This is my interpretation and my idea of what I think this could look like, feel free to change/amend!

Key Points

- Brainstorming session resulting in a rough influence diagram, separated into two models, and a proposed NetLogo model implementation for Benni & Mabel's current ideas
- From Mabel's research, she doesn't believe that hybrid models are the way forward for the project, at least from current research. This is because the Lake Hybrid example is one ABM (the social side) coupled with one SDM (the lake), as opposed to the previous idea of one ABM (farmers/farms and their interactions) and a SDM for each farm. She is unsure of the feasibility of this
- From Benni's research - found the CRS hydrology model and fell down that rabbit hole. This could be a solution to the issue Mabel pointed out - instead of needing a separate SD model for each farm/field, they become grid cells in the hydrology model

Broad Idea for Model Structure

Our current idea is to implement a coupled model with a NetLogo social system model for the farmers and their practices and a python (-wrapped) grid-based hydrological model for soil moisture/viability. NetLogo patches would represent farmers' fields. The patches would correspond directly to grid cells in the hydrological model so either model component can easily access the state variable values it needs from the other component.
For the hydrological model, we should be able to find a more-or-less ready-made implementation, sparing us coding up equations and finding suitable parameterisations (although we will probably still need to parameterise a few things ourselves).

Proposed NetLogo model implementation - Mini ODD

Agents = farmers/farms

- In charge of a set of patches - fields
- Can interact with patches/each other

Patches = fields

- Each field has an elevation, soil moisture, crop yield, crop type, slope, elevation
- Upon each timestep, each patch has its soil moisture calculated by the coupled Python hydrology Model

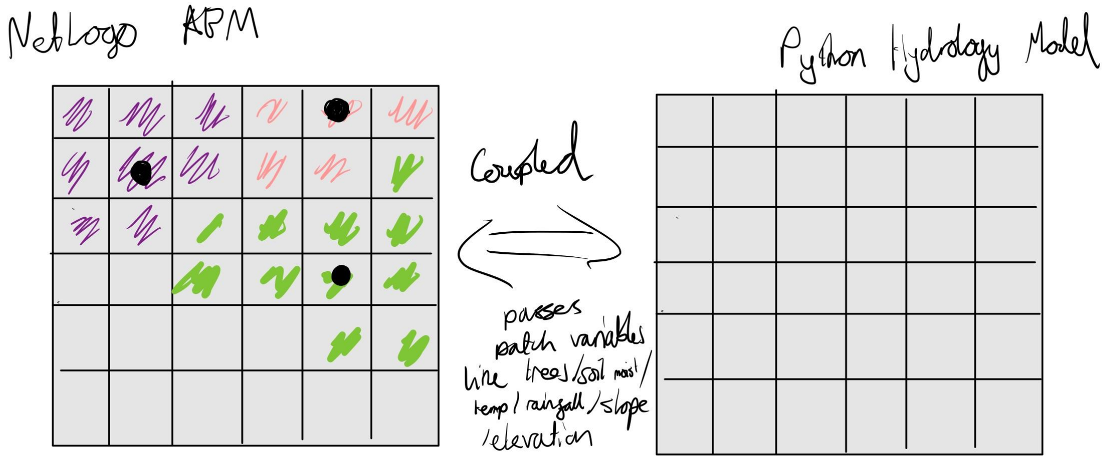
Global Environment

- Temperature - from input of projected temperature rises
    - Same for all patches - impossible to get climate projections at resolutions higher than 1km and hard to get data at resolutions better than ~30km. Local effects are probably small, so it should be fine to apply the same forcing everywhere.
- Rainfall - from input of projected rainfall

        - Same for all patches

Timescales

- Monthly?
    - Definitely need to think about scheduling here - harvesting will only happen once or twice a year. On the other hand, it might be valuable to timestep the hydrology model at steps of 1 day given e.g. surface runoff/percolation can be very different for the same amount of rainfall depending on whether it all falls on the same day or distributed over a couple weeks.


NetLogo ABM
Python Hydrology Model
Coupled
passes patch variables like trees/soil moist/ temp/ rainfall/slope /elevation

All patches / fields of one colour are one farmer

Chat with David about Feasibility

David flagged the following potential issues with our approach:

- The hydrological model might take long to run, so we may need to limit the size of the area modelled
- Finding suitable model code might take a while (too long?), but this is hard to tell. Found a few promising things already.

He also made the following notes and suggestions:

- Our model can just be a proof-of-concept, so having accurate climate data to force it may not be important. We can just come up with a few made-up, but plausible scenarios if that is quicker to do.
- We might want to neglect some effects that lead to complex interactions between grid cells/patches (e.g. elevation and slope of fields) if we find that things are taking long

- Given the potential complexity of the coupler, our paper might end up being more of a model description paper, and less about exploring the model behaviour in depth.