

Matlab

- Variable diverser Typen
- Rechenoperationen
- Funktionen
- Graphik, Animation
- Eingabe, Ausgabe von Daten
- Schleifen, Verzweigungen
- Tools
 - Gleichungslöser
 - Optimierer
 - Differentialgleichungslöser
 - Eigenwertproblemlöser

Was möchte man weniger gern?

- mystische Präambeln
- Einbinden von Headern, Bibliotheken
- Deklaration von Variablen
- Bereitstellen von Speicher
- kompilieren, linken

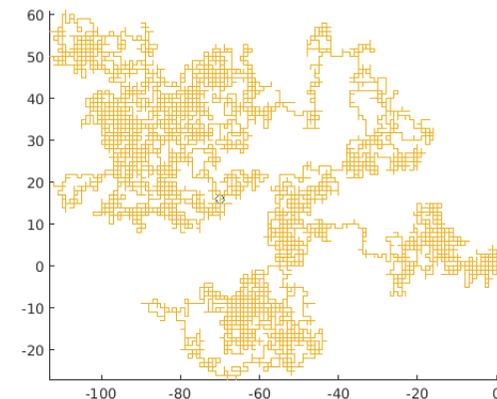
Matlab (und vergleichbare Entwicklungsumgebungen wie Octave, SciLab, julia) ersparen uns weitgehend die Sorge um lästige Nebensachen.

Dafür sind sie meist deutlich langsamer – was bei großen Problemen störend sein kann.

Modellfall: Irrfahrt

Shizuo Kakutani

A drunk man will find his way home, but a drunk bird may get lost forever.



Annahmen

- gesucht ist Weg – Folge von direkt benachbarten Knoten in kardinalem Gitter
- an jedem Knoten wird der Nachfolger fair unter den vier Nachbarn (in der Ebene) ausgelost
- analog wird im D -dimensionalen Fall unter $2D$ Nachbarn gleichverteilt gelost
- im eindimensionalen Fall wird jeweils ein Schritt nach links bzw. rechts gegangen

Bemerkungen

- keine Abhängigkeit von Vorgeschichte
- Verallgemeinerungen: unterschiedliche Wahrscheinlichkeiten für rechts/links, kontinuierliche Winkel und Schrittlängen
- Problem geht zurück auf Arbeiten von Pearson, Pólya, Rayleigh (stochastische Prozesse, Markovketten)

- in 1D und 2D gibt es eine fast sichere Rückkehr zum Ausgangspunkt
- in höheren Dimensionen ist Rückkehr wenig wahrscheinlich
- auch erneute Treffen unabhängig Herumirrender sind in 1D und 2D fast sicher, in 3D und höher nicht

Programmierung (Matlab, 2D)

- Weg wird als Matrix $m \times 2$ repräsentiert
- erste Spalte: x -Koordinate, zweite: y -Werte
- Start (erste Zeile): $[0, 0]$
- Zuwächse: entweder $\Delta x = \pm 1$ und $\Delta y = 0$, oder $\Delta y = \pm 1$ und $\Delta x = 0$
- wiederholen Schritte bis m erreicht, oder Rückkehr zum Ausgangspunkt erfolgt
- Visualisieren des Weges, ggfs. Animation
- Export der Ergebnisse – Speichern von Bild- oder Filmdateien

Implementierung (naiv)

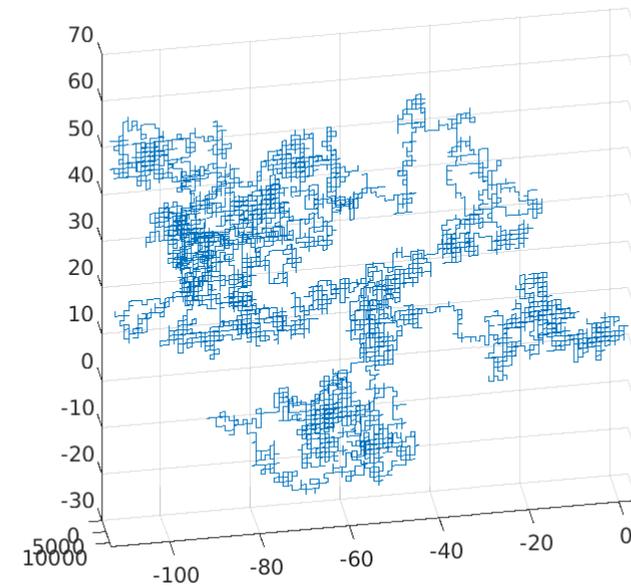
- setze Anfangspunkt
- wiederhole in Schleife
 - wähle mit `rand() < 0.5`, ob in x - oder y -Richtung gegangen werden soll
 - wähle analog, ob Schritt vor- oder rückwärts gehen soll
 - füge neuen Schritt an bisherigen Weg an
 - prüfe Abbruchbedingung
- Ausgabe

```
% Berechnung des Weges
```

```
m=12000; % Festlegen der Weglaenge
o=floor(4*rand(m,1)); % "Wuerfeln" der Richtungswahl,
% Optionen 0 bis 3
z=i.^o; % Nutzen komplexe Zahlen
d=[real(z) imag(z)]; % Zuwaechse
xy=cumsum(d); % Aufaddierte Zuwaechse
```

```
% Ausgabe als Trajektorie
```

```
plot3(1:m,xy(:,1),xy(:,2))
set(gca,'PlotBoxAspectRatio',[0.1,1,1],...
'ygrid','on','zgrid','on')
view(80,30)
print -dpng dogwalking % Speichern als png-Grafik
comet(xy(:,1),xy(:,2)) % simple Animation
```



Navigation icons: back, forward, search, etc.

Navigation icons: back, forward, search, etc.