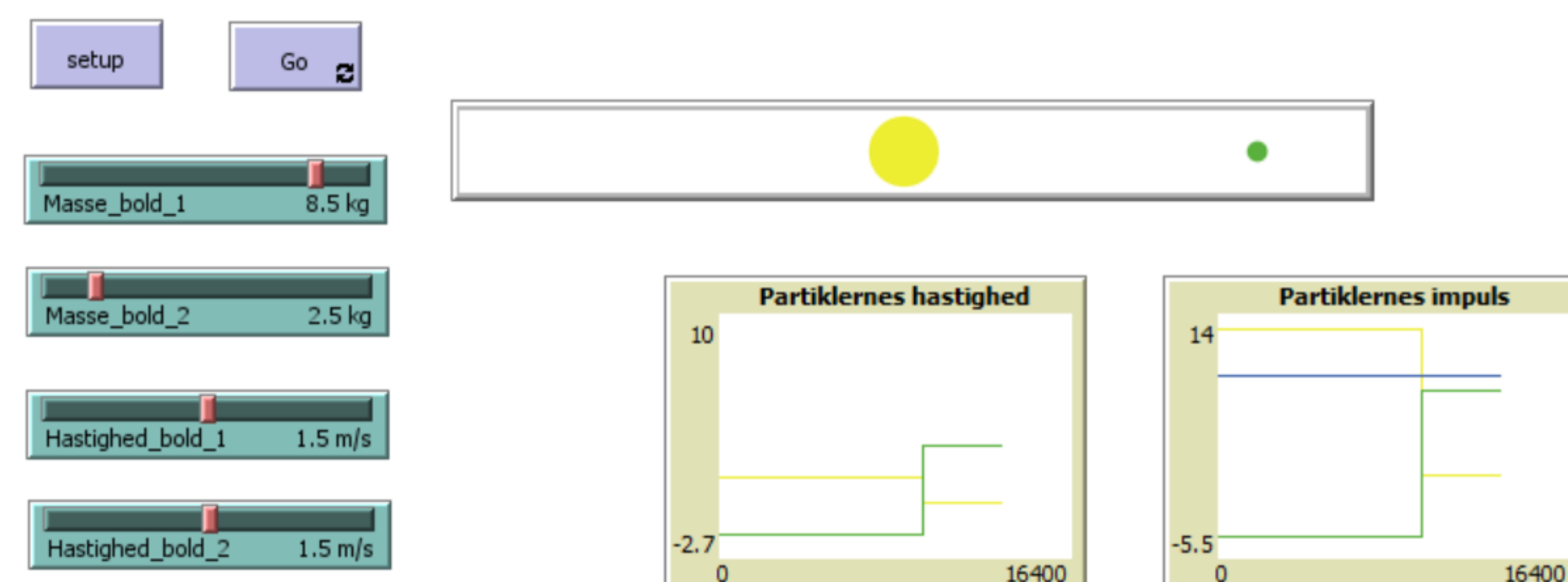


# Stød i en og to dimensioner

## Introduktion

*Stød* er et emne tilhørende kernestoffet på Fysik A. Eleverne er vant til at arbejde med simuleringer, men har måske ikke den store forståelse for, hvad der ligger bag en simulering.

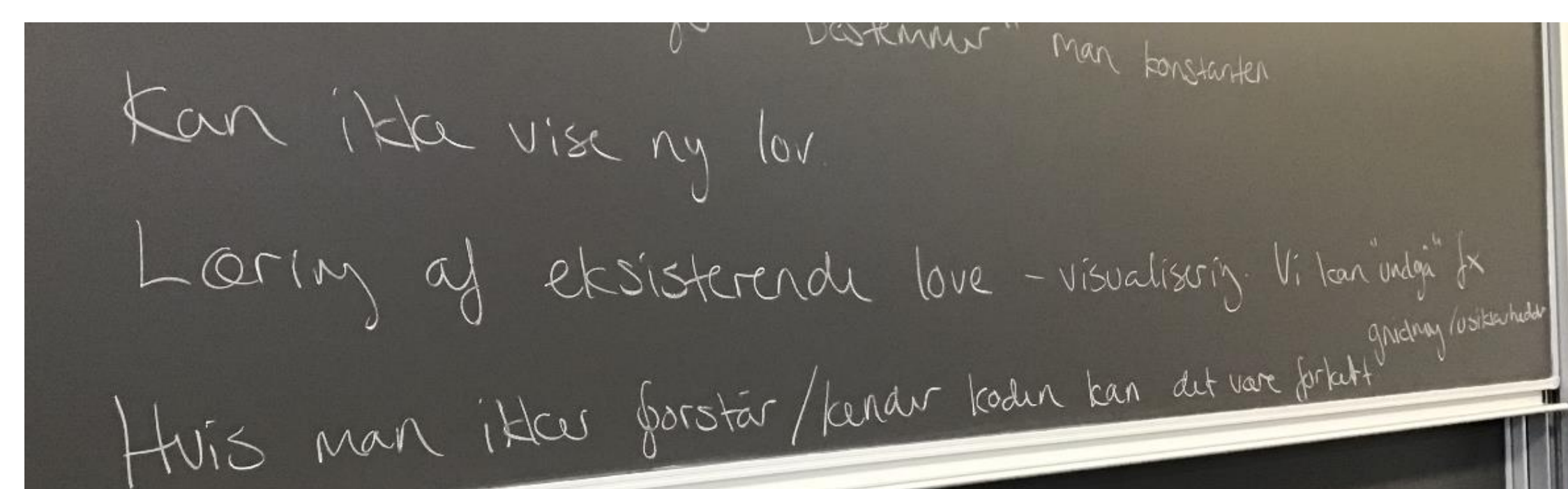
I dette forløb skal eleverne i høj grad selv kode deres simulering. Først arbejdes med stød i én dimension og til slut udvider eleverne modellen til to dimensioner. Elevernes eneste udgangspunkt er en "forkert model". Eleverne skal lave en model med et interface som nedenfor. De skal kode modellen, så den følger og viser fysikkens lov om impulsbevarelse.



## Mål med aktiviteten

Det fagfaglige mål med forløbet er først og fremmest at lære teorien omkring stød. Underformålet relaterer til CT og handler om at give eleverne et kritisk blik på simuleringer.

- Hvad kan man bruge en simulering til og hvad kan man ikke bruge en simulering til?
- Kan man stole på en simulering?



Uddrag fra opsamling på klassen.

## En forkert model

*Model 1* skal blot skabe tryghed med NetLogo og er særligt relevant hvis dette forløb er første gang eleverne møder programmet.

*Model 2* simulerer to bolde, der støder sammen. Når de støder sammen vendes boldenes retning. I koden ser det således ud:

```
ask turtles [
  if any? other turtles-here[
    set speed -1 * speed
  ]
]
```

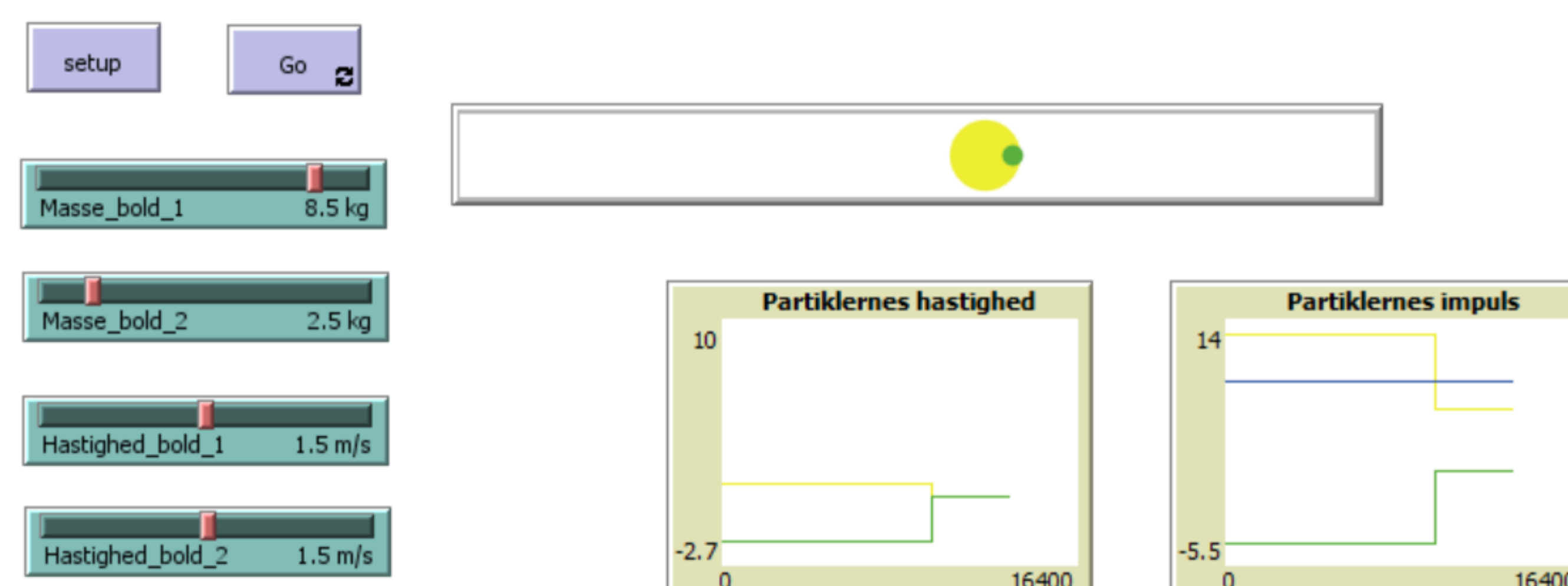
Dette er ikke, hvad der sker i virkeligheden. Formålet med denne forkerte model er at få eleverne til kritisk at overveje, hvorvidt simuleringer er til at stole på. Et hovedbudskab er, at en simulering ikke er bedre end den person, der har kodet simuleringen.

## Redigering af modellen

Eleverne skal selv redigere modellen, så den bliver rigtig – dvs. den skal afspejle impulsbevarelse. Et fuldstændigt uelastisk stød er lettest at kode, da der er en formel for hastigheden efter sammenstødet.

```
to støde-sammen
  let v1 [speed] of turtle 0
  let v2 [speed] of turtle 1

  ask turtles [
    if any? other turtles-here[
      set speed ([mass] of turtle 0 * v1 + [mass] of turtle 1 * v2) / (masse_bolid_1 + masse_bolid_2)
    ]
  ]
end
```



## Udvidelse af modellen

Der er flere forskellige muligheder for udvidelse.

Et forslag er at kode et elastisk stød, men det kræver lidt formelmanipulation at nå frem til at hastigheden  $v$  efter sammenstødet er givet ved

$$v = \frac{p + \Delta p}{m},$$

hvor  $p$  er impulsen inden sammenstødet,  $\Delta p$  er impulsændringen og  $m$  er boldens masse.

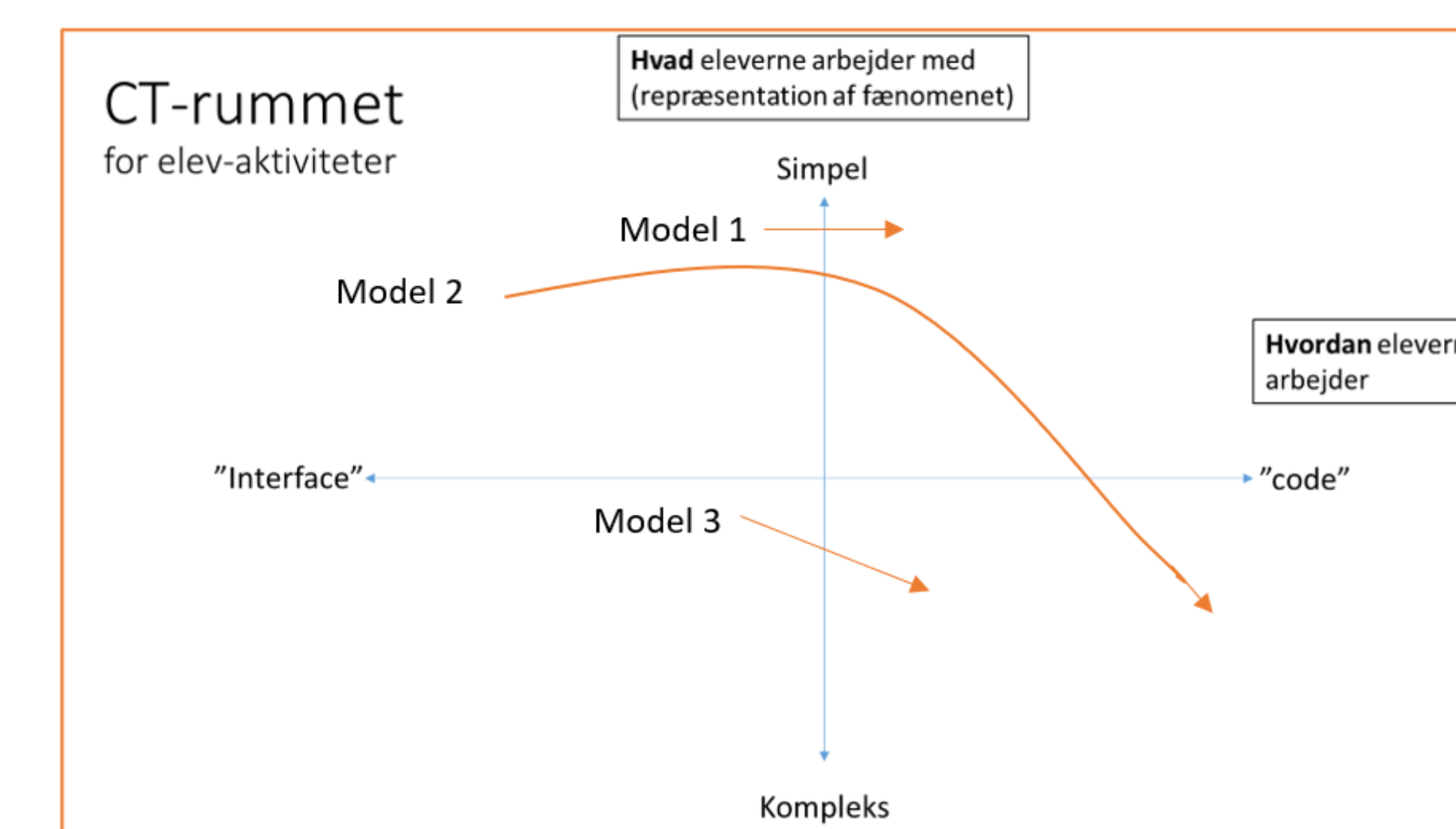
```
let impuls-ændring 2 * ([mass] of turtle 0 * [impulse] of turtle 1 - [mass] of turtle 1 * [impulse] of turtle 0) / ([mass] of turtle 0 + [mass] of turtle 1)

ask turtle 0 [set speed (impulse + impuls-ændring) / masse]
ask turtle 1 [set speed (impulse - impuls-ændring) / masse]
```

Man kan også vælge at udvide til stød i to dimensioner.

## Computational thinking

Forløbet er tilrettelagt så eleverne selv skal skrive kode. De skal ikke bare lave små justeringer – de skal kastes på dybt vand. Det at kode skal afmystificeres, så eleverne har gåpåmod overfor de krav til kodning, som muligvis vil møde senere i livet.



## Kreditering

NetLogo-modellen og undervisningsmaterialet er udviklet af **Mie Johannesen**, Viborg Katedralskole, i forbindelse med deltagelse i udviklingsprojektet Computational Thinking i Matematik og Naturfag i skoleåret '19/'20. Projektet køres i samarbejde mellem Danske Science Gymnasier og Center for Computational Thinking & Design, Aarhus Universitet.