Niels Kjølstad Poulsen

Build. 303b, room 016
Section for Dynamical Systems
Dept. of Applied Mathematics and Computer Science
The Technical University of Denmark

Email: nkpo@dtu.dk
phone: +45 4525 3356
mobile: +45 2890 3797

www2.imm.dtu.dk/courses/42111
What is Dynamic Optimization?

Dynamic Optimization has 3 ingredients:

- A performance **index** (cost function, objective function) depending on the states and decisions.

  In our case it is a summation (or integral) of contribution over a period of time of fixed or free length (might be a part of the optimization).

- Eventually some **constraints**

  on the decisions or on the states.

- Some **dynamics**.

  Here (in this course) described by a state space model.

Lets have a look at some examples:
We are producing a product (brand A) and have to determine its price in order to maximize our income.

There is a competitor product B and a problem.

If we are to modest we might have almost all the customers but we will not earn that much.

If we are to greedy then the bulk majority of the customers will buy the other brand B.
We have to decide the price of the product $u_i \sim u$ ($u$ being the production cost) in each interval.

Let $M$ be the size of the marked and let $x_i$ ($0 \leq x_i \leq 1$) be the (A) share of the marked in the $i$'th interval.

Objective: to make some money - i.e. to maximize

$$\max J \quad \text{where} \quad J = \sum_{i=0}^{N-1} M \bar{x}_i (u_i - u) \quad \quad \bar{x}_i = \frac{1}{2} (x_i + x_{i+1})$$

More precisely, $x_i$ is the marked share at the beginning of interval $i$ and $\bar{x}_i$ is the average share of the marked in interval $i$. 
Optimal Pricing - the dynamics

Dynamics:

\[ x_{i+1} = (1 - p[u_i])x_i + q[u_i](1 - x_i) \]

\[ x_0 = x_0 \]
Recap Optimal Pricing

Dynamics:

\[ x_{i+1} = (1 - p(u_i)) x_i + q(u_i) (1 - x_i) \]

\[ x_0 = x_0 \]

Objective:

\[ \text{Max } J \text{ where } J = \sum_{i=0}^{N-1} M \bar{x}_i (u_i - u) \]

Notice: This is a discrete time model. No constraints. The length of the period (the horizon, \(N\)) is fixed.
If $u_i = u + 5$ ($u = 6, N = 10$) we get $J = 8$ (rounded to integer).
Optimal pricing (given correct model): $J = 27$ (rounded to integer).

Notice different axis for $x$. 
Dynamics (described by a state space model):

\[ x_{i+1} = f_i(x_i, u_i) \quad x_0 = x_0 \]

Objective (to optimize the index):

\[ J = \phi_N(x_N) + \sum_{i=0}^{N-1} L_i(x_i, u_i) \]

Here \( N \) and \( x_0 \) are fixed (given), \( J, \phi \) and \( L \) are scalars. \( x_i \) and \( f_i \) are \( n \)-dimensional vector and vector function and \( u_i \) is a vector of decisions.

Notice: no constraints (except given by the dynamics).
Dynamics:

\[ x_{i+1} = x_i + u_i - s_i \quad x_0 = x_0 \]

Stock: \( x_i \) \quad 0 \leq x_i \leq \bar{x}
Production: \( u_i \) \quad 0 \leq u_i \leq \bar{u}
Sale: \( s_i \) \quad 0 \leq s_i \leq \min(x_i, w_i)
Order: \( w_i \)

Notice: constraints on decisions and states. Stochastics involved.
Goals:
- to earn some money
- to avoid situation with no stock
- to reduce stock charge
- to obtain an even production.

Objective (index to be maximized):

\[
J = \sum_{i=0}^{N-1} (p s_i - c u_i - k x_i - h \text{Max}(w_i - s_i, 0))
\]

where \(p\), \(c\), \(k\) and \(h\) are constants (prices).
Constrained Dynamic Optimization

Dynamics (described by a state space model):

\[ x_{i+1} = f_i(x_i, u_i) \quad x_0 = x_0 \]

Objective (to optimize the index):

\[ J = \phi_N(x_N) + \sum_{i=0}^{N-1} L_i(x_i, u_i) \]

Constraints:

\[ g(x_i, u_i) \leq C_i \]
Variations of the problem

Dynamic optimization with:
- Terminal constraints (take the system from one place to another).
- Constraints (on $u_i$ and $x_i$ within the horizon).
- Continuous time problems
- Open final time (Minimum time problems).
- Stochastic elements (orders in the inventory problem).

2 examples.
Find the shape i.e. $r(x)$ of an axial symmetric nose, such that the drag is minimized.

The decision $u(x)$ is the slope of the profile:

$$\frac{\partial r}{\partial x} = -u = -\tan(\theta)$$

$r(0) = a$
Find the shape i.e. $r(x)$ of a axial symmetric nose, such that the drag is minimized.

$$D = q \int_0^a C_p(\theta)2\pi r dr$$

$$q = \frac{1}{2} \rho V^2 \quad \text{(Dynamic pressure)}$$

$$C_p(\theta) = 2\sin(\theta)^2 \quad \text{for} \quad \theta \geq 0$$
Dynamic:

\[
\frac{\partial r}{\partial x} = -u \quad r_0 = a \quad \tan(\theta) = u
\]

Cost function (drag coefficient, including a blunt nose):

\[
C_d = \frac{D}{q\pi a^2} = 2r_i^2 + 4 \int_0^l \frac{ru^3}{1 + u^2} \, dx \leq 1
\]
Minimum drag nose shape (Newton)

\[ r/a \]

\[ x/a \]

\[ C_d = 0.098 \]
Find a function $u_t \in [0; T]$ which takes the system

$$\dot{x} = f_t(x_t, u_t)$$

from its initial state $x_0$ along trajectories such that the performance index

$$J = \phi_T[x_T] + \int_0^T L_t(x_t, u_t) \, dt$$

is minimized.
Thrust direction program for minimum time transfer from Earth orbit to Jupiter orbit.

\[ \dot{r} = u \]

\[ \dot{u} = \frac{v^2}{r} - \frac{1}{r^2} + a \sin(\theta) \]

\[ \dot{v} = -\frac{uv}{r} + a \cos(\theta) \]
Min. Time Orbit Transfer

\[
\begin{bmatrix}
\frac{d}{dt} r \\
\frac{d}{dt} u \\
\frac{d}{dt} v
\end{bmatrix} =
\begin{bmatrix}
u \\
v^2 - \frac{1}{r} - \frac{1}{r^2} + a \sin(\theta) \\
-uv + a \cos(\theta)
\end{bmatrix}
\]

Initial conditions
Terminal conditions
\[J = T\]
42111/02711 Static and Dynamic Optimization

- Course description (in Danish)
- Course description (in English)

Lecture slides for Static Optimization are found on CampusNet in the folder Static Slides

Lecture slides for Dynamic Optimization:

- L7: Free dynamics optimization-D (pdf).
- L8: Free dynamics optimization-(D+C) (pdf).
- L9: DO with end point constraints (pdf).
- L10: DO with control constraints (pdf).

Dynamic exercise slides:


The note "Dynamic Optimization" is found here in pdf.
Mark Gockenback: A Practical Introduction to MATLAB (as ps) or (as html)

Last modified: September 3, 2012 by nkp
Concluding remarks

- have 42111 and your study number in the subject field when emailing us
- Matlab available on Gbar download site