Inleiding programmeren

1e jaar wis-, natuur- en sterrenkunde

Universiteit van Amsterdam

oktober 2014

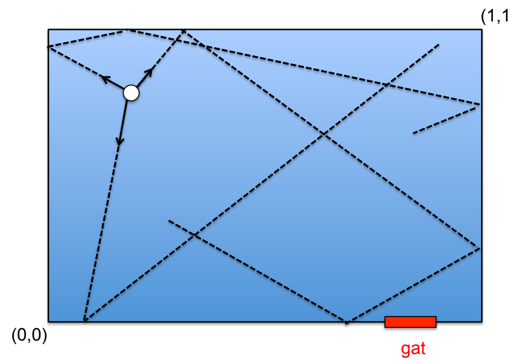
Opgaves bij college 6

simulaties

opgave 1: deeltjes in een doos (basics)

In een doos (afmeting $0 \leq x \leq 1$ en $0 \leq y \leq 1$) worden op een plek $(x_{\rm source}, y_{source}) = (0.25, 0.75)$ een aantal deeltjes geproduceerd met een random snelheid en richting. Voor elk deeltje i geldt dus:

- snelheid (v_i): $0 < v_i < 0.10$
- hoek(α_i): $0 < \alpha_i < 2\pi$



We gaan de positie van een groot aantal deeltjes volgen. In deze opgave zullen we een aantal aannames maken die de natuurkunde versimpelen, maar ons toch in staat stellen wat interessante fenomenen te onderzoeken. Onze aannames:

- De deeltjes ketsen elastisch tegen de wanden en kunnen de doos niet uit
- De deeltjes hebben geen afmeting en kunnen niet botsen

algemeen: Genereer een beginsituatie voor een aantal deeltjes i en hou voor elk zijn x-positie (x), y-positie (y), snelheid in de x-richting (v_x) en die in de y-richting (v_y) bij. Je kan lists gebruiken, maar ook de Tuples van vorige week. Neem steeds stapjes in de tijd en gebruik daarbij: $x_{i+1} = x_i + v_{i(x)}\Delta t$ etc. Zelfde idee voor snelheden.

a) Maak een grafiek van het aantal deeltjes aan de rechterkant van de doos ($x_i > 0.5$) als functie van de tijd.

b) Maak een grafiek van de gemiddelde afstand tussen de deeltjes als functie van de tijd.

Stel nou dat er een gat in de doos zit ($y_{\rm gat} = 0$ en $0.8 \leq x_{\rm gat} \leq 0.9$). Het is dan mogelijk dat deeltjes uit de doos ontsnappen.

c) Maak een grafiek van het aantal deeltjes in de doos als functie van de tijd. Wat is de gemiddelde tijd waarop de helft van de deeltjes uit de doos ontsnapt is: $t_{1/2}$? Probeer ook zonder computerprogramma een schatting te geven.

d) Stel nou dat de deeltjes gemiddeld met een 2x hogere hogere snelheid beginnen (v_i is random tussen 0 en 0.20). Maak weer dezelfde grafiek als bij c) en bepaal opnieuw $t_{1/2}$. Hoe verschilt deze van die bij c) ? Wat verwachtte je ?

Hacker: deeltjes in een doos (realisme)

Animatie, afmeting, snelheidsverdeling

Er zijn verschillende mogelijkheden om deze simulatie uit te breiden met meer realisme. Het visualiseren van een simulatie is erg leuk en interessant omdat het je meer inzicht geeft in mogelijke programmeerfouten en fenomenen die soms niet gelijk opvallen als je naar de vergelijkingen zelf kijkt. We zullen de deeltjes ook een 'echte' afmeting geven en de deeltjes te laten botsen.

a) Animatie puntdeeltjes. Gebruik het voorbeeld `animation_template_circles.py` om om de deeltjes door de doos te zien bewegen als functie van de tijd.

b) Animatie deeltjes met afmeting. Geef de deeltjes een afmeting en laat ze netjes van de wand ketsen als de rand van het deeltje de wand raakt

c) Laat de deeltjes realistisch botsen, dus met de echte afmeting.
Tip voor afleiding: *2-dimensional elastic collisions without trigonometry*

Zodra twee deeltjes botsen veranderen zowel de snelheid als de bewegingsrichting van de deeltjes. Interessant is dat de snelheidsverdeling uiteindelijk de Boltzmannverdeling zal volgen, onafhankelijk van de begincondities. We willen dat graag bewijzen door de verdeling te bekijken na een tijd lang botsen.

d) Begin met dezelfde condities als in vraag 1a) en maak een grafiek van de snelheidsverdeling na 100 en na 1000 botsingen.

e) Probeer (om fluctuaties te voorkomen) een 'gemiddelde' snelheidsverdeling te maken. Dus bijvoorbeeld het gemiddelde profiel tussen botsing tijdstap 95 en 105.

f) Andere begincondities. Begin nu eens met een beginsituatie waarin alle deeltjes dezelfde snelheid hebben en vergelijk de uiteindelijke snelheidsverdeling met die uit de vorige vraag. Wat valt je op ? Zijn ze inderdaad hetzelfde ?