Discrete Optimisation
Project 1: Simulating Economic Agents on Isolated Islands

Estimated duration: 10 hours

2nd November 2016

The Anno game series proposes to impersonate an explorer that arrives in a new area. They settle on islands, and must then fulfil the needs of the burgeoning colony. The first goods are easy to produce, as they are directly extracted from natural resources; as the game progresses, these goods require more elaborate production chains, starting from natural resources (such as crop) to finished products (like beer).

Deriving mathematical models for this game may serve multiple purposes, the first one being a player determining an optimal way of playing the game. As formal optimisation algorithms (such as those studied in this course) *implicitly* check all existing solutions for the best one, this kind of model can be used to check whether the game is balanced: making certain choices do not bring enormous advantages, so that the player is free to choose their strategy. To this end, the game designers may tune parameters and observe the effects by solving again the model or by performing new simulations. Paul Tozour has written a series of blog posts about this topic.

This project asks you to determine the best decisions a player may take during the game for a series of consecutive objectives.

Remark. You may compute by hand quite easily the expected results for the three first questions in order to check your model. However, this way of solving the problem is not sufficient for this project, you must present one optimisation model for each of the four steps. The proposed progression should help you to solve the steps cumulatively.

1 Statement

The game evolves as follows:

1. **Peasants (no chains):** the main need for the first settlers is food, more precisely fish. Each minute, 100 peasants consume `consumption`[: `peasant`, : `fish`] tonnes of fish, and one fisherman's hut produces `production`[: `fisherman`, : `fish`] tonnes (these values are all defined in Section 2).
   What is the minimum number of fisherman's huts that are required to feed 500 peasants?
   When the game starts, the colony has twenty tonnes of fish.
   There is no need to take into account dynamic aspects for this part (i.e. you are not required to include time in your model). The number of buildings is not allowed to vary in time. You may use the data from Table 1.

2. **Citizens (simple chains):** as the time goes by, the colony has more complex needs. Some of your inhabitants are now citizens and need linen garments. These garments are produced by transforming hemp in a weaver's hut; if the weaver has a delivery of hemp, they need one complete minute before the first tonne of linen garments is output (this will be true for all transformation buildings from now on).
   When the game starts, the colony has twenty tonnes of fish and twenty tonnes of linen garments.
   What is the minimum number of fisherman's huts, hemp plantations, and weaver's huts to fulfil the needs for fish and linen garments of 2,000 peasants and 800 citizens?
   There is no need to take into account dynamic aspects for this part, even though this might help to solve the next points. You may use the data from Table 2.

3. **Patricians (multiple islands):** less than a few hours of gameplay later, a new kind of inhabitant gets into the city—patricians. They require more elaborate goods, made of wheat, a crop that cannot be

grown on your main island. Therefore, this wheat must be sown on another island, and then imported by boat (you have two of them at your disposal). The harvested crop is piled up on this second island, then moved by boat, before being transferred into your main island's warehouses and then transformed. The new required good is beer, which is obtained by fermentation of wheat. The wheat is sown and harvested on your second island, then brought back to your main island, where it is transformed. Moving the wheat by boat takes some time (i.e. the wheat storage is decreased on the second island, then increased on the main one a few minutes later); a boat has a maximum capacity; the two boats may start at any time, irrespectively of their load. Loading and unloading a boat is instantaneous.
When the game starts, the colony has twenty tonnes of fish, twenty tonnes of linen garments, and twenty tonnes of beer.
1. What is the minimum number of each building to meet the needs of 500 peasants, 2000 citizens, and 800 patricians?
2. Is the number of boats a limitation for the growth of the colony? (In other words: if the colony's wheat needs increase, would those two boats still be enough?)
You may use the data from Table 3.

4. **Nomads (money, money, money):** you discover another civilisation, living in the Orient, and they teach you the value of money. You now realise that building a good-producing infrastructure costs money, but this is even worse when considering the maintenance. However, your colony can also levy tax from your inhabitants, and sell some of its production to merchants.
The price of goods evolves over time (some goods are more needed than others at some periods of time); you can only sell them from your main island; one merchant stops by your island each minute, with only one boat, whose capacity is the same as that of the player's boats (all goods count towards this capacity). Each inhabitant pays taxes depending on its level (e.g., patricians pay more than peasants); as the size of the island is limited, the maximum population is capped. The population must respect a given pyramid, so that each inhabitant level (including higher ones) is limited to a given proportion of the total population (more precisely, at most 80% of the population can be citizens or patricians, at most 56% of the population can be patricians)—see Figure 1. Buildings only have a maintenance cost that must be paid each minute; these costs are proportional to the number of buildings (i.e. two fisherman's huts cost twice as much as one fisherman's hut); the construction costs of a building are ignored. Due to the limited space available on each island, the number of buildings is limited: the fisherman's huts are built on the coast; the other buildings share the inland, and the farms occupy much more space than the other buildings, due to their fields; housing takes no space.
When the game starts, the colony has twenty tonnes of fish, twenty tonnes of linen garments, and twenty tonnes of beer.
1. What configuration (in terms of producing buildings and inhabitants) is the most profitable one?
2. Are all production chains equivalent in terms of profitability (gold from selling the products of this chain minus the total maintenance costs for this chain)?
3. Is is profitable to have a high population?
You may use the data from Table 4.

Remark. All your models will be approximations of the game's reality. When details are left unspecified, any way of modelling the situation is acceptable.

2 Available data

The Tables 1 to 4 list the available data for this project, associated with a Julia expression to retrieve them in your code. This data is given incrementally: e.g., for the third part, you may use the data from Tables 1 to 3.

Symbol	Unit	Description
`production[:fisherman, :fish]`	tonnes per minute	Quantity of fish a fisherman's hut can produce per minute
`consumption[:peasant, :fish]`	tonnes per minute	Quantity of fish 100 peasants consume per minute

Table 1: Available data for the first question.

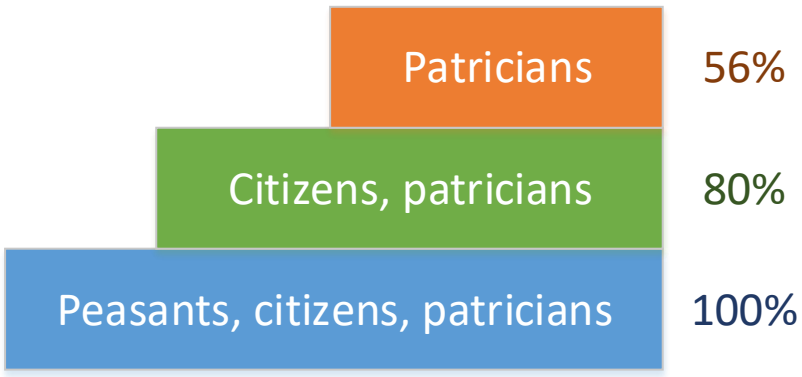

Figure 1: Population pyramid.

Symbol	Unit	Description
`production[:hempplantation, :hemp]`	tonnes per minute	Quantity of hemp a hemp plantation can produce per minute
`production[:weaver, :linen]`		Quantity of linen garments a weaver's hut can produce per minute (when fully supplied with hemp)
`consumption[:citizen,` *good*`]`	tonnes per minute	Quantity of fish (*good* = `:fish`) or linen garments (*good* = `:linen`) 100 citizens consume per minute
`consumption[:weaver, :hemp]`	tonnes per tonne	Quantity of hemp a weaver's hut requires to produce a tonne of linen garments

Table 2: Available data for the second question.

Symbol	Unit	Description
`production[:cropfarm, :wheat]`	tonnes per minute	Quantity of wheat a crop farm can produce per minute
`production[:brewery, :beer]`		Quantity of beer a brewery can produce per minute (when fully supplied with wheat)
`consumption[:patrician,` *good*`]`	tonnes per minute	Quantity of fish (*good* = `:fish`), linen garments (*good* = `:linen`), or beer (*good* = `:beer`) 100 patricians consume per minute
`consumption[:brewery, :wheat]`	tonnes per tonne	Quantity of wheat a brewery requires to produce a tonne of beer
`boat_time[:secondary, :main]`	minutes	Time for a boat to travel from the secondary island to the main one (ensured to be an integer number)
`boat_time[:main, :secondary]`		Time for a boat to travel from the main island to the secondary one (ensured to be an integer number)
`boat_capacity_max`	tonnes	Maximum load per boat

Table 3: Available data for the third question.

Symbol	Unit	Description
`maintenance[`*building*`]`	gold coins per building and per minute	Maintenance cost for one *building* (chosen in the list: `:fisherman`, `:hempplantation`, `:cropfarm`, `:weaver`, `:brewery`)
`tax[`*inhabitant*`]`	gold coins per inhabitant and per minute	Tax levied from *each* inhabitant of level *inhabitant* (chosen in the list `:peasant`, `:citizen`, `:patrician`)
`price[`*good*`, `*t*`]`	gold coins per tonne	Price the merchant stepping by after t minutes of game play pays for a tonne of good *good* (chosen in the list `:fish`, `:hemp`, `:wheat`, `:linen`, `:beer`)
`maximum_population[:main]`	number of inhabitants	The maximum population of the main island (sum of peasants, citizens, and patricians)
`proportion_max[:peasant]`	percentage	Maximum proportion of peasants, citizens, and patricians in the total population
`proportion_max[:citizen]`	percentage	Maximum proportion of citizens and patricians in the total population
`proportion_max[:patrician]`	percentage	Maximum proportion of patricians in the total population
`coast_space_max[`*island*`]`	length	Available coastal length available on *island*
`land_space_max[`*island*`]`	surface	Available inland surface on *island*
`space_use[`*position*`, `*building*`]`	length if *position* = `:coast`, surface if *position* = `:land`	Surface (when *position* = `:land`) and coastal length (when *position* = `:coast`) a *building* occupies

Table 4: Available data for the fourth question.

3 Code structure

You are expected to fill the given skeleton (available on the course's web page), with one model for each of the four parts; the required data for each question is already available within the skeleton. A model corresponds to one function to implement, whose name starts with `Q` followed by the number of the part (1 to 4). Each function is supposed to take as its only argument a JuMP model; for example, the first function can be called as `Q1(Model())`. The return values should be ordered as follows:

1. For the first part: the number of fisherman's huts.
2. For the second part: the number of fisherman's huts, hemp plantations, and weaver's huts.
3. For the third part (question 1): the number of fisherman's huts, hemp plantations, crop farms, weaver's huts, and breweries.
4. For the fourth part (question 1): the number of fisherman's huts, hemp plantations, crop farms, weaver's huts, and breweries; the number of peasants, citizens, and patricians.

The exact data structures to return are described in the skeleton.

Apart from these requirements, you are free to organise your Julia code however you want.

4 Instructions

Individually, develop a sequence of mixed-integer linear models for this problem. You are asked to write two files:

1. A **brief** report explaining your models. It should include the definition of variables, the mathematical statement of the objective function and of the constraints, with brief explanations, following the steps defined in Section 1. Clear answers to the questions (highlighted with an italic in Section 1) must be provided; ideally, the answer to the non-first questions (in parts 3 and 4) should be obtained by an analysis of the solutions obtained to your models or a small alteration of the model for the main questions, you are not required to write code to solve them. No code shall be shown in the report.
   The best way of providing useful explanations is to use lists, and not large blocks of text; related constraints are better presented in contiguous groups.

2. A Julia implementation (one model per step), following the requirements in Section 3. Clearly indicate which version of Julia you used (e.g., the one installed on the MS800 machines or on the Thalès compute server).

Both the report (in PDF format) and your source code must be handed in on or before Friday 4 November 2016 (23:59:59) on the submission platform, as a ZIP or 7Z archive.

The evaluation criteria are the following:

- for the optimisation models:
  - model adaptation to the described situation
  - linearity of both the constraints and the objective function
  - links between the variables
  - quality of the report (including the language quality, organisation, and justification of the needed approximations if any)
- for the implementation:
  - respect of specified input and output formats (Section 3)
  - concordance with the optimisation model in the report
  - quality of the code (including clarity and brevity of the code); copy-paste is allowed if justified

Both parts have the same weight in the total score for this project.