



# Engineering, computermodeller og teknologiforståelse

Slutrapport, 2022

**Engineer** the future

KØBENHAVNS  
PROFESSIONS  
HØJSKOLE **XP**



VIA University  
College

 **Teknologipagten**

Engineering, computermodeller og teknologiforståelse – slutrapport

Udgivet 2022

Forfattere: Mikkel Hjorth, Vibeke Schrøder, Keld Nielsen, Line Have Musæus og Mads Joakim Sørensen

Redaktion: Mads Joakim Sørensen

Grafik & layout: Grethe Kofoed

Slutrapporten offentliggøres af Engineer the Future i samarbejde med Københavns Professionshøjskole og VIA University College og er resultatet af pilotprojektet "*Engineering, computermodeller og teknologiforståelse*" med støtte fra Teknologipagten.

Tak til lærere og eleverne på Sankt Helene Skole, Gribskov kommune, Tingkærskolen, Odense kommune, Lille Egede Friskole, Slagelse kommune, Holmegaardskolen afd. Fensmark, Næstved kommune, Hornbæk Skole, Helsingør kommune, Utterslev Skole, Københavns kommune for at praksisafprøve projektets didaktiske ideer og bidrage med feedback til denne slutrapport.

Tak til de øvrige projektdeltagere Maria Damlund, Simon Olling Rebsdorf, Jesper Heidemann Langhoff og Jonas Ørbæk Hansen for udvikling af prototypeforløb og for frugtbare didaktiske diskussioner, der har bidraget til at kvalificere projektet.

# Resumé

Projektet *Engineering, computermodeller og teknologiforståelse* udviklede og testede undervisning, som kombinerer de to kendte fagområder engineering og teknologiforståelse under inddragelse af faglig viden fra et eller flere af grundskolens STEM-fag.

I projektet blev udarbejdet og afprøvet tre undervisningsforløb. I hvert forløb blev eleverne præsenteret for en konkret udfordring fra deres omverden, som de skulle finde løsning på ved at arbejde med en simpel, men fagligt relevant, computermodel, der var udarbejdet specifikt til forløbet.

Undervisningsforløbene blev startet op i 11 klasser (fra 6. til 8. klasse), der alle – bortset fra tre, der måtte standse undervejs på grund af ydre, praktiske omstændigheder – gennemførte et forløb og bidrog på forskellig vis til dataindsamlingen.

***Projektet har vist, at det er muligt at arbejde projektorienteret i grundskolen med en kombination af engineering og teknologiforståelse under inddragelse af faglig viden fra et eller flere STEM-fag.***

Engineering-design-processen (EDP) fungerede hensigtsmæssigt som didaktisk ramme for den faglige integration. Eleverne blev generelt motiverede af at arbejde med de autentiske problemfelter og med de konkrete udfordringer. EDP-tilgangen reducerede behovet for stilladsering og gjorde dermed lærerens opgave mere overskuelig. Nogle af de deltagende lærere fandt desuden, at EDP gør det nemmere at inddrage naturfag som en integreret del af en designproces.

Nye didaktiske ideer med centrale elementer fra kompetenceområderne fra teknologiforståelse, *computational tankegang* og *teknologisk handleevne*, blev i en grundskolekontekst afprøvet med positive resultater. Eleverne arbejdede konkret og i detaljer med simulering, modellering, vurdering af modeller, programmering, brug af data og præsentation af data.

Det anvendte tekstbaserede programmeringssprog (NetLogo) har vist sig velegnet til at inddrage computermodeller i en problembaseret undervisning i udskoling. Eleverne læser og tolker computerkoden uden større besvær. Desuden laver eleverne meningsfulde ændringer i koden og dermed i modellen. 88 % af eleverne rapporterer, at de selv ændrede i koden. I en klasse har flertallet af eleverne sagt til læreren, at de ønsker at arbejde videre med NetLogo i andre undervisningsmæssige sammenhænge.

I forbindelse med kodning i NetLogo udviser nogle elever en motivation for selv at rette fejl og lære af fejl, som de rapporterende lærere ikke har set før. Tilsvarende har nogle elever også udvist motivation for at programmere i et tekstbaseret programmeringssprog, hvilket var overraskende for deres lærere.

Hver computermodel blev udarbejdet på grundlag af relevant matematisk og naturfaglig viden. På den ene side motiverer den direkte anvendelse af viden eleverne for selv at finde ny faglig viden, så de kan forbedre deres model. På den anden side kan det faglige indhold i computermodellen nemt blive for komplekst for eleverne, og der forestår en opgave med at afgrænse og udvælge faglige elementer, der med rimelighed kan indgå i forløb af denne type.

Nogle lærere rapporterede, at de havde vanskeligt ved at gennemføre alle dele af forløbet, fordi de manglede tid. Det skyldes til dels, at afprøvningen skulle foregå over kun fem uger i foråret 2022, men det peger muligvis også på et behov for at begrænse omfanget af forløbene.

Mens de fleste elever hurtigt kom i gang med programkoden i NetLogo, rapporterer flere lærere om usikkerhed over for at skulle undervise i et nyt og for dem ukendt programmeringssprog, om end der i den generelle tilbagemelding fra lærere tegner sig et billede af, at tekstprogrammering i NetLogo var nemmere, end de havde forventet. I forbindelse med projektet blev det ikke afklaret, præcis hvilke NetLogo-kompetencer der er nødvendige for lærerne.

# Indhold

Indledning .....	4
Eksempel på et undervisningsforløb .....	6
De andre undervisningsforløb .....	11
Projektets struktur og deltagere.....	13
Projektplan .....	13
Organisering .....	13
Projektdeltagere.....	14
Projektskoler .....	14
Fremgangsmåde .....	15
Erfaringer og konklusioner på tværs af de tre undervisningsforløb .....	17
Potentialet ved at kombinere engineering med brug af computermodeller .....	17
Elevernes udvikling af kompetencer inden for modellering og programmering .....	20
Behov for stilladsering af eleverne og kompetenceudvikling af lærerne.....	23
Tværfaglige sammenhænge - STEM-fagene og teknologiforståelse.....	24
Fremadrettede anbefalinger .....	26
BILAG .....	28
Fokusgruppeinterviews .....	28
Analyse af spørgeskemaer.....	31
Analyse af elevartefakter .....	35
Referencer .....	39

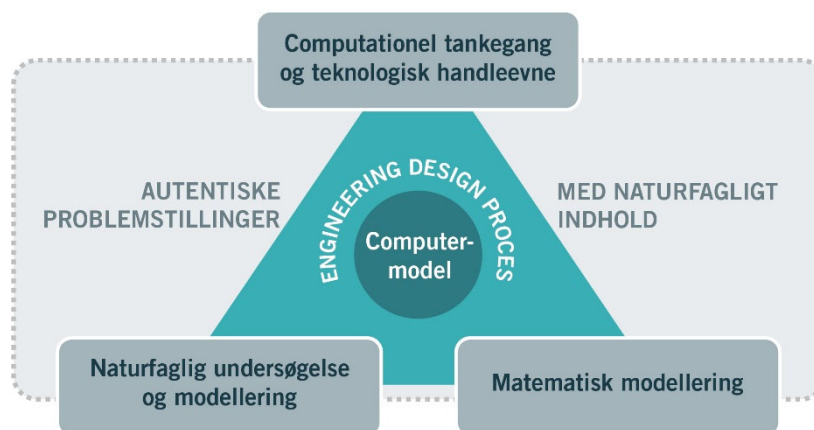
# Indledning

Formålet med projektet *Engineering, computermodeller og teknologiforståelse* var at udvikle og teste undervisningsforløb, der kombinerer de kendte fagområder engineering og teknologiforståelse under inddragelse af faglig viden fra et eller flere af grundskolens STEM-fag.

I projektet er der udarbejdet og afprøvet tre undervisningsforløb, hvor eleverne bliver præsenteret for en konkret udfordring fra deres omverden, som de skal finde løsninger på ved at arbejde med en fagligt relevant computermodel.

Der er ikke noget nyt i, at elever får en udfordring, hvor de både skal udføre undersøgelser og designe løsninger på en konkret udfordring. Det nyskabende i forløbet er inddragelsen af en computermodel og tilhørende elevaktiviteter, både som en del af designprocessen og i forbindelse med løsningen.

Hvert forløb kombinerer altså en *engineering-design-proces* – gennem præsentation af en udfordring og krav til en løsning – med *teknologiforståelse*, fordi eleverne i forløbet arbejder med en *computermodel*. Eleverne forbedrer selv modellen ved at ændre direkte i computerkoden. Modellen producerer data, som eleverne bruger i deres analyse af udfordringen og deres tests af mulige løsninger. Dermed arbejder eleverne med kompetenceområderne undersøgelse og modellering i naturfag og matematik, samt med to ud af fire kompetenceområder fra teknologiforståelse: computationel tankegang og teknologisk handleevne.



Figur 1. Modellen illustrerer hvordan elever kan udvikle kompetencer inden for hhv. naturfag, matematik og teknologiforståelse gennem en engineering design proces med udgangspunkt i autentiske problemstillinger med naturfagligt indhold, og med inddragelse af en computermodel som en del af løsningen.

Denne kombination af elevaktiverende og problemorienteret projektarbejde (engineering) med programmering, modellering og brug af data (teknologiforståelse) er – så vidt vi er orienteret – ikke før afprøvet i grundskolen. Trods tidspress og de løse ender, der altid optræder i et pilotprojekt, gennemførte 8 ud af de 11 deltagende klasser som planlagt et af de tre undervisningsforløb.

Baseret på spørgeskemasvar fra lærere og elever, fokusgruppeinterviews med seks lærere og vurdering af slutprodukterne fra tre tilfældigt valgte klasser vurderer vi, at projektet har leveret ”proof of concept” for, at elevgrupper kan gennemføre en engineering-proces, hvor de gennem brug af viden og færdigheder fra naturfag og matematik, samt tilpasning af computermodellen, kommer frem til en løsning på deres udfordring.

**Dermed har projektet vist, at det er muligt at arbejde projektorienteret i grundskolen med en kombination af engineering og teknologiforståelse under inddragelse af faglig viden fra et eller flere STEM-fag.**

(digital) teknologiforståelse og et eller flere STEM-fag. For at illustrere de nye didaktiske ideer har vi i nedenstående beskrevet ét af de tre nyudviklede undervisningsforløb udførligt.

Dernæst beskrives projektets omfang og organisering. Til slut gives en oversigt over det potentiale, der er identificeret gennem de erfaringer, projektet har givet, samt muligheder og udfordringer for videre arbejde med de afprøvede didaktiske ideer. Som bilag findes de forskellige analyser, der har dannet grundlag for vurderingen af projektets resultater.



## Eksempel på et undervisningsforløb

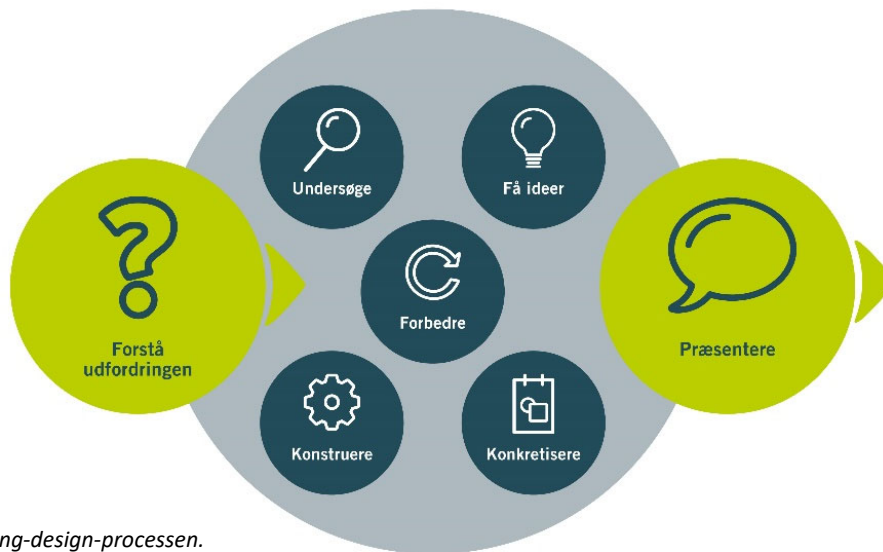
Et af de tre undervisningsforløb havde madspild som tema. Forløbet havde en estimeret varighed på ca. 14 lektioner og er afprøvet i en 6.-klasse, en 7.-klasse og to 8.-klasser. De indgående fagligheder var – ud over engineering og teknologiforståelse – biologi og matematik. Beskrivelsen af undervisningsforløbet er et "running example" på hvordan de nye didaktiske ideer blev gjort konkrete gennem et undervisningsforløb. Der blev lavet yderligere to undervisningsforløb med udgangspunkt i hhv. fysik/kemi og geografi.

### Udfordringen: Kan vi reducere madspild?

Som introduktion til forløbet diskuterede eleverne, hvorfor madspild er et problem, og hvordan en del af problemet skyldes, at store mængder af madvarer bliver kasseret, når sidste salgsdato er overskredet. Kassationen sker selvfølgelig for at undgå, at folk bliver syge af at spise fordærvet mad, men mange fødevarer kan fint spises, selv om holdbarhedsdatoen er overskredet.

Eleverne blev præsenteret for følgende udfordring:

*Design en teknologisk løsning (en prototype), som i hverdagen kan anvendes til at reducere madspild, ved at vejlede forbrugeren (dig) om fødevarers anvendelighed, efter datomærkningen er overskredet. Prototypen skal nemt kunne hjælpe dig til at vurdere, om en fødevarer er sundhedsmæssigt forsvarlig at spise.*



Figur 2. Engineering-design-processen.

### Engineering som ramme

Forløbet er planlagt ud fra engineering-design-processen (se figur 2). Eleverne stilladseres fra begyndelsen til at være nysgerrige og stille spørgsmål til problemfelt og udfordring. Både for at belyse udfordringen og for at afdække, hvilken viden der vil være relevant for deres design af en løsning. De syv delprocesser er med til at understøtte lærerens planlægning og elevernes arbejde med systematik og iterationer i en designproces, så eleverne ender med at præsentere en løsning på udfordringen.

### Undersøgelser

I engineering arbejder eleverne generelt med at forstå udfordringen og de bagvedliggende mekanismer ved selv at udføre naturfaglige undersøgelser. I dette tilfælde var det oplagt at undersøge forskellige fødevarer og relevante forhold vedr. mikroorganismers vækst, hvilket er biologisk viden, som er central for at kunne designe en løsning.

Undervejs konkluderede eleverne, at de ikke kunne få tilstrækkeligt med data fra deres biologiske undersøgelser. Det er både tidskrævende og omfangsrigt, og så er sikkerhed ved håndtering af mikroorganismer en begrænsning for, hvad de kunne lave af eksperimenter i grundskolen. Det begrænser elevernes mulighed for at opsamle data fx gennem undersøgelser med mikroskop.

### Computermodellen

For at supplere eller erstatte de naturfaglige undersøgelser fik eleverne adgang til en simpel computermodel, der kan simulere cellevækst på et næringsfyldt underlag. Modellen gav dem helt nye muligheder for at udføre undersøgelser – herunder at teste hypoteser – som de normalt ikke har mulighed for i undervisningen. Elevundersøgelserne og computermodellen er designet ud fra CMC-princippet, som integrerer kodningsaktiviteter (Coding) med modellering (Modeling) og faglige sammenhænge (Content) (Musaeus & Musaeus, 2019).

Der er ikke noget nyt i, at elever får en udfordring, hvor de både skal udføre undersøgelser og designe løsninger på en konkret udfordring. Det nyskabende i forløbet er inddragelsen af en computermodel og tilhørende elevaktiviteter, både som en del af designprocessen og i forbindelse med løsningen.

Den kode, der ligger bag modellen, er skrevet i NetLogo (Wilensky, 1999), som er et intuitivt forståeligt, tekstbaseret programmeringsmiljø. Eleverne brugte modellen til at simulere cellevækst, parallelt med at de kunne se, læse og ændre i computerkoden. Når eleverne havde ændret i koden, kunne de med det samme teste ændringens effekt på simuleringen.





## Om valg af Netlogo

Lærernes og elevernes arbejde med model og modelkode er baseret på programmeringsmiljøet NetLogo (se <https://ccl.northwestern.edu/netlogo>).

NetLogo er et tekstbaseret programmeringssprog, der er designet, så det er forholdsvis simpelt at bruge for både elever og lærere. Det er muligt at komme i gang med at programmere uden at kunne overskue hele sproget og syntaksen i detaljer.

NetLogo er desuden et såkaldt agentbaseret programmeringssprog. Det betyder, at udgangspunktet for konstruktionen af en computermodel er en eller flere typer af agenter, der optræder i modellen.

Afhængigt af, hvilket fag der arbejdes med, og hvad det er for et fænomen, der skal modelleres, kan agenter f.eks. være atomkerner, ioner, fugle, får, borgere, molekyler, penge, etc.

Det centrale i at programmere en agentbaseret model består i at definere agenterne og udstyre dem med egenskaber og adfærd, der definerer, hvordan de ser ud og opfører sig, samt hvordan de interagerer med hinanden og deres omgivelser.

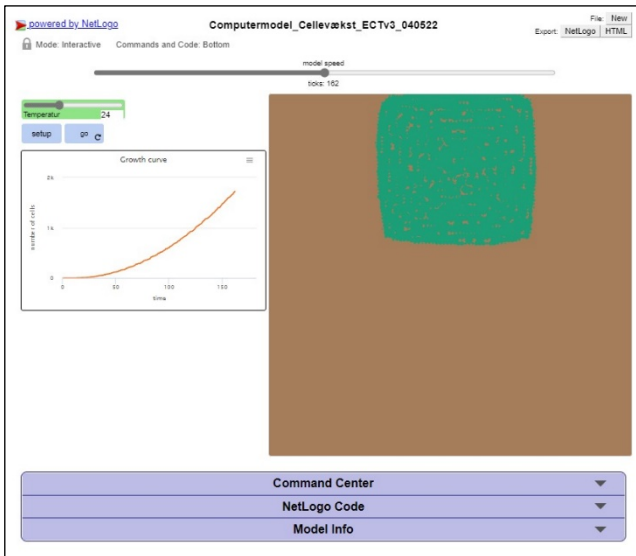
Egenskaber kan f.eks. være agentens størrelse, udseende, farve og position. Adfærd kan f.eks. være et bevægelsesmønster, hvordan der træffes beslutninger, frastødning eller tiltrækning eller henfald, hvis agenten er en atomkerne. Herved skabes måske to nye typer af agenter.

Agentbaseret modellering (ABM) har den styrke, at programmeringen lægger op til intuitiv identifikation mellem agenterne på den ene side, og den, der skaber eller modificerer agentens udseende eller opførsel, på den anden. Denne egenskab ('Jeg forstår godt, hvorfor agenterne opfører sig, som de gør') er med til at give NetLogo en lav indlæringsstærskel med hensyn til at bruge, forstå, ændre og skabe nyt i modellen, og dermed opbygge et tilhørsforhold til den.

Agenterne er håndgribelige og konkrete, og det er ofte lettere for eleverne at forholde sig til og arbejde med end f.eks. objekter og klasser i objektorienteret programmering. Med ABM kan store, komplekse fænomener altså reduceres til et spørgsmål om, hvordan de enkelte agenter agerer i bestemte situationer.

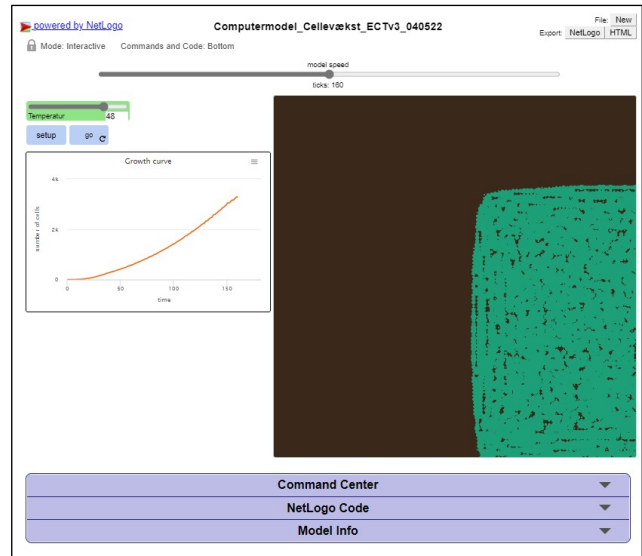
Efter en kort introduktion til brugen af computermodellen (interfacet - se billede 1 og 2), fandt eleverne i grupper selv ud af, hvordan modellen er opbygget. De lærte modellen at kende ved at gennemføre en række undersøgelser – nogle gange som hypotesetests. Processen anskueliggjorde sammenhænge mellem deres fysiske undersøgelser og de modelleringer, de kunne udføre med computeren.

Den udleverede computermodel var meget simpel, med det formål, at eleverne på egen hånd indså, hvilke begrænsninger og forenklinger modellen indeholdt, således at de blev motiveret til selv at videreudvikle modellen ved at ændre i koden.



Billede 1. Computermodellens interface 1.

Et eksempel på den computermodel, som eleverne får adgang til, og som viser mikroorganismers vækst på en fødevarer over tid. Grafen viser udviklingen ved 24 grader som funktion af tiden. Grafikken til højre giver en visuel formidling af samme udvikling. Fødevarer er det brune felt, mikroorganismer de grønne.



Billede 2. Computermodellens interface 2.

Billedet viser samme simulering af mikroorganismers vækst, men efter at en elevgruppe har ændret temperaturen til 48 grader. Væksten af mikroorganismer er fordoblet pga. temperaturstigningen, hvilket fremgår af både graf og grafik.

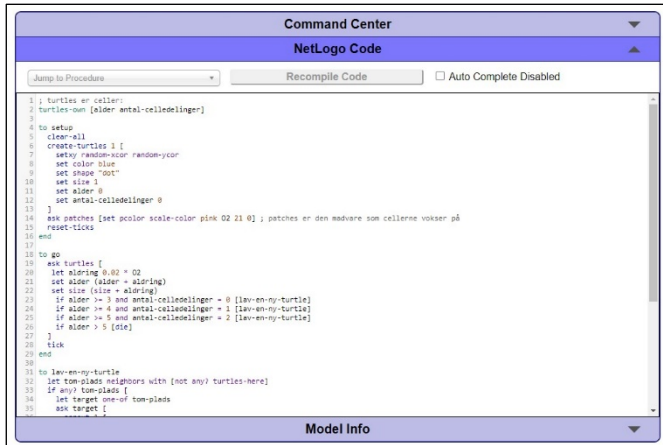
### Use-Modify-Create

Et andet didaktisk nybrud ved denne type undervisningsforløb er tilgangen Use-Modify-Create (Lee et al., 2011), der her indgår som en del af elevernes engineering-proces. Det indebærer, at eleverne – ud over at anvende computermodellen til undersøgelser (*use*) – også ændrede og tilpassede koden, der styrer modellen (*modify*), og – for nogle grupper – selv redesignede koden til et nyt formål (*create*).

I den udleverede model for bakterievækst var det kun muligt for eleverne at undersøge sammenhængen mellem temperatur og bakterievækst. Men langt de fleste elevgrupper modificerede koden, så de også kunne undersøge vækstens afhængighed af lysindfald og tilstedeværelsen af oxygen.

### Brug af data

Eleverne kunne altså ændre modellen ved hjælp af relevant input (data). Som det fremgår af billede 3 og 4 producerer modellen også output (data, vist på en graf), som eleverne dels skulle tolke, dels skulle bruge for at designe deres løsning.



Billede 3. Kode i NetLogo. Billedet viser, hvordan koden ser ud for eleverne i Netlogo. I eksemplet har en elevgruppe redesignet computermodellen ved bl.a. at ændre variabel, så det ikke er temperaturen, men mikroorganismernes adgang til oxygen i atmosfæren, som modellen behandler.



Billede 4. Computermodellens interface efter elevændringer. Billedet viser resultatet en elevgruppes redesign af computermodellen. De har tilpasset det grafiske udtryk, så farven minder om den kødpølse, de har valgt som fødevarer. Gruppen er også på vej til at redesigne computermodellens adfærd, så den viser mikroorganismers vækst som funktion af oxygenindholdet i luften omkring fødevarer.

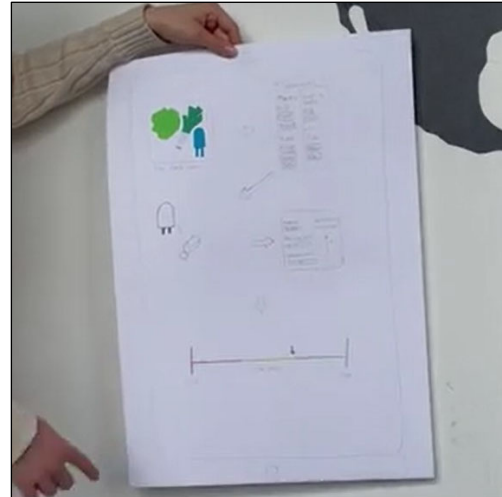
Eleverne blev gennem hele forløbet stilladseret til at arbejde iterativt med engineering-delprocesserne – få ideer, konkretisere, konstruere og forbedre – idet de løbende havde behov for at gå tilbage til computermodellen for at ændre og forbedre deres løsning. Computermodellen blev udviklet, så den bidrog med viden om mikroorganismers vækst på forskellige fødevarer og under forskellige forhold, alt efter hvilken viden elevgruppen fandt var relevant for forbrugeren og modellen blev dermed en central del af deres løsning på engineering-udfordringen.

### Et eksempel på en gruppes løsning

Mange – muligvis alle – grupper nåede frem til at foreslå en brugerrettet app til at løse problemet med vurdering af madvarens kvalitet, efter at sidste holdbarhedsdato er overskredet. Billede 5 og 6 er to screenshots fra en gruppes fremlæggelse af deres løsning.



Billede 5. Billede fra elevfremlæggelse.  
Billedet er fra en video hvor en elevgruppe præsenterer deres redesignede computermodel. Gruppen viser og fortæller i videoen, hvordan de har tilføjet nye sliders (skydere) til deres model (de blå og grønne bjælker), som gør det muligt for dem at undersøge, hvordan bakterievækst – ud over temperatur – påvirkes af lys og oxygen.



Billede 6. Billede fra elevfremlæggelse.  
Billedet er fra en video hvor samme elevgruppe præsenterer deres færdige løsning på designudfordringen. Billedet er desværre utydeligt, men gruppen har skitseret en app. De forklarer at brugeren skal indtaste pakke dato, udløbsdato, opbevaring og temperatur. En elev siger: "NetLogo-simuleringen kører i baggrunden. Appen viser grøn, hvis simuleringen viser færre end 300 mikroorganismer, gul, hvis der er mellem 300 og 450, og rød, hvis der er over 450".

## Organisering og deltagere

I fremlæggelserne kan man se, hvordan elevgruppen har arbejdet med modellen og fænomenet bakterievækst. De har læst og tolket modelkoden og udbygget deres viden om bakterievækst gennem eksperimenter med modellen. Undervejs har de udvidet koden ved at kopiere, tilføje og ændre dele af den. De har altså, som forventet, brugt modellen i deres undersøgelser af den faglighed, der ligger bag udfordringen. Samtidig har de fået indsigt i forholdet mellem koden og modellens opførsel. Endelig har eleverne brugt computermodellen som en del af deres løsning.

## De andre undervisningsforløb

Som omtalt blev der udarbejdet yderligere to undervisningsforløb med tilhørende NetLogo-modeller. Begge forløb havde en estimeret varighed på ca. 12 lektioner. De er nedenfor beskrevet med fokus, på hvordan de differentierede sig fra biologiforløbet, og de gav mulighed for at afprøve forskellige tilgange til at inddrage computermodeller i en problembaseret undervisning.

### Forløb om skovbrande

Et forløb om skovbrande i Australien arbejder med engineering, computermodeller og teknologiforståelse i geografi og inddrager også matematikfaglighed. Den faglige kontekst, dvs. engineering-narrativen, er det øgede antal skovbrande, der på grund af menneskeskabte klimaudfordringer flere steder er kraftigere, mere ødelæggende og har en højere hyppighed end de skovbrande, der er en naturlig del af skovens regeneration. Forløbet er bygget op omkring arbejde med agentbaserede modeller i NetLogo og viden om skovbrande i Australien. I forløbet kommer særligt vidensmål og færdighedsmål, der relaterer sig til elevernes udvikling af undersøgelses- og modelleringskompetencen, i spil.

Udfordringen i forløbet er at sikre en by mod en skovbrand, samtidig med at skovens biodiversitet bevares. Skovbranden skal slukkes, og skoven skal kunne komme sig. Det betyder, at modellen, der løser udfordringen, skal

sikre byen mod at brænde og samtidig efterlade et mosaikmønster i skoven, så den kan vokse videre. Mosaikmønstreet efter skovbrande genfindes i de økosystemer, der som i Australien (og andre steder) har tilpasset sig skovbrande.

Forløbet er bygget op om at lære at bruge modellen for skovbrand i NetLogo (*use*) og om at ændre en model, så både by og skov sikres (*modify*). Progressionen i forløbet er stilladseret, så eleverne arbejder med i alt fire versioner af modellen. Der arbejdes indledningsvis med en følsomhedsanalyse af skovens tæthed og procentvis brændt skov. Det danner en klassisk S-kurve, som eleverne matematisk modellerer og tolker ind i en geografisk sammenhæng for skovbrande i økosystemer, der danner mosaikmønster.

Derefter arbejder eleverne med *modify*, dels i forhold til at "fikse" en NetLogo-kode, der ikke fungerer, dels i forhold til løsning af skovbrandsudfordringen i en mere kompleks version af skovbrandsmodellen, der indeholder muligheder for at etablere brandbælter og iværksætte vandfly for at sikre både byen og skoven fra skovbranden. Skovbranden i denne version spredes afhængigt af vindretning og vindstyrke samt eksplosive eukalyptustræer, som i Australien (og andre lande) gør skovbrande særlig farlige.

Gennem hele forløbet arbejder eleverne med en digital logbog, der skal belyse refleksioner om, hvordan de har løst udfordringen gennem deres modificerede program, der er resultatet af forløbet. I kontekst af engineering svarer det til at konstruere prototyper og forbedre disse indtil den endelige version, produktet, der afslutningsvis præsenteres.

### **Forløb om energiteknologier**

Dette forløb var rettet mod fagligt indhold fra fysik/kemi omkring energiteknologier. Udfordringen var, med baggrund i fremskrivninger af introduktion af forskellige energiteknologier, at rådgive om, hvordan man lokalt eller nationalt kan vælge energiteknologier, så udledningen af CO<sub>2</sub> formindskes. I forløbet anvender eleverne modeller, hvor CO<sub>2</sub>-mængden i atmosfæren er bestemt af de valgte energiformer.

I den første del arbejder eleverne med en model af et kulkraftværk, hvis CO<sub>2</sub>-udledning svarer til Danmarks samlede udledning fra energisektoren i 2020. For at eleverne skal blive fortrolige med kode-delen af modellen, bliver de blandt andet bedt om at ændre i det grafiske udtryk – herunder farverne. Samtidig skal de bruge modellen til at fremskrive den samlede danske CO<sub>2</sub>-udledning i årene 2030, 2050 og 2100 ved hjælp af modellen (*use*).

I anden del af forløbet arbejder eleverne med at introducere alternative energiteknologier. Her er udgangspunktet vindmølleparker, som eleverne kan introducere ved tryk på en knap (*use*) og efterfølgende se, hvordan CO<sub>2</sub>-udledningen falder. Samtidig viser modellen, at CO<sub>2</sub>-molekylerne ikke forsvinder fra atmosfæren, selv hvis al kulkraft i modellen erstattes med vindkraft. Eleverne bliver bedt om at kode deres egne energiteknologier i NetLogo ved at modificere de eksisterende og i det hele taget at ændre på modellen, som det gav bedst mening for at afprøve deres bud på løsninger (*modify*).

Endelig skulle eleverne i del tre foreslå en sammensætning af energiteknologier, som de kunne argumentere for – enten på lokalt eller på nationalt niveau (afhængigt af lærerens valg). Eleverne skulle slutteligt præsentere deres argumentation på en video. Igennem processen udviklede og begrundede eleverne deres valg gennem idegenereringsprocesser, informationssøgning og afprøvninger.



# Projektets struktur og deltagere

Projektet blev påbegyndt i december 2021 og har været opdelt i fire faser:

1. Fastlæggelse af de didaktiske rammer og kontakt til testlærere om afprøvning
2. Udvikling af tre undervisningsforløb. I forbindelse hermed en første, kort afprøvning af elevers arbejde med computermodeller skrevet i NetLogo
3. Møder med testlærere og præsentation af didaktikken og undervisningsforløbene
4. Afprøvning af undervisningsforløbene, indsamling og analyse af data samt udarbejdelse af denne rapport.

## Projektplan

ECT Projektplan i fire faser											
december 2021 til oktober 2022											
	december	januar	februar	marts	april	maj	juni	august	september	oktober	
<b>Fase 1 - Udvikling af didaktisk ramme og aftaler</b>											
Fastlæggelse af didaktisk ramme											
Udarbejdelse af undersøgelsesspørgsmål											
Kontakt til konsulenter/skoler/klasser om projektet og informationsmøde											
Informationsmøde for lærere om praksisafprøvning											
Udvælgelse og aftaler med klasser/lærere											
<b>Fase 2 - Udvikling af prototyper</b>											
Udvikling af samlet prototype											
Fastlæggelse af tema for prototype											
Første skitse af prototype udarbejdes (fælles skabelon)											
Sparring og feedback på prototype på tværs (fokus på manuskraft)											
Samarbejde med programmer om udvikling af computermodel											
1. praksisafprøvning efterfulgt af feedback fra lærer (egen organisation)											
Tilretning af prototype på baggrund af 1. afprøvning											
Justering af endelig computermodel hos programmer											
Prototype klargøres inden for ensartet formidlingsramme											
Sparring og feedback på tværs (fokus på formidlingskrav og evaluering)											
Prototype færdig og klar til workshop og 2. praksisafprøvning											
<b>Fase 3 - Formidling af prototyper og endeligt evaluering</b>											
Tilpasning af evalueringdesign ud fra "færdige" prototyper											
Planlægning af workshop											
Workshop med testlærere (en samlet, evt. virtuelt)											
Forberedelse af evaluering ved 2. praksisafprøvning											
<b>Fase 4 - Afprøvning af prototyper, analyser og slutrapport</b>											
2. praksisafprøvning (aftales mellem udviklere og klasser)											
Sparring mellem udviklere og lærere											
Dataindsamling og feedback fra lærere/elever (ifm. 2. praksisafprøvning)											
Analyse og rapportskrivning 1. udkast											
1. udkast slutrapport til kommentering											
Indarbejdelse af kommentarer og færdiggørelse af slutrapport											
Slutrapport til korrektur											
Slutrapport til grafiker											
Slutrapport færdig											

Tabel 1. ECT projektplan i fire faser.

## Organisering

Projektejer er Engineer the Future, og projektledelsen blev varetaget af Mads Joakim Sørensen (ETF), som i samarbejde med Michael E. Caspersen (direktør It-vest) og Keld Nielsen (projektkoordinator for "Computational Thinking i matematik og naturfag" i gymnasiet) udviklede og beskrev projektet i foråret 2021. Projektet er finansieret gennem midler bevilliget af Teknologipagten og gennem egenfinansiering og er blevet gennemført i samarbejde med Københavns Professionshøjskole og VIA University College både med relevante medarbejderressourcer og med medfinansiering. Desuden har projektet haft tilknyttet it-didaktiske eksperter fra Center for Computational Thinking & Design på Aarhus Universitet samt didaktikere og praktikere inden for STEM-fagene.

I tabel 2 er der først en oversigt over projektets gennemgående deltagere ift. projektledelse, udvikling, formidling, analyser og udarbejdelse af slutrapport. Dernæst findes en oversigt over, hvilke skoler deltog i hele afprøvningen af de tre forløb og efterfølgende bidrog til evalueringen.

## Projektdeltagere

Deltagere	Projektrolle
Vibeke Schrøder Docent i teknologiforståelse, KP	Projektkoordinator/udvikler/evaluering
Jesper Heidemann Langhoff Lektor geografi, KP	Udvikler
Mikkel Hjorth Docent i teknologiforståelse, VIA UC	Projektkoordinator/udvikler/evaluering
Simon Olling Rebsdorf Adjunkt i fysik/kemi, VIA UC	Udvikler
Maria Damlund Lærer i naturfag og teknologiforståelse, Hornbæk Skole	Udvikler
Keld Nielsen Lektor emeritus, CCTD, AU	Udvikler/evaluering
Line Have Musaeus Forsker, CCTD, AU	Udvikler/evaluering
Jonas Ørbæk Hansen Lektor i fysik og kemi, Silkeborg Gymnasium	NetLogo-programmør
Mads Joakim Sørensen Seniorkonsulent, STEM-undervisning, ETF	Projektleder/udvikler/evaluering

Table 2. Oversigt over projektdeltagere og deres rolle i projektet.

## Projektskoler

I projektet deltog lærere og elever fra følgende skoler: Sankt Helene Skole, Gribskov kommune, Tingkærskolen, Odense kommune, Lille Egede Friskole, Slagelse kommune, Holmegaardskolen afd. Fensmark, Næstved kommune, Hornbæk Skole, Helsingør kommune, Utterslev Skole, København kommune.

# Fremgangsmåde

Projektet har været et eksplorativt pilotprojekt. Der eksisterer ikke nogen praksis for agentbaseret modellering i grundskolen, som projektet kunne undersøge, og som følge heraf har det fra start været hensigten, at projektet selv skulle eksperimentere med en sådan praksis gennem interventioner i skoler. I projektets første fase blev derfor udviklet en didaktisk ramme for udvikling og evaluering, herunder formuleret fire didaktisk funderede domæner med en række undersøgelsesspørgsmål, som interventionerne skulle bidrage med svar på:

## 1. Didaktik – Engineering og computermodeller

- Hvad er engineering-design-processens potentiale som metode til at integrere computationel tankegang i problemløsning i STEM, herunder afdække behovet for tilpasninger af engineering-didaktikken.
- Hvornår og hvordan computermodeller inddrages og er anvendelige i en designproces.
- Hvilke problemer/udfordringer giver det mening for elever at håndtere med computermodeller?
- Med hvilke forskellige formål kan computermodeller anvendes i undervisningen i naturfag, matematik og/eller i teknologiforståelse?

## 2. Elevernes udvikling af kompetencer

- I hvor høj grad evner eleverne at undersøge, vurdere og redesigne med simulering/digital modellering?
- I hvor høj grad evner eleverne at genkende og dekonstruere eksisterende kode/kodesprog i en computermodel og efterfølgende konstruere nye kodelinjer med udgangspunkt i naturfaglig viden og/eller undersøgelser?

## 3. Graden af understøttelse

- Hvilket behov har eleverne for computationel tankegang og stilladsering for funktionelt at kunne redesigne eksisterende computermodeller i problemløsende designprocesser?
- Hvilken viden og færdigheder har lærerne behov for i forhold til at få denne form for computermodellering til at fungere i deres praksis?

## 4. Nye tværfaglige sammenhænge

- Hvilke undervisningsmål i STEM/teknologiforståelse kan understøttes med at inddrage computermodeller i en problembaseret undervisning?

Undersøgelsesspørgsmålene har i projektet bidraget til at skabe en sammenhæng mellem projektets formål, udvikling af undervisningsforløb, indsamling af data fra praksisafprøvningen samt analyser og konklusioner i slutrapporten.

I projektet er der – som omtalt ovenfor - udviklet tre forskellige problembaserede engineering-forløb, der inddrager agentbaseret modellering med NetLogo som værktøj. Forløbene er udviklet af tre udviklingsgrupper og har taget udgangspunkt i problemfelter fra hhv. fagene geografi, biologi og fysik/kemi. Der blev i den indledende fase på baggrund af bl.a. undersøgelsesspørgsmålene udviklet et prototypemanus med en række ens kriterier for udvikling af undervisningsforløbene, men med frihed hos udviklerne til at afprøve forskellige didaktiske ideer.

De tre undervisningsforløb blev formidlet til testlærerne på en praksisnær heldagsworkshop med hovedfokus på at give lærerne erfaringer med at anvende undervisningsmaterialet samt NetLogo. Herefter havde lærerne ca. 5 uger til at gennemføre undervisningsforløbene i egen praksis.

Ved begyndelsen af praksisafprøvningen var der lavet aftaler med i alt 11 lærere med tilhørende klasser. Desværre var det ikke alle lærere, der deltog på workshoppen, som endte med at gennemføre praksisafprøvningen. I tabel 3 kan aflæses, på hvilket datagrundlag der er analyser og konklusioner til denne rapport.

Deltagere	Antal
Elever	138
Lærere	8
Klasser i geo-forløb	1
Klasser i fys/kemi-forløb	3
Klasser i bio-forløb	4

Tabel 3. Antal elever, lærere og klasser, der deltog i projektet og i dataindsamlingen.

Data, i form af spørgeskemabesvarelser, blev indsamlet fra både lærere og elever efter gennemførelse af forløbene. Desuden blev der indsamlet data gennem besvarelser af et spørgeskema uddelt til lærerne inden afvikling af forløbene. Lærerne besvarede således spørgeskemaer før og efter forløbene. Eleverne besvarede et spørgeskema efter deltagelse i et af forløbene. Alle spørgeskemaer indeholdt både åbne spørgsmål med tekstbesvarelser og lukkede spørgsmål i form af skalaer med tre svarmuligheder ('nej', 'ja lidt', 'ja meget').

12 lærere har besvaret før-spørgeskemaet fuldt ud. 8 lærere har besvaret efter-spørgeskemaet fuldt ud. I alt var 138 elever deltagere i projektet, hvoraf 129 elever gennemførte hele spørgeskemaundersøgelsen efter at have deltaget i et af de tre forløb. De besvarende elever var fordelt med 20 elever fra geografiforløbet, 19 elever fra fysik/kemi-forløbet og 90 elever fra biologiforløbet.

Efter implementeringsfasen blev alle lærere inviteret til fokusgruppeinterviews om deres erfaringer med forløbene. Der blev gennemført tre interviews med to lærere i hver. Gruppeinterviewene foregik virtuelt med repræsentanter fra hvert af de tre afprøvede undervisningsforløb, henholdsvis 3 fra biologiforløbet, 2 fra fysik/kemi-forløbet og 1 fra geografiforløbet. På grund af det lave antal lærere er det ikke hensigtsmæssigt at lave en komparativ analyse på tværs af fagene, og derfor er interviews fortrinsvis behandlet samlet. Hertil kommer, at både lærere og elever havde meget forskellige udgangspunkter for at gennemføre forløbet, hvilket medførte, at oplevelsen af det samme undervisningsforløb kunne variere meget.

# Erfaringer og konklusioner på tværs af de tre undervisningsforløb

I opsamlingen af erfaringer har vi særligt haft fokus på fire didaktiske domæner:

- Didaktiske spørgsmål – potentialet ved at bygge på engineering og anvendelsen af computermodeller i denne kontekst
- Elevernes udvikling af kompetencer inden for modellering og programmering med udgangspunkt i faglig viden og undersøgelser
- Behov for stilladsering af eleverne og kompetenceudvikling af lærerne
- Nye tværfaglige sammenhænge, afdækning af de kompetencemål i STEM-fagene og teknologiforståelse, der kan dækkes via tilgangen.

Den overordnede konklusion er, at projektet har demonstreret, at det er muligt at arbejde projektorienteret i grundskolen med en kombination af engineering og teknologiforståelse under inddragelse af faglig viden fra et eller flere STEM-fag. For eksempel siger en lærer:

*”Umiddelbart har forløbet foregået rigtig godt, alle er kommet godt i mål med at lave noget fedt med den her computermodel”.*

## Potentialet ved at kombinere engineering med brug af computermodeller

### Motivation og processtyring

Der er tydelige tegn på, at det motiverer eleverne at arbejde med en konkret udfordring. I fokusgruppeinterviews forklarer flere lærere, at problemstillingen optog eleverne og udgjorde en god basis for den interesse og energi, som eleverne generelt lagde i arbejdet.

Efter gennemførte forløb rapporterede lærerne, at de skulle arbejde mindre med at motivere eleverne, end de på forhånd havde forventet. En lærer udtrykker det på denne måde:

*”... selve casen er god, og den giver noget til det her med at arbejde med modellen, at man hele tiden har et eller andet at vende tilbage til: Okay du skal huske, når du laver det her i din model, at du skal kunne forklare det”.*

I det hele taget rapporterer lærerne efter at have gennemført deres forløb, at der generelt var brug for mindre ad hoc stilladsering af grupperne, end de havde forventet. Det kan muligvis forklares med – som det er set i andre engineering-sammenhænge – at engineering-design-processen er med til at strukturere forløbet for både lærere og elever, samt at elevernes arbejde med tekstprogrammering i NetLogo var lettere tilgængeligt og mere intuitivt forståeligt end lærerne havde forudset.

Mindst én lærer har dog også haft problemer med EDP-tilgangen. Læreren fortæller, at forløbet blev for kompliceret af det produktkrav, der hører til engineering-processen. Læreren valgte derfor at frafalde produktkravet og begrænsede brug af modellen til at undersøge det faglige fænomen, hvorved forløbet mere lignede et IBSE-inspireret



forløb. Det ser altså ud til, at man som udvikler af et forløb bør være opmærksom på at balancere mellem graden af faglig kompleksitet (i udfordring, computermodel og produktkrav) og forventninger til elevernes arbejde med en engineering-design-proces.

## Computermodellen og relevans af faglig viden

Den computermodel, som eleverne får adgang til i forbindelse med undervisningsforløbet, modellerer et fagligt fænomen. Der er altså bygget faglig viden ind i modellen, og vidensbaserede ændringer af modellen kræver, at man arbejder med fagligheden og dermed enten respekterer denne faglighed og/eller udbygger modellen på grundlag af ny faglig viden. Der er tegn i projektet på, at dette direkte anvendelsesaspekt gør eleverne nysgerrige efter at forstå den tilknyttede faglighed og/eller at finde nye oplysninger, som de kan bruge, når de arbejder med modellen. Engineering koblet med brug af en faglig model giver således eleverne en reel oplevelse af, at faglig viden kan anvendes på konkrete problemer og er en forudsætning for at kunne løse en udfordring. Dette illustreres blandt andet ved at 58 % af eleverne fra spørgeskemaundersøgelsen syntes, at "Interfacet (simuleringen)" var en hjælp til at finde en løsning, og et tilsvarende antal elever (60 %) syntes, at computerkoden var en hjælp til at finde en løsning. Det var altså et flertal af eleverne, der oplevede, at modellerne med fordel kunne inddrages, og at de var anvendelige i processen.

Her følger et uddrag fra et fokusgruppeinterview, hvor to lærere reflekterede over deres tidligere erfaringer med engineering-design-processer:

Lærer A:

*"... jeg synes, at engineering er lige så god som alle de andre innovationsmodeller".*

Lærer B:

*"... den er faktisk bedre. Det synes jeg, fordi mange af de innovationsmodeller, der findes, de går aldrig rigtig fagligt i dybden, hvor der i engineering er en undersøgelsesdel, som faktisk går ind i naturvidenskaben, og der synes jeg nogle gange, så bliver de andre, de bliver sådan lidt for ikke-fagfaglige".*

Inddragelse af faglig viden har voldt problemer i nogle klasser. Nogle lærere fortæller, hvordan undervisningen blev splittet i på den ene side en computationel del, hvor det mest drejede sig om modellen og programkoden, afkoblet fra det faglige fænomen, og på den anden side arbejde med en mere klassisk faglig forståelse, som ikke blev knyttet til modellen.

Denne splittelse er ikke ualmindelig i forsøg med ny didaktik, hvor lærerne skal kombinere to fagligheder og i klassen bliver usikre på, om de nu underviser med sigte på den ene eller den anden faglighed. Erfaring fra bl.a. arbejde med modeller i gymnasieklasser tyder dog på, at lærerens erfaring er en kritisk faktor. Når en lærer opnår mere erfaring med den kombinerede undervisning, opnår læreren kompetencer til at kombinere de to fagligheder, så de styrker hinanden. (Yadav et al., 2019)

Der er tegn på, at det er vigtigt at holde modellen – og dermed de faglige problemstillinger – meget simple. En lærer siger:

*”Jeg tror måske, en god ide ville være at simplificere den [modellen] så meget som overhovedet muligt. Ikke fordi den har været kompleks, eller jo, lidt kompleks er den selvfølgelig. Men jo mere simpel den er, jo nemmere er den også at lege med og bevæge sig rundt i og differentiere, så man rammer sine elever rigtigt”.*

En sammenligning af elevernes produkter/løsninger fra de tre forløb bekræfter, at jo enklere man kan udforme modellen og dermed den bagvedliggende faglige problemstilling, jo nemmere er det for eleverne at gennemskue, hvordan de kan bruge modellen som redskab til at udvikle deres produkt.

I nogle klasser viste det sig at være udfordrende for eleverne (og lærerne) at finde relevante data og relevant faglig viden, hvilket kan skyldes, at undervisningsforløbets faglige udfordring var vanskelig for eleverne at overskue. En lærer siger ligeud:

*”Det var megahårdt at stå som lærer og sige fuck, jeg har givet dig en opgave [med søgning efter faglig information], du ikke kan klare”.*

Alle de adspurgte lærere vurderede dog, at eleverne gennem forløbene havde opnået læring i de respektive fag, idet alle fag bliver nævnt som en faglighed, eleverne fik styrket. Altså hhv. biologi, geografi, matematik, fysik/kemi, engelsk, STEM, teknologiforståelse. Men der nævnes også mere specifikke faglige områder som programmering, naturvidenskabelig undersøgende tilgang, nedbrydere og atmosfæren. Desuden nævnes elevernes forståelse af modellen som en simulator eller et billede

Mange elevfremlæggelser demonstrerer, at eleverne har fået fornemmelse for, hvordan computermodellen viser noget om den virkelige verden, hvad enten der er tale om bakteriers vækst, hvordan skovbrande breder sig, eller hvordan atmosfæren fyldes op med CO<sub>2</sub>.

Desuden får eleverne direkte indsigt i, hvad modeller er, og hvordan de kan bruges. Selvom det ikke direkte var et mål med forløbet, vurderer en lærer endda, at eleverne har fået bedre forudsætninger for at forholde sig til computationelle modeller i samfundet mere generelt. Netop det at kunne forholde sig kritisk og analytisk til digitale artefakter i sin omverden er en del af kompetenceområdet digital myndiggørelse i forsøgsfagligheden teknologiforståelse.

En lærer siger om elevernes arbejde med modellen:

*”... det giver jo et perspektiv på, at alle kan sidde og lave sådan nogle modeller her. Altså, hvis vi [i klassen] kan lave dem, hvem har så lavet de modeller, vi finder ude på nettet ... [det] giver dem et blik for, at det er faktisk ikke så svært at lave de her modeller og få dem til at gøre lige præcis det, du gerne vil have. Så kan man stole på alle de modeller, man ellers finder?”*

En lærer kommenterer på forholdet mellem elevernes løsning/produkt og den model, de har arbejdet med: *”Men mange endte med [en] app, ... hos dem alle sammen endte koden med at være tankegangen bag appen”.*

Samtidig er der også lærere, som fortæller, at det var vanskeligt for eleverne at forbinde modellen med den virkelighed, som udfordringen krævede, de skulle forholde sig til. Specielt gav det udfordringer for nogle grupper at se, hvordan modellen kunne relateres til deres omverdensproblem og derfra til de så kunne løse det.

## Elevernes udvikling af kompetencer inden for modellering og programmering

### Erfaringer med programmeringsmiljøet NetLogo

I forbindelse med undervisning i programmering bliver det løbende diskuteret, om det er mest hensigtsmæssigt at lade elever starte med tekstbaseret programmering (Visual Basic, Python etc.), eller det er bedre at starte med blokprogrammering (Scratch, Blockly etc.).<sup>1</sup>

I lyset af den diskussion må man sige, at elever, der deltog i afprøvningskerne, havde overraskende nemt ved at læse, afkode og ændre den tekstbaserede computerkode og viste stort engagement. Det gælder selv i en 6.-klasse, som kom med i afprøvningskerne på et afbud.

Denne iagttagelse gælder ikke kun afprøvningskerne i de klasser, som vi refererer her. En forudgående, indledende afprøvning af NetLogo-modeller i tre klasser viste samme resultat, hvor projektdeltagere i udviklingen af biologiforløbet havde følgende observation: *Efter den forvirring, der altid optræder, når man bliver præsenteret for et nyt programmeringssprog, gav eleverne sig uimponerede i gang med at læse, diskutere, fortolke og ændre i koden.*

Eleverne havde nemt ved at arbejde med koden. 88 % af de deltagende elever rapporterer, at de selv ændrede i computerkoden og dermed i modellen. Det tal er højere end andre lignende projekter, hvilket vi anser som stærkt opmuntrende. Tallet skal selvfølgelig fortolkes i lyset af, at nogle ændringer i NetLogo – såsom at ændre farver eller form på de elementer (agenter), som man ser på skærmen – er ret nemme at lave, mens ændringer, der ændrer eller tilføjer funktionalitet, er mere udfordrende.



<sup>1</sup> Se fx <https://codingpirates.dk/wp-content/uploads/2021/10/FA%CC%8A-BOERNENE-TIL-AT-LEGE-MED-KODNING-e8768952.pdf>

I det hele taget har brugen af den tekstbaserede NetLogo-kode vist sig at fungere godt. En lærer siger:

*”Mine elever, som gik fra aldrig at kunne kode noget som helst til at introducere sliders, farver og former osv. og kan analysere og ændre og skrive koder – dér er der jo sket en kæmpe, virkelig fed udvikling”.*

En anden lærer siger:

*”Jeg synes faktisk, at den første del af forløbet, hvor de skal stifte bekendtskab med koden første gang, dér var de superexcitede. Vi havde en hel fagdag med dem, hvor vi brugte det her med egentlig bare at sidde og arbejde ind i koden og begynde helt fra ... bare følge arbejdsgangene. Det var de altså hooked på”.*

Også elever, som læreren på forhånd ikke havde forventet ville give sig i kast med tekstprogrammering, overraskede. Med udgangspunkt i deres forhold til matematik blev en lærer positivt overrasket over en gruppe piger:

*”De piger, som er lidt svage i matematik, de har en enorm styrke ... når man koder. Selvom de ikke har set den her form for kodning før, så fandt de lynhurtigt ud af at dele det ind i blokke, og sagde: ’Den her blok, den koder jo fabrikken, denne her blok, den koder baggrunden, denne her ...’. Altså, de delte det hurtigt op i blokke og sagde: ’Jamen, altså, rent tekstuel, kan vi så dele det ind i blokke?’ Og det gjorde de altså lynhurtigt og fik et overblik”.*

Også en smule uventet rapporterer flere lærere, at deres elever fandt det mere tilfredsstillende at arbejde med tekstbaseret kode (som i NetLogo) end med blokbasert programmering (som fx i Scratch). Eleverne følte ifølge læreren, at de havde mere frihed og flere muligheder for selv at lave ændringer i NetLogo. En lærer siger:

*”Det øjeblik de sætter sig ned og har forstået koden og kan gå ind og ligesom omsætte de oplysninger, de har fundet, der skal lægges ind i koden, når ... de skal ind og redigere i koden og ser: Virker det? Virker det ikke? Når de opdager, at det virker faktisk, det her, så er de simpelthen ... altså så er det bare pissefedt for dem”.*

Efter at have gennemført et af de tre forløb mente 44 % af lærerne, at eleverne havde lært om brug af NetLogo, mens 22 % mente, at eleverne havde fået styrket deres faglighed i programmering. En lærer fortalte desuden, at hans klasse ønsker at arbejde med NetLogo-modeller næste år, men med nogle andre modeller og anden faglighed.

## Elevernes brug af data

Vores analyse af 26 videoer med elevfremlæggelser fra tre forskellige klasser med tre forskellige undervisningsforløb viser, at eleverne har været fortrolige med at bruge den simulering, som er modellens fremtrædelse på skærmen. Langt de fleste grupper er omhyggelige med at forklare, hvilke inddata modellen arbejder med (fx temperatur, lysintensitet, iltindhold, vindhastighed, antal vindmølleparker etc.), og de demonstrerer god forståelse for den rolle, inddata spiller.

Alle de modeller, eleverne arbejder med, producerer også uddata, der enten vises numerisk eller som en graf med tilhørende akser. De fleste af grupperne henviser til modellens uddata og bruger dem i deres argumentation. For eksempel har flere af de grupper, der foreslår at lave en app på grundlag af modellens simulering, overvejelser over, hvordan uddata kan præsenteres overskueligt og anvendeligt for en bruger.

Denne konkrete brug af data harmonerer godt med målene for databehandling i forsøgsfaget teknologiforståelse, som det er beskrevet i Fælles Mål, og peger på et potentiale for en mere systematisk inddragelse af matematik i denne type af forløb.

Der er dog også indikationer af, at hvis modellen producerer for mange data – fx i form af flere simultane grafer – vælger eleverne at afstå fra at bruge data og tolker simuleringen direkte på grundlag af grafikken.

## En ny fejlretningskultur?

Der er flere lærerudsagn om, at hvis en elev ændrer i koden, så modellen ikke virker, men returnerer en fejlmelding, så taber eleven ikke modet. Mange elever forsøger straks at rette fejlen uden opmuntring fra læreren. Det har nogle af de deltagende lærere aldrig set før. Lærerne er vant til, at eleverne giver op, når de laver en fejl og opdager dens frustrerende konsekvenser.

En lærer siger om denne tilgang til at håndtere fejl:

*”... det er ikke noget, jeg ser normalt ved dem [eleverne], det her med, når de laver en fejl, at de så bare kommer tilbage direkte igen og prøver at finde ud af, hvad er fejlen, og hvordan kan vi gøre, så den ikke er der længere.*

*... Jeg har dem også i matematik, og når de får at vide, at der er noget, der er forkert, så kan du se, hvordan humøret bare fuldstændig [indikerer et fald med hånden].*

*... det var nyt for mig, det synes jeg var fedt personligt og for mig som underviser for dem, at de har rent faktisk den her tankegang om, at fejl, det er noget, der kan rettes, og det er noget, vi kan gøre bedre. Det er ikke, fordi vi er dårlige.*

*Så det var en af de største ahaoplevelser både for mig og for dem, håber jeg da også”.*

Den anden lærer, der deltager i interviewet, giver den første lærer ret. Denne lærer har også set sine elever finde og rette fejl i koden og har ikke set lignende fejlhåndtering hos elever i andre sammenhænge, heller ikke i forbindelse med programmering.

*”At hun opsøger hendes egen fejl og selv får den rettet med det samme, inden jeg overhovedet når at sige noget. Det er ikke noget, jeg har oplevet før, og det var sådan generelt for dem alle sammen her”.*



## Behov for stilladsering af eleverne og kompetenceudvikling af lærerne

Generelt var der ved spørgeskemaundersøgelsen lidt flere elever, der vurderede at have mere brug for hjælp fra læreren til computermodellen end til de øvrige dele af engineering-processen, men denne forskel var ikke statistisk signifikant. Det kan betyde, at eleverne ikke opfattede arbejdet med computermodellen som væsentlig sværere end de andre arbejdsprocesser i forløbene.

De deltagende lærere besvarede en række spørgsmål, før de gik i gang med afprøvningen, og igen, efter at afprøvningen var gennemført.

Spørgeskemaundersøgelsen fra lærerne viser et par meget interessante perspektiver:

Før lærerne gennemførte deres afprøvning, blev de spurgt om deres forventninger til, hvor stort behov der ville være for at stilladsere eleverne med hensyn til:

- Computerkoden/programmering
- Det faglige
- Data fra modellen
- Praktiske undersøgelser
- Undersøgelser med computermodellen
- Sammenhængen mellem model og udfordring.

Efter forløbene blev lærerne spurgt om deres konkrete erfaringer med stilladsering på de samme områder. I forbindelse med programmering rapporterede lærerne, at deres erfaringer for behovet for stilladsering svarede til deres forventninger. På de øvrige områder rapporterede lærerne et *mindre* behov for stilladsering end forventet. I betragtning af hvor nyt det var for både elever og lærere at arbejde med computermodeller i en problemløsende kontekst, er det positivt, at omfanget af og behovet for stilladsering ser ud til at være overskueligt for lærerne.

I spørgeskemaundersøgelsen kunne lærerne skrive kommentarer i fritekst. I forbindelse med spørgsmål om elevernes læring ved at arbejde med modeller skrev lærerne i præ-testen, at de havde forventninger til, at eleverne ville lære "computational tankegang" og "teknologiforståelse". I post-testen er lærernes ordvalg anderledes, og de skriver konkret om læring i "programmering" og "modellering".

Der kan være flere forklaringer på dette sproglige og begrebsmæssige skifte hos lærerne, når de beskriver elevernes læring før og efter praksisafprøvningen. En nærliggende forklaring er, at lærerne – efter at have gennemført et forløb – har fået en mere konkret opfattelse af, hvad det er, eleverne kan arbejde med i teknologiforståelse. Hvis det er tilfældet, er det ganske positivt.

Før forløbene havde lærerne desuden forudset, at deres egne største udfordringer ville blive elevmotivation (42 %), programmering (8 %) og tid (8 %). Efter forløbene beskriver lærerne imidlertid deres udfordringer i kategorierne informationssøgning (33 %), programmering (33 %), motivation (22 %) og tid (11 %). Lærerne selv var dermed langt mindre udfordrede i kategorien 'elevmotivation', end de selv havde forudset, mens de var mere udfordrede i forhold til programmering og informationssøgning, end de havde regnet med at blive. Det peger på et behov for yderligere kompetenceudvikling af lærerne, der især retter sig mod kompetencer omkring programmering og kodning.

## Tværfaglige sammenhænge - STEM-fagene og teknologiforståelse

### Teknologiforståelse

Projektet har givet lærere og elever gode, konkrete og nye muligheder for at arbejde med helt centrale områder i teknologiforståelse som modellering, programmering og brug af data. Herunder arbejde med, hvordan man præsenterer relevante data for brugergrupper.

De konkrete opgaver med at arbejde med tekstbaseret programmering og udformning af modeller, der har relationer til virkelige fænomener, har styrket elevernes teknologiske handleevne. Det ser ud til, at eleverne fik erfaringer med, at computerkode har en rigid syntaks, og en indsigt i den rolle, som koden spiller for en computers eller en models opførsel. De fik ligeledes mulighed for selv at ændre på koden.

Konkret indsigt i, hvordan man bygger en model op og tilpasser den og får data og prognoser, der er relateret til virkeligheden dermed til analyse af presserende problemer, har styrket elevernes computationelle tankegang.

Dette afspejles, når eleverne bliver bedt om med egne ord at beskrive, hvordan computermodellen kan anvendes til løsning af et tværfagligt problem. I den sammenhæng beskriver nogle elever næsten eksperimenter med computermodellen, hvor fx temperaturen varieres (som i et køleskab, i en fryser og ved stuetemperatur). Andre elever nævner også, at det at forstå fænomenet "mere præcist", "forstå problemet" og "visualisere/se" problemet er vigtige egenskaber, som computermodellen bidrager med. Enkelte elever nævner, at modellen kunne bruges til at "få nye tanker", "lave en plan", "se, hvordan fremtiden måske kunne ende ud" og "give forkerte data". Det tyder altså på, at nogle elever i hvert fald selv har haft en oplevelse af, at de er blevet bedre i stand til at anvende modellering til at forstå deres omverden og til at forudsige noget om fremtiden.

### STEM-fagene

Alle tre udarbejdede undervisningsforløb arbejdede inden for den didaktiske ramme, der er sat af projektet: En udfordring skulle løses i et engineering-design-forløb under brug af en fagligt relevant computermodel. Men med hensyn til elevernes brug af den konkrete model og inddragelse af fag og nødvendig faglighed blev der afprøvet tre forskellige tilgange.

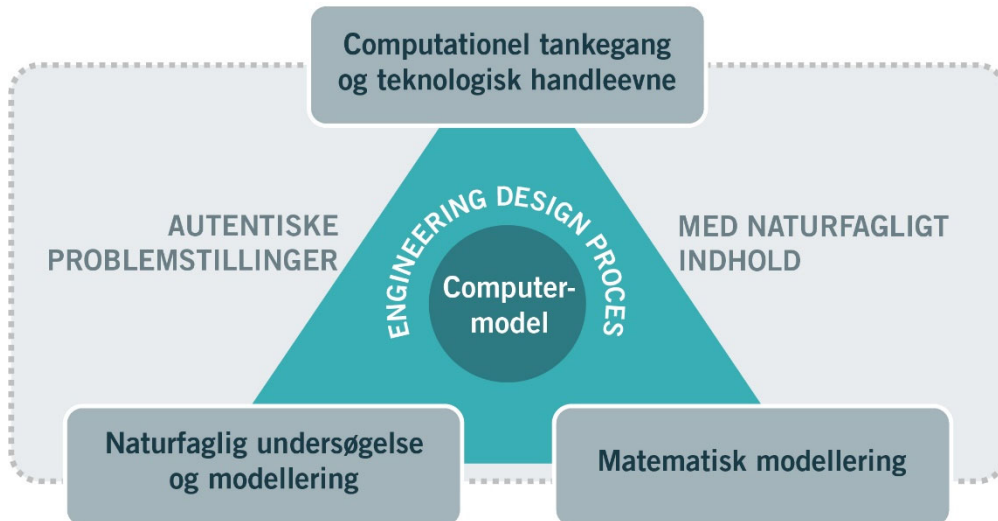
Derfor var den rolle, som matematik og de indgående naturfag spillede, også forskellig fra forløb til forløb, og det er vanskeligt på grundlag af erfaringerne at uddrage noget generelt.

Dog er der positive tegn på, at eleverne i arbejdet med modellen ser nye sider af kendte fag. Fx siger en lærer:

*"... jeg har diskuteret vækst med dem, og lige pludselig så gav det mening noget af det, vi har diskuteret [om vækstfunktioner] i matematik".*

Der er også tegn på, at eleverne direkte oplever, at de – for at forstå modellen og for at bruge den – er nødt til at have styr på faglig viden. Og for at udvide modellen på en meningsfuld måde opstår der behov for at finde ny, relevant faglig viden.

Lærerne beskriver i besvarelsen af spørgeskemaerne, at eleverne får styrket faglighederne inden for teknologiforståelse, STEM-fag og den naturvidenskabelige undersøgelsesmetode ved arbejdet med disse forløb. Mht. hvilke kompetencer eleverne får styrket i disse forløb, nævner lærerne oftest undersøgelseskompetencen, samarbejdskompetencen og modelleringskompetencen.



Figur 3. Modellen illustrerer hvordan elever kan udvikle kompetencer inden for hhv. naturfag, matematik og teknologiforståelse gennem en engineering design proces med udgangspunkt i en naturfaglige problemstillinger, og med inddragelse af en computermodel som en del af løsningen.

## Fremadrettede anbefalinger

Pilotprojektet har – som det er konkluderet ovenfor – vist, at det er muligt at arbejde projektorienteret i grundskolen med en kombination af engineering og teknologiforståelse under inddragelse af faglig viden fra et eller flere STEM-fag.

Resultatet er opmuntrende, fordi det bidrager med erfaringer og ny didaktisk viden, som gør det muligt at inddrage teknologiforståelse i undervisningen i STEM-fagene (eller at inddrage STEM-fagene i undervisning i teknologiforståelse). Sådanne faglige kombinationer har stort potentiale, dels fordi brug af data og computermodeller spiller et stadig større rolle i STEM-fagene, som de praktiseres uden for skolen, og denne rolle bør afspejles i skolens undervisning, dels fordi det er vigtigt, at teknologiforståelse indarbejdes i STEM-fagene på så bredt et autentisk grundlag som muligt.

Pilotprojektet har desuden demonstreret, at kombinationer af STEM-fag og teknologiforståelse med fordel kan planlægges, gennemføres inden for rammerne af en etableret engineeringdidaktik inden for STEM-fagene, herunder engineering-design proces.

Den innovative didaktik tilfører fagene et nyt, moderniseret grundlag for at udfolde en række centrale kompetencer som modelleringskompetencen og undersøgelseskompetencen samt et autentisk omverdensrelateret aspekt på brug af digitalisering, data og modeller. Vi vurderer endvidere at didaktikken har stort potentiale for at vise relevansen af faglig viden i forhold til problemløsning og anden anvendelse af viden i forbindelse med konkrete omverdensudfordringer.

Endelig finder vi at det er en vigtig observation, at eleverne udfordres positivt af, og mestrer at arbejde med, tekstbaseret kode, i hvert fald når det sker inden for rammerne af programmeringsmiljøet NetLogo.

Det må således forventes, at der vil ske en yderligere udvikling af undervisning, der kombinerer digital teknologiforståelse med STEM fag inden for rammerne af en engineering design proces.

Derfor supplerer vi nedenstående rapporten over pilotprojektets forløb og resultater med en række anbefalinger, der bygger på erfaringerne fra projektet og samtidig afspejler den viden om STEM-fagene og teknologiforståelsesfaget, der er til stede i den ekspertgruppe, som har stået bag pilotprojektet (se gruppens sammensætning side x).

Vi har fundet det naturligt at formulere anbefalingerne som en række punkter vi forventer vil kræve en særlig indsats i en fremtidig udvikling.

Fokuspunkterne er:

- Selvom mange af eleverne har nemt ved at arbejde med koden (i NetLogo), så er det svært for lærerne at vejlede, når eleverne alligevel støder på problemer. Der skal findes en løsning på, hvordan lærerne kan rustes bedre til det.
- Det skal gøres mere tydeligt for lærere og elever, på hvilke forskellige måder computermodeller kan bidrage til løsningen af problemer i engineering-forløb.
- Det er centralt, at der udvælges enkle faglige udfordringer, så forløbene og computermodellerne får den rette grad af faglig kompleksitet, så eleverne reelt kan designe en løsning, herunder finde data, når de laver undersøgelser med modellen.
- Det ser ud til, at matematik har et uforløst potentiale, som bør udvikles yderligere med henblik på at forankre og tydeliggøre matematikken bedre i forløbene.
- Det bør undersøges nærmere, hvordan computermodeller i en problembaseret undervisning kan bidrage til, at eleverne lærer at forholde sig kritisk og analytisk til digitale artefakter i deres omverden og derved opnår færdigheder og viden inden for kompetenceområdet digital myndiggørelse i forsøgsfagligheden teknologiforståelse.

- Det kan undersøges mere grundigt, hvad det er, eleverne lærer ved disse forløb. Det er før vist, at CMC-tilgangen kan bidrage med faglig læring i biologi hos elever i gymnasiet (Musaeus et al., 2022).
- Det ser ud til, at eleverne arbejder bedst og får den bedste designproces, når den computermodel, de får udleveret, er så simpel som overhovedet muligt. Derved har de selv kontrol over, hvad det er, de vil ændre ved modellen, og det forpligter dem.
- De første erfaringer med at udvikle undervisningsmateriale inden for engineering, computermodeller og teknologiforståelse giver grobund for at overveje, hvorvidt undervisningsmaterialet skal opbygges modulerbart, så lærerne i højere grad kan forme undervisningsforløbet selv, og hvorvidt materialet skal bygges op omkring en enkelt simpel model eller en række af modeller med stigende kompleksitet.
- Integreringen af flere didaktiske tilgange er krævende for lærere, særligt dem, der ikke er kompetenceudviklet i engineering i forvejen. Kompetenceudvikling skal kunne rumme lærere både med og uden engineering-baggrund.
- Kompetenceudviklingen skal være mere langvarig og grundig end den ene onlineworkshop, der blev tilbudt i praksisafprøvningen.
- Kompetenceudviklingen af lærere skal udvikles, så de løbende udvikler deres NetLogo-kompetencer. Der bør være fokus på, hvordan lærerne skal klædes på til at stilladsere eleverne, når der opstår udfordringer med syntaksen i NetLogo, herunder hvordan lærerrollen kan varetages faciliterende, selvom man som lærer, ikke nødvendigvis har svarene.

# BILAG

## Præsentation af indsamlet empiri

I forbindelse med pilotprojektet har vi indsamlet erfaringer gennem interviews med lærere, analyse af elevers afsluttende videoer/produkter og gennem tre spørgeskemaundersøgelser. Herunder findes et ekstrakt af pointer fra de tre forskellige kilder til empiri.

### Fokusgruppeinterviews

#### 1. Didaktiske spørgsmål – potentialet ved at bygge på engineering og anvendelsen af computermødeller i denne kontekst

Nogle lærere har haft meget engagerede klasser, og nogle klasser vil gerne arbejde videre med modeller i NetLogo – og meget gerne med lokalt funderede modeller. De interviewede lærere nævner, at de ser mange fagligt spændende perspektiver i at arbejde med modeller, og de erfaringer, lærerne nu har fået, har bragt dem til at tænke videre om modeller og modellering i både naturfagene og matematik. Der er altså lærere i projektet, som i interviews nævner, at de forestiller sig at arbejde videre med agentbaseret modellering i NetLogo, hvilket vi ser som et tegn på, at lærerne har oplevet agentbaseret modellering såvel som NetLogo som meningsfulde i deres undervisning.

### Sammenhæng mellem problem og model

I interviewene blev der præsenteret en bred vifte af erfaringer i forhold til at skabe sammenhæng mellem de naturfaglige problemstillinger, som eleverne skulle arbejde med, og de computermødeller, som eleverne skulle anvende i forhold til problemstillingerne.

Nogle af lærerne fortalte, at de havde udfordringer med at få eleverne til at forbinde computermødellen med virkeligheden, mens andre oplevede, at eleverne bragte modellen i brug på måder, som lærerne ikke havde forudset. En lærer pegede på, at arbejdet med modellen aktualiserede undervisning i det naturfaglige indhold: "de syntes, det var en fed måde at lære det på, at det er den tilbagemelding, de kom med: 'Hey, nu giver det meget mere mening, det her med CO<sub>2</sub>'". En anden lærer pegede på, at casen med CO<sub>2</sub> med fordel kunne have været mere nær for eleverne: "Jeg synes, selve casen er god. Jeg kunne godt tænke mig, at man måske i stedet for at gøre den national så gøre den lokal". Samtidig pegede denne lærer (ligesom den anden med samme forløb) på, at der havde været store vanskeligheder forbundet med at søge relevante data frem om det nationale energiforbrug, og at netop det at gøre engineering-udfordringen mere lokal måske kunne afhjælpe en sådan problematik.

Andre lærere oplevede, at de i deres undervisning kom til at fokusere enten på det faglige indhold eller på computermødellerne. En talte om, at elevernes naturfaglige udfordring kom til at tage en stor del af deres opmærksomhed, så det var den faglige problematik i sig selv, der var i fokus, og ikke det at arbejde med modellering af en faglig problematik. En anden lærer oplevede, nærmest modsat, at programmeringen kom til at fylde så meget, at det overskyggede den naturfaglige udfordring. Denne opdeling opstod, da to lærere delte forløbene (og dermed kompetenceområderne) op i et med fokus på programmering og et med fokus på madspild. Eleverne endte med at kvittere for denne opdeling ved at aflevere i 2 spor til hver sin lærer. Ikke fordi lærerne ikke samarbejdede, men fordi eleverne havde så forskellige faglige niveauer på de to områder – madspild og programmering. Det tyder altså på, at det er væsentligt, at lærere har fokus på at integrere model og faglighed, hvis eleverne skal opleve en sammenhæng mellem model og naturfaglig problemstilling. Det er derfor et relevant opmærksomhedspunkt at arbejde med, hvordan undervisningsforløb i engineering, teknologiforståelse og modellering kan planlægges og gennemføres, så der bliver en faglig sammenhæng mellem den digitale model og det naturfaglige domæne.



Endvidere pegede flere lærerne også på, at det er vigtigt at styrke fokus på, at der skal veksles mellem arbejde med kode i computermodellen og den faglige udfordring, som eleverne var blevet stillet, for at udvikle et fokus på sammenhængen mellem kode og virkelighed. Der bør altså sikres et didaktisk fokus på sammenhænge mellem model og virkelighed – ellers kommer NetLogo som ny udfordring til at skygge for det – komplekse – faglige indhold. En lærer peger i denne sammenhæng på, at hvis NetLogo havde været introduceret i tidligere forløb, ville det virke bedre, mens to lærere fremhæver, at NetLogo er ét blandt virkelig mange forskellige digitale værktøjer, som eleverne introduceres for, og at NetLogo måske ikke er det, der har det mest intuitive og lækre brugerinterface, og at dette kan hæmme anvendelsen på sigt. NetLogo har ifølge disse lærere ikke samme woweffekt som fx Makey Makey eller 3D-briller. Til gengæld oplevede nogle af lærerne også, at koblingen mellem tekstbaseret programmering og visualisering af modeller var meget motiverende for eleverne, som var "superexcited". Som nævnt ovenfor lader det til, at NetLOGO, på trods af at det ikke har et specielt indbydende brugerinterface, har et stort potentiale som værktøj til tekstbaseret programmering og agentbaseret modellering i grundskolen.

## Undervisningsmaterialet

Mens vi kan se, at næsten alle elever gennemførte forløbene og kom frem til et produkt, hæftede lærerne sig i interviews ved, at det havde været svært at nå at gennemføre forløbene, og mange lærere endte med at blive så tidspressede, at de ikke gennemførte alle delelementer af forløbene. For bedre at kunne tilpasse forløbene til den klasse, hvor de skulle gennemføres, efterspurgte en lærer et mere fleksibelt materiale. En anden lærer efterspurgte instruktionsvideoer, så fx elever, der har været fraværende, nemt kan komme i gang på egen hånd.

En anden udfordring ved materialet var ifølge en lærer, at eleverne på et tidspunkt skulle tegne deres ide, før de implementerede den, men det kunne de ikke se pointen i: "... hvor jeg sagde: 'Hey, prøv at tegne jeres forslag først på det her stykke papir', og det var der bare ingen, der tiltog sig, de var bare: 'Jamen, vi går da bare i gang med det her, der er da ikke nogen grund til først at tegne det og så lave det inde i ...'". Det er en kendt udfordring, at elever har en tendens til at ville fokusere på deres første indskydelse i kreative processer (Smith et al., 2015; Christensen et al., 2016), og lærerens erfaringer tyder på, at der kan være behov for en yderligere stilladsering af elevernes idegenerering, så de kan se formålet med at eksternalisere deres ideer (Dix & Gongora, 2011).

## 2. Elevernes udvikling af kompetencer inden for modellering og programmering med udgangspunkt i faglig viden og undersøgelser

Lærerne havde generelt oplevet, at eleverne udviklede deres evner til at programmere, og lærerne var positivt overraskede over, at en del elever, der ellers kunne være svære at engagere i undervisningen, lærte at arbejde med NetLogo-kode. En lærer pegede på et behov for at stilladser eleverne i deres første møde med NetLogo, men de fleste lærere oplevede, at eleverne blev bedre til at programmere: "Mine elever, som gik fra aldrig at kunne kode noget som helst til at introducere sliders, farver og former osv. og kan analysere og ændre og skrive koder – dér er der jo sket en kæmpe, virkelig fed udvikling". Ifølge lærerne tilegnede eleverne sig altså erfaringer med at programmere i et tekstbaseret programmeringssprog, hvilket må antages at styrke elevernes teknologiske handleevne, som defineret i teknologiforståelse som forsøgsfaglighed i folkeskole (BUVM, 2018).

## Eleverne har fået et fagfagligt udbytte

De fleste af lærerne oplevede, at eleverne igennem forløbene fik et fagfagligt udbytte. En lærer fra geografi fortalte fx om, hvordan han tidligere har prøvet at tale med eleverne om signaturforklaringer på et kort, hvor det røg "ind ad det ene øre, og ud af det andet", mens eleverne i dette forløb "... begyndte at få øjnene op for, den del i hvert fald, at man går lige, man kigger lige rundt først, og ser, hvordan, hvor er der noget, der kan hjælpe mig lidt på vej". Brugen af digitale kort med viden, der var relevant for modelleringen, gjorde det meningsfuldt at forstå signaturforklaringerne. En lærer oplevede imidlertid, at eleverne ikke selv mente, at de havde lært noget af det fagfaglige indhold i deres forløb. En anden lærer fra samme forløb mente, at det måske var, fordi programmeringen havde været svær og derfor kom til at fylde meget, mens en tredje lærer oplevede, at eleverne netop havde lært noget fagfagligt, men at de ikke

havde lært at programmere. Billedet er altså ikke helt entydigt, men som nævnt oplevede lærerne generelt, at eleverne også fik et fagfagligt udbytte. Det tyder til gengæld også på, at for stor kompleksitet i programmeringselementerne kan have en negativ effekt på elevernes fagfaglige udbytte.

## Har eleverne lært noget om at vurdere computermodeller?

En enkelt lærer mente, at arbejdet med modeller i NetLogo i dette forløb (omkring energiforsyninger og klimaløsninger) kunne have givet eleverne "... et perspektiv på, der giver dem et blik for, at det er faktisk ikke så svært at lave de her modeller og få dem til at gøre lige præcis det, du gerne vil have. Så kan man stole på alle de modeller, man ellers finder?" Læreren peger altså på, at der potentielt kan være en læring i forløbet, som kan danne baggrund for elevernes udvikling af evner til kritisk stillingtagen i forhold til digitale modeller i deres omverden. Det tyder på, at arbejdet med agentbaseret modellering har potentiale til at forberede eleverne til det, der i forsøgsfagligheden teknologiforståelse er blevet døbt digital myndiggørelse.

## Elevernes engagement

De interviewede lærere havde generelt oplevet eleverne som motiverede i forløbene – både ift. programmering i NetLogo og i forhold til at skulle optage videoer, hvor de skulle "sælge" deres produkt. En lærer omtalte elever som "superexcitede" ift. at arbejde i NetLogo. En anden lærer pegede også på, at de havde gjort følsomhedsanalysen i geografiforløbet til en slags leg, og at det havde været motiverende for eleverne.

Nogle lærere mente, at det havde været engagerende, at NetLogo-koden var synlig og at processen var synlig, så det ikke kun foregik "inde i hovedet". Læreren kunne se processen og anerkende arbejdet, også når der var fejl i koden. En lærer nævnte også, at det var godt for elever, der havde "et had til at læse", at de i stedet kunne arbejde med at programmere og vise, hvad de fandt frem til.

To lærere pegede på, at eleverne faktisk var motiverede for at rette fejl, og det var lærerne ikke vant til: "når de opdager, at der er noget, der ikke virker, så bliver de frustrerede, men jeg synes også samtidig, de bliver motiveret til at finde ud af – okay, hvad virker så? (...) det er ikke noget, jeg ser normalt ved dem". En lærer nævnte, at man måske kunne blive inspireret til at tage denne fejlmodighed med ind i andre fag: "Der er et eller andet her, der er en motiverende faktor, som vi ikke bruger i det daglige, og den skal vi til at bruge". Samtidig omtalte denne lærer også en vanskelig gruppe af piger, som i dette projekt havde engageret sig helt i programmeringen. Samlet set havde lærerne altså oplevet eleverne som motiverede i en grad, som de var blevet overraskede over.

## 3. Behov for stilladsering af eleverne og kompetenceudvikling af lærerne

Erfaringerne hos de lærere, vi interviewede, var, at så godt som alle eleverne gennemførte undervisningsforløbene og producerede de digitale artefakter, der var forventet i de tre forløb. Også selv om nogle af eleverne blev udfordret, da de blev præsenteret for NetLogo:

*"det ser meget farligt ud, det NetLogo, og uh, der en kodedel. Og en masse udtryk, vi skal have interface, vi skal have input og output, og ting, de pludselig skal forstå. Og fordi vi prøver at holde os i det, i de begreber og den begrebsverden, så var der nogle, der svømmede lidt, og nogle var med hele vejen. Så jeg tror lidt, at det var en blandet landhandel, men overordnet set med stilladsering fik vi jo dem alle sammen igennem, og skifte farve og lave de her ting, og det virkede rigtig godt, når man så kom derhen til".*

Eleverne deltager med meget forskellige forudsætninger i forhold til kodning. Eleverne har mødt kodning i skolen, for eksempel i form af Scratch, men ofte i et mindre omfang. Men nogle elever har interesse for kodning og arbejder engageret med det uden for skoleregi, og det giver et stort spænd af elevkompetencer at skulle stilladser. En lærer

pegede på et behov for at stilladsere eleverne i deres første møde med NetLogo, som de syntes så "farligt ud", men de fleste lærere oplevede, at eleverne blev bedre til at programmere:

Med hensyn til kompetenceudvikling peger en lærer på, at det var vigtigt, at der var tid til at sætte sig ind i programmet, før man brugte det i undervisningen. En anden mente, at hvis man ikke i forvejen var vant til at programmere (det var han), ville det "tage meget lang tid at lære", og en tredje mente, at det var gavnligt på forhånd at have en forståelse for programmeringssprogs modus operandi. Til gengæld var der også en lærer, der mente, at vores lille kursus havde været rigeligt til at komme i gang, og at det ikke ville have været bedre at starte med et udbygget kursus i NetLogo.

En lærer peger på, at han tror, at "hvis man skal have det her til at lykkes, så skal man blive ved med at tage fat i de lærere, der gerne vil, de behøver ikke at kunne det, men bare det, at de gerne vil".

Dette peger på en af de store udfordringer i at arbejde med teknologiforståelse i skolen, da teknologiforståelsesfagligheden ikke er en obligatorisk del af læreruddannelsen. Ingen lærere i skolen er altså uddannet i fagligheden på et niveau, der svarer til et undervisningsfag i skolen (gap-analysen:

<https://xn--danskeprofessionshjskoler-xtc.dk/wp-content/uploads/2021/01/gap-analyse.2021.pdf>).

Ud over computationel kompetence peger en lærer også på, at undervisningen havde krævet, at man som lærer "... kunne fungere i et dynamisk rum ...", mens en anden omtalte det som at "agere i kaos".

#### **4. Nye tværfaglige sammenhænge (afdækning af de kompetencemål i STEM-fagene og teknologiforståelse, der kan dækkes via tilgangen)**

De tre nyudviklede undervisningsforløb i engineering, teknologiforståelse og naturfagene (& matematik) skaber en ny unik tværfaglighed, der kan tilbyde problemorienterede faglige forløb, der sigter mod at bidrage til at løse problemet gennem et digitalt artefakt udviklet af eleverne. Den problemrettede og problemløsende del af arbejdet rammesættes af engineering-processen, og en lærer pointerer, at engineering-modellen tilbyder noget særligt i forhold til mange af de øvrige innovations- og designmodeller, der er kommet ind i skolens praksis gennem de sidste 10 år: "mange af de innovationsmodeller, der findes, de går aldrig rigtig fagligt i dybden, hvor der i engineering er en undersøgelsesdel, som faktisk går ind i naturvidenskaben, og der synes jeg nogle gange, så bliver de andre, de bliver sådan lidt for ikke fagfaglige".

Lærerne vægter dog engineering forskelligt i håndteringen af det tværfaglige forløb, blandt andet fordi de oplever, at elevernes kompetencer inden for det naturfaglige område og det teknologifaglige område er på forskellige niveauer.

Det bliver også beskrevet, at teknologifagligheden og det computationelle arbejde skaber fornyet interesse for det naturfaglige område: "Jeg vil sige, det var lidt sjovt med brandpunktskortet i forhold til Australien. Der var der en del, der begyndte at få øjnene op for signaturforklaringer, for det havde jeg prøvet før, og så refererer det til, hvor der er byer og togsinker og jernbaner henne, og det ryger ind ad det ene øre og ud af det andet. Men her der skulle de bruge det til at analysere decideret: 'Kig på signaturforklaringen, så kan I jo godt selv regne ud, hvad der foregår', og den der forståelse af, at når der bliver lavet et eller andet form for temakort, så – de signaturforklaringer dér, er det en hjælp og oftest skrevet på en måde, så man kan komme til dem (...)".

#### **Analyse af spørgeskemaer**

Både elever og lærere besvarede spørgeskemaer som en del af evalueringen af projektet. Lærerne besvarede skemaer både før og efter afprøvningen af forløb, mens eleverne besvarede efter deltagelse i forløbene. Lærerbesvarelserne er baseret på 8 lærere, hvoraf 70 % har over 6 års erfaring og derfor kan betegnes som erfarne. Kun 19 % af lærerne angiver at have erfaring med undervisning i teknologiforståelse, mens andre STEM-fag er ligeligt repræsenteret med ca. 60-80 % af lærerne som undervisere.

I alt gennemførte 129 elever hele spørgeskemaundersøgelsen efter at have deltaget i et af de tre forløb. 20 elever fra geografiforløbet besvarede, 19 elever fra fysik/kemi-forløbet og 90 elever fra biologiforløbet. Besvarelsene fra de tre elevgrupper testes med en Anova one-way statistisk analyse i Excel (signifikansniveau .95).

Uheldigvis var der ikke tid til at gennemføre en test af spørgsmålene til lærernes før- og efter-spørgeskemaer og til elevernes spørgeskemaer. Derfor blev spørgsmålene ikke afprøvet på en uafhængig gruppe af elever og efterfølgende tilrettet, inden de blev anvendt i spørgeskemaet. Dette burde forbedres ved en eventuel opfølgende undersøgelse. Det er velkendt, at det i forbindelse med spørgeskemaundersøgelser er hensigtsmæssigt at teste spørgsmålene, før man bruger dem til dataindsamling, da faren for, at spørgsmålene bliver misforstået eller slet ikke forstået, er ret stor, og graden af misforståelser kan ikke bagefter vurderes. Derfor skal nedenstående konklusioner på baggrund af spørgeskemaer tages med forbehold.

### **1. Didaktiske spørgsmål – potentialet ved at bygge på engineering og anvendelsen af computermodeller i denne kontekst**

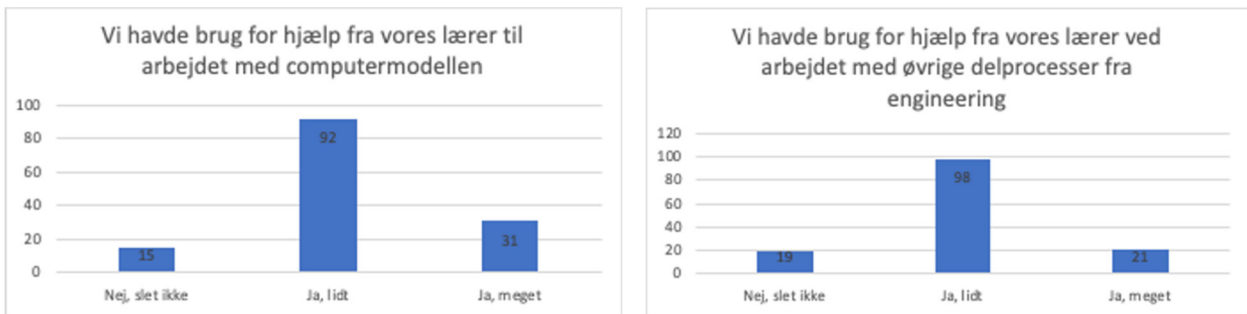
Eleverne blev i spørgeskemaundersøgelsen spurgt til, om de oplevede, at computermodellerne var anvendelige i deres designprocesser. 58 % af eleverne syntes, at "Interfacet (simuleringen)" var en hjælp til at finde en løsning, og et tilsvarende antal elever (60 %) syntes, at computerkoden var en hjælp til at finde en løsning. Det var altså et flertal af eleverne, der oplevede, at modellerne med fordel kunne inddrages, og at de var anvendelige i processen.

I vurderingen af, hvorvidt computermodellen var en hjælp til at finde en løsning på problemstillingen, svarede i alt 64 % af eleverne, at den var en hjælp. Heraf svarede 8 %, at den var en meget stor hjælp. Der var til gengæld forskel på elevernes oplevelse af, hvorvidt arbejdet med computermodellen var en hjælp til at designe en løsning. Computermodellerne var meget forskellige i kompleksitet og anvendelsesmuligheder ift. at finde løsninger på den centrale problemstilling i forløbene. Forskellene i spørgeskemaundersøgelsen tyder på, at en meget simpel og direkte anvendelig model lader til at være nemmest for eleverne at arbejde med og bruge til design af løsninger. Dermed tyder det på, at hvis computermodeller skal anvendes i naturfag i et EDP-perspektiv, kan det være en fordel, at modellerne er meget simple.

### **2. Elevernes udvikling af kompetencer inden for modellering og programmering med udgangspunkt i faglig viden og undersøgelser**

I spørgeskemaundersøgelsen målte vi elevernes evner ved at spørge til deres behov for hjælp fra deres lærer i processen. Som det kan ses ved sammenligning af de to delfigurer i figur 1, var der lidt flere elever, der vurderede at have mere brug for hjælp fra læreren til computermodellen end til de øvrige dele af engineering-processen, men denne forskel var ikke statistisk signifikant. Det tyder altså på, at eleverne ikke opfattede arbejdet med computermodellen som væsentlig sværere end de andre arbejdsprocesser i forløbene. Samtidig var det kun i alt 31 af de i 138 elever (22 %), der i alt besvarede dette spørgsmål, som vurderede, at de havde meget brug for læreren. Det tyder på, at eleverne i gennemsnit ikke har opfattet computermodellerne som specielt svært tilgængelige, men omvendt var det kun 15 elever (11 %), der slet ikke havde haft brug for læreren til arbejdet med computermodellerne, så læreren er stadig et nødvendigt stillads i den proces. Samlet set tyder dataene på, at eleverne generelt set oplevede, at de var i stand til at anvende modellerne i engineering-processerne.





Tabel 4. Elevernes vurdering af behov for hjælp fra deres lærer til arbejdet med hhv. computermodellen og øvrige engineering-processer i forløbet. Y-aksen angiver antal elever.

I gennemsnit rapporterer 88 % af alle elever, at de selv lavede ændringer i computermodellen, og 79 % af alle elever, at modellen blev bedre af deres ændringer. Der var dog forskel mellem de tre grupper af elever. Det tyder ikke overraskende på, at jo mere komplekse computermodellerne er, jo vanskeligere er det for eleverne at lave meningsfulde ændringer i dem. Endelig er konklusionen altså, at eleverne overordnet selv mener, at de er i stand til at lave ændringer i computermodellerne, hvis disse modeller er tilstrækkelig simple.

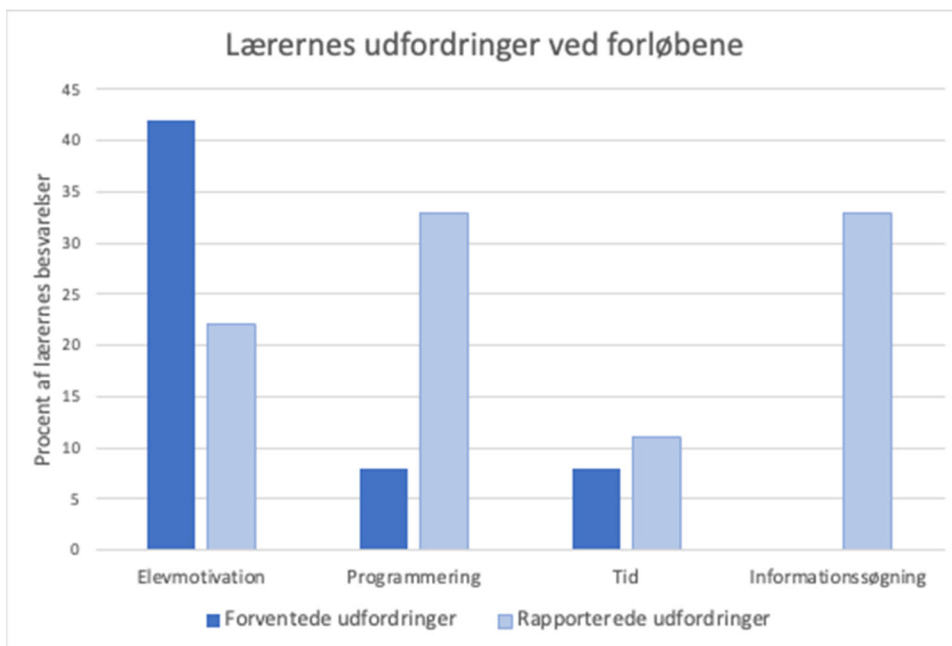


### 3. Behov for stilladsering af eleverne og kompetenceudvikling af lærerne

Som beskrevet ovenfor udfyldte lærerne spørgeskemaer både før og efter forløbet for at se, om deres oplevelse af forløbene levede op til lærernes forventninger. Det viste sig, at lærerne oplevede, at eleverne havde haft mindre brug for stilladsering end forventet, både i forbindelse med elevernes arbejde med computerkoden, det faglige, modellens output af data, fysiske undersøgelser, undersøgelser ved hjælp af computermodellen og til at tydeliggøre sammenhængen mellem model og den fysiske verden.

Lærerne havde for eksempel forventet brug af "ad hoc"-stilladsering i forbindelse med elevernes arbejde med nødvendig, som lærerne havde forventet. Dataene tyder altså på, at eleverne med en stilladseringsgrad, som var håndterbar for lærerne, var i stand til funktionelt at kunne redesigne eksisterende computermodeller i problemløsende designprocesser.

Ud over at forholde sig til elevernes behov for stilladsering blev lærerne også bedt om at forholde sig til deres egne kompetencer og udfordringer (se tabel 5). Før forløbene havde lærerne forudset, at deres største udfordringer ville blive elevmotivation (42 %), programmering (8 %) og tid (8 %). Efter forløbene beskriver lærerne imidlertid deres udfordringer i kategorierne informationssøgning (33 %), programmering (33 %), elevmotivation (22 %) og tid (11 %), se tabel 5. Eleverne var altså tilsyneladende mere motiverede for forløbene, end lærerne havde regnet med. Til gengæld var lærerne selv mere udfordrede i forhold til programmering og informationssøgning, end de havde regnet med at blive. Det tyder på, dels at der er et behov for yderligere kompetenceudvikling af lærerne, dels at forløbene med fordel kan laves, så der er mindre behov for informationssøgning (medmindre man vil tilføje det som et mål med undervisningen).



Tabel 5. Lærernes udfordringer ved forløbene. De forventede udfordringer er beskrevet af lærerne inden deres arbejde med forløbene i klasserne, og de rapporterede udfordringer er beskrevet af lærerne efter arbejdet med forløbene.

Efter forløbene nævner lærerne, at de har opnået kompetencer i programmering (22 %), forståelse for værktøjet NetLogo (44 %) og modellering (22 %), så det tyder på, at kompetenceudvikling af lærere i ECT-lignende forløb kan ske i forbindelse med læreres afprøvning af forløb som dem, der er beskrevet her, i deres egen praksis.



#### 4. Nye tværfaglige sammenhænge (afdækning af de kompetencemål i STEM-fagene og teknologiforståelse, der kan dækkes via tilgangen)

Lærerne beskriver i besvarelsen af spørgeskemaerne, at eleverne får styrket faglighederne inden for teknologiforståelse, STEM-fag og den naturvidenskabelige undersøgelsesmetode ved arbejdet med disse forløb. Mht. hvilke kompetencer eleverne får styrket i disse forløb, nævner lærerne oftest undersøgelseskompetencen, samarbejdskompetencen og modelleringskompetencen.

Det kan være svært at udlede, hvilke præcise undervisningsmål der understøttes af en given tilgang, efter et pilotprojekt som dette, men når eleverne bliver bedt om med egne ord at beskrive, hvordan computermodellen kan anvendes til løsning af et tværfagligt problem, beskriver nogle elever næsten eksperimenter med computermodellen, hvor fx temperaturen varieres (som i et køleskab, en fryser og ved stuetemperatur). Andre elever nævner også, at det at forstå fænomenet "*mere præcist*", "*forstå problemet*" og "*visualisere/se*" problemet er vigtige egenskaber, som computermodellen bidrager med. Enkelte elever nævner, at modellen kunne bruges til at "*få nye tanker*", "*lave en plan*", "*se, hvordan fremtiden måske kunne ende ud*" og "*give forkerte data*". Det tyder altså på, at nogle elever i hvert fald selv har haft en oplevelse af, at de er blevet bedre i stand til at anvende modellering til at forstå deres omverden og til at forudsige noget om fremtiden, og måske tyder det også på, at nogle af eleverne er blevet bedre i stand til at forholde sig kritisk til andres anvendelse af modeller.

#### Analyse af elevpræsentationer

De tre undervisningsforløb var alle tilrettelagt efter engineering-design-processen. Derfor afsluttede elevgrupperne forløbet med en præsentation af deres forslag til løsning på den udfordring, de havde arbejdet med (se figuren af engineering-design-processen på side 6).

I det følgende analyseres 26 gruppepræsentationer fra tre klasser: 9 præsentationer fra en klasse, der havde arbejdet med madspild, 7 fra en klasse, der havde arbejdet med CO<sub>2</sub>-udledning, og 10 fra en klasse, der havde arbejdet med skovbrand. Alle grupper præsenterede i form af en video af 2-4 minutters varighed.

Der eksisterer mange erfaringer med undervisningsforløb, som er tilrettelagt efter engineering-tilgangen. Derfor var der på forhånd visse forventninger til, hvordan grupperne kunne eller "burde" præsentere deres løsning. Men fordi der i ECT-projektet er tale om et allerførste forsøg med nye elementer i engineering-forløb, blev analysen holdt åben på den måde, at der ikke på forhånd var formuleret temaer for en analyse.

Præsentationsvideoerne blev analyseret af en deltager med kendskab til projektet og til de tre undervisningsforløb. Efter et første gennemsyn blev der formuleret 7 temaer, som var centrale i projektsammenhæng og dækkende i forhold til de 26 præsentationer.

Der blev formuleret følgende temaer:

1. Forholdet mellem udfordringen og løsningen
2. Gruppens brug af data, databehandling og datarepræsentation
3. Modellens rolle i forhold til løsningen
4. Gruppens arbejde med justeringer/ændringer af computerkoden
5. Gruppens brug af praktiske undersøgelser
6. Gruppens brug af STEM-faglig viden
7. Præsentationen bedømt i forhold til engineering-processen.

Derefter blev de 26 videoer gennemgået igen, nu med henblik på de 7 temaer.

En lang række forskelle mellem videoerne skyldes (naturligvis), at eleverne/grupperne ikke var ens, at klasserne ikke var ens, og at de lærere, der gennemførte undervisningsforløbene, ikke var ens. Der er i forbindelse med analysen ikke

gjort forsøg på at kortlægge denne type forskelle, og en mulighed kunne være, at især forskelle mellem lærerholdninger og mellem de 26 elevgrupper kunne overdøve andre forskelle og dermed give et mudret billede af fremlæggelserne, så det ikke var muligt at uddrage mere generelle karakteristika.

Imidlertid viste det sig, at der – på tværs af de omtalte forskelle mellem klasserne – var tydelige og karakteristiske forskelle mellem præsentationerne fra hvert af de tre undervisningsforløb. Derfor blev analyseresultaterne samlet i tre grupper – en gruppe for hvert de tre forløb. Resultaterne for biologiforløbet er vist i tabel 6 side 38.

De tre undervisningsforløb var alle udarbejdet med udgangspunkt i EDP-tilgangen og med henblik på at afprøve de samme elementer i undervisningen. Inden for denne ramme afveg de tre forløb på en række punkter. Fx var der forskel på, om eleverne i et givet forløb kun fik udleveret én computermodel, eller de fik flere modeller af voksende kompleksitet.

Der var også forskel på graden af faglig kompleksitet i forløbene, idet ét forløb tilstræbte, at modellen, og dermed den indgående faglighed, skulle være så simpel som muligt, mens et andet forløb tillod større kompleksitet ud fra et ønske om at gøre model og problemstilling mere virkelighedstro. En yderligere forskel var, at ét forløb tog udgangspunkt i, at grupperne som introduktion udførte (fysiske) naturfaglige undersøgelser, hvorimod dette aspekt var helt udeladt i et andet forløb. Der var også forskelle i de krav, der blev stillet til gruppernes løsning – blandt andet i detaljeringsgraden af formuleringen af kravene.

## Konklusioner

Følgende konklusioner på baggrund af analysen af gruppernes præsentationer bygger på den formodning, at gruppernes fremlæggelser afspejler sådanne forskelle mellem forløbene, og at præsentationerne dermed også implicit genspejler de processer, grupperne havde været igennem i forbindelse med deres forløb.

Det betyder, at forskellene i de tre forløb afdækker og illustrerer betydelig flere sammenhænge, end det ville have været muligt at lokalisere, dersom vi kun havde testet ét forløb. Det betyder også, at flere af konklusionerne ikke gælder på tværs af alle tre forløb, men bygger på gentagne iagttagelser fremlæggelser hørende til ét forløb og derfor ikke har præg af tilfældigheder.

En vigtig og lidt uventet erfaring er, at elevernes fremlæggelser og dermed – må man formode – de processer, grupperne har været igennem, er meget følsomme for, hvorledes et undervisningsforløb udformes, herunder kompleksiteten af den indgående computermodel, kompleksiteten af den indgående STEM-faglighed og formuleringer af krav til elevernes produkt/løsning.

Nedenstående konklusioner er grupperet i forhold til de allerede indførte didaktiske kategorier. Da der er tale om elevprodukter, giver det dog ikke mening at konkludere i forhold til område 3 om behov for stilladsering af eleverne og kompetenceudvikling af lærerne.<sup>2</sup>

### 1. Potentialet ved at bygge på engineering og anvendelsen af computermodeller i denne kontekst

Flertallet af grupperne demonstrerer, hvordan de kunne bruge en fagligt baseret model i en engineering-proces, hvor de kom frem til en løsning. Nogle grupper fremlagde deres løsning som et egentligt produkt i form af en skitse til en app. Appen var bygget over en udgave af modellen, som gruppen selv havde forbedret ved at ændre i koden. Andre grupper præsenterede deres løsning ved at ændre i computerkoden, så simuleringen viste en løsning. Andre grupper igen argumenterede for, hvordan udfordringen kunne løses ved at ændre på parametre i simuleringen.

<sup>2</sup> Der skelnes i det følgende mellem elevernes arbejde med hhv. "model" og "simulering". Modellen er den kode (det program), som styrer computeren, herunder hvordan den behandler relevante data. Simuleringen er modellens fremtrædelse i form af det billede på skærmen, som computeren viser, når den eksekverer koden. Ændringer i modellen/koden bliver synlige i simuleringen. På den anden side er det muligt at arbejde med en simulering uden at have kendskab til den bagvedliggende kode.

Ud fra præsentationsvideoerne er det ikke muligt at afgøre i detaljer, hvorledes grupperne har arbejdet med alle delprocesser i engineering-processen. Men fremlæggelserne godtgør, hvorledes grupperne har arbejdet med simuleringen som et instrument i fasen *Undersøge* for at forstå det faglige fænomen, der ligger bag deres udfordring – hhv. bakterievækst, udledning af CO<sub>2</sub> og forløb af skovbrande. Det fremgår implicit, men tydeligt, at mange grupper også har arbejdet intenst – og computationelt – med delprocesserne *Få ideer*, *Forbedre* og *Konkretiser*. Grupper, der præsenterede skitser til en app, har desuden arbejdet med delprocessen *Konstruer*.

### **Elevernes udvikling af kompetencer inden for modellering og programmering med udgangspunkt i faglig viden og undersøgelser**

I så godt som alle fremlæggelser forklarer og/eller demonstrerer gruppen, hvordan de har arbejdet med at læse, fortolke og ændre i computerkoden. Mange grupper har ændret eller forbedret modellen (koden) gennem indførelse af nye fagligt begrundede modelparametre. Nogle grupper omtaler eksplicit, hvordan de har kopieret en subrutine i koden, ændret den efter egne overvejelser og derefter kopieret den nye rutine ind i koden.

Nogle grupper har tydeligt arbejdet med at forstå det faglige fænomen dybere (fx bakterievækst eller skovbrand) og omsætte denne forståelse til ændringer i modellen eller simuleringen.

Flere grupper bruger simuleringens uddata i form af tal eller grafer som input i deres app eller i deres argumentation for, at de har fundet en løsning. Et mindretal af grupper begrænser sig til at bruge grafikken i simuleringen som argument for deres løsning.

Nogle grupper er meget bevidste om, at inddata er afgørende for modellens opførsel og dermed, hvorledes simuleringen forløber.

Nogle grupper har desuden overvejelser over, hvordan de bedst kan organisere og præsentere uddata for en eventuel bruger af deres app.

En række grupper omtaler deres behov for at få mere faglig viden for bedre at forstå modellen og dermed som grundlag for at arbejde videre med at forbedre den.

### **Nye tværfaglige sammenhænge (afdækning af de kompetencemål i STEM-fagene og teknologiforståelse, der kan dækkes via tilgangen)**

På baggrund af fremlæggelserne er det – på den ene side – vanskeligt at få et konkret indtryk af, hvordan en given gruppe har arbejdet med deres faglige viden. På den anden side er det tydeligt i fremlæggelserne, at eleverne har levet sig ind i det faglige problem og har mange faglige overvejelser om betydningen af fx tid, temperatur, oxygenniveauer, vindhastighed, antændelsestemperatur osv.

En konklusion er, at arbejdet med modellen/simuleringen gør det konkret for eleverne, hvordan modellen bygger på en faglig forståelse af det modellerede fænomen. Det samme gør sig gældende for den matematik, der bruges i koden. Derved bliver anvendelsesaspektet af faglig viden tydeligt.

Fremlæggelserne demonstrerer, at i lignende forløb vil det være muligt og relevant at opstille konkrete mål både for udvikling af faglig viden og for kompetencer i naturfag (undersøgelse, modellering, kommunikation), i matematik (problembehandling, modellering, ræsonnement og tankegang, kommunikation) samt i teknologiforståelse (computationel tænkning, teknologisk handleevne, digital design og designprocesser).

Eksempel på analyse af elevernes videopræsentationer

Analysetema	Forløb om Madspild (9 fremlæggelser)
<b>Udfordring/ Løsning/ produkt</b>	Langt de fleste grupper har skitseret en app, som de fremlægger som et bidrag til at løse udfordringen med at nedbringe madspild.
<b>Databehandling og datarepræsentation</b>	De fleste grupper har gode overvejelser om både input af data og output, herunder præsentation og organisering af output.
<b>Brug af modellen i løsningen</b>	Nogle grupper siger det ligeud, andre tager for givet, at vi forstår, at appen er baseret på modellen. En gruppe siger: "modellen kører i baggrunden i appen". Det er tydeligt, at grupperne har styr på, hvad det er, modellen/simuleringen gør i forhold til vækst af bakterier.
<b>Ændring/justering af koden</b>	Fem af grupperne omtaler og forklarer, hvordan de har været inde i koden og lavet ændringer/tilføjelser.
<b>Praktiske undersøgelser</b>	Fire af grupperne refererer til praktiske undersøgelser af bakterievækst.
<b>Faglig indsigt</b>	Det er vanskeligt at få et indtryk af elevernes brug af biologisk viden. Til at afgøre, om en fødevarer er spiselig, bruger de det totale antal mikroorganismer. Men de overvejer tydeligt betydningen af temperatur, lys og ilt. Der er ikke eksempler på egentlige faglige misforståelser.
<b>Engineering-proces</b>	Flertallet af grupperne har demonstreret, hvordan de kan bruge en fagligt baseret model i en engineering-proces, hvor de kommer frem til en løsning i form af et produkt.

Tabel 6. Eksempel på analyse af elevernes videopræsentationer som afslutning på madspildsforløbet. Der er lavet tilsvarende analyser af elevpræsentationer til forløb fra hhv. fysik/kemi og geografi.

# Referencer

- Christensen, K. S., Hjorth, M., Iversen, O. S., & Blikstein, P. (2016). Towards a formal assessment of design literacy: Analyzing K-12 students' stance towards inquiry. *Design Studies*, 46, 125-151.
- Dix, A., & Gongora, L. (2011, October). Externalisation and design. I: *Proceedings of the second conference on creativity and innovation in design* (31-42).
- Lee, I. et al. (2011). Computational thinking for youth in practice. *Acm Inroads*, 2(1), 32-37.
- Musaeus, L. H., & Musaeus, P. (2019, February). Computational thinking in the Danish high school: Learning coding, modeling, and content knowledge with NetLogo. I: *Proceedings of the 50th ACM technical symposium on computer science education* (913-919).
- Musaeus, L. H., Tatar, D., & Musaeus, P. (2022). Computational modelling in high school biology: A teaching intervention. *Journal of Biological Education*, DOI: 10.1080/00219266.2022.2118353
- Smith, R. C., Iversen, O. S., & Hjorth, M. (2015). Design thinking for digital fabrication in education. *International Journal of Child-Computer Interaction*, 5, 20-28.
- Undervisningsministeriet (2018). Læseplan for forsøgsfaget teknologiforståelse. Hentet d. 18/10-2022 på <https://emu.dk/sites/default/files/2019-02/GSK.%20L%C3%A6seplan.Tilg%C3%A6ngelig.%20Teknologiforst%C3%A5else.%20pdf.pdf>
- Wilensky, U. (1999). NetLogo, Center for Connected Learning and Computer-Based Modeling, Northwestern University. <http://ccl.northwestern.edu/netlogo/>
- Yadav, A., & Berges, M. (2019). Computer science pedagogical content knowledge: Characterizing teacher performance. *ACM Transactions on Computing Education (TOCE)*, 19(3), 1-24.