

Übungsblatt 01

Aufgabe 03

Aufgabenstellung

Lernen Sie NetLogo kennen.

Erweitern Sie das Modell
„Ökosystem aus Pflanzenfressern“
im 3. Teil des Tutorials.

Was ist NetLogo?

- Programmierumgebung zur Modellierung von selbstorganisierenden adaptiven Systemen
- einfache Programmierung durch vorgefertigte Bedienelemente
- große Simulations-Bibliothek

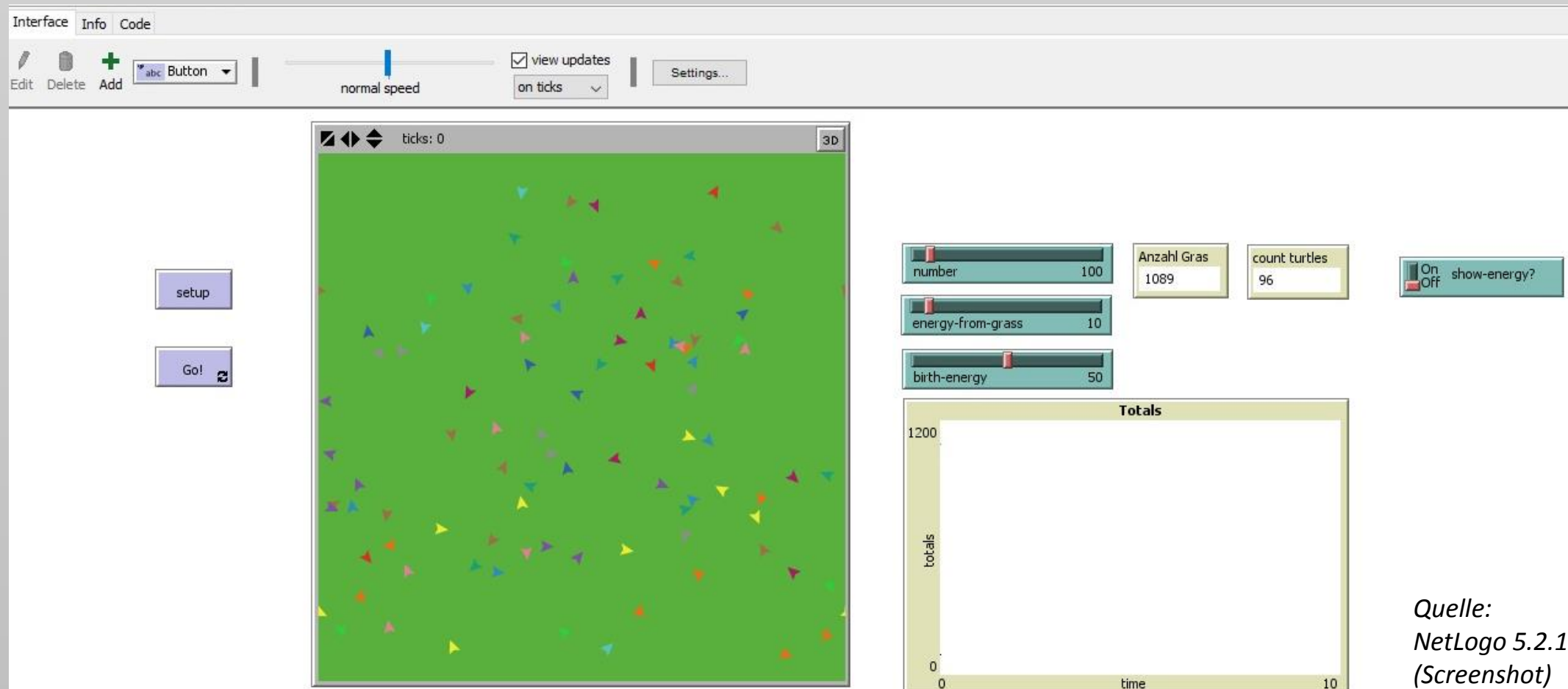
Modell „Ökosystem aus Pflanzenfressern“

Ausgangssituation aus Tutorial:

- 100 „Turtles“ (Pflanzenfresser, z.B. Schafe)
- ein großes Feld voller Gras (grüne Felder = „patches“)
- Schafe laufen in jeder Iteration (Tick) einen Schritt in zufällige Richtung
- jedes Schaf hat Energielevel
- Schaf frisst Gras → 10 mehr Energie, Feld wird schwarz
- Schaf läuft einen Schritt → 1 weniger Energie
- Schafe ohne Energie sterben
- Schafe mit Energie >50 gebären neues Schaf und geben 50 an dieses ab
- Gras wächst in jeder Iteration zu 3 % nach

Modell „Ökosystem aus Pflanzenfressern“

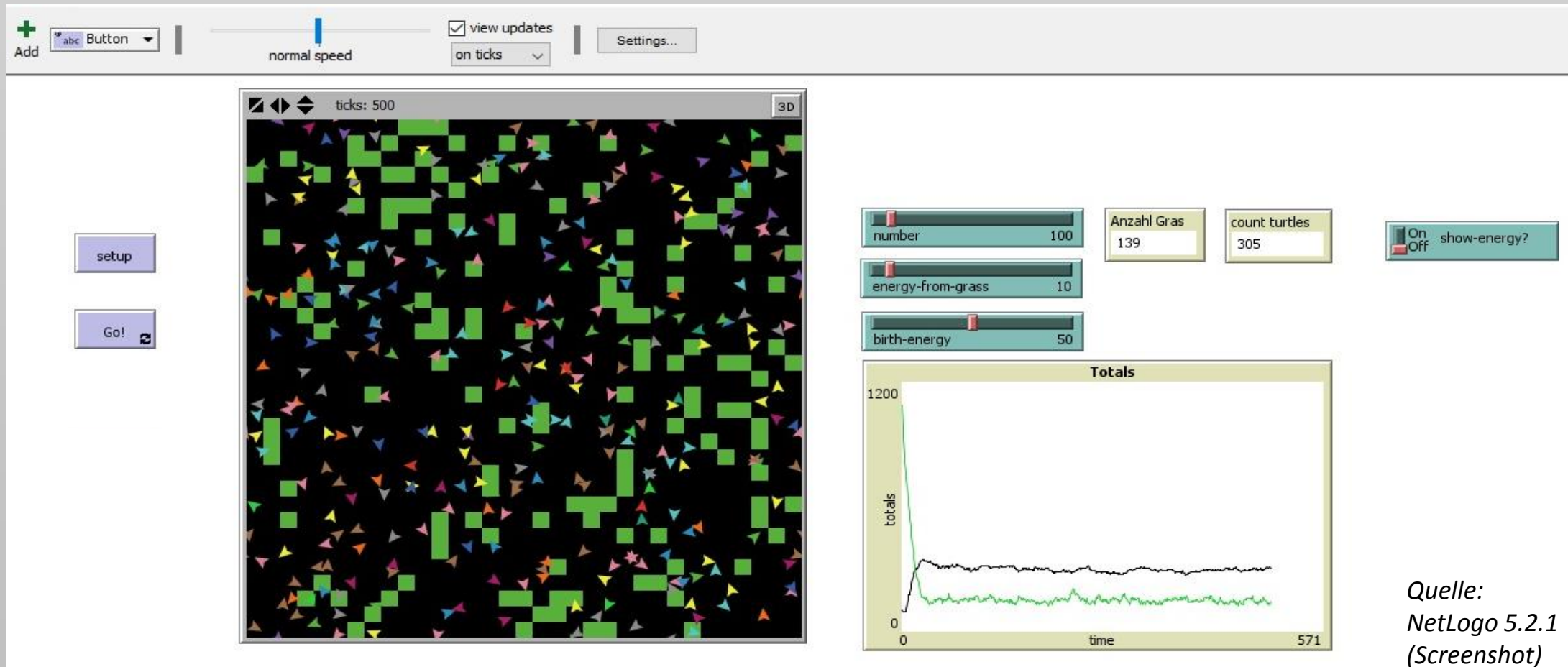
Ausgangssituation aus Tutorial:



Quelle:
NetLogo 5.2.1
(Screenshot)

Modell „Ökosystem aus Pflanzenfressern“

Ausgangssituation aus Tutorial (einmal ausgeführt, 500 Iterationen):

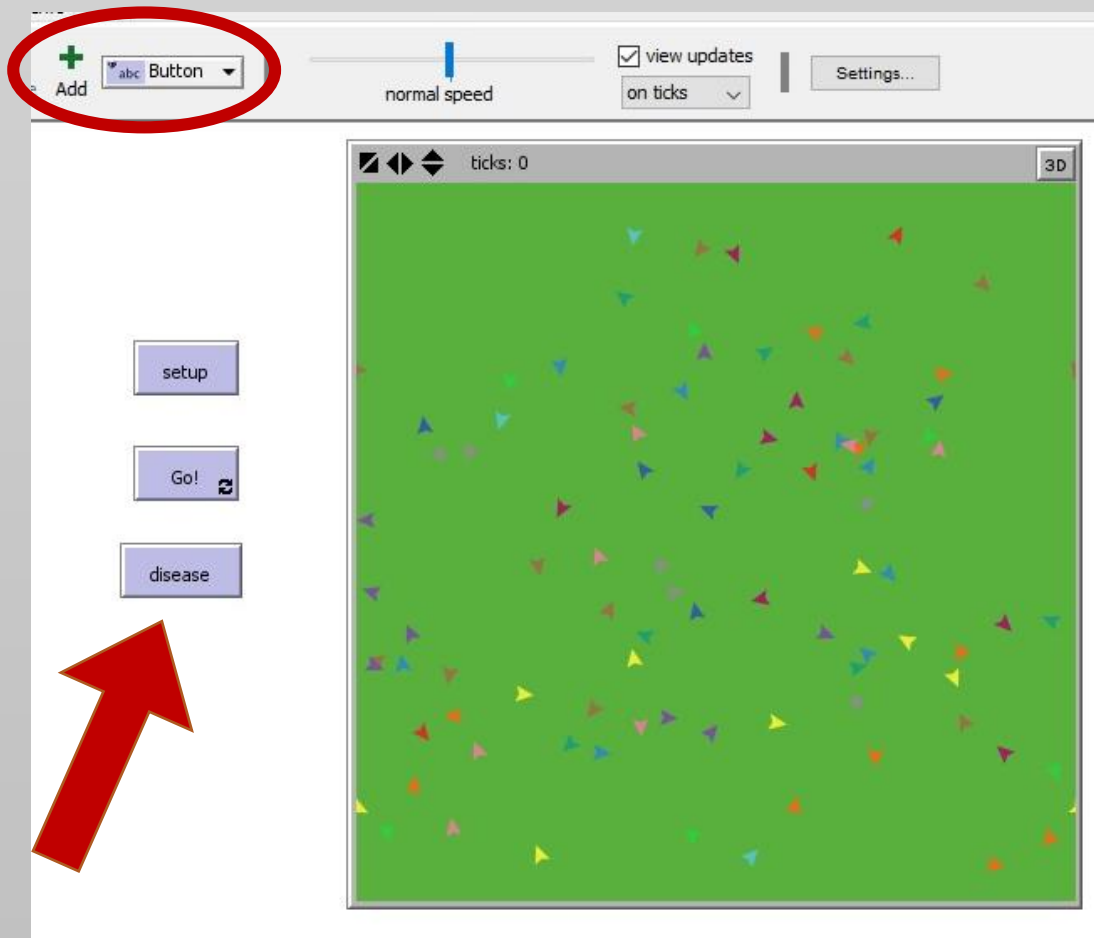


Aufgabenstellung

(a.1)

- Erstellen Sie einen Button, der eine Krankheit auslöst
- Krankheit tötet alle Schafe mit Energie < 30

Teilaufgabe (a.1) – Button mit Krankheit



to disease

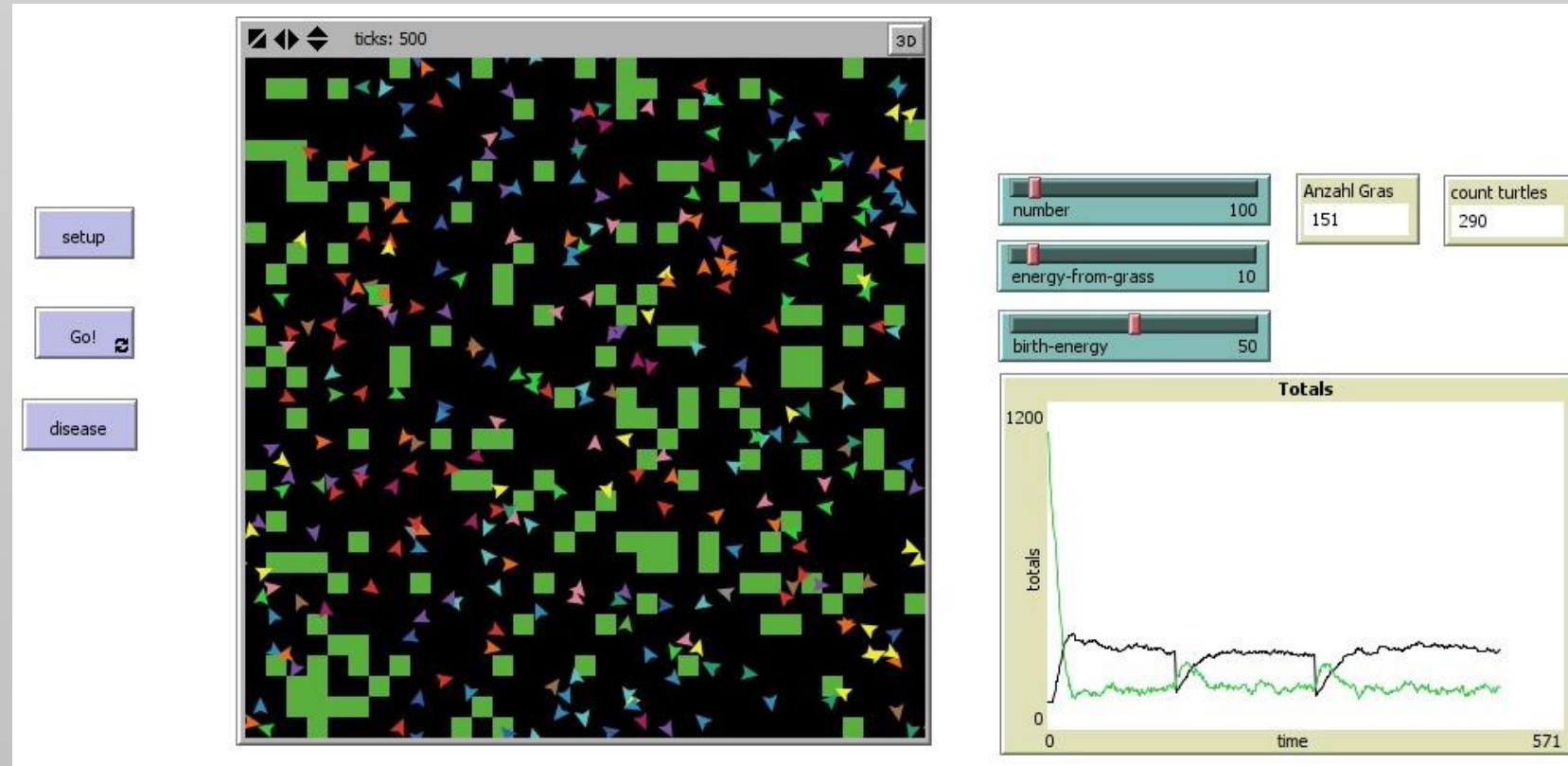
```
ask turtles [  
  if energy < 30 [ die ]
```

```
]   
end
```


Teilaufgabe (a.1) – Button mit Krankheit

Disease-Button-Auslösung

- Schafpopulation schrumpft sofort auf etwa die Hälfte
- Grasflächen steigen um etwa ein Drittel an (wird weniger gefressen)
- beide Werte pendeln sich nach kurzer Zeit wieder ein



Aufgabenstellung

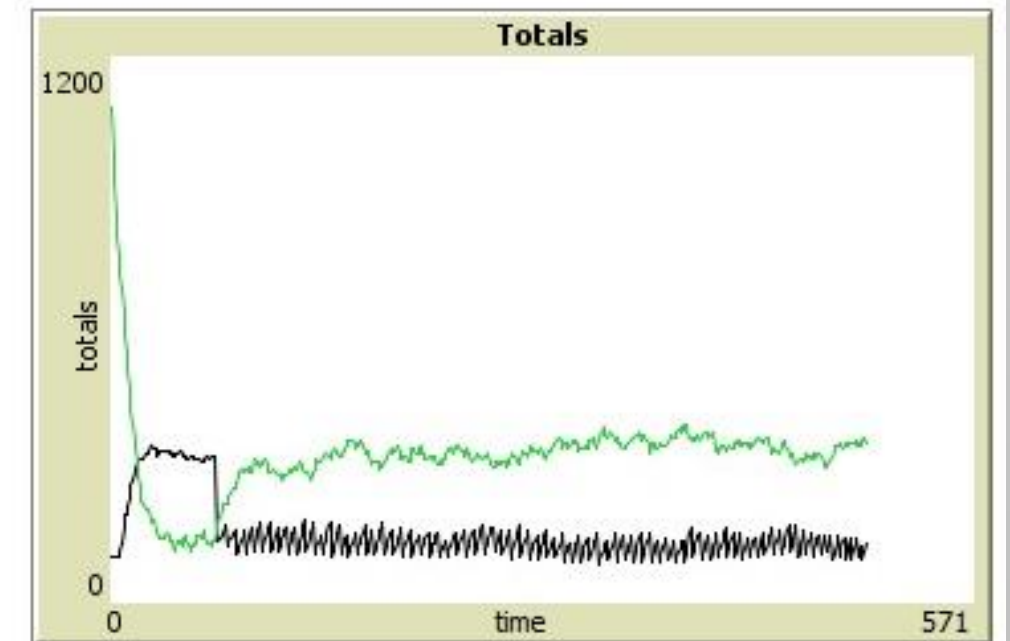
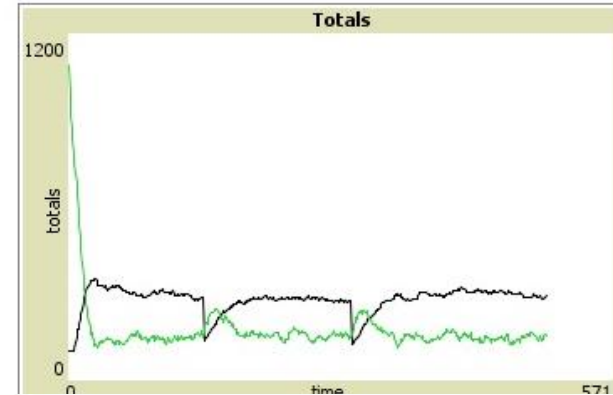
(a.2)

- Was passiert bei wiederholter Auslösung?

Teilaufgabe (a.2) – Button mit Krankheit

mehrmalige Button-Auslösung

- Einpendeln auf Ausgangszustand wird verhindert
- dauerhaft weniger Schafe (etwa ein Viertel so viele)
- dadurch mehr Gras (wird weniger gefressen)
- bei gleichmäßigem Drücken bleibt dieser Zustand erhalten



Aufgabenstellung

(a.3)

- Wie wirkt sich eine Änderung der 30 auf 40 aus?

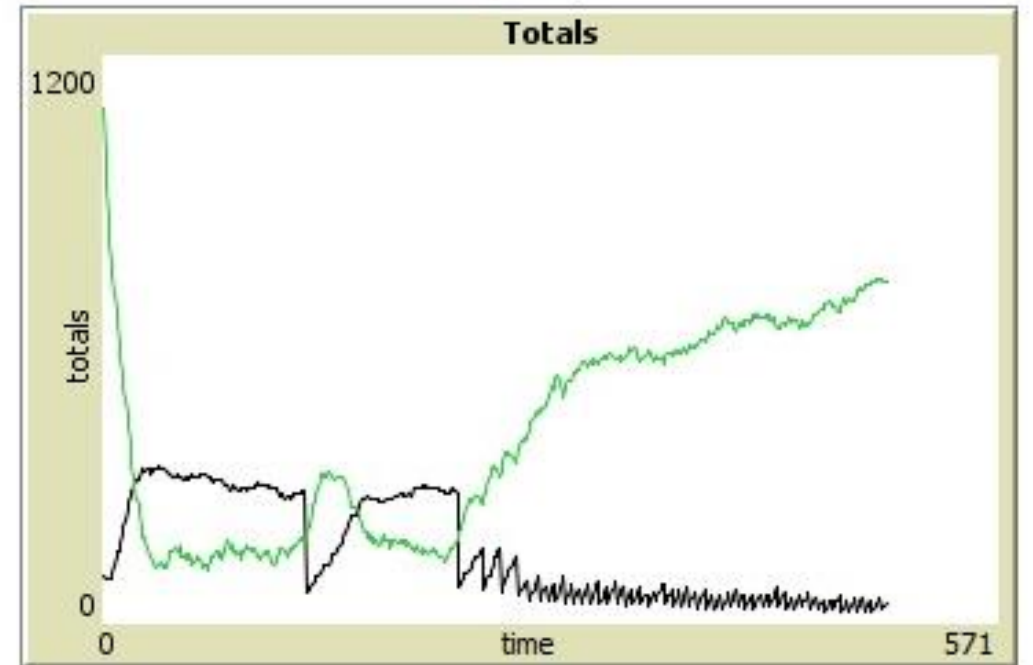
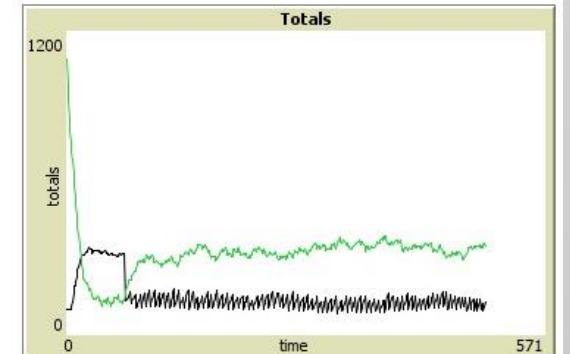
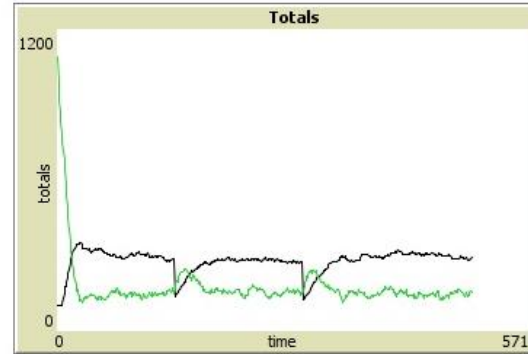
Teilaufgabe (a.3) – Button mit Krankheit

Krankheit aggressiver

- mehr Schafe sterben
- noch mehr Gras (weniger wird gefressen)
- bei mehrmaligem Auslösen gewaltigere Auswirkungen

to disease

```
ask turtles [  
  if energy < 40 [ die ]  
]  
end
```

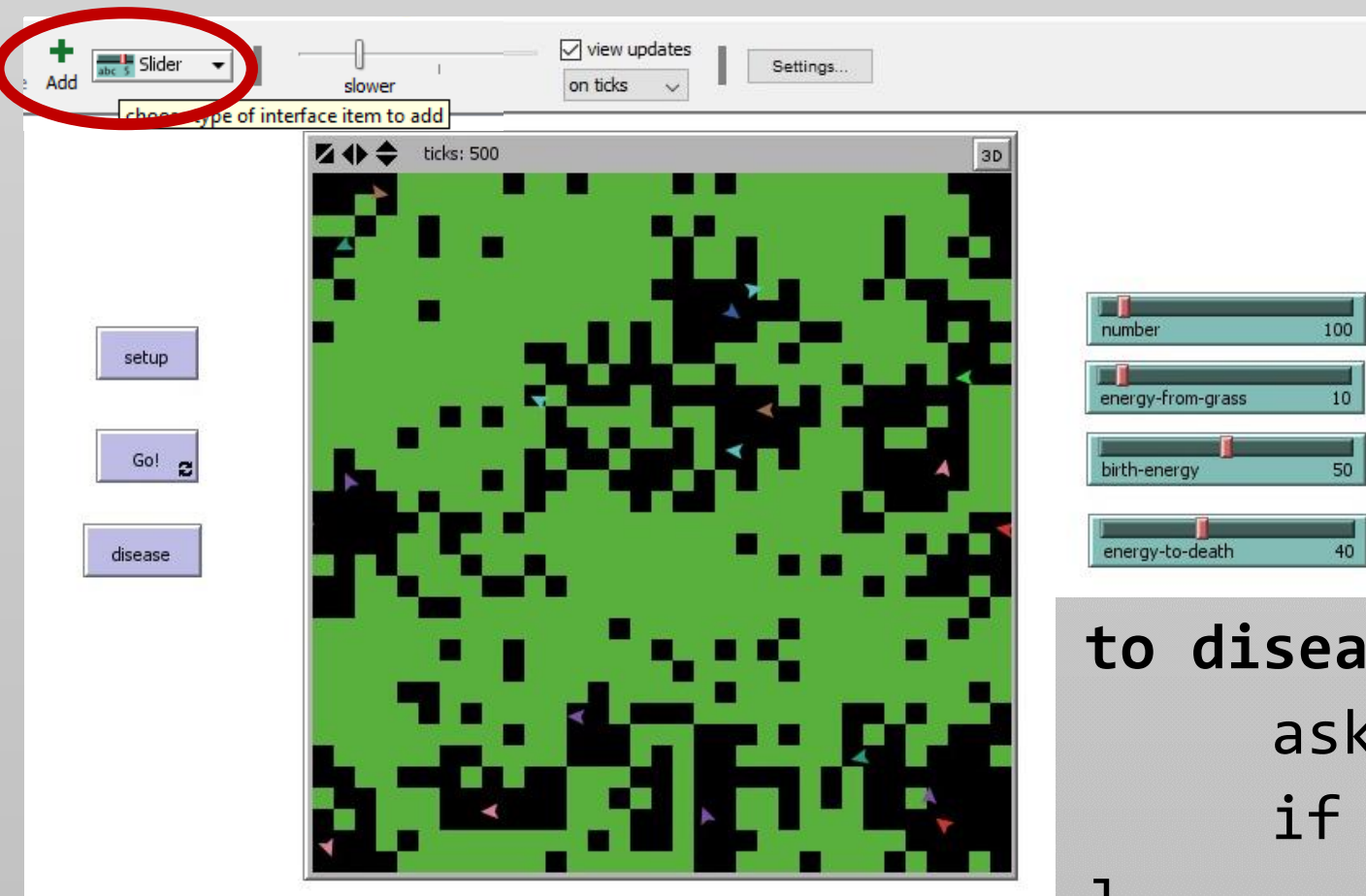


Aufgabenstellung

(b)

- Erstellen Sie einen Slider
- Aggressivität der Krankheit soll einstellbar sein

Teilaufgabe (b) – Aggressivitäts-Slider



to disease

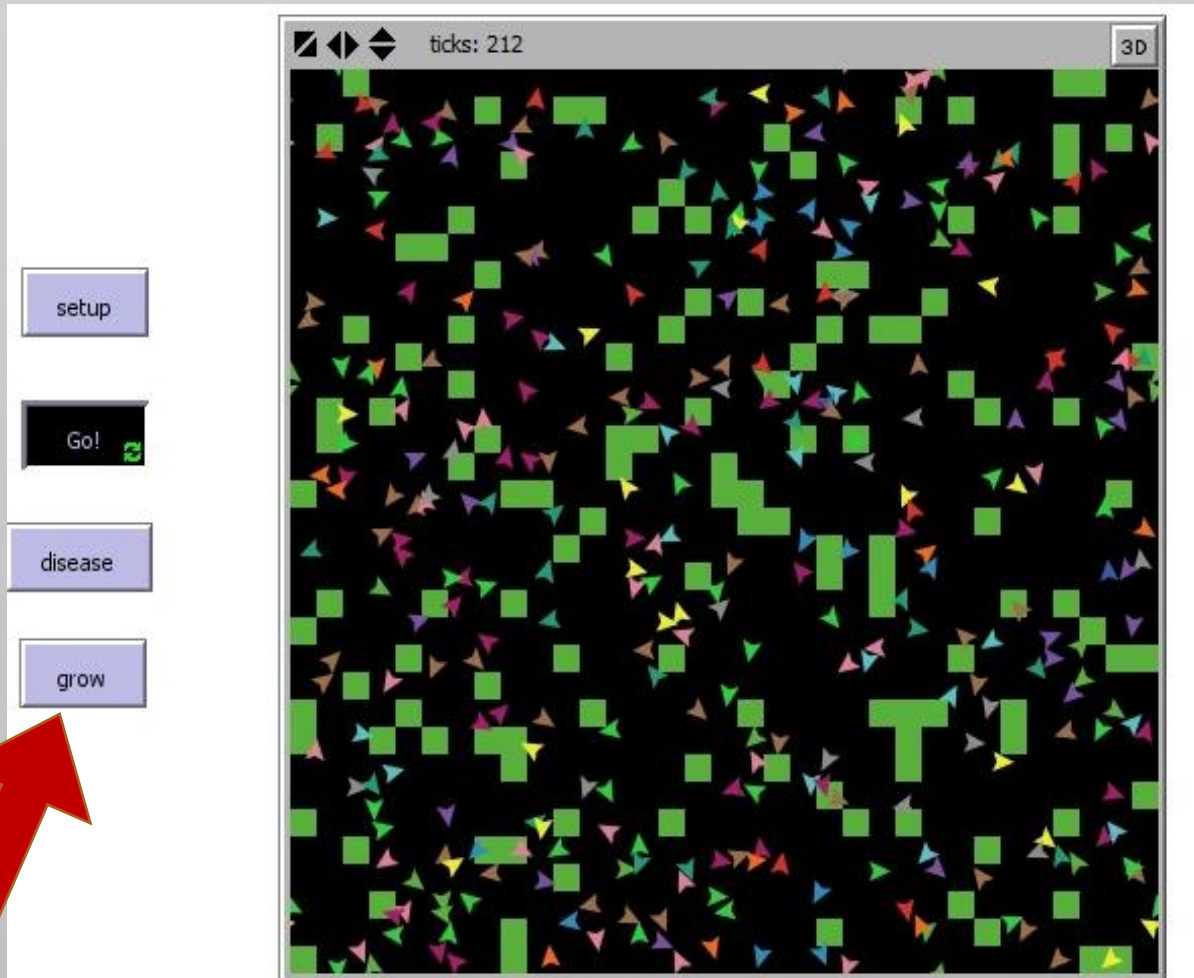
```
ask turtles [  
  if energy < energy-to-death [die]  
]  
end
```

Aufgabenstellung

(c.1)

- Erstellen Sie einen Button, der eine Regenzeit simuliert
- Bei Regenzeit wächst 50 % der abgegrasten Fläche sofort nach
- Auswirkung des unerwarteten Essensnachschubs auf die Schafspopulation

Teilaufgabe (c.1) – Button für Regenzeit

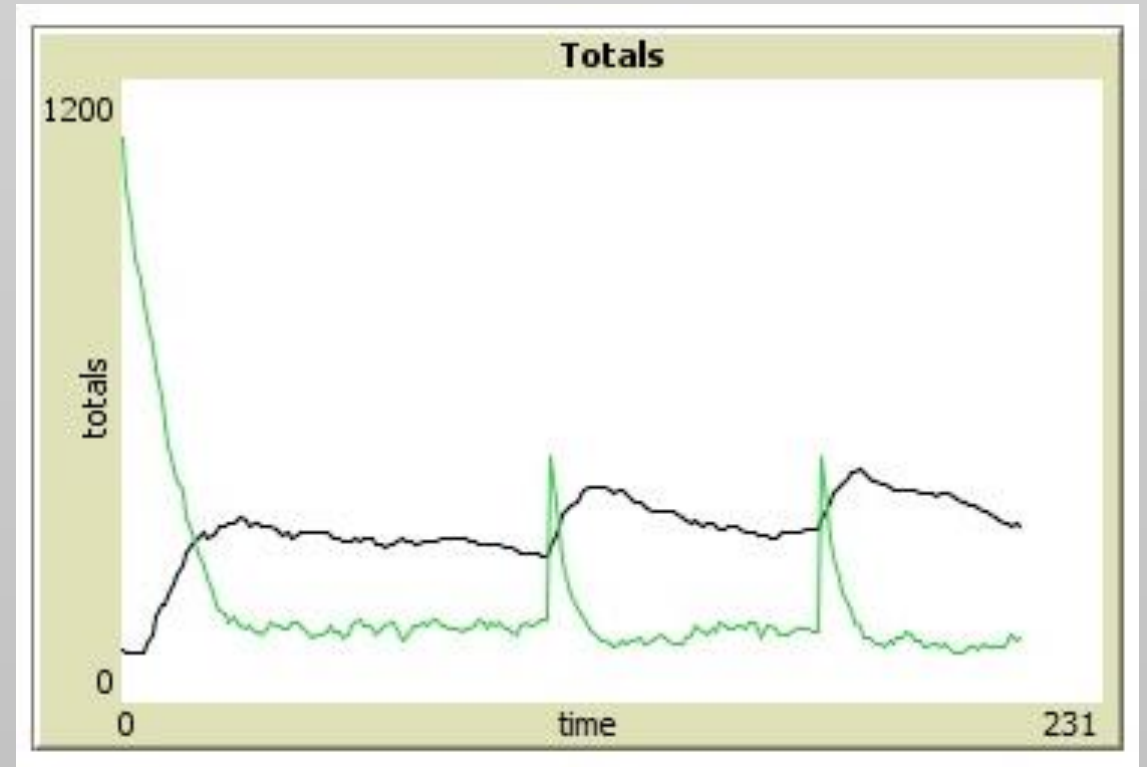


```
to grow
  ask patches [
    if pcolor = black [
      if random 100 < 50 [
        set pcolor green
      ]
    ]
  ]
end
```

Teilaufgabe (c.1) – Button für Regenzeit

Auslösung des Essensnachschubs:

- Population der Schafe steigt mit der Zeit etwas an
- Grasangebot sinkt schnell wieder ab
- Schafpopulation nähert sich langsam wieder dem „Ausgangswert“ an



Aufgabenstellung

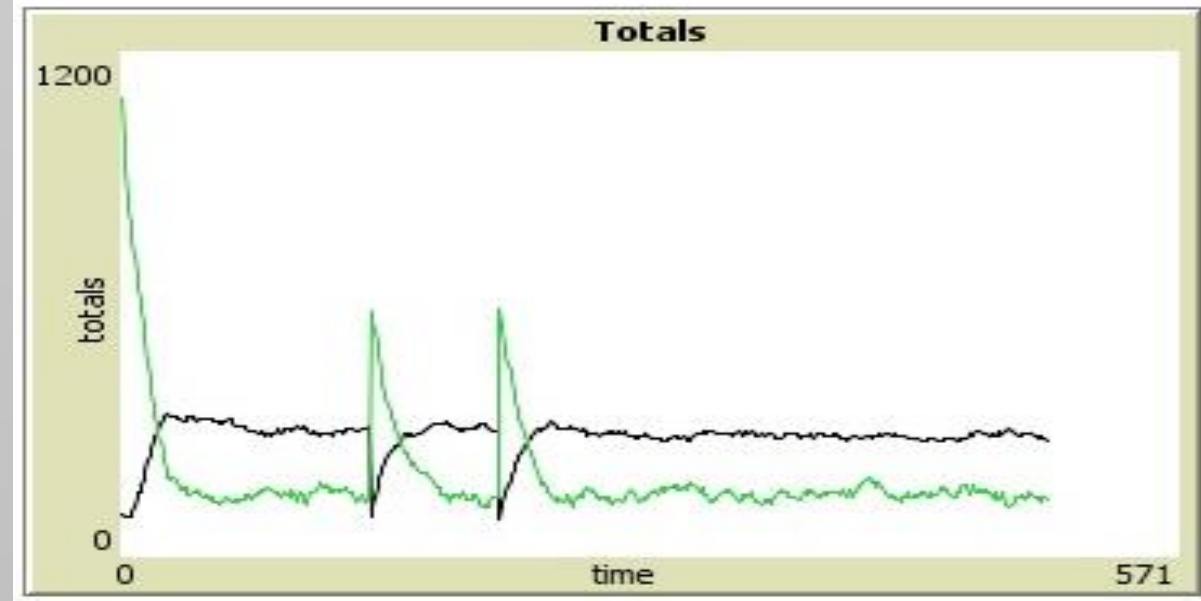
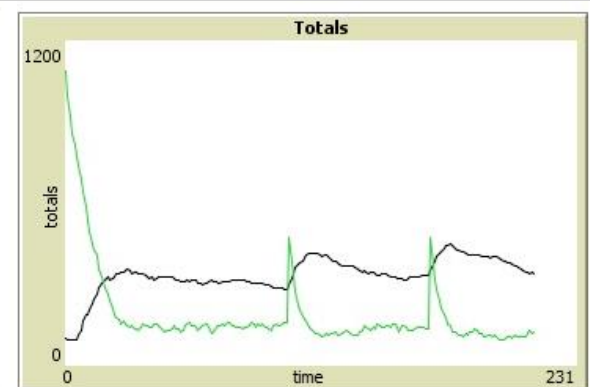
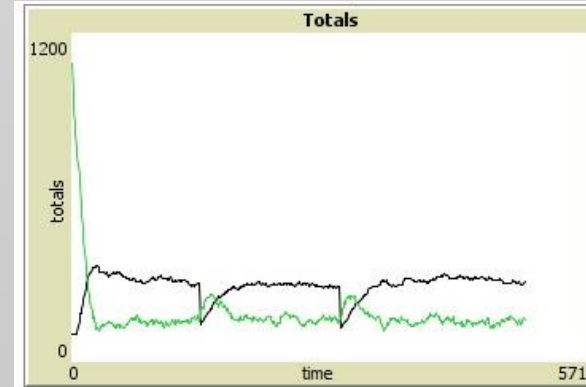
(c.2)

- Was passiert, wenn gleichzeitig Schafe erkranken und die Regenzeit Essensnachschub liefert?

Teilaufgabe (c.2) – Simultanität

Zusammenfall von Krankheit und Regenzeit

- Auswirkungen verstärken sich
- Schafpopulation pendelt sich schneller wieder ein (da mehr Gras vorhanden)
- Grasangebot pendelt sich langsamer wieder ein (da weniger Schafe fressen)



Übungsgruppe 02 sagt

Danke!