Carbon Farming Optimiser

Testing the feasibility of assessing on-farm carbon management changes

NSW GOVERNMENT | Department of Primary Industries

Modelling farm-scale trade-offs between production and carbon farming

Oscar Cacho

UNE Business School

Final Report

1 May 2020

NSW Department of Primary Industries

Contents

Background

Research as part of this Agreement supports two projects (Project 4 and 5) under the Carbon Opportunities theme within the NSW DPI Climate Change Research Strategy (CCRS).

Project 4: Provide guidance, advice and assistance to DPI to develop mathematical programming models for spatially distributed representative farming systems, which will assess economic potential of abatement options, for the development of emissions reductions pathways for NSW.

Project 5: Develop simulation models for three Emissions Reduction Fund methods. These models will be based on real-farm case studies to test the ability of a simulation to optimise land use for production and carbon. DPIE will support UNE in collating on-farm data, to develop models and interpret the implications of model outcomes. Model evaluation by the project team will determine the basis for ongoing development of a web-based decision support tool, which is beyond the scope of this Agreement.

Figure 1 shows the relationship between the wider project (to design web-based tools) and this Agreement; activities indicated in green are for this Agreement and those activities for future project deliverables are indicated in orange.

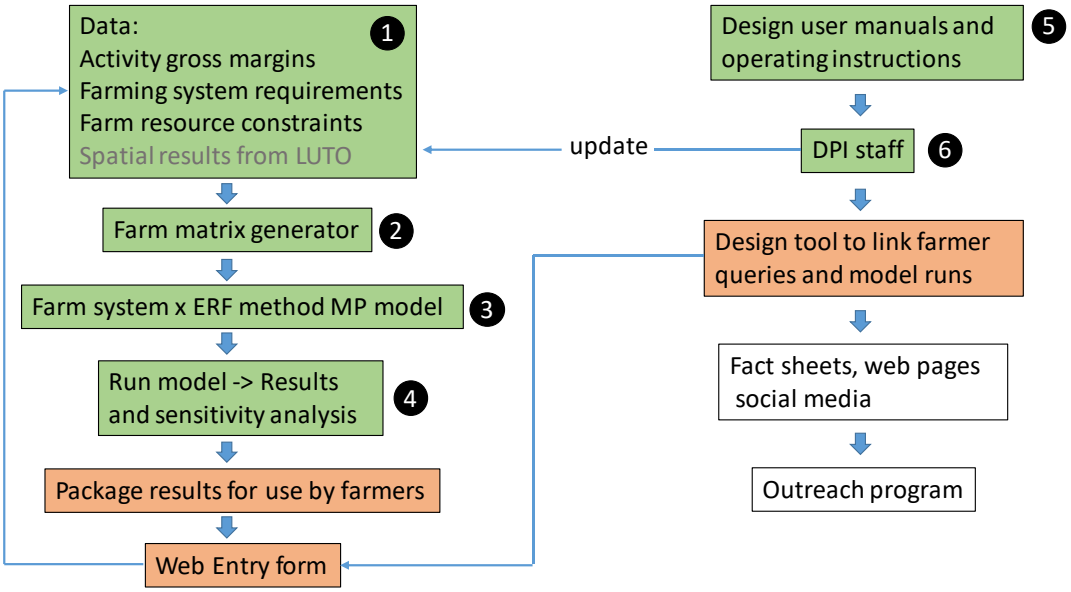


Figure 1. Project workflow, numbered circles represent activities related to this contract.

Collecting on-farm data

The first step in building a whole-farm model is to estimate the gross margins and input requirements of each enterprise on the farm. This means that a template to collect on farm data should be based on a gross margin budget. Additional information related to the constraints faced by the farm is also required, this could consist of seasonal labour requirements, feed requirements for livestock or feed supply from crops and pastures, rotational constraints etc.

The gross margin budgets produced by NSW DPI provide a good starting point in designing templates to collect farm data. PDF versions of the budgets can be accessed at https://www.dpi.nsw.gov.au/agriculture/budgets. Excel worksheets for the most common cropping and livestock activities are available to the project from various DPI staff in charge of maintaining and updating them. The data are scattered around the state and there does not seem to be a central depository.

Designing an activity record

Estimating activity gross margins

Gross margin (*GM*) is defined as the difference between the revenue (*R*) and the variable costs (*C*) for an enterprise. In general terms the gross margin for an enterprise is:

$$GM = R - C = \sum_i p_i y_i - \sum_j c_j x_j \qquad (1)$$

Where the outputs are denoted by *i* and the inputs by *j*; y_i is the yield of output *i* and p_i is its price received per unit; c_j is the cost per unit of input *j* and x_j is the amount used. *R* is the sum of the value of all the outputs, which may include grain and lint for a cotton enterprise, lambs and wool for sheep and so on. *C* is the sum of the variable costs, such as seed, fertiliser, labour etc.

For crops, *GM* is normally stated on a per ha basis. For livestock *GM* is expressed on a per head or per DSE basis. For self-replacing operations, the number of breeding females is used to calculate *GM* per head. All the variables required for equation (1) can be taken directly from the DPI gross margin budgets and then adjusted for the particular farm circumstances.

At the heart of a whole-farm linear programming (LP) model is a matrix where the columns are activities (or decision variables) available to the farmer and the rows are constraints. In this sense, an **activity** represents an enterprise on the farm, but a gross margin budget can be split into different activities. For example, pasture and forage production can be treated as different activities (columns) from the grazing livestock enterprise to provide flexibility in selecting the optimal mix of feed sources. The link between the activities is created through seasonal feed requirements for livestock and feed production by pastures and forages.

To create a functional LP model, in addition to *GM* for each enterprise we need information on:

- Requirements for limiting resources (land, labour, feed)
- Net CO_2 emissions
- Environmental indicators (i.e. soil erosion, biodiversity)
- Other constraints such as rotations and quotas

Some of this information may be available from the literature, from FullCAM simulations of ERF methodologies, or from previous DPI work. The template for on-farm data collection will need to consider the variables and constraints listed above. Initial estimates for carbon and environmental indicators may be obtained for each region based on Land-use Trade-offs (LUTO) modelling as part of the project 5 and/or sensitivity analysis.

The need for an activity database

The project will need to consider a large number of activity combinations, including variants to represent adoption of ERF methods. Each activity is a separate column in the LP whole-farm model. Considering different regions, farming systems and climate change scenarios will result in thousands of options that need to be tracked and managed to produce policy analysis that is useful and timely. The same is true when the model is used to provide advice to farmers. This means that coming up with a sound data management strategy is key to the success of the project.

The activity database will consist of records that contain combinations of variables common to all activities (see equation 1) as well as variant components that depend on the type of activity (i.e, crop, livestock, trees). Each record will represent one column of the LP model, and the template used for collection of on-farm data must be based on the relevant record for each activity on the farm.

Composition of activity records

A record in a database can be viewed as a ópackage represent one observation, or one activity in our case. In Matlab and other modelling packages this is called a structure variable (or a *struct* for short). The variables required in a structure for crop and livestock activities are listed below.

General variables:

- Unique identifier
- Type (crop, livestock, pasture, tree)
- Unit of activity (ha, hd, DSE, etc)
- Yield of output (y)
- Price of output (p_y)
- Unit of output (kg, tonne, bales)
- Total revenue (R) defined as in equation (1)
- Variable cost (C) defined as in equation (1)

Constraints (by season if applicable)

- Land
- Labour
- Capital
- Feed produced (MJME)
- Feed required (MJME)
- Interaction with other activities (i.e. rotations)

Environmental variables

- Net CO_2 emissions (SimaPro derived)
- Environmental indicators (biodiversity ï structural integrity and soil carbon loss)

Estimating technical coefficients for new activities

Designing a model that meets the needs of the project will require the creation of new activities that result in changes such as reduced or negative net emissions, improved environmental outcomes, or higher profits. The data for some options will not be available from the sample farms, because the options have not been adopted. In these cases, the best scientific information available will be used to come up with plausible and defensible relationships. Those relationships could be based on bioenergetics, mass balance, nutritional principles, or modelling with packages such as FullCAM, 3PG, APSIM, GrassGro, etc.

Template development

The purpose of the template is to collect data from farmers to populate the LP model at the centre of this contract. To ensure that all the data required are collected, it was necessary to develop and solve a prototype LP model of a mixed farm over a period of several decades to capture carbon accumulation in biomass over time. This involved developing a matrix generator and solving the optimisation model for several scenarios for testing.

In the process of designing the model and the templates, several Zoom meeting were held with the DPI team. These meetings were recorded and they are available in the shared folder *Meetings*. The document *Carbon DPI Project Zoom Meetings.docx*, provides summaries of these meetings.

The first version of the template was designed to be read from Excel into the LP model. This simplifies the process of collecting and exchanging information within the team, as each activity on the farm is described in a separate worksheet. The *LP_Matlab_template.xlsx* contains an example for a mixed farm. This is referred to as the Matlab Template.

The Matlab Template consists of a Model worksheet that describes the farm in general and a set of activity worksheets representing the enterprises available for selection on the farm. Each of these worksheets has a set of named ranges that represent the variables of the model. These named ranges can be accessed through the Name Manager in the Formulas menu of Excel.

The variables in the **Model worksheet** are:

act_names: Names of activities available for selection. Each activity has a worksheet with the same name.

c_avail: The amount of resource available associated with each constraint, these become the RHS variable sin the LP mode (see Training Guide).

eq_names: Names of the equality constraints in the model. These are accounting variables to keep track of gross margins, carbon emissions, and other environmental indicators through time.

max_t: Planning horizon for the model (maximum time to model).

p_e: price of environmental variables, used when there are rewards or taxes for improving or degrading the environment.

plot_area: Area (ha) of each distinct plot of land on the farm.

plot_mat: Matrix showing the plots where each of the activities can be undertaken.

plot_names: Names of plots on the farm.

Note that the term plot is used in a general sense to represent specific types of land, so several paddocks suitable for cropping can be classified together as the cropland plot even if they are not adjacent.

For each of the activities listed in act_names, there is an activity worksheet with the same name. The variables in each of the **activities worksheets** are:

e_mat: environment matrix. Contains coefficients for CO_2 emissions associated with the activity as well as other environmental indicators.

info: text providing information about the activity.

max_age: maximum age for the activity (in years).

name: name of the activities for use in the LP model.

p_y: matrix containing prices of outputs .

v_cost: total variable cost for the activity.

x_mat: matrix of coefficients for the inputs (resources) used in the production of the activity.

x_names: names of the resources on the farm for use in the LP model.

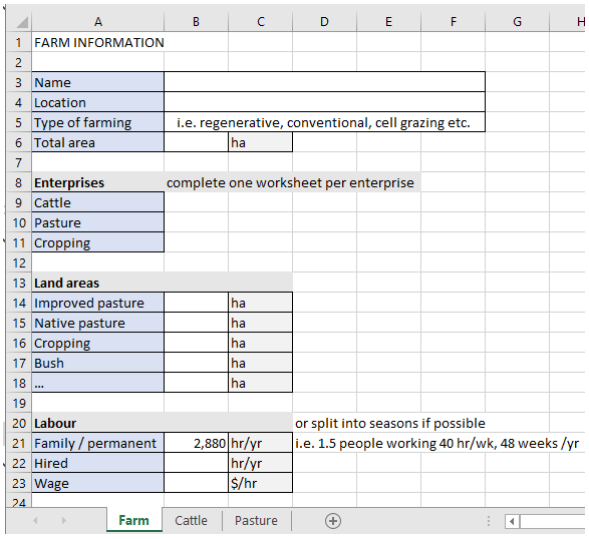
y_mat: matrix of yields for the outputs produced by the activity.

y_names: names of outputs produced by the activity.

The design of this template has been tested and validated by solving a set of optimisation scenarios, but not all information in the template can be obtained directly from farms. In particular, CO_2 emissions and environmental co-benefits need to be obtained from other sources, including the literature, and other models. For example, FullCAM can be used for carbon sequestration in tree-based systems and GrassGro can be used to estimate seasonal pasture yields.

The actual template for on-farm data collection is a simplified version of the full template described above, but it contains only information that can be provided by farmers. Different versions of the template will be produced for different types of farms, and these will be refined as interviews with farmers progress.

The team agreed that the best way to proceed is to design a template for the simplest type of livestock farm; a cattle trading farm on rangeland, and adapt this template for more complex operations as we learn more from the process of interviewing farmers. The draft version of this simple template is in file *Cattle_trading_template.xlsx*.



	A	B	C	D
1	FARM INFORMATION			
2				
3	Name			
4	Location			
5	Type of farming	i.e. regenerative, conventional, cell grazing etc.		
6	Total area		ha	
7				
8	**Enterprises**	complete one worksheet per enterprise		
9	Cattle			
10	Pasture			
11	Cropping			
12				
13	**Land areas**			
14	Improved pasture		ha	
15	Native pasture		ha	
16	Cropping		ha	
17	Bush		ha	
18	...		ha	
19				
20	**Labour**			or split into seasons if possible
21	Family / permanent	2,880	hr/yr	i.e. 1.5 people working 40 hr/wk, 48 weeks /yr
22	Hired		hr/yr	
23	Wage		$/hr	
24				

Farm | Cattle | Pasture

Figure 2. Draft template to collect on-farm data. General farm information

Figure 2 shows the Farm worksheet. In addition to general farm information, we need a listing of all the activities on the farm and the total areas of distinct types of land. For each activity listed here there must be a corresponding activity worksheet. We also need information about the limiting resources on the farm (in addition to land). In this worksheet there is only labour, distinguishing between permanent or family labour and hired labour. If the distribution of labour through the year is important, then labour can be split into seasons. Other potential limiting resources could also be included, for example limited capital may constrain the speed at which tree planting can be undertaken. It may not be as easy to get information on capital constraints from farmers, but we can ask them what other constraints they face and get whatever details they are willing to share.

Figure 3 shows the activity worksheets. In this simple example we have only two, cattle and pasture. In the case of Cattle, information on initial weight and target weight should allow us to calculate the seasonal energy requirements, but any additional information on rates of weight gain that the farmer wants to share would be useful. A selection of the main variable costs is included, but other costs can be added to the list is needed.

	A	B	C	D	E	F
1	BEEF CATTLE TRADING GROSS MARGIN and activity details					
2	Date					
3	Description:	i.e. growing out steers 240kg - 460kg in 12 months				
4	Enterprise Unit:	100	hd			
5	Purchase weight	150	kg/hd			
6	Target weight	400	kg/hd			
7	Over time period	12	months			
8	Pasture:	ha				
9	improved	108				
10	native	0				
11						
12	INCOME:	hd	$/hd	Total value		
13	Steers	98	1,403	137,494		
14						
15	VARIABLE COSTS:	hd	$/hd	Total value		
16	Steer Purchase	100	670	67,000		
17	Cartage to Property	100	15	1,500		
18	Livestock and vet costs*			1,224		
19	Livestock selling cost	98	80	7,840		
20		ha	$/ha	Total value		
21	Fodder crops (12 ha)	12	350	4,200		
22	Hay & Grain or silage			-		
23	Drought feeding costs.			-		
24	Total Variable costs			81,764		
25	GROSS MARGIN			55,730		
26			per hd	557.30		
27						
28	Other variables		Unit			
29	CO2 emissions	2.46	t/hd			
30	Labour required	4.35	hr/hd			

Farm | Cattle | Pas ...

	A	B	C	D
1	Pasture type	improved		
2	Area	200	ha	
3	Seasons could be divided differently			
4	Season	feed prod	units	This will have to be calculated based
5	Spr	4420	MJME	on grazing days, kg DM, or other
6	Sum	6800	MJME	measures.
7	Aut	4080	MJME	
8	Win	2040	MJME	
9				
10				
11	Variable cost	100	$/ha	
12				
13	Other variables		Unit	
14	CO2 emissions	-0.5	t/ha	
15	Labour required		hr/ha	
16				
17	**Environment**			
18	Veg Cover			
19	Biodiversity			
20				
21	**Capital costs**			
22	**Fencing**			
23	Purchase cost		$	
24	Useful life		yr	
25	Salvage value		$	
26	Mantenance		$/yr	
27	**Water points**			
28	Purchase cost		$	
29	Useful life		yr	
30	Salvage value		$	
31	Mantenance		$/yr	
32	**Pasture establishment**			
33	Cultivation	0	$/ha	
34	Sowing	90	$/ha	
35	Fertiliser	95	$/ha	
36	Lime	68	$/ha	
37	Chemicals	20	$/ha	
38				

... | Cattle | Pasture

Figure 3. Draft template to collect on-farm data. Cattle and Pasture activities

In the case of Pasture (right panel in Figure 3), ultimately we want to know how much metabolisable energy is produced per season, but this will need to be calculated based on grazing days, DSEs or whatever unit the farmer uses to describe their pasture production. Regarding seasons, the default is 4 seasons, but this could be changed if the farmer thinks there is a better way of distributing their pasture production through the year. The pasture worksheet also include capital costs. This is important when they have substantial fencing costs for cell grazing for example. We will use this information to estimate annual depreciation rates to be included in the model.

Regarding other variables in the activity worksheets, it is important to obtain a realistic value of the amount of labour required per unit of activity. Another important variable for this project is CO_2 emissions per unit of activity, this probably won't be obtained but is listed here for completeness. The Pasture worksheet also includes environmental variables (vegetation cover and biodiversity) that could be collected from other sources. However, these variables are not required for the operation of the optimisation model.

Brief description of the Carbon Farming Optimiser

The Carbon Farming Optimiser is a multi-period linear programming (MPLP) model of a farming system. It maximises the net present value (NPV) of total gross margins over a desired planning horizon, subject to a set of constraints. Two types of constraints exist:

- resource constraints are inequalities (Ò) in the model to represent limited availability of land, labour, capital, feed etc.
- non-resource constraints are equalities (=) in the model. They are used for three different purposes:
    - accounting for variables such as carbon emissions, soil loss, gross margins etc. from year to year
    - providing *permissions* for multi-year activities to account for the establishment phase that requires investment before the enterprise becomes active.
    - Setting the initial conditions of the farm in terms of enterprise mix at the start of the planning horizon.

The linear programming model can be expressed algebraically as:

Maximise

$$Z = \sum_{t} \sum_{j} f_{tj}\, x_{tj}\, (1 + r)^{-t}$$

Subject to

$$\sum_{t} \sum_{j} A_{tij}\, x_{tj} \leq b_{ti}$$

$$\sum_{t} \sum_{k} E_{tkj}\, x_{tj} = 0$$

$$x_{1m} = x_{0m}$$

$$x_{tm} \geq 0$$

Where the subscripts represent:

- j (1 , n_j) columns of the model for each time period;
- i (1 , n_i) rows related to inequality constraints for each time period;
- k (1 , n_k) rows related to equality constraints for each time period;
- m (1 , n_m) are the columns related to activities that represent farm enterprises;
- t (1 , T) year counter in the panning horizon of T years,

The decision variables x_{tj} represents the decision variable related to activity j in year t, and f_{tj} is the objective function coefficient (the gross margin) for the activity. A_{tij} is an element in the matrix of technical coefficients, representing the requirement of resource i by activity j, in year t, and b_{ti} is the constraint associated with that resource (i.e. land, labour, capital). Finally x_{tm} 0 is a non-negativity constraint indicating that negative solutions are invalid for agricultural outputs. Note that the activity counter j represents all the columns for a given

year, whereas the counter m includes only activities that represent farm enterprises. By convention in the model design all the entries for activities that represent accounting columns ($n_m + 1$, n_t), followed by activities that represent farm enterprises (n_m).

The model is implemented as a set of matrices (see next section) and solved in Matlab. The model matrices are generated automatically based on the model templates explained previously.

Structure of the model

The matrices that represent the body of the optimisation model are illustrated graphically in Figure 12. Note the block-diagonal nature of the multi-period model, where each block of coefficients represents one year in the operation of the farm. These matrices (A and E) contain many zero entries, which allows them to be represented as sparse matrices in Matlab. This option has not been implemented in the current model but can be used in future to save memory with very large models.

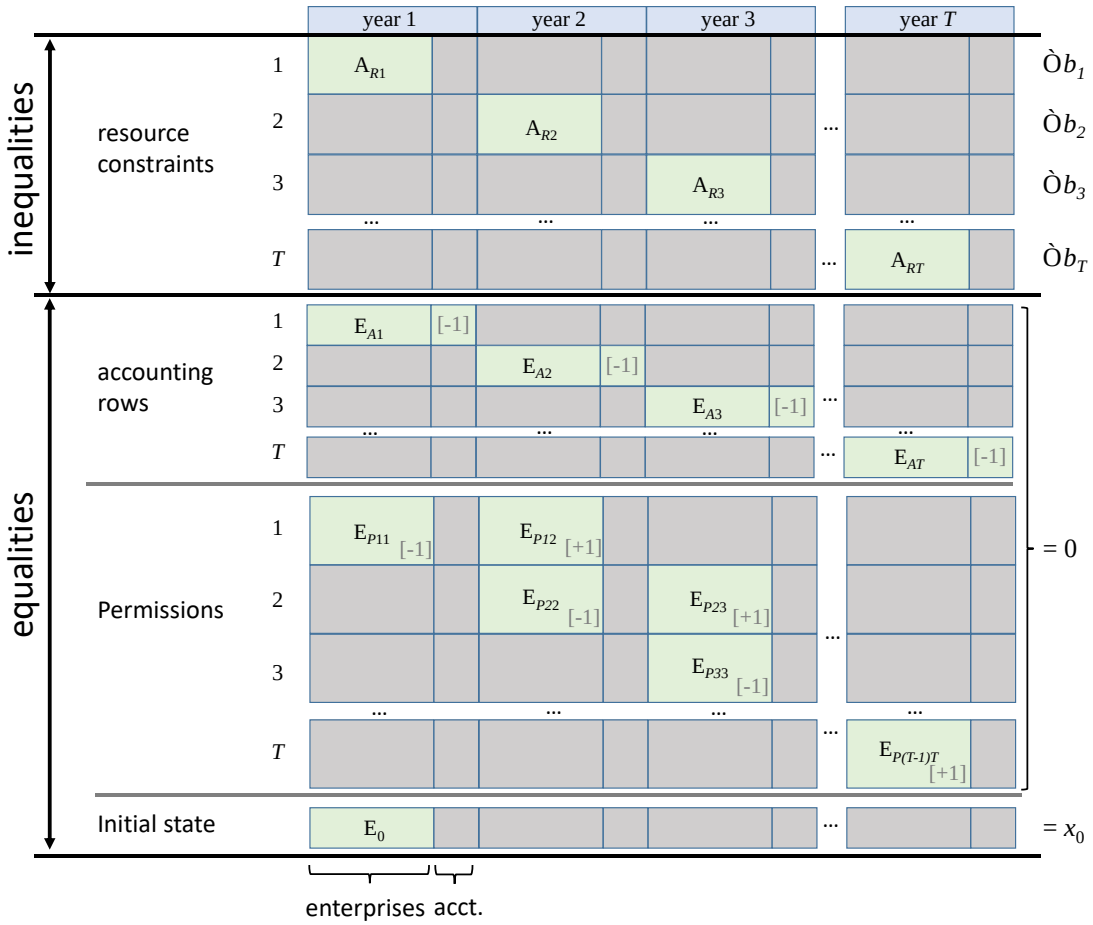


Figure 12. Structure of the matrices in the Carbon Farming Optimiser. Green cells contain models coefficients, grey cells contain zeros.

The matrix generator and the optimisation model have been written using the Matlab language. Scripts and functions are all available in the *Matlab_v2* shared folder. Within that folder, the file *READ_ME.docx* contains a description of the model components and their

roles. More detailed information for each model component is available in the header for the corresponding *.m file.

All Matlab files have the extension *.m, they are text files that can be opened with any text reader. However, in order to run the model it is necessary for the user to have access to the Matlab Base software and the Optimization Toolbox.The original input is read from Excel templates, but this process is relatively slow, so when a new Excel file is read it is then saved as a *.mat file. This is in the binary format used by Matlab and it cannot be opened by a text reader. In subsequent analyses the *.mat file is read before running the model, rather than the Excel file.

The matrix version of the MPLP model is:

Maximise:

$$Z = \mathbf{f}\,\mathbf{x}\,(1+r)^{-\mathbf{t}}$$

Subject to:

$$\mathbf{A}\,\mathbf{x} \leq \mathbf{b}$$

$$\mathbf{E}\,\mathbf{x} = [0]$$

$$\mathbf{x}[1, \mathbf{m}\,\cdot] = \mathbf{x}_0$$

$$\mathbf{x}[., \mathbf{m}] \geq 0$$

Where **A** and **E** are matrices containing elements A_{tij} and E_{tkj}, **b** is a column vector of resource constraints containing elements b_{ti}, **x** is a column vector containing elements x_{tj} and **t** is a vector of time periods and **m** is a vector containing indexes of **x** that correspond to farm enterprises.

Information on three Emissions Reduction Fund methods for integration into Carbon Farming Optimiser

The three ERF methods selected are Human-induced Regeneration; Environmental Plantings; and Soil Carbon. The requirements for information on each of these will be compiled in consultation with DPI staff. This section presents preliminary information and options for modelling each method.

Maps of areas that have potential for establishment of each method are being prepared by DPI. These maps are then to be used by the LUTO team to identify areas where the methods have economic potential. Results from their analyses will guide the selection of case study farms for this project.

Human-induced Regeneration (HIR)

Projects under this method capture carbon by changing land management practices to facilitate regeneration of a native forest. Landholders can assist regeneration through activities such as excluding livestock from the project area, managing the timing and extent of grazing, managing feral animals and non-native plants in the project area and stopping activities such as mechanical destruction of natural regrowth.

The method uses the Full Carbon Accounting Model (FullCAM) to work out the carbon captured by the project. Impacts of disturbances such as fires as also accounted for.

Projects are subject to permanence obligations; the project must be maintained for a nominated period of either 100 or 25 years.

https://www.environment.gov.au/climate-change/government/emissions-reduction-fund/methods/human-induced-regeneration-native-forest

Information requirements for modelling

Data for modelling this option will be obtained from FullCAM simulations for the region of interest and using the parameters for the specific farm when possible. Geoff Cockfield has data for semiarid areas, where all the uptake has happened. He shared all the information with Jeff Connor.

Environmental Plantings

Environmental plantings are covered under the *Reforestation by Environmental or Mallee Plantings ï FullCAM* methodology. Projects under this method capture carbon by permanently planting native trees or Mallee to establish forest cover.

The method uses the Full Carbon Accounting Model (FullCAM) to work out the carbon captured by the growing trees. Impacts of disturbances such as fires are also accounted for.

Projects are subject to permanence obligations: the project must be maintained for a nominated period of either 100 or 25 years.

Eligibility and restrictions

Projects need to meet the following requirements to be eligible under this method:

- The plantings must include tree or mallee species native to the local area.
- Mallee species must only be planted where annual rainfall is under 600 mm.

- The project area must have been clear of forest cover for at least five years prior to project commencement.
- The plantings must have the potential to reach forest cover (20 per cent crown cover consisting of trees that are at least two metres tall).
- The trees may not be harvested except for in very limited circumstances such as hazard reduction.
- Tree products such as firewood, fruits and nuts cannot be commercially harvested from the planting.
- If the land has been lawfully cleared in the past, it must have occurred more than seven years ago, or five years ago if the land was cleared by previous holders.
- Projects cannot be established on land that has been cleared unlawfully.

Monitoring, reporting and auditing:

Projects require regular reporting (at least once every five years) to demonstrate project requirements are being met, and to report on carbon abatement. Projects must also be audited by a registered national greenhouse and energy reporting (NGER) auditor.

https://www.environment.gov.au/climate-change/government/emissions-reduction-fund/methods/quantifiying-carbon-sequestration-permanent-native

Information requirements for modelling

Modelling environmental plantings will be confined to higher rainfall areas. Information required will be obtained from FullCAM modelling.

Soil Carbon

There are two soil carbon methodologies:

1. Estimating sequestration of carbon in soil using default values
2. Measurement of soil carbon sequestration in agricultural systems

The ERF rules for measured soil carbon projects allow a wide range of interventions in cropping and livestock operations to increase soil C. The actual C credits granted depend on the C accumulated in soils at a point in time, measured relative to the baseline when the project was registered. Before crediting, measured soil C is adjusted by subtracting any additional emissions arising from the intervention (i.e. N_2O from extra N fertiliser or additional livestock emissions from increased stocking rate). This C figure is then reduced by 25% or by 5% for permanence periods of 25 years and 100 years respectively. There are other discounts to measured C depending on the variability of the soil C estimates.

The crediting period is fixed at 25 years for soil carbon projects, but the activity must remain in place for the length of the permanence period. The frequency of payments is determined in the contract based on **project reporting periods** that can be between 1 and 5 years. Participants also need to provide **forward abatement estimates** (FAE) from their proposed management change. This is the ñbest estimateò of the number of carbon credits likely to be earned during the 25 year crediting period. So, although the real method relies on actual soil samples, we still need modelling to obtain FAE. There is no requirement to use FullCAM for this, and that is where APSIM and AusFarm come into the picture for us.

Among the eligible activities *altering stocking rate, duration or intensity of grazing*ò. This has been our focus for the measu

case study, but modelling these accounting rules introduce some complexities for the CFO for at least two reasons:

1. C sequestration in soils is treated differently from livestock C emissions. Extra livestock emissions relative to the baseline are penalised, but livestock emission reductions do not seem to be credited under this method.
2. C sampling and crediting does not occur annually as in our current case studies, but frequency of sampling and C sales becomes a decision variable which needs to be solved as part of the optimisation. The optimal solution will be affected by the cost of sampling and certifying soil C relative to C price.

To account for 1 we need to distinguish sequestration in soils from emissions in livestock. To account for 2 we need to introduce new variables to keep track of C-credit accumulation and payments according to the rules. Then we need to add extra columns and extra rows to the LP matrices.

New variables:

c_ls_t: CO_2 sequestration from livestock in year t.

c_land_t: CO_2 sequestration in land in year t.

c_credit_t: carbon credits accumulated in year t.

c_stock_t: carbon credits accumulated from last sale and up to year t

c_sell_t: carbon credits sold in year t.

c_sample_t: soil carbon sampling activity, required to enable c_sell_t

New equations:

$$c\_credit_t = \ _c\, c\_land_t + c\_ls_t - c\_ls_1 \quad (1)$$

$$c\_stock_t = c\_stock_{t-1} + c\_credit_t - c\_sell_t \quad (2)$$

$$c\_sell_t \leq c\_stock_{t-1} \quad (3)$$

for $t \in (2, \dots, T)$

Where $_c$ is the proportion of measured carbon that is eligible for credits. The default value is $_c = 0.75$. For $t = 1$ the equations become:

$$c\_credit_t = 0.75 c\_land_t \quad (1a)$$

$$c\_stock_t = c\_credit_t - c\_sell_t \quad (2a)$$

$$c\_sell_t \leq 0 \quad (3a)$$

The sampling cost requires integer constraints to be introduced. Sampling is a fixed cost, as the whole area under contract needs to be sampled. Sampling gives permission to sell:

$$c\_sell_t \leq \propto_{smp} c\_sample_t \quad (4)$$

$$c\_sample_t \in (0,1) \quad (5)$$

Where $_{smp}$ is a large number that gives permission to sell when c_sample_t = 1.

The gross margin rows are adjusted to account for C payments and sampling costs

$$gm_t = gm_t + pc_t\, c\_sell_t - cs_t\, c\_sample_t \qquad (6)$$

Where cs_t is the fixed cost of sampling and auditing the soil C estimates for the whole area under contract, and gm_t on the RHS denotes the existing coefficients in the original LP matrix.

To incorporate these new equations into the model first we need to move all the terms denoting decision variables to the left and then use the indexing system in the *ids* variable to insert coefficients in the appropriate elements of the LP matrices.

Additional code is needed to insert the extra columns and rows in the matrices before running the model.

In addition, incorporating *c_sample* as an integer variable will require a Mixed Integer LP algorithm which may not be as fast and easy to solve as straight LP, but it is easy to implement. The solution of the model will determine whether it is worth participating and, if so, the optimal frequency of sampling and sale of C credits.

These features have been implemented in the new version of the Matlab model, available in folder *Matlab_ERF* in the shared folder.

Introducing the ERF rule that sampling must occur every 3-5 years will require additional variables and constraints.

$$count\_smp_t = count\_smp_{t-1} + 1 - reset_t \qquad (7)$$

$$count\_smp_t \leq 5 \qquad (8)$$

$$count\_smp_t \leq 5\, c\_sample_t \qquad (9)$$

When t=1

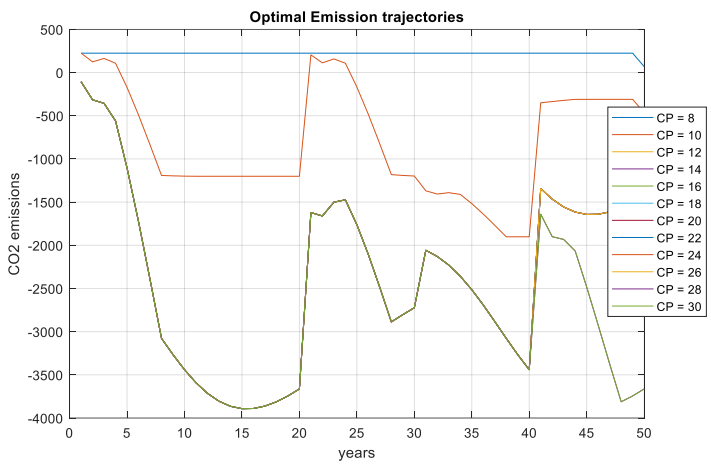
$$count\_smp_t = 1 \qquad (7a)$$

These last feature has not been added to the model yet. It will require similar steps to those detailed above to expand the matrices and insert the coefficients.

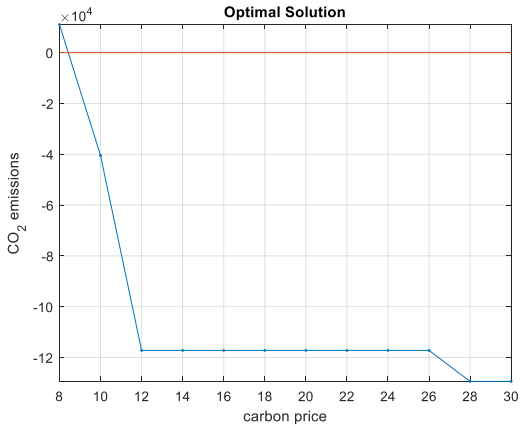
Example results

The representative farm template used in these simulations represents a fictitious farm, with technical coefficients that are plausible but not based on a real operation.

In this example carbon price (CP) ranges from \$8-\$30 per CO_2e. An optimisation is run for each CP, resulting in an optimal plan that maximises profits for the given assumptions. Each optimal plan is associated with a different trajectory of emissions through time. The emission trajectories shown below change because of decisions made regarding trees and pastures in order to maximise profit as CP changes.



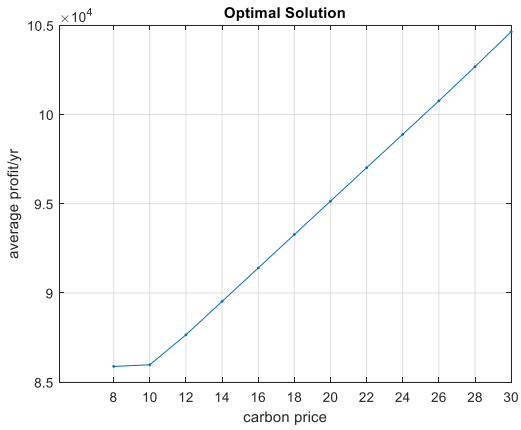
The sum of CO_2 emissions over the planning horizon shows the CP at which the optimal solution switches:



Total emissions (blue line) change from positive (cattle emissions) to negative (sequestration by trees and pasture) at $10/t CO2e, then decrease again at $12 and $26 as the optimal solution changes.

From the figure one can visualise how introducing other potential activities on the farm and aggregating results from other farms, we will be able to estimate the potential mitigation for a region under the assumption that farmers optimise profits given the best information available and no constraints on labour or capital availability to undertake the changes.

The optimal solution shows increasing profits as C price increases and farmers switch their practices. Average profit per year is calculated as NPV divided by the planning horizon of 50 years.



The stream of annual profits changes over time (not shown), sometimes reaching negative values when capital is required for an investment. This means that the optimal solution may change if we introduce constraints on capital available for the farmer to invest in planting trees.

Validation and adjustment of case studies

Creating experiments and processing data

The CFO model estimates optimal farm plans for long time horizons and considers options for carbon farming as well as allowing biodiversity and other environmental variables to be included. The model is written in the Matlab language, so access to this software and some familiarity with its environment is required to run CFO.

The CFO model requires a farm template as its only input. The template consists of a set of Excel worksheets describing the farm and the enterprises (activities) available for inclusion into the optimal plan. Details must be included for each activity regarding labour and capital requirements, feed required or feed produced, annual carbon emissions or sequestration, yields and prices of outputs produced. Other variables relevant to the particular problem can also be included in the farm template. The design and application of the template is explained in Project Brief #2 and applied to two case studies in Project Brief #3. The final version of the Excel template is explained in detail in Section 4.2 of the CFO User Guide (*CFO_guide.docx*).

The model output is in the form of a data structure within the Matlab environment that contains all the details of the optimal solution:

- The net present value (NPV) of the stream of annual profits over the planning horizon
- The optimal level of each activity (hectares of crops and trees or head of livestock) for each year in the planning horizon.
- The level of resources (land, labour, capital, feed etc.) used under the optimal solution for each year of the planning horizon.
- The shadow prices of the resources on the farm for each year of the planning horizon.

Shadow prices are explained in detail in the user guide, but briefly, they are the marginal value of the resource within the production system. They can be compared to market prices to

decide whether it is worth hiring an additional unit of the resource (i.e. borrow money or hire labour). They can be very useful when interpreting optimisation results and planning further experiments to run with the model.

The interpretation of CFO results is explained in the CFO_guide (Section 4.4). Results can be saved to Matlab (*.mat) files and accessed for future analysis. Matlab code is available to plot and analyse results (see the Technical Appendix in the User Guide).

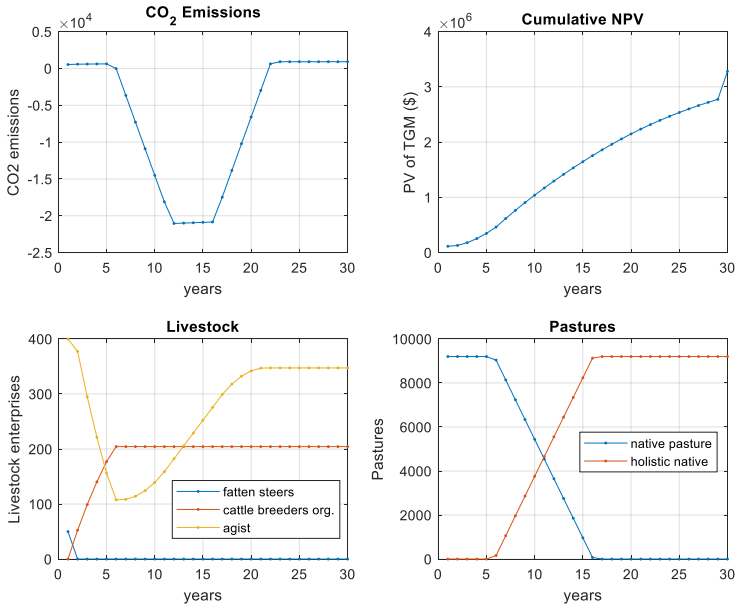
Results of a single model run

The CFO model consists of a number of functions and scripts written in the Matlab language. The model is run through the main script. *CFO_mixed_farm.m* is the name of the main script in the User Guide. The user would take this script and modify it to represent a different farm or different assumptions. The new script would then be saved under a different name before running, so different versions of the model can be created to represent different farms (this was done in the case studies in Brief #3).

As mentioned in the previous section, CFO creates a data structure containing the optimal solution. The variables contained in this data structure, named ***OS***, are used to plot results such as those illustrated in Figures 2 and 3. These results are for Bokhara Plains, one of the case study farms reported in Brief #3. To create these figures CFO was run twice, first with a carbon price of zero and then with a carbon price of $15 per tonne of CO_2.

Experiments in CFO, as illustrated above, where one or more model parameters are modified, the model is run, and the results are saved. Using this strategy we will have one results file (with extension .mat) for each run of the model. Running multiple experiments to represent different scenarios can be time consuming and result in a proliferation of output files. To avoid this, we can run experiments within loops, where each loop represents a different treatment. The results from all treatments can then be saved in a single file, as explained in the following section.

(A) *K* and *L* constrained – no C market



Assumptions:

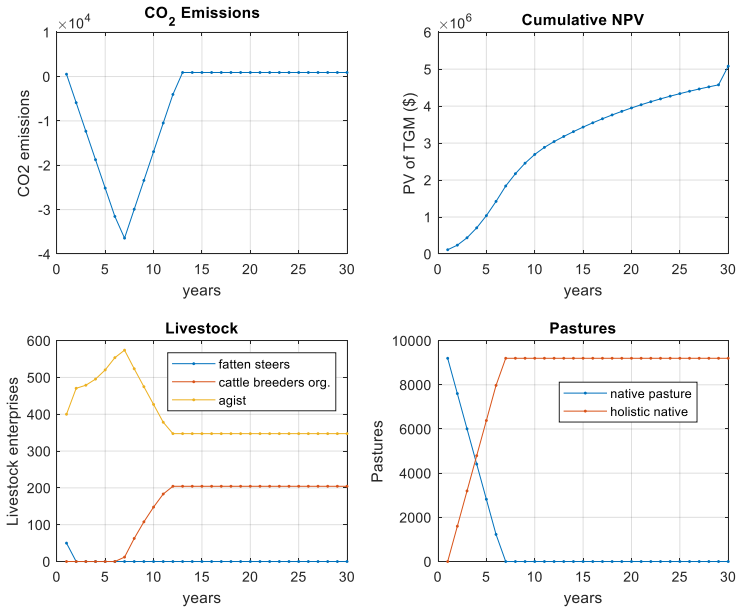
K = \$80,000
L = 1,500 hr
P_C = \$0/CO_2e

Results:

emissions = -199,492 t CO_2
NPV = \$3.28 million

Figure 2. Optimisation results for Bokhara Plains when both capital (K) and labour (L) are constrained and carbon price (P_C) is zero.
For details see ACM_Brief_3_progress_report.docx

(B) *K* and *L* constrained – C price=\$15



Assumptions:

K = \$80,000
L = 1,500 hr
P_C = \$15/CO_2e

Results:

emissions = -197,932 t/CO_2
NPV = \$5.08 million

Figure 3. Optimisation results for Bokhara Plains when both capital (K) and labour (L) are constrained and carbon price (P_C) is \$15 per tonne of CO_2.
For details see ACM_Brief_3_progress_report.docx

Experiments with multiple treatments

An experiment consists of treatments that represent different sets of inputs into the model. The experimental variables that change between treatments can be model parameters, constraints, prices of carbon, costs of inputs, yields and prices of outputs, etc. So the concept of an experiment in this context is very flexible. In addition, the treatments may follow a particular experimental design, where different combinations of parameter values are selected to extract useful information. The user guide presents an example of a full factorial design. Some of that material is replicated here to gain an overview of the process, but see Section 5.3 of the CFO User Guide for detailed instructions.

A simple factorial experiment to explore the effects of two variables: carbon price and initial stocking rate of the farm is illustrated as a two-way table:

	Initial DSE	
C Price	0	4,320
0	Case 1	Case 3
20	Case 2	Case 4

where Initial DSE is a measure of the stocking rate of the farm (dry sheep equivalents) at the start of the planning horizon and C Price is carbon price in $ per tonne of CO_2 equivalents. If we run this experiment ***OS*** will now have four elements instead of a single one. So we can refer to *OS*(1) to access the optimal solution for Case 1, *OS*(2) for Case 2 and so on.

The *CFO_experiment.m* script illustrates how to run a more complex experiment, where C prices were changed at intervals of $2 per tonne of CO_2e, and two different levels of capital (K) and labour (L) were considered, resulting in 44 different treatments representing the different levels of the three factors tested (11×2×2). In this case the *OS* data structure contains 44 elements. This an efficient way of running multiple experiments and saving the results in a single file. The Matlab script also plots summary results for the experiment (Figure 4).

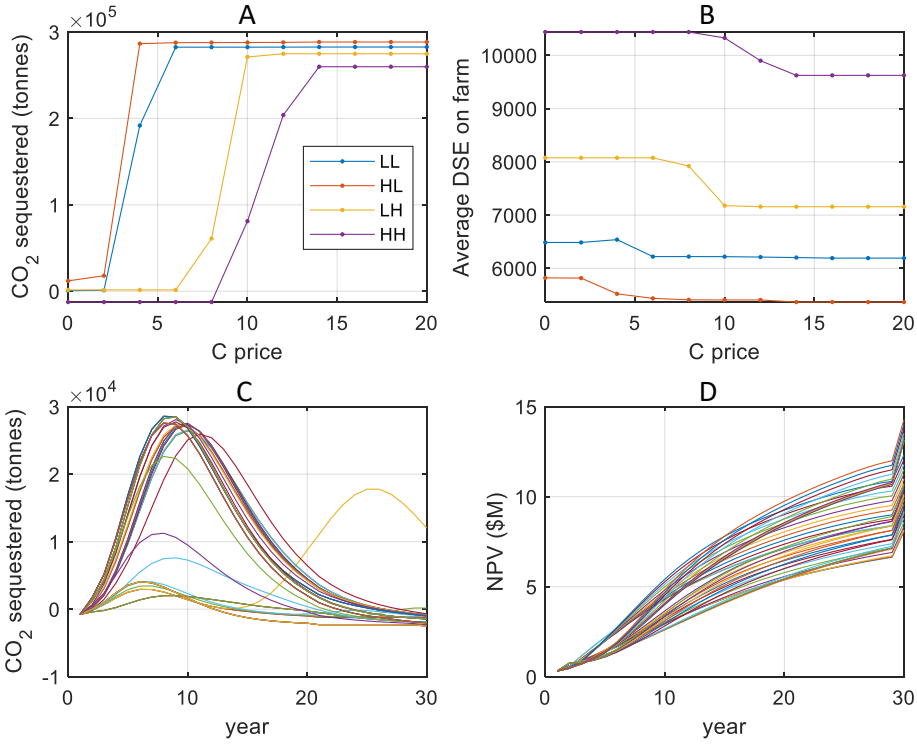


Figure 4.Summary results from running the *CFO_experiment* script. The legend applies to panels A and B, it represents high (H) and low (L) levels of capital and labour repectively.

Briefly, these results show how carbon prices affect the optimal level of CO_2 sequestered (Figure 4A), which occurs as increasing areas of perennial pastures and trees are established to capture carbon in biomass and soil and obtain C credits that can be sold. The optimal number of livestock on the farm (Figure 4B) is determined based on pasture availability and the C price. The amounts of capital and labour available to the farm influence the optimal solution, as these resources determine the type and level of activities that can be carried out in the process of maximising profit. The time trajectories of CO_2 sequestered and NPV (Figures 4C and 4D) are presented to show the variation in the variables that are packaged within the *OS* structure.

Deriving Marginal Abatement Cost (MAC) curves

A marginal abatement cost curve shows the potential reduction in CO_2 emissions depending on the Carbon price. Conceptually, the MAC represents the supply curve for C abatement for a particular region or industry. In our case, we can derive a MAC for the example farm using the data plotted in Figure 4. To expand this estimate to a whole region we need information on the number and types of farms in the region. Section 5.4 of the CFO User Guide explains this.

A MAC for a particular region can be generated by following these steps:

1. Identify the main farm types in the region and create a representative farm template for each of them.

2. Supplement the templates in 1 with data on the distribution of resource endowments (land, labour and capital) in the farm population, perhaps from ABARES data.
3. Combine both data sources to generate a distribution of representative farms of each type, which differ in terms of resources they have available. This can be refined if additional information on debt levels is available. Essentially, we are creating an artificial population of farm templates that reflects the distribution of farms in the region.
4. Run CFO for alternative C prices (as above) for each farm in the representative farm population, then aggregate results and derive the MAC for the region.

Essentially this process consists of running a complex experiment where treatments involve different farm types and variation in resources available.

A simplified illustration using this process with the example mixed farm is presented in Section 5.4 of the CFO User Guide. In this case we have a single farm type, but a full distribution of resource endowments in terms of land, labour and capital available.

The ***CFO_MAC.m***. script is similar to *CFO_Experiment*, but instead of running a factorial experiment it samples from normal distributions to generate the experimental variables. We use land, labour and capital as the limited resources that vary across the population of farms. In this arbitrary example we used a population of 50 farms to illustrate how to derive the MAC (Figure 5).

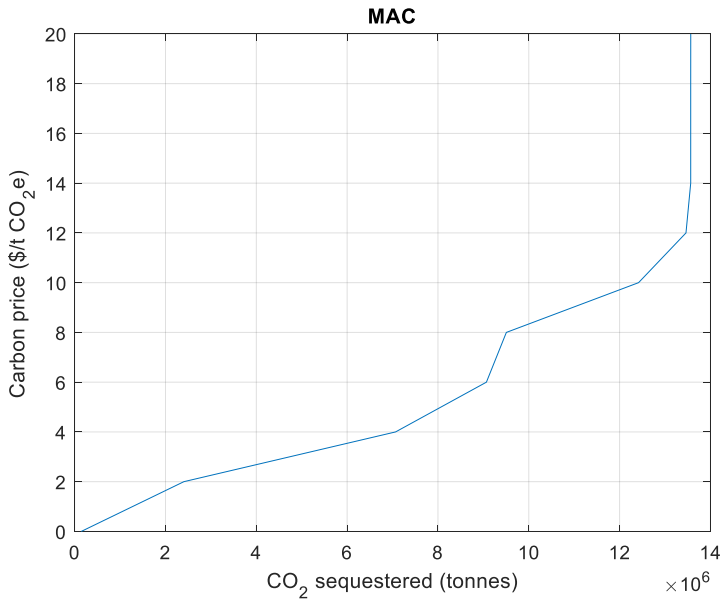


Figure 5. Marginal abatement cost curve derived from a hypothetical distribution of mixed farms.

MAC curves can be useful for policy analysis to indicate the potential carbon abatement that could be achieved at any given carbon price and using different technologies. Introduction of new technologies or policies to encourage adoption would result in shifts of the original MAC curve.

Getting the most out of the Carbon Farming Optimiser

CFO is very flexible in terms to the variety of farm systems and enterprises that can be modelled. It can be used for analysis of individual farms, to understand how carbon farming may fit within the operation while maximising profit over any given planning horizon. It can also be used to evaluate different technologies or policies for a given sector or region. The key is to have a database of representative farming systems for the region, so that the aggregate effects of individual farmer decisions can be simulated.

It is relatively straightforward to update and modify model parameters (although improvements can always be made). For example, a new technology that reduces livestock emissions but requires an initial investment can be represented by changing coefficients for costs and CO_2 emissions within the farm template for the new activity. Policies that affect prices, costs or access to resources can also be accommodated by changing the appropriate coefficients.

The challenge in applying the model at a large scale is in obtaining the data to represent the main farming systems. For each activity on the farm we require a gross margin budget supplemented with additional information on carbon emissions, labour requirements, feed requirements or other variables (eg. rotations), depending on the type of enterprise. For long-lived activities, such as tree planting, annual coefficients are required for a whole productive cycle, so that carbon sequestration, costs and profits can be estimated for any farm plan. This means we need to obtain carbon growth curves (Figure 6A) for each perennial activity.

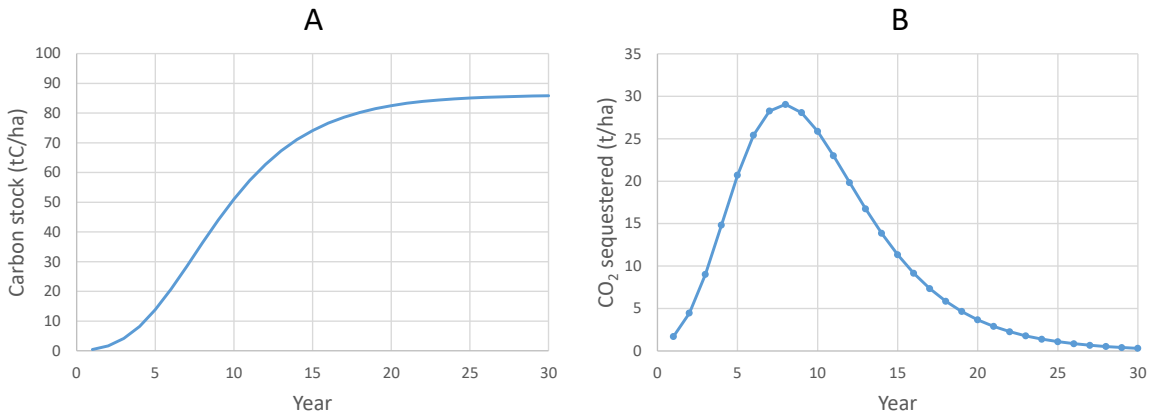


Figure 6. Biomass carbon growth curve (A) and CO_2 sequestration rate (B) for hypothetical forest restoration under the Human Induced Regeneration method of the Emissions Reduction Fund of the Australian Government.

The carbon growth curve can be generated using FullCAM[1], the software required under the ERF to estimate carbon sequestration. The results from FullCAM runs can then be used to generate CO_2 sequestration curves (Figure 6B) for the given activity. The points on this curve indicate the values that need to be inserted into the appropriate row in the farm template for this activity.

[1] https://www.industry.gov.au/data-and-publications/full-carbon-accounting-model-fullcam

In addition to carbon curves, we also need information on the operating capital and labour requirements of the activity for each year. In the case of human induced regeneration (HIR), these variables could be estimated by calculating the costs of fencing in year 1 and the costs of fence maintenance plus any other annual costs (i.e. weed and pest control) for the remaining years. If any family labour is required for these activities (as opposed to hired labour), these requirements also need to be included in the template.

To use CFO effectively you will need:

- Realistic information about the representative farms and their enterprises.
- A strategy to design, store and access multiple files representing different experiments and/or farming systems
- Enough familiarity with Matlab to be able to modify code within scripts to create and modify experiments. There is no need to be familiar with the functions contained in CFO, but for those interested the CFO User Guide contains a Technical Appendix.
- A strategy to access multiple files containing optimisation results and process them as required for the particular analysis. The initial access needs to be done within the Matlab environment to read the *.mat binary files, but the relevant data can be extracted for further analysis with other software such as R for statistical analysis.

There is the potential for the user to become overwhelmed with results once a few experiments are run and multiple scenarios are introduced. This means it is important to have a sound strategy for naming, storing and documenting input and output files.

Recommendations for refinement of the Carbon Farming Optimiser

Although CFO is quite flexible and relatively easy to use, as shown above and in the user guide, there is some room for improvement in several aspects of the model.

1. Excel farm templates are used to simplify the process of obtaining information from farmers and exchanging files between members of the team. However, once the Excel template is read into the CFO model, we recommend that the farm file be saved as a *.mat file for quick future access. To apply the model at large scale it will be necessary to have a large number of farm files and at that stage the use of Excel to manage farm templates is not recommended, as this will hamper automation options.
2. The prices of outputs and costs of inputs for each enterprise are contained within activity worksheets. This means that changing assumptions on prices and costs requires relevant model coefficients to be changed for each activity within the code. There is scope for creating a standardised list of inputs and outputs, so that prices can be modified in one place. This will require modifications to the code of CFO so that activities can cross-reference the specific products (i.e breed of cattle or variety of cereal) to obtain prices. This may be a worthwhile change to simplify the use and maintenance of the model in the future.
3. Obtaining model coefficients for a new farm system or activity is not trivial, a gross margin budget plus other variables are required as explained above. Some coefficients can be obtained through modelling, using models such as FullCAM, APSIM and AUSFarm. The outputs from those models still need to be manually transformed and inserted into the farm template. This can be cumbersome, especially when many scenarios are modelled. Developing an automated system to read results from those models and inserting the new model coefficients in the appropriate sections of the template will enable large scale application of the model while making efficient use of analysts time.

NSW GOVERNMENT