

Appendix 5: Finding which initial conditions and model parameters lead to which IPM strategies

It is unlikely a given IPM strategy will perform well in all scenarios, for example strategy that works well in cases where the yield penalty (Y_D) of the weed is very low is unlikely to be good in cases where the yield penalty is very high. To find the parameters and initial conditions (n_0) that most influence what makes a good IPM strategy we extend the meta-modelling approach outlined in (Coutts & Yokomizo, 2014) to complex multivariate time series outputs (i.e. the sequences of the four sub actions). The meta-modelling approach to global sensitivity analysis is very flexible, and only requires building a function that predicts a models output given a set of parameters. The complication in this case is that the model output is a time series of actions, rather than a single number. To address this we carry out the sensitivity analysis as follows:

1. We use latin hypercube sampling to generate 15000 parameter combinations. We treat the initial conditions as parameters (Rr_{int} , Qq_{int} and N_{int}).
2. For each parameter combination we run the genetic algorithm (Appendix 3) for 100 iterations, experience showed that improvements to the best solution found dramatically slowed down after 60 iterations with a solution set size of 100 and mutation rate or 0.03. After 100 iterations we choose the action sequence with the highest reward as the best action sequence.
3. This resulted in 15000 action sequences. We only use the first 10 time steps of each action sequence, although the reward was calculated over 25 time steps. This ensured that the action chosen at each time step was done so considering rewards from at least 15 time steps in to the future. To make sense of this mass of data we organised these action sequences by finding how they differed in relation to one another. The first step in this process was to build a dissimilarity matrix of each action sequence to all the other action sequences. We use a distance metric from text analysis, longest common sub-sequence, calculated with the 'SimilarityMeasures' (Toohey, 2015) package in R. We allow a positional displacement of one time step, that is actions in sequence 1 are allowed to be matched to actions in sequence 2 that are separated in time by at most one time step. This allows out of phase cycles to have very high similarity. Two action sequences where the same combinations of actions are used, but at different times will have low similarity. We use non-metric multi-dimensional scaling (NMDS), implemented with the 'ecodist' package (Goslee & Urban, 2007) to project this very high dimensional dissimilarity matrix into a lower dimensional space. We call this the solution space and we found that we could project the dissimilarity matrix into an eight dimensional solution space, with a stress of 0.09. Stress measures the difference between the relative location of each solution in the solution space, and the dissimilarity matrix lower values are better. We found that 8 dimensions was a good trade-off between low stress and a manageable number of dimensions. Figure S1 shows how the matches up to a separate hierarchical clustering (using the wardD2 method) we carried out to confirm and aid visualisation of the NMDS.
4. We recorded the eight co-ordinates of each solution in the solution space, along

with the parameter set that generated that solution. This built a dataset 15000 rows long, where each row was a parameter set and a location in the solution space.

5. We use multi-variate boosted regression trees (mvBRT) as our meta-model. mvBRTs were fit using the 'mvtboost' package (Miller *et al.*, 2016). mvBRTs fit a separate boosted regression tree to each dimension of the response, in our case the eight dimensional solution space. At each iteration in each tree the split that most reduced the co-variance between responses is choose. For further details see (Miller *et al.*, 2016). We used a learning rate of 0.01, maximum interaction depth of 5. These tuning parameters lead to a optimum of 12229 trees in each BRT.
6. We interrogate the mvBRT to find the most influential parameters. We also calculate partial dependence plots of the mvBRT to predict where in the solution space a given set of parameters leads to, and use the closest solution to that location to estimate what a solution in the predicted location would look like. We use this as guide to help us explore and and interrogate the complex solution space. However, we have access to the data generating process and so do not need to rely on this estimate. For the actual plots in the results we re-run the genetic algorithm with the desired parameter combinations.

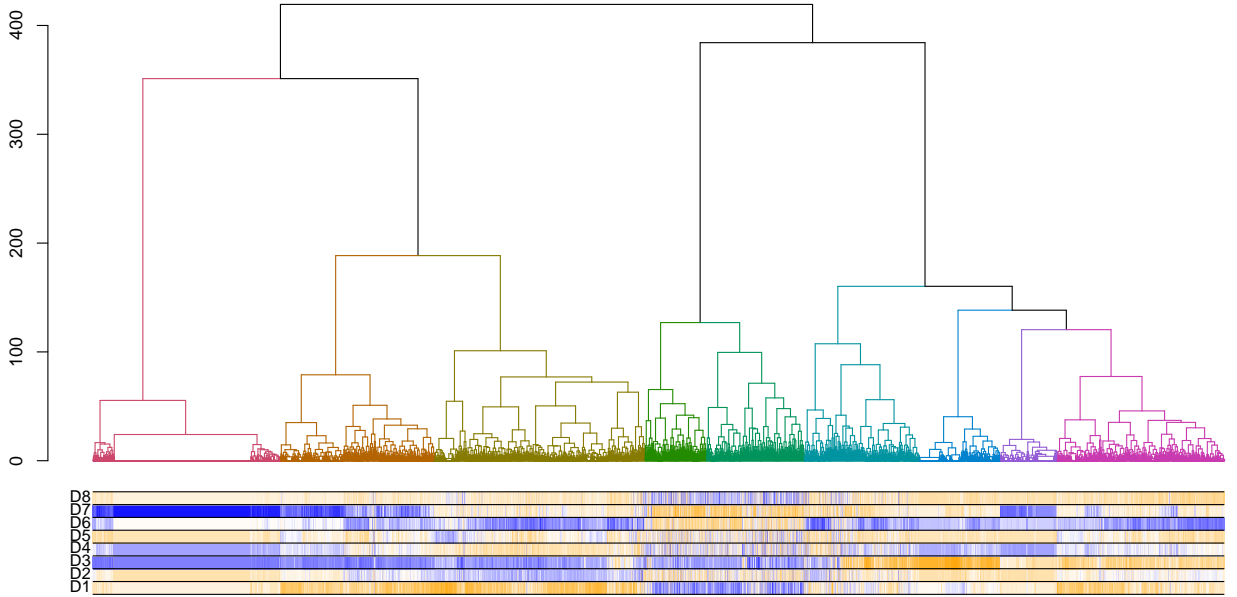


Figure S1: Position of all 15000 IPM strategies found by the genetic algorithm in the eight dimensional solution space and dendrogram built using the wardD2 method. Different colours in the dendrogram highlight the 9 most distinct clusters. Colours in the bars show each solutions position in the solution space. All dimensions are centred on 0 (white), with dark blues being at one extreme and dark oranges a the other (only relative position is meaningful).

Appendix 3 Genetic Algorithm

References

- Coutts, S.R. & Yokomizo, H. (2014) Meta-models as a straightforward approach to the sensitivity analysis of complex models. *Population Ecology*, **56**(1), 7–19.
- Goslee, S.C. & Urban, D.L. (2007) The ecodist package for dissimilarity-based analysis of ecological data. *Journal of Statistical Software*, **22**, 1–19.
- Miller, P.J., Lubke, G.H., McArtor, D.B. & Bergeman, C. (2016) Finding structure in data using multivariate tree boosting. *Psychological methods*, **21**(4), 583.
- Toohey, K. (2015) *SimilarityMeasures: Trajectory Similarity Measures*. URL <https://CRAN.R-project.org/package=SimilarityMeasures>. R package version 1.4.