

Obliczenia komputerowe

Michał Kołodziej

1 Równania i układy równań nieliniowych

Do czego potrzebne jest nam rozwiązywanie równań i układów równań funkcji nieliniowych?

Z równaniami spotykamy się wszędzie. W Elektronice w **stanie ustalonym!** (stan ustalony to stan w którym wszystkie pochodne po jakimś czasie przyjmą wartość zero 0) relacje pomiędzy prądami i napięciami w dowolnym miejscu obwodu będą równaniami funkcji (w prostych przypadkach liniowych, w bardziej technicznych nieliniowymi).

(W problemach dynamicznych w Elektronice, koniecznym i pełnym opisem układu elektrycznego będą równania różniczkowe zwyczajne i ich układy.)

1.1 Co rozumiemy przez równanie?

Przez równanie rozumiemy równość dwóch funkcji zależnych od jednego parametru. Funkcja to przyporządkowanie jednego zbioru na drugi gdzie możemy przetestować to przyporządkowanie punktowo. Przyporządkowanie ma mieć jeden stopień swobody, ale może być różnie opisane, np.:

$$f: \mathbb{R} \rightarrow \mathbb{R}, f(x) = x^2 = x \cdot x = \sum_{i=0}^{i=\frac{x}{h}} h(2i-1)h = \{(0,0), (1,1), (2,4), (3,9) \dots\} \quad ,$$

$$g: \mathbb{R} \rightarrow \mathbb{R}, g(x) = 0 = 0 \cdot 0 = \sum 0 = \{(0,0), (1,0), (2,0), (3,0) \dots\} \quad .$$

(Każdy z tych opisów jest tak samo dobry i tyle samo z niego wynika informacji!, aczkolwiek może być mniej lub bardziej wygodny dla takich czy innych manipulacji matematycznych.

Powyższe funkcje są zapisane w postaciach odpowiednio: algebraicznej, iloczynowej, jako sumy przyrostów od 0, tablicowej.

Możliwe są też inne zapisy (fromy) np. w postaci szeregu furiera (bazą / odniesieniami są funkcje $\sin(nx)$ i $\cos(nx)$ dla n z liczb naturalnych), po transformacji Furier'a (bazą / odniesieniami są funkcje $\sin(rx)$ i $\cos(rx)$ dla r z liczb rzeczywistych), po transformacji Laplace'a (bazą / odniesieniami są funkcje wykładnicze), szeregów rekurencyjnych, i innych.

Ważne jest to, że te wszystkie postacie to precyzyjne (choć inne) opisy tego samego przedmiotu.)

Równanie wtedy było by:

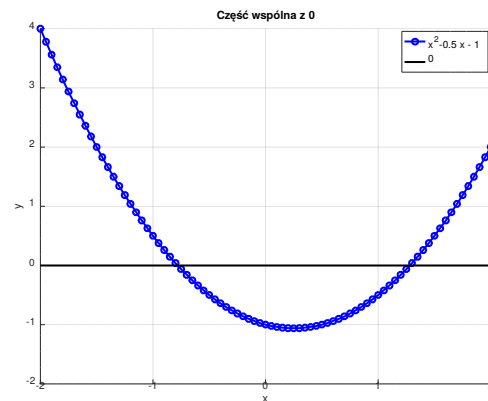
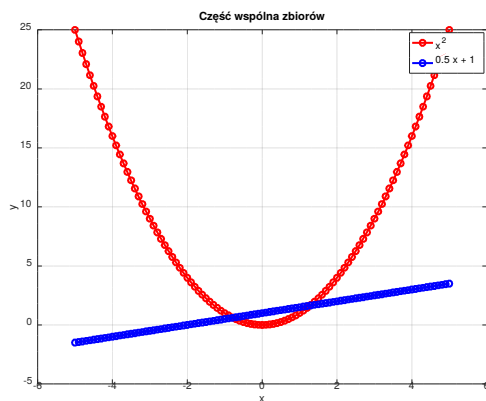
$$f(x) = g(x) \Leftrightarrow f(x) - g(x) = 0 \Leftrightarrow h(x) = 0$$

Rozwiązaniem równania z jednym parametrem swobody jest zbiór wszystkich punktów x dla których spełniony jest warunek że funkcje f i g przyjmują dla tych punktów te same wartości.

1.2 Jak obliczyć część wspólną zbiorów?

Zbiór 1 to obraz jakiejś zadanej funkcji $f(x)$, zbiór 2 to obraz innej zadanej funkcji $g(x)$. Na

przykład: $f(x) = x^2$ oraz $g(x) = \frac{1}{2}x + 1$.



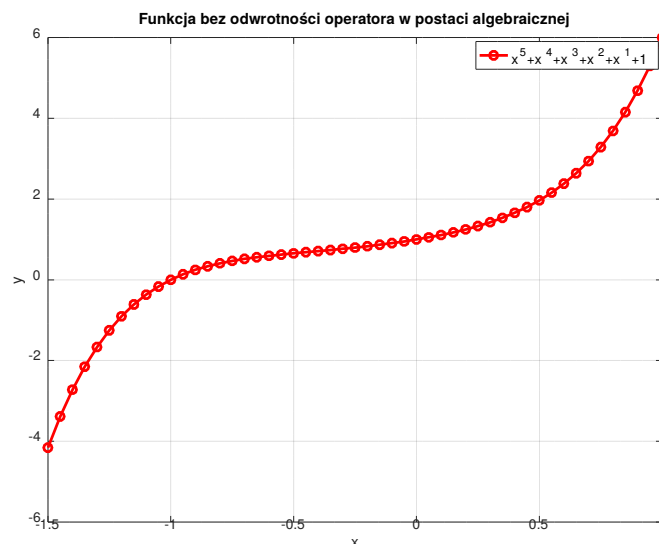
1.2.1 Manipulacja algebraiczna

$$x^2 - \frac{1}{2}x - 1 = 0 \Leftrightarrow (x + 0.78) \cdot (x - 1.28) = 0 \Leftrightarrow \begin{cases} x = -0.78 \\ x = 1.28 \end{cases}$$

Ale co jak nie znamy wzorów na odwrotności? np.:

$$x^5 + x^4 + x^3 + x^2 + x + 1 = 0$$

Wiadomo, że rozwiązanie jest, ale również wiadomo, że nie ma wzorów w postaci algebraicznej do wyznaczania miejsc zerowych (/odwrotności) wielomianów stopnia 4 bądź wyższych.

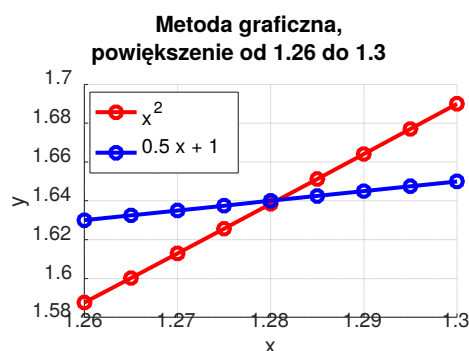
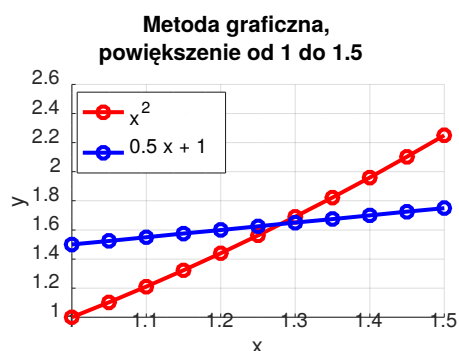
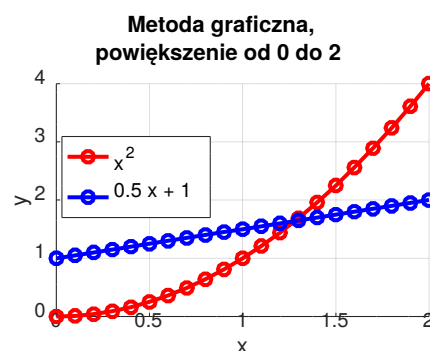
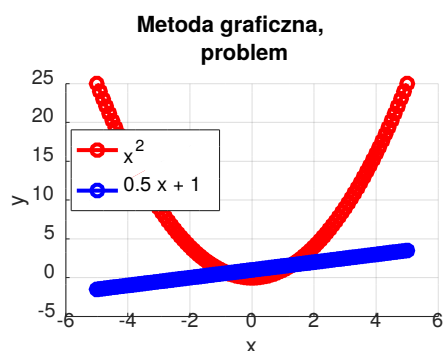


I co z tego wynika? Czy naprawdę nie istnieje odwrotność operatora wielomianowego 4 stopnia, skoro wiadomo, że jest rozwiązanie, i jak ono generalnie wygląda? Oczywiście istnieje, ale wzór musiałby również korzystać z funkcji nieelementarnych i niealgebraicznych. Więc czy można by rozwiązać ten problem algebraicznie, odpowiedź brzmi, że oczywiście, że tak, nie było by w tym błędu logicznego, jednak zazwyczaj tak się nie robi.

(Zazwyczaj ale nie zawsze, przykładem mogą być funkcje Bessel'a).

1.2.2 Graficzne punkty przecięcia się wykresów. Metoda powiększenia

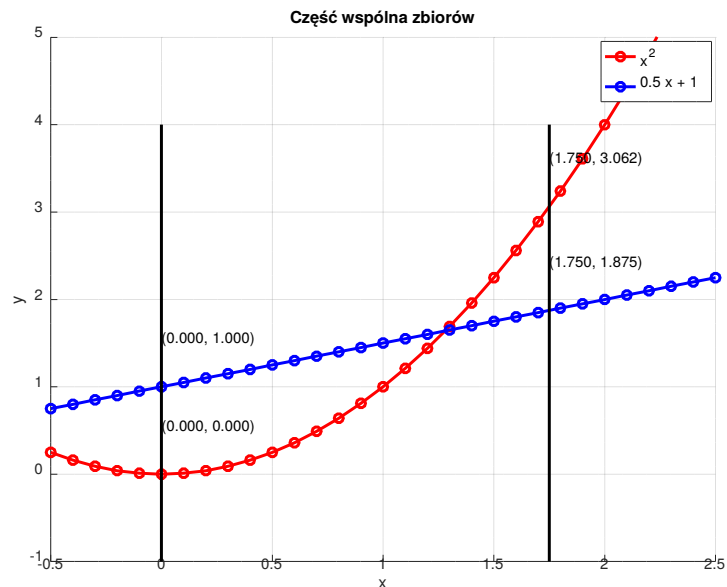
Nie ważne jaki funkcja ma wzór, ważne że możemy ją wykreślać w dowolnym powiększeniu, z dowolną dokładnością. Procedura jest oczywista: od ogółu do szczegółu, jak wyznaczymy region przecinania się wykresów, to powiększamy, i powtarzamy procedurę.



(ale co gdyby funkcji nie można zobrazować „w całości”, np. funkcję zależną od 4 parametrów?)

1.2.3 Logicznie

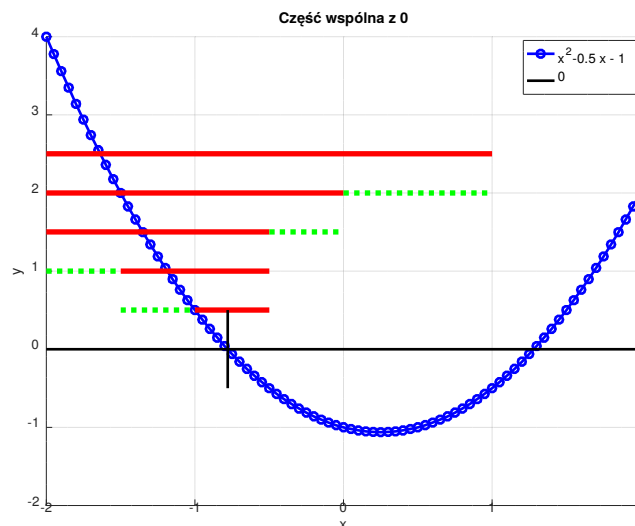
Punkty wspólne dwóch wykresów (zbiorów) możemy znaleźć posługując się pewnym założeniem i wynikającą z niego logiką. Jeśli przyjmiemy, że wykresy są ciągłe to wynika z tego, że jeżeli wybierzemy dwa punkty wykresu, to wykres odwiedzi wszystkie wartości pomiędzy z osobna:



Np. założmy że wykres jest ciągły oraz przebiega przez punkty (1,2) oraz (3,4) to oznacza że koniecznie wykres przejdzie przez punkty o wartościach (współrzędnych) (1-3, 2-4).

(Dowolny punkt ale o postaci (1-3, 2-4) zawsze będzie można znaleźć na krzywej łączącej (1,2) z (3,4).)

Trochę innym problemem ale dokładnie z tym samym rozwiązaniem będzie problem przecięcia się dwóch wykresów z których jeden będzie osią X. Wtedy rozwiązaniem naszego problemu będą miejsca zerowe.



Kiedy ciągły wykres na danym przedziale ma miejsca zerowe? => Oczywiście wtedy gdy wykres na końcach przedziału przybiera różne znaki. Jeżeli tak jest to co dalej? Oczywiście szukamy mniejszego przedziału ale z takimi własnościami. Czy zawsze możemy znaleźć taki przedział? Odpowiedź brzmi tak, ponieważ każdy przedział możemy podzielić na dwa przedziały (dowolnie), wtedy jeden z tych nowych przedziałów będzie miał taką własność, albo punkt podziału będzie miejscem zerowym.

Uwaga: większość algorytmów numerycznych rozwiązywania równań nieliniowych działa na tej zasadzie (usprawnienia polegają na optymalnie lepszym punkcie podziału przedziału w związku z przyjętymi dodatkowymi własnościami funkcji, np. że lokalnie funkcja będzie hiperbolą etc.).

Zwróćmy jednak uwagę na kilka spraw:

1. Co gdy badany przedział na końcach ma te same znaki? (Należy wtedy znać skalę szczegółowości problemu i przeszukać przedział liniowo, co jest podmiotem następnego pytania.)
2. Jakie własności musi mieć funkcja, żeby skanowanie przedziału od lewej do prawej mogło wychwycić przedział z końcami o różnych znakach (zakładając że jest miejsce zerowe)?
3. Jak wybrać punkt podziału? (Mamy tu dowolność, jednak wybór będzie wpływał na szybkość zbieżności metody. Strategią optymalną dla jedynie założenia ciągłości funkcji będzie oczywiście środek przedziału.)
4. Czy ta metoda pomija miejsca zerowe? (Gwarantowane jest że metoda znajdzie miejsce zerowe, ale tych miejsc może być więcej.)
5. Jeśli jest kilka miejsc zerowych w przedziale, które zostanie znalezione, od czego jest to uzależnione? (Od doboru pierwszego i późniejszych przedziałów.)

1.3 Potrzeba minimalnej wiedzy: ciągłość, różniczkowalność, przedziałowość

Jeżeli nic nie wiemy o badanej funkcji (nie jesteśmy w żaden sposób jej oszacować) to jedynym algorytmem jest przeglądnięcie wszystkich punktów funkcji co jest **nie możliwe**. Właściwie dla każdego problemu inżynierskiego należy przyjąć, że:

1. Podstawowym założeniem jest że funkcja jest ciągła (przynajmniej przedziałowo).
2. Kolejnym założeniem jest, że funkcja nie zmienia się zbyt szybko, czyli wartości sąsiadujących punktów są podobne (przynajmniej przedziałowo).
3. Kolejnym jest skala funkcji (problemu) i przedział zainteresowania.

Należy dobrze rozumieć te założenia w kontekście konkretnego problemu, w przeciwnym wypadku stosowanie metod numerycznych nie ma sensu.

1.4 Dokładne szukanie zera gdy „wiemy” gdzie jest zero, metody czasem zbieżne (ale szybciej)

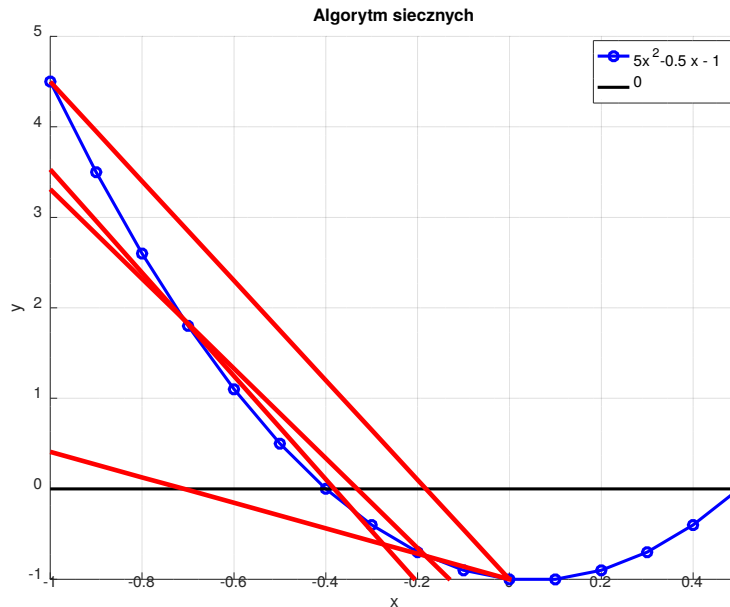
Założenie przyjęte dla tych metod jest następujące:

Funkcja w „szerszym” otoczeniu badanego punktu wygląda tak samo jak w tym punkcie.

Jest to dosyć optymistyczne, ale pamiętajmy, że **optymizm jest dobry!** Dodatkowo optymizm na komputerze dużo nie kosztuje, a zyskać można wiele.

1.4.1 Metoda Siecznych

W metodzie siecznych, ustalamy dwa punkty początkowe na krzywej, i zakładamy, że funkcja będzie miała przebieg bardzo podobny do prostej łączącej te punkty. Wtedy miejscem zerowym badanej funkcji byłby punkt bardzo blisko punktu w którym prosta ma miejsce zerowe. Przyjmujemy więc, że nowym oszacowaniem miejsca zerowego jest właśnie ten punkt. Mając nowy punkt oraz poprzedni procedurę powtarzamy.



Pamiętajmy jednak, że jeżeli kształt badanej funkcji w obrębie siecznej będzie „znacząco” inny od liniowego, procedura może nie znaleźć rozwiązania.

Procedurę tą można łatwo zapisać algebraicznie. Krok wstępny to wyznaczenie stosunku przyrostów dla pierwszej siecznej, jest to:

$$\left. \frac{\Delta y}{\Delta x} \right|_{x_{\text{teraz}}} = \frac{f(x_{\text{teraz}}) - f(x_{\text{wcześnie}})}{x_{\text{teraz}} - x_{\text{wcześnie}}}$$

i znajdujemy się w x_{teraz} , ($y_{\text{teraz}} = f(x_{\text{teraz}})$) to jeśli chcemy zaktualizować przybliżenie to powinniśmy się przesunąć o Δx , wtedy dla przybliżenia liniowego jesteśmy w punkcie ($x_{\text{teraz}} + \Delta x$, $y_{\text{teraz}} = 0$) możemy to łatwo zrobić, mnożąc odwrotność stosunku przyrostów przez $\Delta y = f(x_{\text{teraz}})$:

$$\Delta x = \left. \frac{\Delta x}{\Delta y} \right|_{x_{\text{teraz}}} \Delta y = \frac{x_{\text{teraz}} - x_{\text{wcześnie}}}{f(x_{\text{teraz}}) - f(x_{\text{wcześnie}})} f(x_{\text{teraz}})$$

x_{nowe} będzie miało wartość:

$$x_{\text{nowe}} = x_{\text{teraz}} - \Delta x = x_{\text{teraz}} - \frac{x_{\text{teraz}} - x_{\text{wcześnie}}}{f(x_{\text{teraz}}) - f(x_{\text{wcześnie}})} f(x_{\text{teraz}})$$

1.4.2 Metoda Newtona

Procedura metody Newtona jest identyczna do procedury siecznych, z tym że sieczne zastępujemy stycznymi. Metoda ta zakłada, że badana funkcja w otoczeniu zadanego punktu w paraktyce zachowuje się jak pierwsze dwa wyrazy rozwinięcia tej funkcji w szereg Taylora.

Procedura w wersji geometrycznej wygląda następująco: po wyborze punktu startowego, rysujemy styczną do badanej krzywej w tym punkcie, miejsce przecięcia się stycznej z osią X to współrzędna x-owa „lepszego” przybliżenia, współrzędna y-owa to wartość badanej funkcji dla nowego x, mamy w ten sposób nowy punkt. Powtarzamy procedurę względem nowego punktu. Proszę to sobie zobrazować na własnym przykładzie.

Procedurę tą można również łatwo zapisać algebraicznie:

$$x_{\text{nowe}} = x_{\text{teraz}} - \Delta x = x_{\text{teraz}} - \frac{dx}{dy} \bigg|_{x_{\text{teraz}}} f(x_{\text{teraz}}) = x_{\text{teraz}} - \frac{f(x_{\text{teraz}})}{f'(x_{\text{teraz}})}$$

W przypadku metody Newtona pamiętajmy również, że w zależności z jakiego punktu zaczniemy, może się okazać, że na przykład styczne będą wybierały na osi X punkty które będą tworzyły cykl, i wtedy niezależnie od liczby iteracji, nie uda się zbliżyć do rozwiązania. Oczywiście jest też, że gdy wartość pochodnej w mianowniku będzie zbliżała się do zera, albo będzie zerem, algorytm nie da wyniku.

1.4.3 Jak ująć matematycznie, że coś się „mało” zmienia?

Pytanie jest podchwytliwe, właściwie chodzi o to, że coś nie zmienia się mało tylko cały czas podobnie. Typowo zapisujemy to tak, że **cała!** badana funkcja ma wzór taki jak rozwinięcie tej funkcji w szereg Taylor’a w jakimś zadanym punkcie.

Niektóre funkcje mają taką własność (np. funkcja wykładnicza, której jedna z definicji jest szeregiem Taylor’a dla punktu 0), inne nie.

Twierdzenie Taylora mówi, że wartości funkcji **tylko w otoczeniu** punktu są idealnie równe rozwinięciu w szereg Taylora. Pamiętajmy jednak, że otoczenie jest wielkością niemierzalną w takim sensie, żeby można je przyporządkować jakimukolwiek mierzalnemu odcinkowi.

1.5 Jak czasami znaleźć część wspólną zbiorów wielowymiarowych? (Układy równań nieliniowych)

W przypadku efektywnego (relatywnie szybkiego) obliczania miejsc zerowych układów równań nieliniowych optymizm jest wymagany. Algorytmy oparte o metodę przedziałową dla układów równań nieliniowych istnieją, ale są dużo bardziej skomplikowane (matematycznie i koncepcyjnie), wymagają też dużej mocy obliczeniowej i dlatego nie będą tutaj prezentowane.

1.5.1 Jak wygląda geometrycznie układ równań nieliniowych?

Czym jest zbiór (x_1, x_2) spełniający warunek $f_1(x_1, x_2) = 0$ i $f_2(x_1, x_2) = 0$? Przyjrzyjmy się składnikom:

- $f_1 : (x_1, x_2) \rightarrow \mathbb{R}$, \leq to reprezentuje płaszczyznę.
- $f_2 : (x_1, x_2) \rightarrow \mathbb{R}$ \leq to również reprezentuje płaszczyznę.
- $f_1(x_1, x_2) = 0 \leq$ to reprezentuje przecięcie się płaszczyzny f_1 z płaszczyzną $0_{x_1x_2}$ (krzywą, w przypadku funkcji liniowych przecięcie będzie prostą).
- $f_2(x_1, x_2) = 0 \leq$ to reprezentuje przecięcie się płaszczyzny f_2 z płaszczyzną $0_{x_1x_2}$ (krzywą, w przypadku funkcji liniowych przecięcie będzie prostą).
- $f_1(x_1, x_2) = 0$ i $f_2(x_1, x_2) = 0 \leq$ to reprezentuje przecięcie się krzywych (prostych w przypadku funkcji liniowych) na płaszczyźnie $0_{x_1x_2}$.

1.5.2 Metoda Newtona-Raphsona

Metodą pozwalającą lokalnie rozwiązać układ równań nieliniowych jest metoda Newtona-Raphsona. Polega ona na optymistycznym założeniu, że badany układ równań wygląda globalnie tak jak układ równań liniowych.

Zapisując układ równań nieliniowych w postaci macierzowej:

$$\mathbf{f}(\mathbf{x}) = 0$$

Traktujemy ten układ w badanym punkcie jako układ równań liniowych stosując twierdzenie Taylora (pamiętajmy że pochodna to współczynniki nachylenia)

$$\mathbf{f}(\mathbf{x}) \Big|_{\mathbf{x}_{\text{teraz}}} = \mathbf{A}(\mathbf{x} - \mathbf{x}_{\text{teraz}}) + \mathbf{b} = \frac{d\mathbf{f}(\mathbf{x})}{d\mathbf{x}} \Big|_{\mathbf{x}_{\text{teraz}}} (\mathbf{x} - \mathbf{x}_{\text{teraz}}) + \mathbf{f}(\mathbf{x}_{\text{teraz}})$$

Nowe położenie punktu rozwiązującego układ przybliżamy rozwiązaniem układu liniowego, szukamy takiego \mathbf{x} żeby $\mathbf{A}(\mathbf{x} - \mathbf{x}_{\text{teraz}}) + \mathbf{b} = 0$, czyli:

$$\mathbf{x}_{\text{nowe}} = \mathbf{x}_{\text{teraz}} - \frac{d\mathbf{x}}{d\mathbf{f}(\mathbf{x})} \Big|_{\mathbf{x}_{\text{teraz}}} \mathbf{f}(\mathbf{x}_{\text{teraz}})$$

W przypadku dwóch zmiennych pochodna wektorowa, będzie macierzą 2x2 nazywaną Jakobianem

$$\frac{d\mathbf{f}(\mathbf{x})}{d\mathbf{x}} \Big|_{\mathbf{x}_{\text{teraz}}} = \begin{bmatrix} \frac{\partial f_1(x_1, x_2)}{\partial x_1}(\mathbf{x}_{\text{teraz}}) & \frac{\partial f_1(x_1, x_2)}{\partial x_2}(\mathbf{x}_{\text{teraz}}) \\ \frac{\partial f_2(x_1, x_2)}{\partial x_1}(\mathbf{x}_{\text{teraz}}) & \frac{\partial f_2(x_1, x_2)}{\partial x_2}(\mathbf{x}_{\text{teraz}}) \end{bmatrix}$$

Odwrotną pochodną będzie odwrotność macierzy. (Do obliczenia odwrotności macierzy możemy użyć algorytmu Gauss'a).

1.6 Kiedy przestać?

Podane wcześniej algorytmy w każdej iteracji próbują znaleźć miejsce zerowe równania albo układu równań. Jak ocenić czy algorytm poprawia rozwiązanie? Jak ocenić czy rozwiązanie jest dostatecznie dokładne?

Stosuje się najczęściej porównanie względne, jeżeli zmiana względna oszacowań zera jest mała to przestajemy, algebraicznie:

$$\frac{\|x_{\text{teraz}}\|}{\|x_{\text{wcześnie}}\|} < \varepsilon$$

1.7 Co z tego wiedzieć, co umieć?

Wiedza:

1. Jakie są reprezentacje przyporządkowania zbiorów.
2. Jaką metodą co można uzyskać.

Umiejętności podstawowe:

1. Konsekwentnie wyznaczyć graficznie rozwiązanie problemu metodami:
 1. Przedziałów.
 2. Siecznych.
 3. Newtona.
 4. Newtona-Ramphsona.
2. Z komentarzem o zbieżności i globalności rozwiązania.

Umiejętności techniczne (ćwiczenia/laboratoria):

1. Powiązanie geometrii algorytmów Przedziałów, Siecznych, Newtona, Newtona-Ramphsona z zapisem algebraicznym.
2. Zastosowanie algorytmów w środowisku Matlab/Octave.
3. Opisanie/interpretacja/napisanie kodu źródłowego dla algorytmów Przedziałów, Siecznych, Newtona, Newtona-Ramphsona.

1.8 Przykładowa implementacja algorytmów w Matlab/Octave

Przykład dla metody siecznych:

```
f = @(x) 5.*x.^2;      %%%% Definicja funkcji anonimowej
g = @(x) x.*0.5 + 1;   %%%% Definicja funkcji anonimowej
h = @(x) f(x) - g(x);  %%%% Definicja funkcji anonimowej
zero = @(x) x.*0;
x = -1:0.1:0.5;

x_prev = -1;
x_curr = 0
nextSecantGuess = @(f, x_curr, x_prev) x_curr - (f(x_curr)*(x_curr - x_prev)/(f(x_curr) -
f(x_prev)));
next = @(x_curr, x_prev) nextSecantGuess(h, x_curr, x_prev);
secant = @(f, x_curr, x_prev, x) (f(x_curr) - f(x_prev))/(x_curr - x_prev).*(x.-x_prev) .+
f(x_prev);

figure;      %%%% nowe okno na wykresy
hold "on";  %%%% operacje na wykresie nie będą tworzyły nowych okien
plot(x,h(x), "-ob;5x^2-0.5 x - 1;", 'LineWidth', 2); %%% wyrysowanie wektorów x i f(x) - g(x)
plot(x,zero(x), "-k;0;", 'LineWidth', 2);           %%% wyrysowanie x-> 0
xlabel("x"); %%% nazwa osi x ma być x
ylabel("y"); %%% nazwa osi y ma być y
title("Algorytm siecznych"); %%% ustawienie tytułu wykresu
grid "on"; %%% wyświetlenie siatki
ylim([-1, 5]);
xlim([-1, 0.5]);

plot(x,secant(h, x_curr, x_prev, x),"linestyle", "--", "color", "r", "linewidth", 3);

x_news = next(x_curr, x_prev);
x_prev = x_curr;
x_curr = x_news
plot(x,secant(h, x_curr, x_prev, x),"linestyle", "--", "color", "r", "linewidth", 3);

x_news = next(x_curr, x_prev);
x_prev = x_curr;
x_curr = x_news
plot(x,secant(h, x_curr, x_prev, x),"linestyle", "--", "color", "r", "linewidth", 3);

x_news = next(x_curr, x_prev);
x_prev = x_curr;
x_curr = x_news
plot(x,secant(h, x_curr, x_prev, x),"linestyle", "--", "color", "r", "linewidth", 3);

print -dsvg SecantGeometry.svg; %%% zapisanie okna do pliku
```