ACTA UNIVERSITATIS AGRICULTURAE ET SILVICULTURAE MENDELIANAE BRUNENSIS

Volume LIX 18 Number 2, 2011

THE DYNAMIC PROGRAMMING APPROACH TO LONG TERM PRODUCTION PLANNING IN AGRICULTURE

J. Janová

Received: December 17, 2010

Abstract

JANOVÁ, J.: *The dynamic programming approach to long term production planning in agriculture.* Acta univ. agric. et silvic. Mendel. Brun., 2011, LIX, No. 2, pp. 129–136

The production planning in agriculture is one of the most important decision problems of the farmer. Although some decision support tools based mainly on linear programming and addressed to agriculture authorities were presented, their direct application by a farmer is not possible. This is mainly due to the local character of the models developed for particular agricultural conditions and also due to the complexness of underlying mathematical programming models.
This paper aims to develop dynamic programming model for the long run crop plan optimization covering the typical conditions of Czech farms, which could serve as a platform for further enlargements and changes according to needs and conditions of particular farm. The dynamic programming algorithm is developed in detail for model case of four areas to be planted by four crops each year. The possibility of covering different constraints by generating the state space is discussed, and the generating procedure for crop rotation rules is shown. The goal function reflects the farmers objective of profit maximization and it is defined with respect to harvests' randomness. The case study is solved for the data from South Moravian agriculture cooperative and the optimal solution is presented and discussed.

production planning, MATLAB algorithm, crop rotation, dynamic programming

For the past several decades the production planning problems in agriculture has been extensively studied from the mathematical point of view. Aiming to develop the credible decision support tools for agricultural authorities, various models and mathematical approaches were employed. Beginning with linear programming in 50's, the models subsequently followed the development of mathematical programming techniques to the current dynamic optimization applications which reflects the dynamic structure of planning problems in agriculture. Recently the dynamic programming and optimal control techniques were applied in several studies concerning agroecosystem modelling (see Seppelt, 1999, Seppelt *et al.*, 2002, Chikumbo *et al.*, 2003, Bond and Farzin, 2007, Parsons *et al.*, 2009). Considering different assumptions and simplifications reflecting the unique local conditions, these contributions deal with modelling of the agroecosystem concerning the biochemical and economical aspects of agriculture processes. Any of the model mentioned intends to serve as a decision support for upper level agriculture authorities. Apart from their local applicability due to the specific features of local agriculture systems, the models mentioned are also very complex from the mathematical point of view and its solutions are uneasy to be obtained, i.e. the models are hardly directly applicable by the decision maker. Moreover, the dynamic programming models are limited by the curse of dimensionality which matters in such a complex systems. On the other hand the dynamic programming offers straightforward user-friendly solution procedure for the medium scale problems- such as long-run crop plan optimization on a farm under agricultural constraints typical for Czech Republic.

In such a case, the simple dynamic programming algorithm can be directly applied by the user and can also serve as a platform for further enlargements stemming from the particular additional constraints and conditions.

The problem is to decide what is to be planted on particular pieces of arable land next period. The farmer's goal is to gain a considerable profit from the yield under the requirements of sustainable farming. Apart from many simple constraints these requirements include also crop succession rules which are uneasy to be covered by mathematical program. Recently, there were studies on solving the problem of crop rotation by linear programming: for the representation of crop succession restrictions via linear programming see (Klein Haneveld, Stegeman, 2004), in (Castelazzi, 2008) the software tool was developed which can be used to create scenarios of cropping systems and land use but it does not answer the question how large areas should be cropped by particular crop-plants. Since the crop rotation issue is of dynamic structure applying the methods of dynamic optimization is appropriate (for the particular case of dynamic programming approach to corn-soybean farm in Illinois see Duffy and Taylor, 1992).

This paper aims to develop unique dynamic programming algorithm for the purpose of long-run crop plan optimization under crop succession rules, which can be used for further enlargements and applications for various conditions and needs of particular farmers. The algorithm is applied in the particular case of South Moravian agriculture cooperative.

In the following section the problem of the farmer's decision making and its model case is described. Then, the method of dynamic programming is briefly summarized and the particular algorithm used is developed in detail, applied on particular case of a farm, the results are discussed in separate section and the conclusions with respect to further development and application of the model are given.

METHODS AND RESOURCES

The representative farm

The problem of crop plan optimization will be solved for the particular case of South Moravian agriculture co-operative (for details about the farm see Janová and Ambrožová, 2009). It is only for part of the arable land at farm that the historical crop pattern is known in detail. Therefore, for the purpose of developing the dynamic programming model only this part of 123 ha of arable land will be considered. The area is naturally split into many small fields; each is cropped as a whole by one crop each year. For simplicity we will consider only four larger areas (A, B, C, D), which cluster the fields into homogenous parts (with respect to crop planted, yields, costs, etc.). For the acreage and initial sow plan see Tab. I.

I: *Description of the field areas*

Area	A	B	C	D
Acreage [ha]	23.9	32.6	26.6	39.4
Initial sow plan	Corn silage	Oilseed rape	Corn silage	Spring barley

For the model case we will assume four crops can be planted each year. These crops represent the typical crops planted at the major part of the arable land at farm (for the particular crops and notatios see Tab. II).

II: *Notation of the crops*

Crop	Winter wheat	Spring barley	Corn silage	Oilseed rape
Notation	1	2	3	4

The goal of the farmer is to gain a considerable profit when the farm is running in the sustainable manner. In the conditions of Czech Republic the main agricultural restrictions to be covered are the crop rotation rules. These will be hold by specific definition of the state space in dynamic program. We will consider one year succession requirements as described in Tab. III. These crop rotation rules represent the basic principles of crop succession in the conditions of Czech Republic. Of course many other, more detailed rules could be considered (such as the multiple-year succession requirements, e.g. for oilseed rape the five year period should be hold between re-sowing the plant on the same piece of land). But, as we will discuss in the *Results* section, these requirements could be involved in addition into the presented functioning dynamic model developed.

III: *Crop rotation rules*

Crop	1	2	3	4
Feasible preceding crops	3, 4	1, 3	1, 2, 3, 4	1, 2

The real data sets from the South Moravian farm were used to obtain the desired parameters of the model. As we shall see in the next paragraph, there is a need for the information about the yields, prices and costs for the plants. The historical yields from 1997 to 2008 can be seen in Tab. IV, the prices and cost per ton were set on values of 2008 and throughout the calculations they were assumed to be constant parameters (see Tab. V, for details see Janová and Ambrožová, 2009).

Dynamic programming

Dynamic programming is a mathematical optimization technique for formulating and numerically solving multi-stage decision problems (Bellman, 1957). In dynamic programming the whole time pe-

IV: *History of yields (t/ha)*

crop	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008
winter wheat	5.41	6.25	5.76	5.70	5.85	5.69	4.76	6.48	6.74	4.54	6.28	7.66
spring barley	4.80	4.28	4.89	3.80	4.31	4.50	4.93	5.39	4.61	3.51	3.28	4.56
corn silage	27.84	34.36	32.35	38.22	26.96	36.50	17.00	23.79	24.95	25.31	29.03	36.34
oilseed rape	2.67	3.14	3.05	4.12	3.30	2.60	2.05	3.91	3.29	3.27	2.73	3.29

V: *Crop plants – Production costs and selling prices*

i	Crop	Costs n_i (Kč/t)	Price p_i (Kč/t)
1	winter wheat	3 500	3 900
2	spring barley	4 000	5 300
3	corn silage	550	600
4	oilseed rape	8 000	8 800

riod over which the decisions are made –the planning horizon – is divided into discrete and finite number of time periods in which the decisions are made. Each period n, $n = 1, 2, \dots N+1$ defines one stage of the sequential decision process and is associated with the state space S_n formed by the set of states s_n, which represent all the feasible states of the system in the period n. The feasibility of states is given by the constraints which must hold throughout the whole planning horizon (for more details about the dynamic programming and its applications see Denardo, 2003). In the crop planning optimization problem we have the following constraints:

- the crop rotation rules (see Tab. III),
- all four areas must be planted each year,
- each of the areas is cropped by one plant as a whole.

The state of the system in our problem can be described by crop plan applied in particular stage (year). For example the initial state is

$$s_1 = (3\ 4\ 3\ 2),$$

which reflects the situation when at the beginning of the planning horizon there is planted corn silage on area A and C, oilseed rape on area B and spring barley on D. The feasible states in the following stage can be generated according to crop rotations rules (see Tab. III) and possibly other constraints, e.g. there are 16 possible crop plans in the second stage. The first task in dynamic programming is therefore to establish the state space S_n for each stage n. In our problem this was made in the first part of the dynamic programming algorithm developed in MATLAB (see appendix). The state space S_n was composed of all states which were feasible for at least one state in stage $n-1$.

The decision maker will determine the state s_{n+1} by invoking management decision $d_n(s_n)$ from finite discrete set $D(s_n)$ of decisions associated with s_n. In our problem the decisions are restricted by the crop rotation rules, hence, given the current state, the decision maker is allowed to plan for the next stage only feasible crop plans from the standpoint of crop rotation. Selecting decision d_n for state s_n earns reward $r(s_n, d_n)$ and cause the transition to the state $s_{n+1} = t(s_n, d_n)$. This reward corresponds to the objective function of the optimization problem. The objective of our problem is to maximize the profit from yields in the long run perspective, which set the reward to:

$$\rho = \sum_{i=1}^{4}(p_i - c_i) \times q_i \times x_i(d_n), \qquad (1)$$

where p_i and c_i is price per ton and costs per ton of crop plant i, respectively, q_i is the yield in tons per hectare of the crop plant i and x_i is the area in hectares where the crop plant i is cropped (the argument d_n refers to the particular decision, i.e. the areas x_i are determined by the decision d_n about the next crop plan). In the reward function (1), one must take into account, that the yields q_i are random variables. Hence, we must face the stochastic nature of the processes involved in the decision making. We will adopt the approach suggested by Freund for the linear programming optimization of a sowing plan for a next period. Denoting the profit per 1 ha of area planted by crop i by

$$z_1 = (p_i - c_i)q_i \qquad (2)$$

we can consider z_i to be a random variable. Following (Freund, 1956), we assume $zi \sim n(\gamma_i, \sigma_i^2)$. Then the reward function (1) with random parameters can be replaced by the Markowitz function

$$r(d_n) = \frac{a}{2}\ x\,\Sigma\,x^T - \gamma x^T, \qquad (3)$$

where Σ denotes the covariance matrix of the random vector (z_1, z_2, z_3, z_4), $x = (x_1(d_n), x_2(d_n), x_3(d_n), x_4(d_n))$ and a is the risk aversion coefficient. The maximization of profit reward (1) can be now replaced by minimization of the deterministic function (3) which represents the difference between the variability and the mean value of the total profit for a crop plan.

Having defined the reward function, the rewards can be counted for each decision in each stage. Note, that in our particular case, the reward depends only on the decision done (i.e. on the crop plan chosen for the next stage, see (3)). Hence, having chosen one particular feasible crop plan for the next stage, the reward from this decision will be the same whatever the current crop plan is.

Let the initial crop plan in the stage 1 be given by the state s_1 and the aim is to plan the crop plans for the next N periods. We do not determine any goal state in stage N+1. Such a formulation represents an initial value problem which can be solved by recursive fixing (see, Denardo, 2003) derived for each state in S_{N+1}. The goal is to identify the sequence of decisions (d_1*, d_2*, d_3*, d_4*) with highest total reward. The recursive relationship describing the system and determining optimal sequence of decisions (d_1*, d_2*, d_3*, d_4*) is then

$$f(s_n) = \begin{cases} 0 & for\, n = N+1 \\ \max\{r(d_n) + f[t(s_n, d_n)]\} & for\, n < N+1 \end{cases},$$

where is the maximum total reward obtainable at stages *n* through *N+1* if state is occupied.

MATLAB algorithm

The algorithm described qualitatively in the preceding paragraph was developed for the particular case of a South Moravian farm in MATLAB (see Appendix 1). Since there is a specific structure of the optimization problem it is possible to assign the reward to each possible state in our problem (for four crops rules and four areas to be planted, there is $4^4 = 256$ possible states of the system) no matter which state was occupied in the preceding stage (varibles *state_space* and *reward_space* contains the possible states and rewards respectively). The rewards were computed using additional function considering the farm data (see Appendix 2). Finally, to each possible state the feasible successive states were assigned (*feasible_next_states* array).

The information about the recursive fixing procedure is contained in the *DP* array. For each stage *t* the (256 x 256 + 2) matrix *DP* {*t*} contains the information about states in the stage (in the first column) and the best successive state in the next stage. In the last column there is a value of the total reward obtained when the optimal path leads through states given at the row (0 means not occupied, 1 means that the state is occupied), see the Fig I.

$$DP\{t\} = \left(\begin{array}{ccccccc} & \multicolumn{5}{c}{\overbrace{\quad best\ \ further\ \ state \quad}} & \\ 1 & 0 & 1 & 0 & 0 & \cdots & 0 & reward \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & \cdots & 0 & reward \\ 1 & & & \vdots & & & & \vdots \\ \vdots & & & \vdots & & & & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \end{array} \right) \Bigg\} current \quad feasible \quad states$$

1: *Example of the matrix DP{t}*

RESULTS AND DISCUSSION

The optimization problem was solved for 10–years planning period in the particular case of the South Moravian farm. The risk aversion coefficient of the farmer was set to a = 1.10^{-6}, which is the value used in (Freund, 1956) and confirmed as realistic for the particular farm in (Janová, 2010). The results of the program is the sequence of crop plans as follows:

3	4	3	2
2	2	2	3
1	3	3	2
4	2	2	3
2	3	1	2
1	2	4	3
4	3	2	2
2	2	1	3
3	3	4	2
2	2	2	1.

The first line represents the initial crop plan (corn silage, oilseed rape, corn silage, spring barley) the second line the optimal crop plan for next year etc.

As we have mentioned above the presented model is a small scale example of the crop plan optimization. The algorithm can be applied directly for different planning horizon and the similar technique could be used for more homogeneous areas and/ or more crop plants involved. For the realistic problem of whole farm planning the knowledge of expected profit for the planning horizon could be of use. Since the goal function used in the dynamic program has no practical meaning due to the stochasticity of initial goal function, the expected profit should be evaluated using the mean values of profits per hectare. Other possibility is to run Monte Carlo simulations of harvests and evaluate the profit performance of optimal solution from the long run perspective. Easily the profit in the initial goal function could be replaced by its present value which would be transferred also into the reward function.

The dynamic programming algorithm presented is appropriate also for more complex optimization of the sowing plan concerning more constraints. The matter of constraints is covered by properly generated state space in each stage of the dynamic program. Hence, more constraints involved more complicated definition of state space and of feasible following states for each state. The dynamic programming algorithm as itself would work in current form up to probably more complex structure of array *DP*. There is a possibility to enlarge the current dynamic programming model by covering additional constraints on

- crop rotation rules considering the longer succession requirements periods than one year,
- limitations on total areas cropped by particular crops,
- limitations on total costs, etc.

In further possible enlargement of the model the randomness of the prices could be considered, since actually the prices are unknown in the moment of planning the crop planting. The Markovian property of these prices mentioned in (Duffy and Taylor, 1993) enables to solve the general stochastic dynamic programming problem via Markov chains. This would change the structure of dynamic programming algorithm, hence for the presented model, the possibility of long run econometric estimation or expert estimation of prices would be more appropriate.

SUMMARY

The aim of the paper was to present the simple dynamic programming algorithm for the problem of long–run crop plan optimization at Czech farm. The initial performance criterion was the profit from yields and the long run sustainability of farming was ensured by considering the crop rotation rules. Since the profits from yields are random variables due to the randomness of harvests (the selling prices of crop plants are considered to be known constants), the performance criterion was changed and Markowitz model was applied. Hence, the reward for the decision about particular crop plans were calculated as the difference between the term representing the variance of the total profit and the mean profit.

The algorithm for particular case of four field areas (each cropped by one crop as a whole) and four typical crop plants were developed in MATLAB. The algorithm was applied for particular case of South Moravian farm for ten years planning horizon, the optimal solution was found and discussed.

The possibilities of further enlargement and applications of the algorithm were listed: there is a possibility to directly apply the algorithm for various planning horizons, the presented algorithm can be simply rebuilt to cover the optimization problem for more areas and/or more crop plants involved and after re–defining the state space generating process, it is possible for more complex restrictions on long-run crop succession requirements to be covered by the dynamic programming algorithm presented.

The MATLAB code is presented, which may serve as a base for individual changes and enlargement with respect to particular needs and conditions of the farm.

Acknowledgements

This work was supported by the grant MSM6215648904 of Czech Ministry of Education, Youth and Sport of Czech Republic.

Appendix 1

```
state_space = npermutek([1 2 3 4],4); %generating the whole set of possible states, function npermutec can
be downloaded at
http://www.mathworks.com/matlabcentral/fileexchange/11462-npermutek
for j=1:256
   v=state_space(j,:);
   reward_space(j)=reward(v);
%function ohodnoceni evaluates each state according to the reward function (3) for the particular case of
agricultural cooperative
end
%initial crop plan
osev0=[3 4 3 2];
%planning horizont in years
years=10;


%generating the feasible states

            for j=1:256
                 pocstavnext(j)=1;
                 for n=1:4
                      if state_space(j,n)==1
                          possibilities(n)= 2;
                          admissible{n}=[3,4];
                      end
                      if state_space(j,n)==2
                          possibilities(n)= 2;
                            admissible{n}=[1,3];
                      end
                      if state_space(j,n)==3
                          possibilities(n)= 4;
                            admissible{n}=[1,2,3,4];
                      end
                      if state_space(j,n)==4
                          possibilities(n)= 2;
```

```
                    admissible{n}=[1,2];
                end

        pocstavnext(j)=pocstavnext(j)*possibilities(n);

        end


        i=1;
         for hon1=1:possibilities(1)
         states{j}(i,1)= admissible{1}(hon1);
            for hon2=1:possibilities(2)
               states{j}(i,2)= admissible{2}(hon2);
                for hon3=1:possibilities(3)
               states{j}(i,3)= admissible{3}(hon3);
                for hon4=1:possibilities(4)
               states{j}(i,4)= admissible{4}(hon4);
                  if i<pocstavnext(j)
                   states{j}(i+1,:)=states{j}(i,:);
                   i=i+1;
                  end
            end
            end
        end
end


[c, ia, ib] = intersect(states{j},state_space,'rows');
feassible_next_states{j}=ib;

end

[c, ia, ib] = intersect(osev0,state_space,'rows');
N=256;
tree{1}=zeros(N,N+1);
tree{1}(ib,1)=1;

%describing the decision tree

for t=2: years
        tree{t}=zeros(N,N+1);

        for j=1:N
           if tree{t-1}(j,1)==1
                for p=1:size(feassible_next_states{j})
                   pom=feassible_next_states{j}(p);
                   tree{t}(pom,1)=1
                   tree{t}(pom,j+1)=1;
                end
           end
        end
end

%dynamic programinc algorithm
DP{years}=zeros(N,N+2);
DP{years}(:,1)=tree{years}(:,1);
DP{years}(:,N+2)=reward_space';

 for a=1:(years-1)
        t=years-a
        DP{t}=zeros(N,N+2);
```

```
        DP{t}(:,1)=tree{t}(:,1);
        for j=1:N
           if DP{t}(j,1)==1
              for p=1:size(feassible_next_states{j})
                 pom=feassible_next_states{j}(p);
              UF=DP{t+1}(pom,N+2)+reward_space(j);
                 if    UF<DP{t}(j,N+2)
                   DP{t}(j,2:end)=zeros(1,N+1);
                   DP{t}(j,pom+1) =1;
                   DP{t}(j,N+2)=UF;
                 end
              end
           end

        end
end

%visualization of the optimal crop plan sequence
        path=zeros(1, years);
        path(1)=ib;
           for t=2:years
           for j=1:N
           if DP{t−1}(path(t−1),j+1)==1
           path(t)=j
           end
           end
        end
        solution=zeros(years,4);
        t=1:years
        solution(t,:)=state_space(path(t),:);
solution
```

Appendix 2

```
function [zisk] = reward(v) %v je vstupni parametr- radkovy vektor konkretniho oseti

zisky=  [575   3851   1940   1706   2291   1667  –1960   4748   5762  –2818   3968   9350;
        7664   4908   8141   2364   5067   6074   8353  10791   6657    827   –392   6392;
         798   4710   3504   7026    270   5994  –5706  –1632   –936   –720   1512   5898;
        –760   3376   2584  12000   4784  –1376  –6216  10152   4696   4520   –232   4696];
Z=zisky';
kovariancni_matice=cov(Z);
prumerne_zisky=[2590; 5571; 1727; 3185];
a=0.000001;% risk aversion coefficient
hony=[23.9 32.6 26.6 39.4];
rozlohy=[0 0 0 0];
for n=1:4
for k=1:4
              if v(n)==k
                 rozlohy(k)= rozlohy(k)+hony(n);%ha
              end
end
end

pom1=rozlohy*kovariancni_matice;
pom2=rozlohy*prumerne_zisky;
pom3=pom1*rozlohy';
zisk=pom3*0.5*a-pom2;

end
```

REFERENCES

BELLMAN, R., 1957: *Dynamic programming*. The Princeton University Press, Princeton, NJ.

BOND, C. A., FARZIN,Y. H., 2008: *Agroecosystem modeling and optimal economic decisions: Implications for sustainable agriculture*, Optimal Control Applications and Methods, 29, pp. 59–80.

CASTELLAZZI, M. S. et al., 2008: *A systematic representations of crop rotations*. Agricultural Systems, 97, pp. 26–33.

CHIKUMBO, O., MAREELS, I. M. Y., 2003: Predicting terminal time and final crop number for o forest plantation stand: Pontryagin's Maximum Principle in Ecosystems and sustainable development IV, pp. 1227–1235.

DENARDO, E. V., 2003: *Dynamic programming, Models and applications, Dover publications*, New York.

DUFFY, P. A., TAYLOR, C. A., 1992: Long-Term Planning on a Corn-Soybean Farm: A Dynamic Programming Analysis, Abriculrural Systems, 42, pp. 57–71.

FREUND, P. J., 1956: *The introduction of risk into a programming mode.* Econometrica, 24, pp. 253–263.

JANOVÁ, J., 2010: *Optimization models in agriculture and natural resources: the validation and verification process*, in Mathematical methods in economics, 2010, University of South Bohemia, České Budějovice.

JANOVÁ, J., AMBROŽOVÁ, P., 2009: *Optimization of production planning in Czech agricultural co-operative via linear programming.* Acta univ. agric. et silvic. Mendel. Brun., 57 (6), pp. 99–104.

KLEIN HANEVELD, W. K., STEGEMAN, A. W., 2004: *Crop succession requirements in agricultural production planning*, European journal of operational research, 166, pp. 406–429.

PARSONS, D. J. et al., 2009: Weed Manager – A model based decision support system for weed management in arable crops, Computers and electronics in agriculture, 65, pp. 155–167.

SEPPELT, R., VOINOV, A., 2002: *Optimization methodology for land use patterns using spatially explicit landscape models*, Ecological modelling, 151, pp. 125–142.

SEPPELT, R., 1999: *Applications of optimum control theory to agroecosystem modelling*, Ecological Modelling, 121, pp. 161–183.

Address

Mgr. Ing. Jitka Janová, Ph.D. Ústav statistiky a operačního výzkumu, Mendelova univerzita v Brně, Zemědělská 1, 613 00 Brno, Česká republika, e-mail: janova@mendelu.cz