rekursive Programme:

$X \xrightarrow{0.4} \text{call } Y$

$Y \xrightarrow{0.3} \text{call } X$

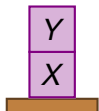
$Z \xrightarrow{0.3} \text{call } X$

$X \xrightarrow{0.6} \text{return}$

$Y \xrightarrow{0.4} \text{call } Z$

$Z \xrightarrow{0.7} \text{return}$

$Y \xrightarrow{0.3} \text{return}$



Probabilistische Rekursive Programme

Einfaches Modell für probabilistische rekursive Programme:

$X \xrightarrow{0.4} \text{call } Y$

$Y \xrightarrow{0.3} \text{call } X$

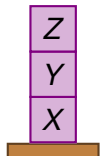
$Z \xrightarrow{0.3} \text{call } X$

$X \xrightarrow{0.6} \text{return}$

$Y \xrightarrow{0.4} \text{call } Z$

$Z \xrightarrow{0.7} \text{return}$

$Y \xrightarrow{0.3} \text{return}$



Probabilistische Rekursive Programme

Einfaches Modell für probabilistische rekursive Programme:

$X \xrightarrow{0.4} \text{call } Y$

$Y \xrightarrow{0.3} \text{call } X$

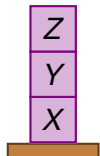
$Z \xrightarrow{0.3} \text{call } X$

$X \xrightarrow{0.6} \text{return}$

$Y \xrightarrow{0.4} \text{call } Z$

$Z \xrightarrow{0.7} \text{return}$

$Y \xrightarrow{0.3} \text{return}$



Probabilistische Rekursive Programme

Einfaches Modell für probabilistische rekursive Programme:

$X \xrightarrow{0.4} \text{call } Y$

$Y \xrightarrow{0.3} \text{call } X$

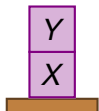
$Z \xrightarrow{0.3} \text{call } X$

$X \xrightarrow{0.6} \text{return}$

$Y \xrightarrow{0.4} \text{call } Z$

$Z \xrightarrow{0.7} \text{return}$

$Y \xrightarrow{0.3} \text{return}$



Probabilistische Rekursive Programme

Einfaches Modell für probabilistische rekursive Programme:

$X \xrightarrow{0.4} \text{call } Y$

$Y \xrightarrow{0.3} \text{call } X$

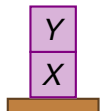
$Z \xrightarrow{0.3} \text{call } X$

$X \xrightarrow{0.6} \text{return}$

$Y \xrightarrow{0.4} \text{call } Z$

$Z \xrightarrow{0.7} \text{return}$

$Y \xrightarrow{0.3} \text{return}$



Probabilistische Rekursive Programme

Einfaches Modell für probabilistische rekursive Programme:

$X \xrightarrow{0.4} \text{call } Y$

$Y \xrightarrow{0.3} \text{call } X$

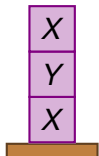
$Z \xrightarrow{0.3} \text{call } X$

$X \xrightarrow{0.6} \text{return}$

$Y \xrightarrow{0.4} \text{call } Z$

$Z \xrightarrow{0.7} \text{return}$

$Y \xrightarrow{0.3} \text{return}$



Probabilistische Rekursive Programme

Einfaches Modell für probabilistische rekursive Programme:

$$X \xrightarrow{0.4} \text{call } Y$$

$$Y \xrightarrow{0.3} \text{call } X$$

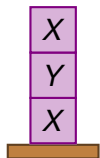
$$Z \xrightarrow{0.3} \text{call } X$$

$$X \xrightarrow{0.6} \text{return}$$

$$Y \xrightarrow{0.4} \text{call } Z$$

$$Z \xrightarrow{0.7} \text{return}$$

$$Y \xrightarrow{0.3} \text{return}$$



Terminierungs-Wahrscheinlichkeit?

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0.4YX + 0.6 \\ 0.3XY + 0.4ZY + 0.3 \\ 0.3XZ + 0.7 \end{pmatrix}$$

Probabilistische Rekursive Programme

Einfaches Modell für probabilistische rekursive Programme:

$$X \xrightarrow{0.4} \text{call } Y$$

$$Y \xrightarrow{0.3} \text{call } X$$

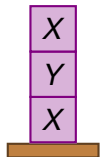
$$Z \xrightarrow{0.3} \text{call } X$$

$$X \xrightarrow{0.6} \text{return}$$

$$Y \xrightarrow{0.4} \text{call } Z$$

$$Z \xrightarrow{0.7} \text{return}$$

$$Y \xrightarrow{0.3} \text{return}$$



Terminierungs-Wahrscheinlichkeit?

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0.4YX + 0.6 \\ 0.3XY + 0.4ZY + 0.3 \\ 0.3XZ + 0.7 \end{pmatrix}$$

Die Terminierungs-W'keit ist die **kleinste Lösung**.

Nicht unbedingt (1, 1, 1)!

- Verifikation probabilistischer rekursiver Programme
[Esparza, Kučera, Mayr, LICS'04],
[Etesami, Yannakakis, STACS'05]
(Terminierungsw'keiten und mehr)
- Analyse stochastischer kontext-freier Grammatiken
 - um die Sekundärstruktur von RNA zu beschreiben
[Knudsen, Hein, Bioinformatics'99]
 - in der Verarbeitung natürlicher Sprache
[Wojtczak, Etesami, TACAS'07]
- Stochastische Analyse von Surf-Verhalten
[Fagin et al., STOC'00]
- Reputationssysteme
[Bouajjani et al., TACAS'08]

⇒ All das führt auf das Bestimmen der kleinsten Lösung von $\mathbf{x} = \mathbf{f}(\mathbf{x})$.

Approximation ist nötig.

Im Allgemeinen kann μf nicht exakt berechnet werden.

Beispiel

Sei $f(X) = \frac{1}{6}X^6 + \frac{1}{2}X^5 + \frac{1}{3}$.

μf kann nicht mit Wurzeln ausgedrückt werden.

⇒ Approximation nötig

$(0.3357037075 < \mu f < 0.3357037076)$

Ein einfaches Approximationsverfahren

Proposition (Fixpunktsatz von Kleene)

Die Kleene-Folge $\mathbf{0}, \mathbf{f}(\mathbf{0}), \mathbf{f}(\mathbf{f}(\mathbf{0})), \dots$ konvergiert zu $\mu \mathbf{f}$.

Proposition (Fixpunktsatz von Kleene)

Die Kleene-Folge $0, f(0), f(f(0)), \dots$ konvergiert zu μf .

Aber sehr langsam!

Beispiel

Sei $f(X) = 0.5X^2 + 0.5$.

Dann $\mu f = 1$, aber $f^k(0) \leq 1 - \frac{1}{k+1}$,

\Rightarrow Man braucht **2^i Iterationen**,
um μf auf **i Bits** genau zu berechnen.

Ein einfaches Approximationsverfahren

Proposition (Fixpunktsatz von Kleene)

Die Kleene-Folge $\mathbf{0}, f(\mathbf{0}), f(f(\mathbf{0})), \dots$ konvergiert zu μf .

Aber sehr langsam!

Beispiel

Sei $f(X) = 0.5X^2 + 0.5$.

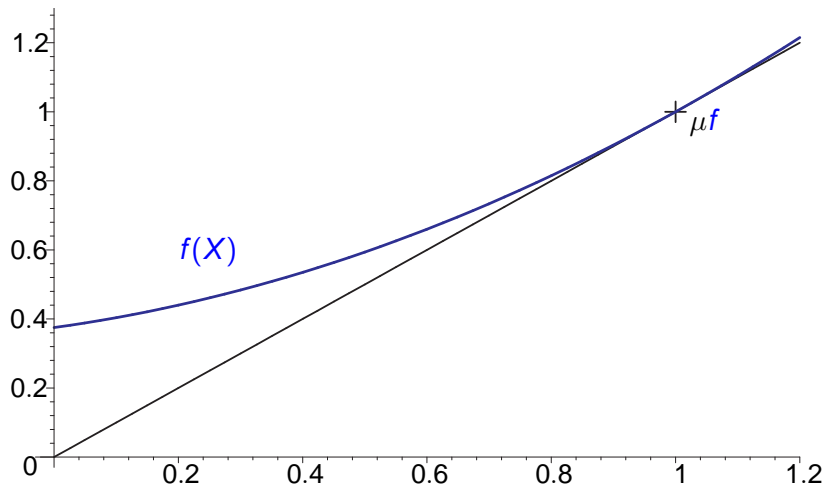
Dann $\mu f = 1$, aber $f^k(0) \leq 1 - \frac{1}{k+1}$,

\Rightarrow Man braucht 2^i Iterationen,
um μf auf i Bits genau zu berechnen.

Lösung: Newton-Verfahren

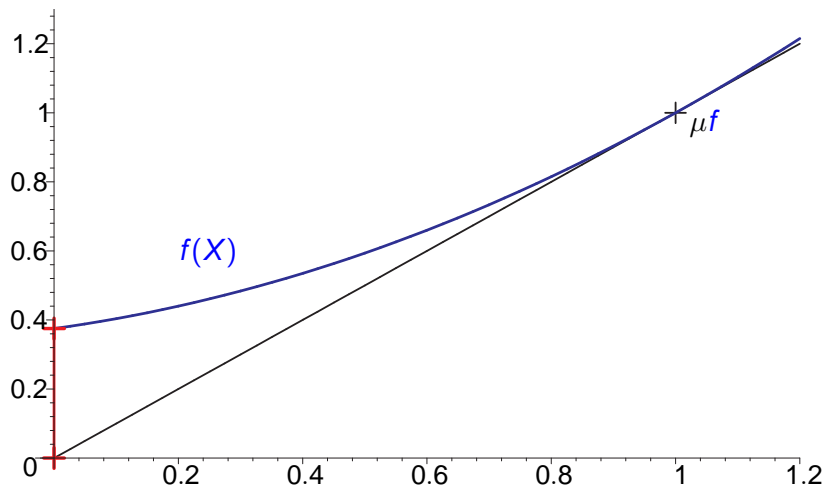
Kleene-Iteration (1-dimensional)

Betrachte $f(X) = 3/8 \cdot X \cdot X + 1/4 \cdot X + 3/8$



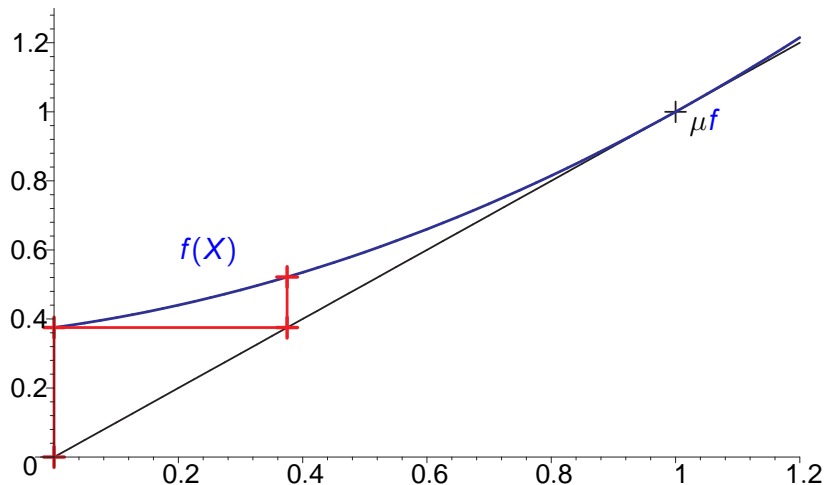
Kleene-Iteration (1-dimensional)

Betrachte $f(X) = 3/8 \cdot X \cdot X + 1/4 \cdot X + 3/8$



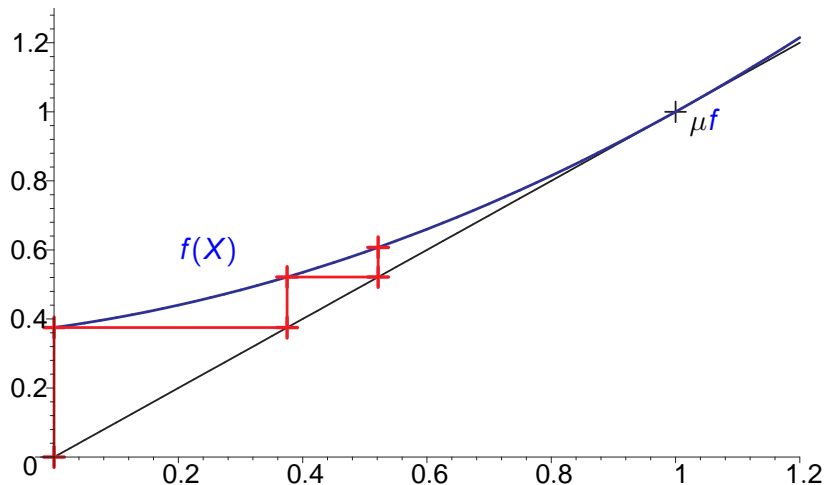
Kleene-Iteration (1-dimensional)

Betrachte $f(X) = 3/8 \cdot X \cdot X + 1/4 \cdot X + 3/8$



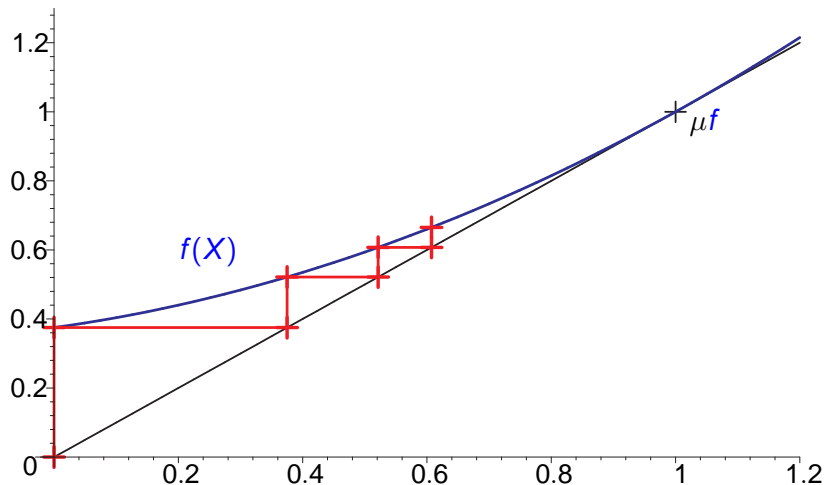
Kleene-Iteration (1-dimensional)

Betrachte $f(X) = 3/8 \cdot X \cdot X + 1/4 \cdot X + 3/8$



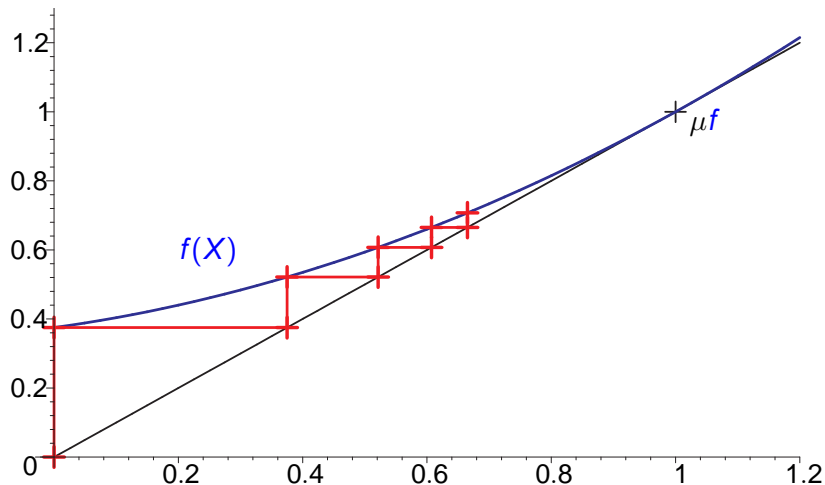
Kleene-Iteration (1-dimensional)

Betrachte $f(X) = 3/8 \cdot X \cdot X + 1/4 \cdot X + 3/8$



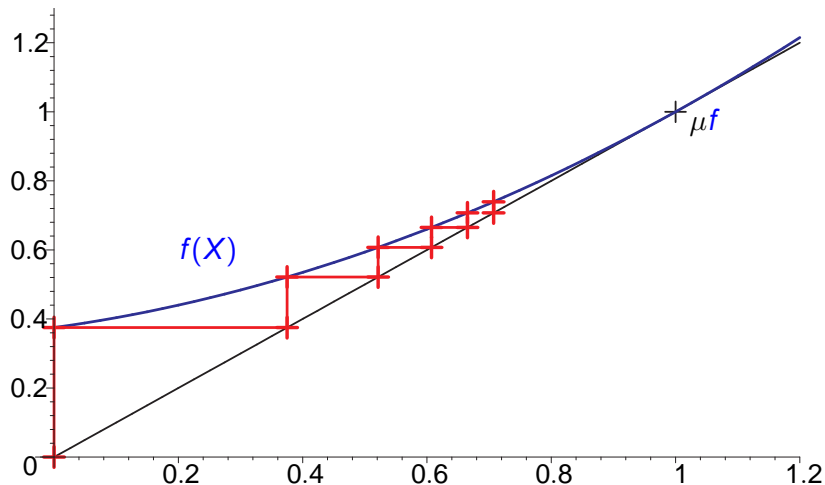
Kleene-Iteration (1-dimensional)

Betrachte $f(X) = 3/8 \cdot X \cdot X + 1/4 \cdot X + 3/8$



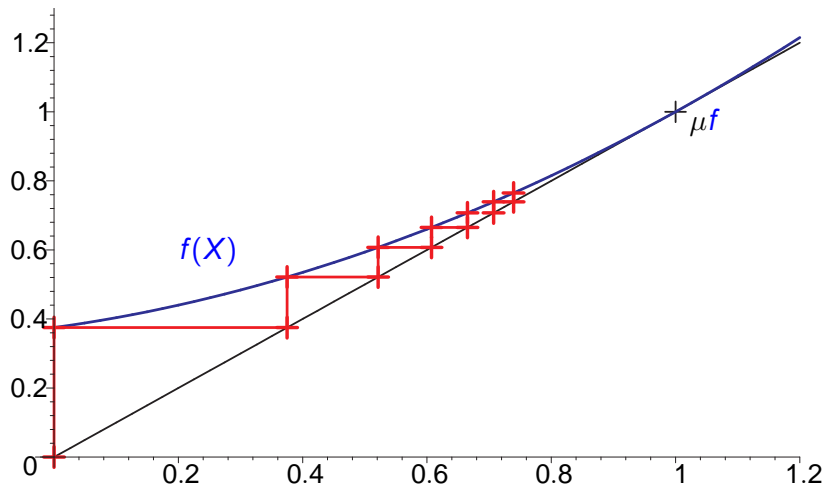
Kleene-Iteration (1-dimensional)

Betrachte $f(X) = 3/8 \cdot X \cdot X + 1/4 \cdot X + 3/8$



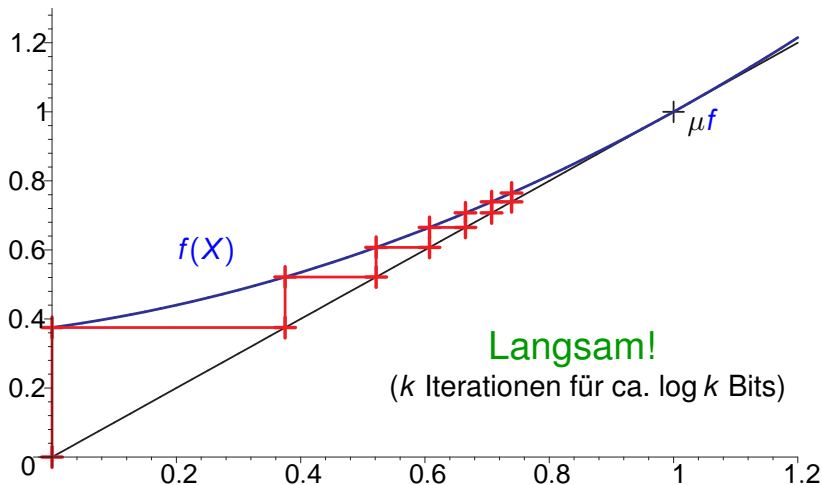
Kleene-Iteration (1-dimensional)

Betrachte $f(X) = 3/8 \cdot X \cdot X + 1/4 \cdot X + 3/8$



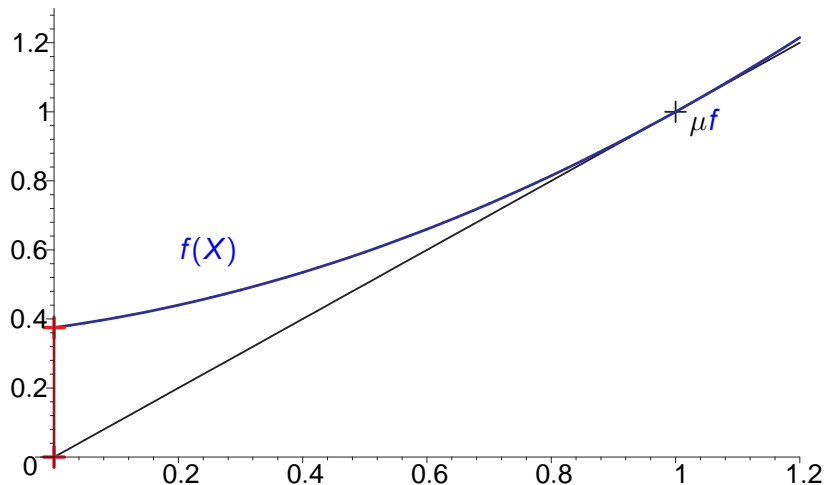
Kleene-Iteration (1-dimensional)

Betrachte $f(X) = 3/8 \cdot X \cdot X + 1/4 \cdot X + 3/8$



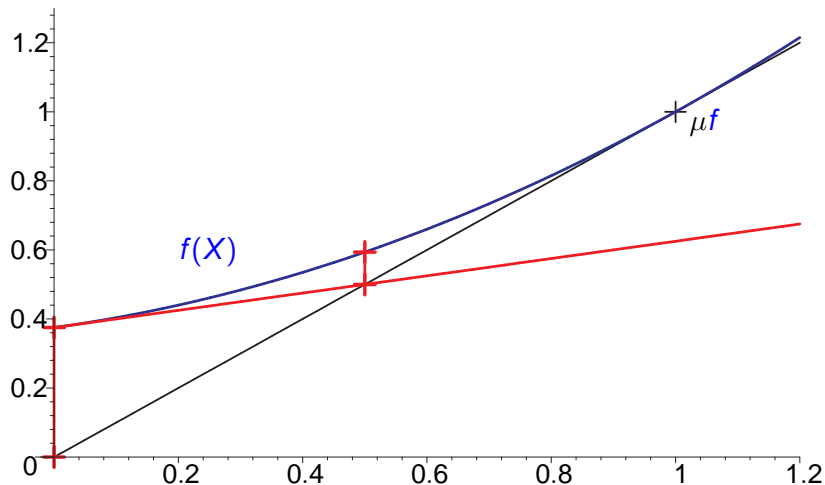
Newton-Verfahren (1-dimensional)

Betrachte $f(X) = 3/8 \cdot X \cdot X + 1/4 \cdot X + 3/8$



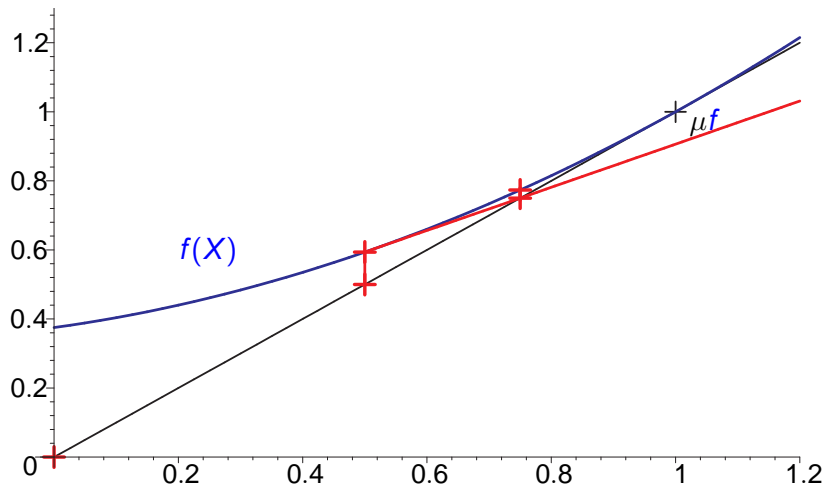
Newton-Verfahren (1-dimensional)

Betrachte $f(X) = 3/8 \cdot X \cdot X + 1/4 \cdot X + 3/8$



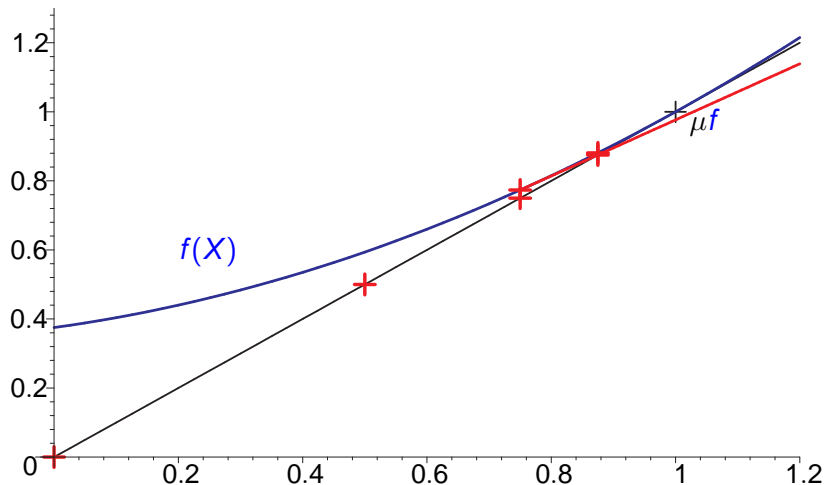
Newton-Verfahren (1-dimensional)

Betrachte $f(X) = 3/8 \cdot X \cdot X + 1/4 \cdot X + 3/8$



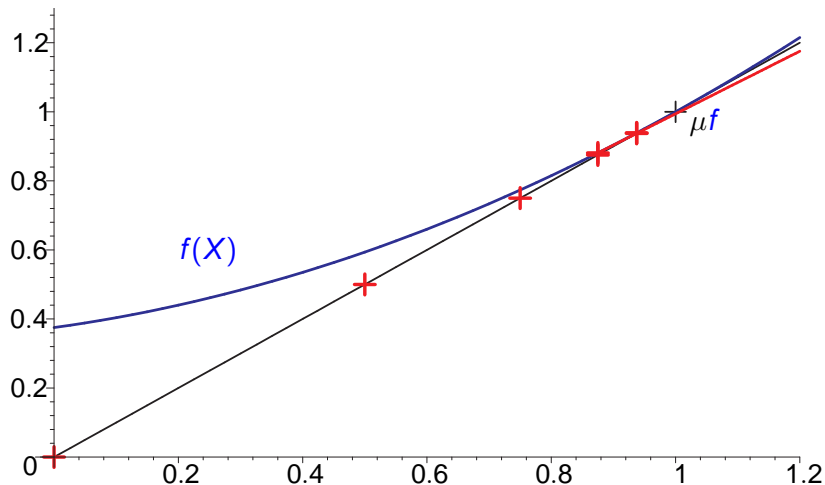
Newton-Verfahren (1-dimensional)

Betrachte $f(X) = 3/8 \cdot X \cdot X + 1/4 \cdot X + 3/8$



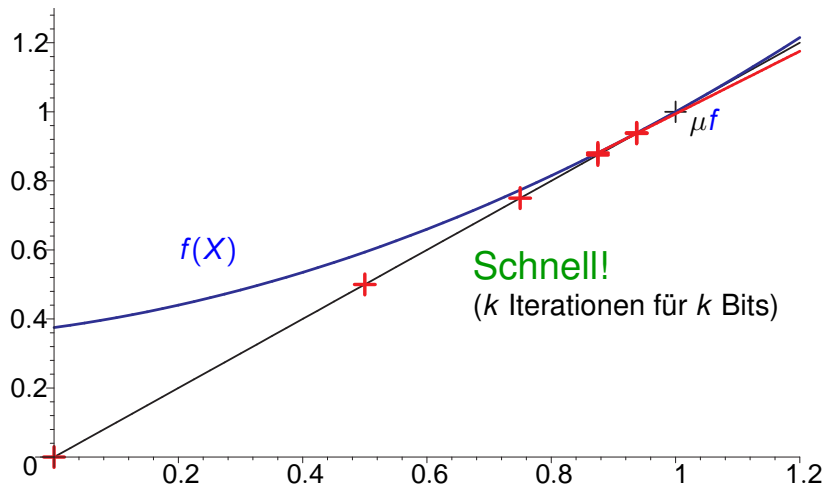
Newton-Verfahren (1-dimensional)

Betrachte $f(X) = 3/8 \cdot X \cdot X + 1/4 \cdot X + 3/8$



Newton-Verfahren (1-dimensional)

Betrachte $f(X) = 3/8 \cdot X \cdot X + 1/4 \cdot X + 3/8$



Das Newton-Verfahren konvergiert linear.

Satz

Sei $\mathbf{x} = \mathbf{f}(\mathbf{x})$ *stark zusammenhängend*.

Dann gibt es eine "Schwelle" $t_f \in \mathbb{N}$,

sodass der $(t_f + k)$ -te Newton-Iterant

mindestens k *gültige Bits* von $\mu\mathbf{f}$ aufweist.

Das Newton-Verfahren konvergiert linear.

Satz

Sei $\mathbf{x} = \mathbf{f}(\mathbf{x})$ *stark zusammenhängend*.

Dann gibt es eine "Schwelle" $t_f \in \mathbb{N}$,

sodass der $(t_f + k)$ -te Newton-Iterant
mindestens k *gültige Bits* von $\mu\mathbf{f}$ aufweist.

Insbesondere gilt das für

$t_f = \lceil 4mn + 3n \max\{0, -\log \mu_{\min}\} \rceil$ und für

$t_f = 4mn2^n$, wobei

- n die Zahl der Gleichungen (= Zahl der Variablen),
- m die Größe des Systems (Koeffizienten binär),
- μ_{\min} die kleinste Komponente von $\mu\mathbf{f}$ ist.

Wenn $\mathbf{f}(\mathbf{0}) > \mathbf{0}$ gilt, noch besser: $t_f = 7mn$.

Nicht stark zusammenhängend: auch lineare Konvergenz.

Anwendungsbeispiel für den Satz

Betrachte das Beispiel vom Anfang:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0.4YX + 0.6 \\ 0.3XY + 0.4ZY + 0.3 \\ 0.3XZ + 0.7 \end{pmatrix}$$

14 Newton-Schritte (z.B. mit Maple) ergeben:

$$\nu^{(14)} = \begin{pmatrix} 0.98283 \dots \\ 0.97380 \dots \\ 0.99269 \dots \end{pmatrix}$$

Ist die Lösung $\mu \mathbf{f} = (1, 1, 1)$?

Anwendungsbeispiel für den Satz

Betrachte das Beispiel vom Anfang:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0.4YX + 0.6 \\ 0.3XY + 0.4ZY + 0.3 \\ 0.3XZ + 0.7 \end{pmatrix}$$

14 Newton-Schritte (z.B. mit Maple) ergeben:

$$\nu^{(14)} = \begin{pmatrix} 0.98283 \dots \\ 0.97380 \dots \\ 0.99269 \dots \end{pmatrix}$$

Ist die Lösung $\mu f = (1, 1, 1)$? **Nein!** Der Satz impliziert:
Der Fehler nach 14 Iterationen ist höchstens 0.004 (8 Bits).
Daher:

$$\mu f \leq \nu^{(14)} + \begin{pmatrix} 0.004 \\ 0.004 \\ 0.004 \end{pmatrix} \leq \begin{pmatrix} 0.987 \\ 0.978 \\ 0.997 \end{pmatrix} < \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

Wir lassen in **positiven polynomiellen Gleichungssystemen**

$$\mathbf{x} = \mathbf{f}(\mathbf{x})$$

jetzt auch Minimum und Maximum zu.

Beispiel

$$X = \max\{1.2X + 0.3Y^2, 0.9Y\}$$

$$Y = \min\{0.7X^2 + 0.2, 0.4X\}$$

Damit besteht jede Komponente von $\mathbf{f}(\mathbf{x})$ aus:

- Variablen
- positiven reellen Zahlen
- und den Operatoren $+$, \cdot , \max , \min

Wie zuvor: Wir möchten $\mu\mathbf{f}$ approximieren.

Beispiel: Grippe

Ein Patient hat Grippe. Die Ärztin hat zwei Möglichkeiten:

- Verschreibe ihm **nichts**.
- **Behandle** ihn mit Muniflu.

N = W'keit, den (ursprünglich **N**icht **b**ehandelten) Patienten zu kurieren, sowie alle, die er evtl. ansteckt

B = W'keit, einen **B**ehandelten Patienten zu kurieren, sowie alle, die er evtl. ansteckt

$$N = \max \left\{ \overbrace{0.3 + 0.7NN}^{\text{nicht behandeln}}, \overbrace{0.9B + 0.1N}^{\text{behandeln}} \right\}$$
$$B = \underbrace{0.35 + 0.65BN}$$

Beispiel: Grippe

Ein Patient hat Grippe. Die Ärztin hat zwei Möglichkeiten:

- Verschreibe ihm **nichts**.
- **Behandle** ihn mit Muniflu.

N = W'keit, den (ursprünglich **N**icht **b**ehandelten) Patienten zu kurieren, sowie alle, die er evtl. ansteckt

B = W'keit, einen **B**ehandelten Patienten zu kurieren, sowie alle, die er evtl. ansteckt

$$N = \max \left\{ \overbrace{0.3 + 0.7NN}^{\text{nicht behandeln}}, \overbrace{0.9B + 0.1N}^{\text{behandeln}} \right\}$$
$$B = \underbrace{0.35 + 0.65BN}$$

Die kleinste Lösung ergibt die gesuchten W'keiten.
(Nicht unbedingt $N = B = 1$!)

Beispiel: Grippe

Ein Patient hat Grippe. Die Ärztin hat zwei Möglichkeiten:

- Verschreibe ihm **nichts**.
- **Behandle** ihn mit Muniflu.

N = W'keit, den (ursprünglich **N**icht **b**ehandelten) Patienten zu kurieren, sowie alle, die er evtl. ansteckt

B = W'keit, einen **B**ehandelten Patienten zu kurieren, sowie alle, die er evtl. ansteckt

$$\begin{array}{l} N = \max \left\{ \overbrace{0.3 + 0.7NN}^{\text{nicht behandeln}}, \overbrace{0.9B + 0.1N}^{\text{behandeln}} \right\} \\ B = \left\{ \underbrace{0.35 + 0.65BN}_{\text{normale Grippe}}, \underbrace{0.5 + 0.2BN + 0.3BNN}_{\text{Schweinegrippe}} \right\} \end{array}$$

Die kleinste Lösung ergibt die gesuchten W'keiten.
(Nicht unbedingt $N = B = 1$!)

Beispiel: Grippe

Ein Patient hat Grippe. Die Ärztin hat zwei Möglichkeiten:

- Verschreibe ihm **nichts**.
- **Behandle** ihn mit Muniflu.

N = W'keit, den (ursprünglich **N**icht **b**ehandelten) Patienten zu kurieren, sowie alle, die er evtl. ansteckt

B = W'keit, einen **B**ehandelten Patienten zu kurieren, sowie alle, die er evtl. ansteckt

$$N = \max \left\{ \overbrace{0.3 + 0.7NN}^{\text{nicht behandeln}}, \overbrace{0.9B + 0.1N}^{\text{behandeln}} \right\}$$
$$B = \min \left\{ \underbrace{0.35 + 0.65BN}_{\text{normale Grippe}}, \underbrace{0.5 + 0.2BN + 0.3BNN}_{\text{Schweinegrippe}} \right\}$$

Die kleinste Lösung ergibt die gesuchten W'keiten.
(Nicht unbedingt $N = B = 1$!)

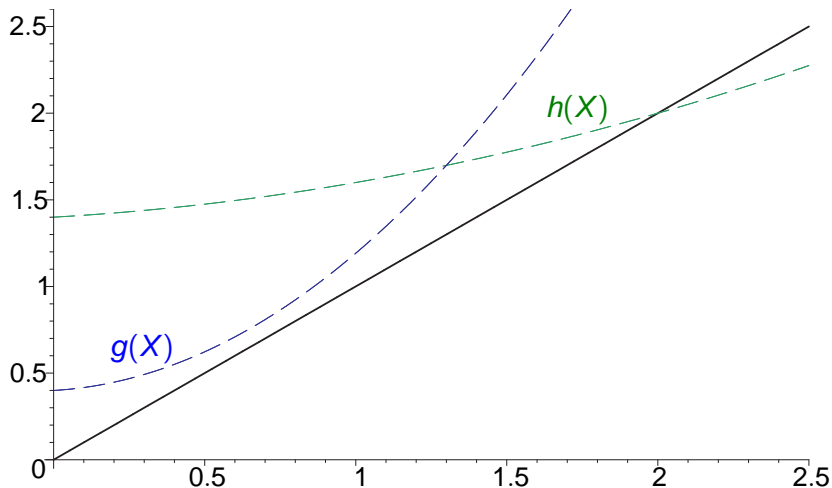
Kleene-Iteration $\mathbf{0}, f(\mathbf{0}), f(f(\mathbf{0})), \dots$

- funktioniert immer noch
- ist immer noch so langsam.

Was ist mit dem Newton-Verfahren?

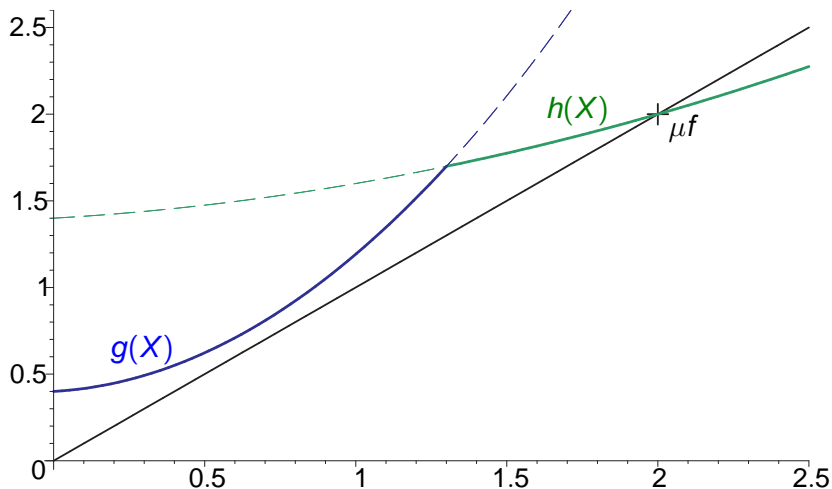
Newton-Verfahren (1-dimensional)

Betrachte Newton-Verfahren mit $f(X) = \min\{g(X), h(X)\}$.



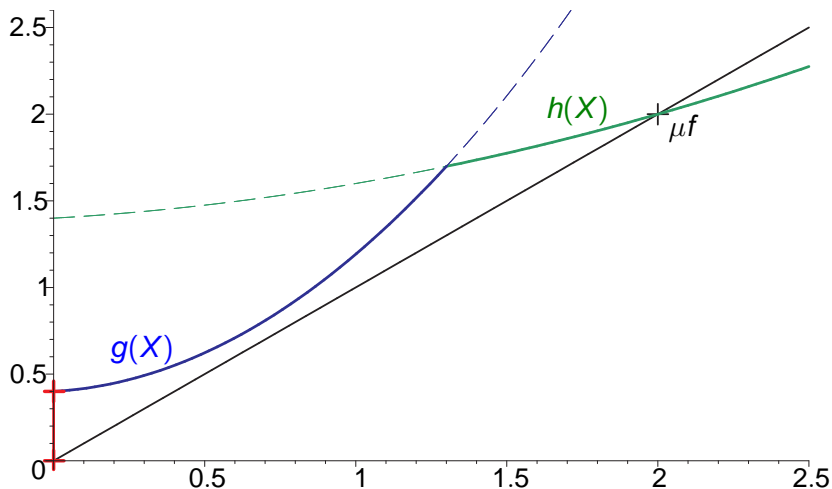
Newton-Verfahren (1-dimensional)

Betrachte Newton-Verfahren mit $f(X) = \min\{g(X), h(X)\}$.



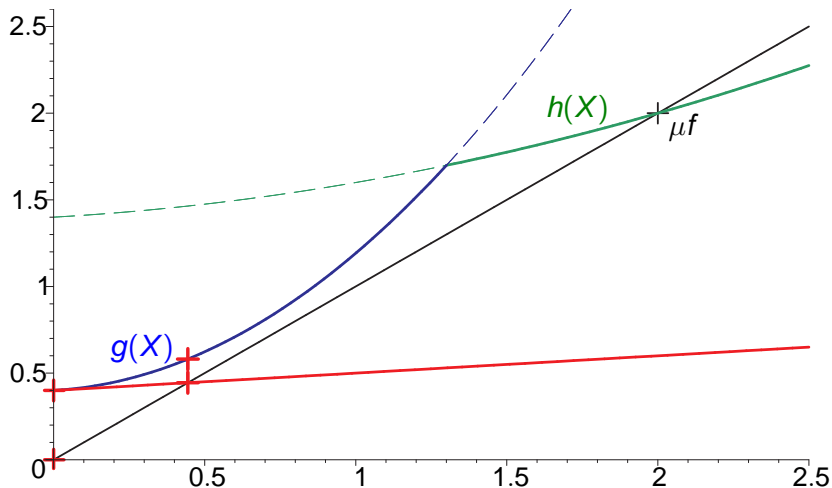
Newton-Verfahren (1-dimensional)

Betrachte Newton-Verfahren mit $f(X) = \min\{g(X), h(X)\}$.



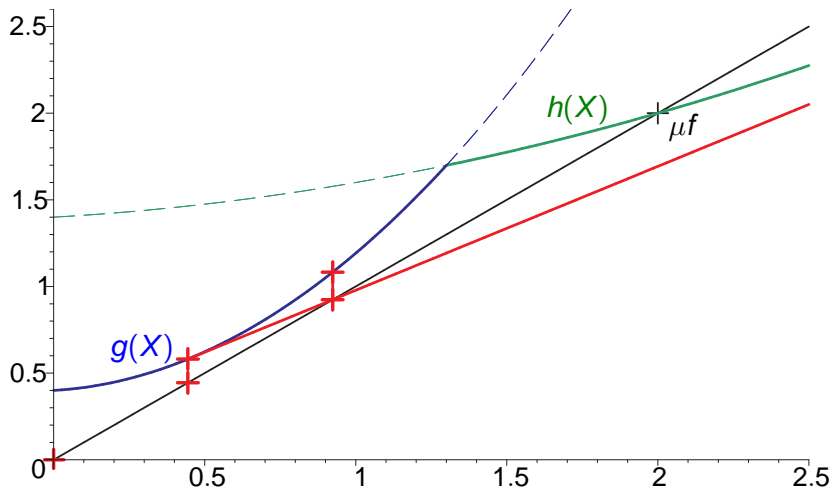
Newton-Verfahren (1-dimensional)

Betrachte Newton-Verfahren mit $f(X) = \min\{g(X), h(X)\}$.



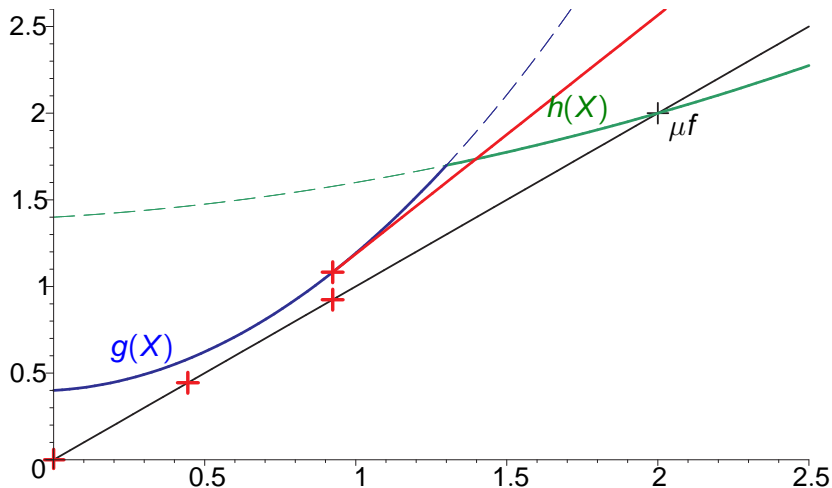
Newton-Verfahren (1-dimensional)

Betrachte Newton-Verfahren mit $f(X) = \min\{g(X), h(X)\}$.



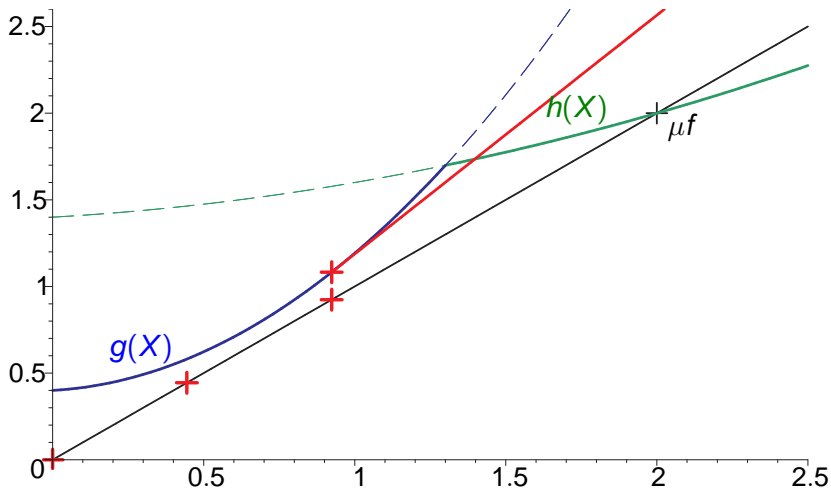
Newton-Verfahren (1-dimensional)

Betrachte Newton-Verfahren mit $f(X) = \min\{g(X), h(X)\}$.



Newton-Verfahren (1-dimensional)

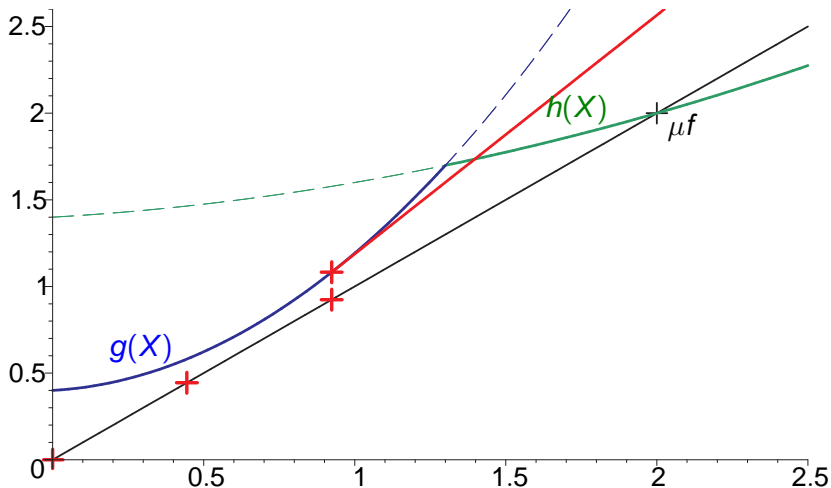
Betrachte Newton-Verfahren mit $f(X) = \min\{g(X), h(X)\}$.



Das Newton-Verfahren funktioniert nicht!

Newton-Verfahren (1-dimensional)

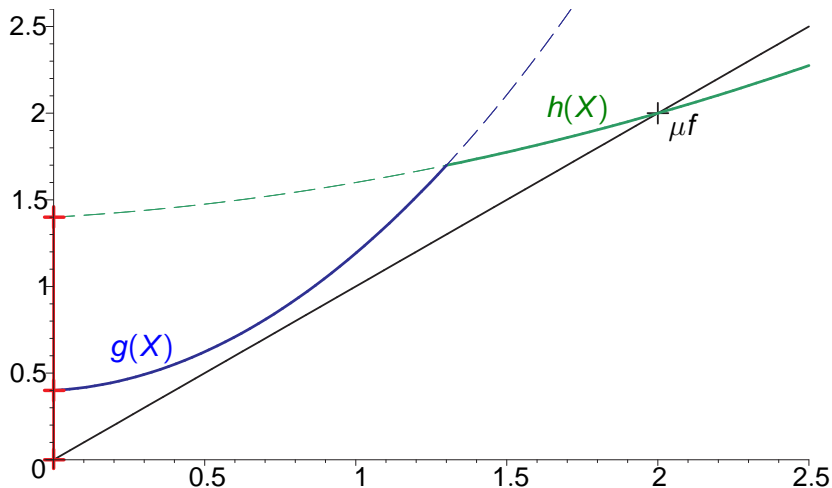
Betrachte Newton-Verfahren mit $f(X) = \min\{g(X), h(X)\}$.



Dann verändern wir es!

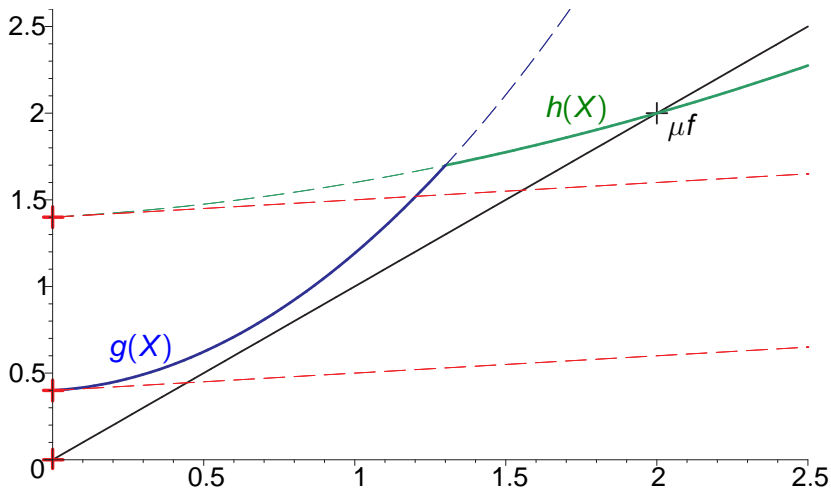
Verändertes Newton-Verfahren

Wieder $f(X) = \min\{g(X), h(X)\}$. Aber beide Teile linearisieren!



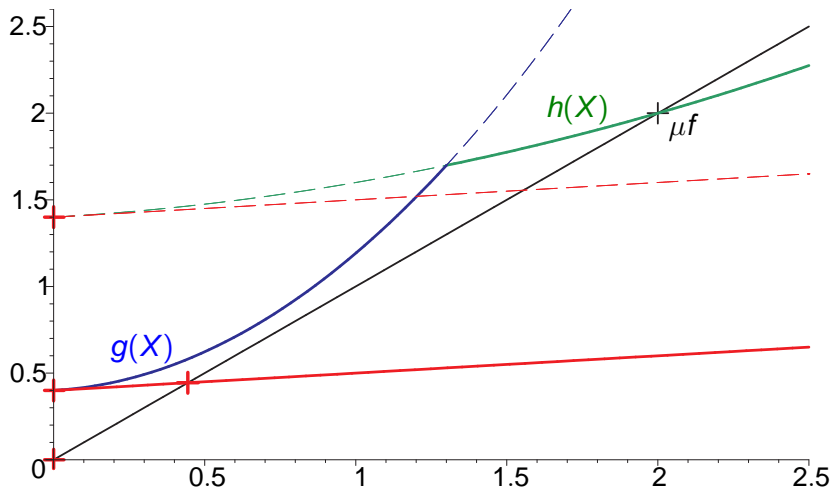
Verändertes Newton-Verfahren

Wieder $f(X) = \min\{g(X), h(X)\}$. Aber beide Teile linearisieren!



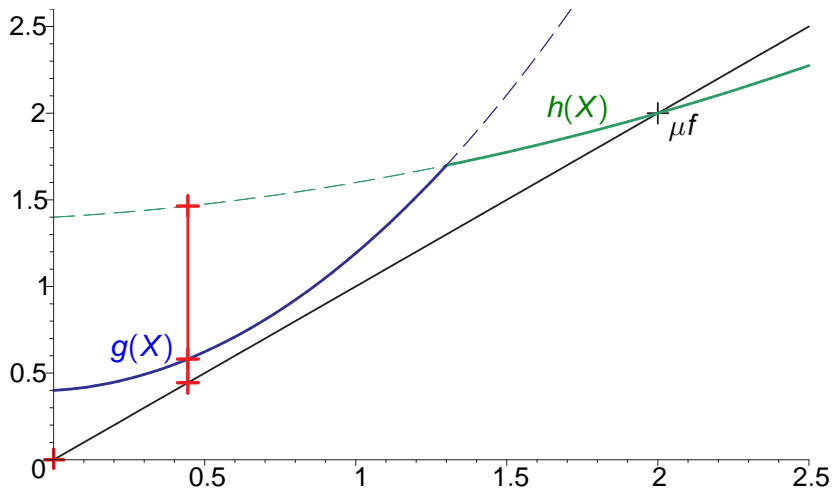
Verändertes Newton-Verfahren

Wieder $f(X) = \min\{g(X), h(X)\}$. Aber beide Teile linearisieren!



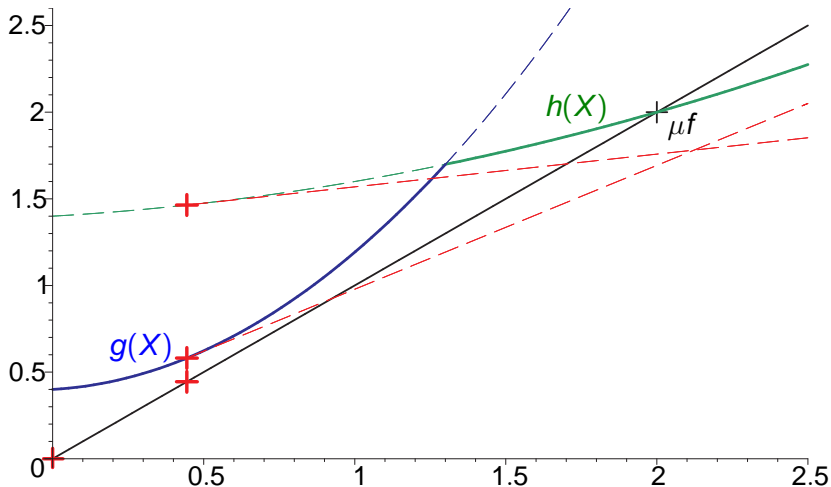
Verändertes Newton-Verfahren

Wieder $f(X) = \min\{g(X), h(X)\}$. Aber beide Teile linearisieren!



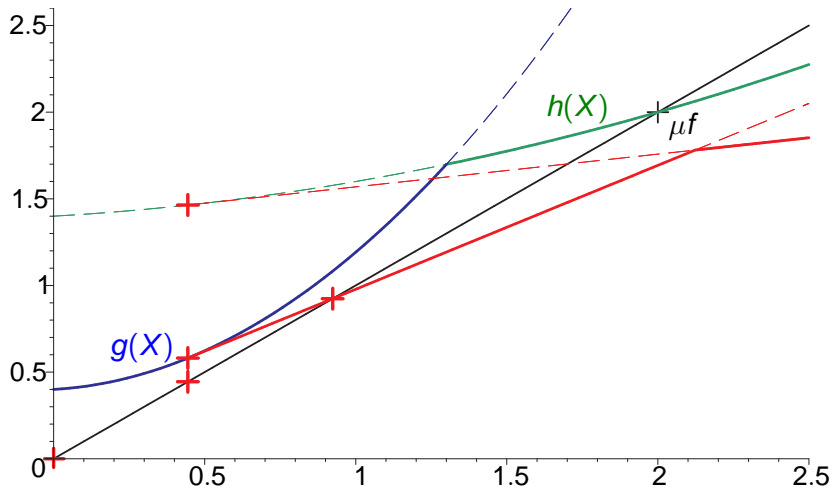
Verändertes Newton-Verfahren

Wieder $f(X) = \min\{g(X), h(X)\}$. Aber beide Teile linearisieren!



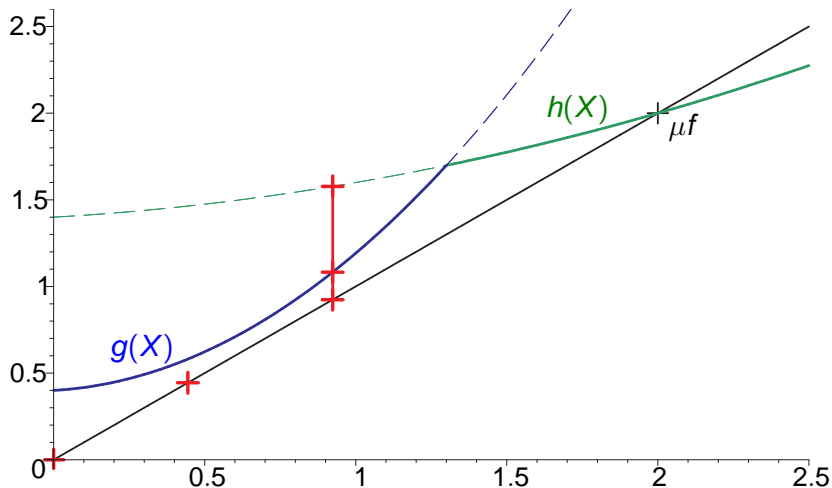
Verändertes Newton-Verfahren

Wieder $f(X) = \min\{g(X), h(X)\}$. Aber beide Teile linearisieren!



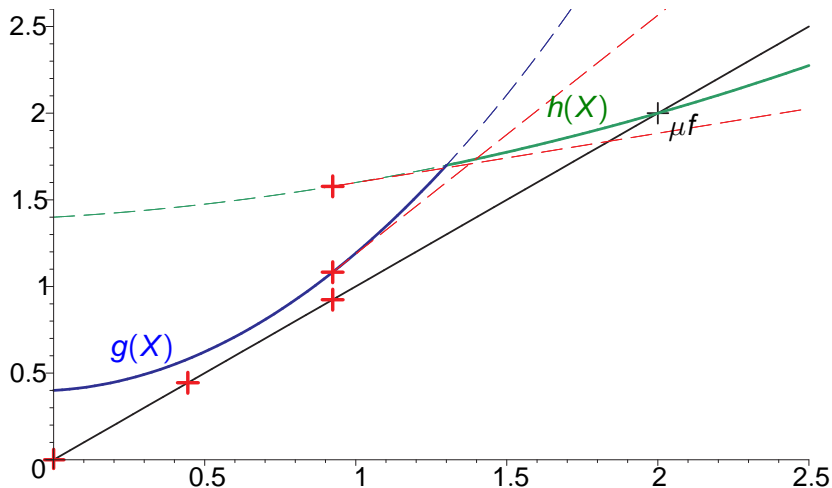
Verändertes Newton-Verfahren

Wieder $f(X) = \min\{g(X), h(X)\}$. Aber beide Teile linearisieren!



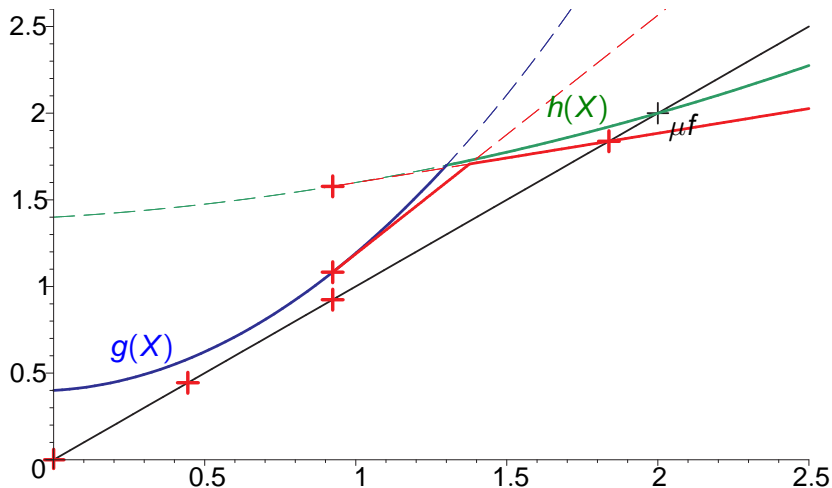
Verändertes Newton-Verfahren

Wieder $f(X) = \min\{g(X), h(X)\}$. Aber beide Teile linearisieren!



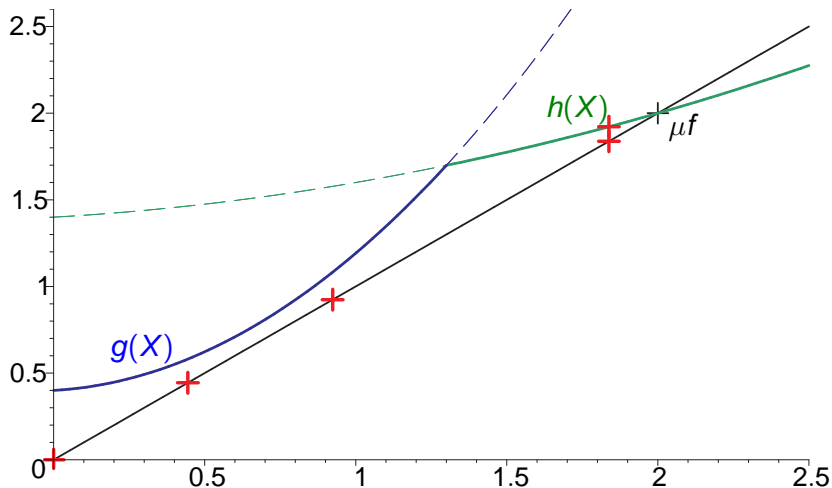
Verändertes Newton-Verfahren

Wieder $f(X) = \min\{g(X), h(X)\}$. Aber beide Teile linearisieren!



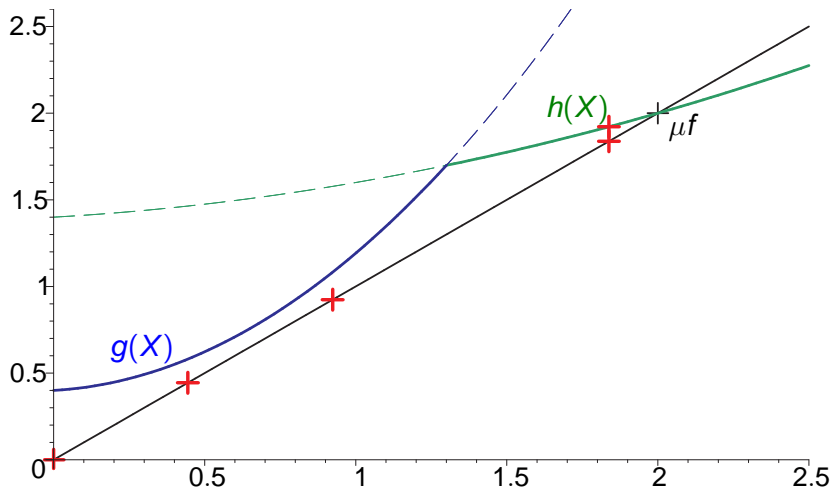
Verändertes Newton-Verfahren

Wieder $f(X) = \min\{g(X), h(X)\}$. Aber beide Teile linearisieren!



Verändertes Newton-Verfahren

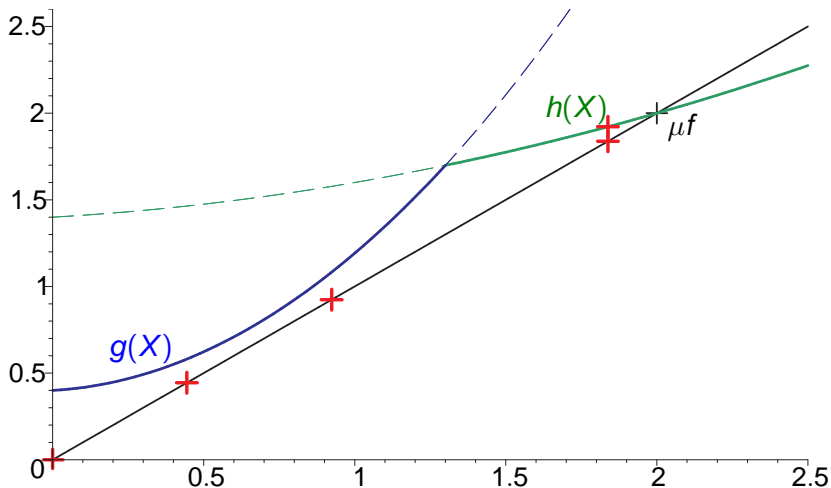
Wieder $f(X) = \min\{g(X), h(X)\}$. Aber beide Teile linearisieren!



Dieses Verfahren funktioniert!

Verändertes Newton-Verfahren

Wieder $f(X) = \min\{g(X), h(X)\}$. Aber beide Teile linearisieren!



Asymptotisch konvergiert es so schnell zu μf
wie das Newton-Verfahren ohne min/max.

Verallgemeinerung auf Maximum

Das beschriebene Verfahren lässt sich auf Systeme mit Minimum **und Maximum** verallgemeinern.

Zwei Varianten:

- In jedem Schritt **min-max-lineares** System lösen

- In jedem Schritt **min-lineares** System lösen.

Das beschriebene Verfahren lässt sich auf Systeme mit Minimum **und Maximum** verallgemeinern.

Zwei Varianten:

- In jedem Schritt **min-max-lineares** System lösen
 - ⇒ lineares Programmieren und Strategie-Iteration
 - ⇒ große, aber teure Schritte
- In jedem Schritt **min-lineares** System lösen.
 - ⇒ lineares Programmieren
 - ⇒ kleine, aber billige Schritte

Verallgemeinerung auf Maximum

Das beschriebene Verfahren lässt sich auf Systeme mit Minimum **und Maximum** verallgemeinern.

Zwei Varianten:

- In jedem Schritt **min-max-lineares** System lösen
 - ⇒ lineares Programmieren und Strategie-Iteration
 - ⇒ große, aber teure Schritte
- In jedem Schritt **min-lineares** System lösen.
 - ⇒ lineares Programmieren
 - ⇒ kleine, aber billige Schritte

Beide Varianten **konvergieren mindestens linear** zu μf .
Die erste berechnet **ϵ -optimale** Strategien.

Zurück zum Grippe-Beispiel

N = W'keit, den (ursprünglich **Nicht behandelten**) Patienten zu kurieren, sowie alle, die er evtl. ansteckt

B = W'keit, einen **Behandelten** Patienten zu kurieren, sowie alle, die er evtl. ansteckt

$$N = \max \left\{ \overbrace{0.3 + 0.7NN}^{\text{nicht behandeln}}, \overbrace{0.9B + 0.1N}^{\text{behandeln}} \right\}$$
$$B = \min \left\{ \overbrace{0.35 + 0.65BN}^{\text{normale Grippe}}, \overbrace{0.5 + 0.2BN + 0.3BNN}^{\text{Schweinegrippe}} \right\}$$

Mit der Variante mit kleinen, aber billigen Schritten:

$$\begin{pmatrix} N \\ B \end{pmatrix} \quad \left| \quad \begin{pmatrix} 0 \\ 0 \end{pmatrix} \begin{pmatrix} 0.300 \\ 0.350 \end{pmatrix} \begin{pmatrix} 0.409 \\ 0.465 \end{pmatrix} \begin{pmatrix} 0.524 \\ 0.524 \end{pmatrix} \begin{pmatrix} 0.538 \\ 0.538 \end{pmatrix} \begin{pmatrix} 0.538 \\ 0.538 \end{pmatrix} \dots \right.$$

Zurück zum Grippe-Beispiel

N = W'keit, den (ursprünglich **Nicht behandelten**) Patienten zu kurieren, sowie alle, die er evtl. ansteckt

B = W'keit, einen **Behandelten** Patienten zu kurieren, sowie alle, die er evtl. ansteckt

$$N = \max \left\{ \overbrace{0.3 + 0.7NN}^{\text{nicht behandeln}}, \overbrace{0.9B + 0.1N}^{\text{behandeln}} \right\}$$
$$B = \min \left\{ \overbrace{0.35 + 0.65BN}^{\text{normale Grippe}}, \overbrace{0.5 + 0.2BN + 0.3BNN}^{\text{Schweinegrippe}} \right\}$$

Mit der Variante mit kleinen, aber billigen Schritten:

$$\begin{pmatrix} N \\ B \end{pmatrix} \quad \left| \quad \begin{pmatrix} 0 \\ 0 \end{pmatrix} \begin{pmatrix} 0.300 \\ 0.350 \end{pmatrix} \begin{pmatrix} 0.409 \\ 0.465 \end{pmatrix} \begin{pmatrix} 0.524 \\ 0.524 \end{pmatrix} \begin{pmatrix} 0.538 \\ 0.538 \end{pmatrix} \begin{pmatrix} 0.538 \\ 0.538 \end{pmatrix} \dots \right.$$

Schnelle Konvergenz!

Zurück zum Grippe-Beispiel

N = W'keit, den (ursprünglich **Nicht behandelten**) Patienten zu kurieren, sowie alle, die er evtl. ansteckt

B = W'keit, einen **Behandelten** Patienten zu kurieren, sowie alle, die er evtl. ansteckt

$$N = \max \left\{ \overbrace{0.3 + 0.7NN}^{\text{nicht behandeln}}, \overbrace{0.9B + 0.1N}^{\text{behandeln}} \right\}$$
$$B = \min \left\{ \overbrace{0.35 + 0.65BN}^{\text{normale Grippe}}, \overbrace{0.5 + 0.2BN + 0.3BNN}^{\text{Schweinegrippe}} \right\}$$

Mit der Variante mit kleinen, aber billigen Schritten:

$\begin{pmatrix} N \\ B \end{pmatrix}$	$\begin{pmatrix} 0 \\ 0 \end{pmatrix}$	$\begin{pmatrix} 0.300 \\ 0.350 \end{pmatrix}$	$\begin{pmatrix} 0.409 \\ 0.465 \end{pmatrix}$	$\begin{pmatrix} 0.524 \\ 0.524 \end{pmatrix}$	$\begin{pmatrix} 0.538 \\ 0.538 \end{pmatrix}$	$\begin{pmatrix} 0.538 \\ 0.538 \end{pmatrix}$	\dots
Aktion d. Ärztin	nicht beh.	nicht beh.	beh.	beh.	beh.	beh.	\dots

Schnelle Konvergenz!

- Die Lösung **positiver polynomieller Gleichungssysteme**
 - ist ein natürliches Problem
 - hat vielerlei Informatik-Anwendungen
- Die entwickelten Lösungsverfahren basieren auf
 - dem **Newton-Verfahren**
 - linearem Programmieren und/oder Strategie-Iteration
- Es wird **lineare Konvergenz** gezeigt
 - mit Schranken für die Konvergenzrate
 - und teilweise **Schwellen**

Danke!