Aalto-yliopisto
Perustieteiden
korkeakoulu

Extending the Decision Programming framework with continuous decisions (valmiin työn esittely)

Antti Ahvenjärvi

01.11.2023

Advisor: *Olli Herrala*

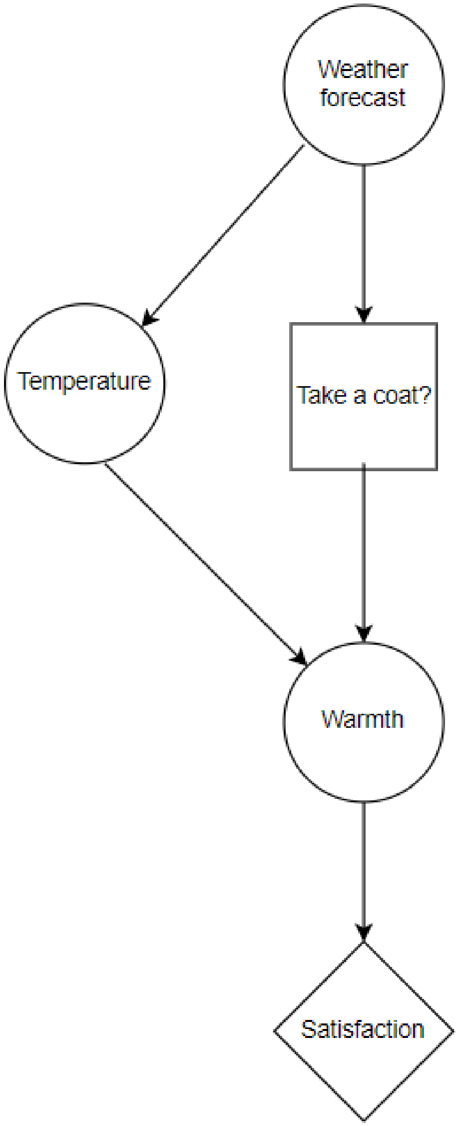
Supervisor: *Fabricio Oliveira*

Systeemianalyysin
laboratorio

Influence diagram

- Acyclic directed graph
- Decision nodes (squares)
- Chance nodes (circles)
- Value nodes (diamonds)
- Arcs the flow of information

- Example diagram: choosing optimal clothing to match the weather



Decision Programming framework

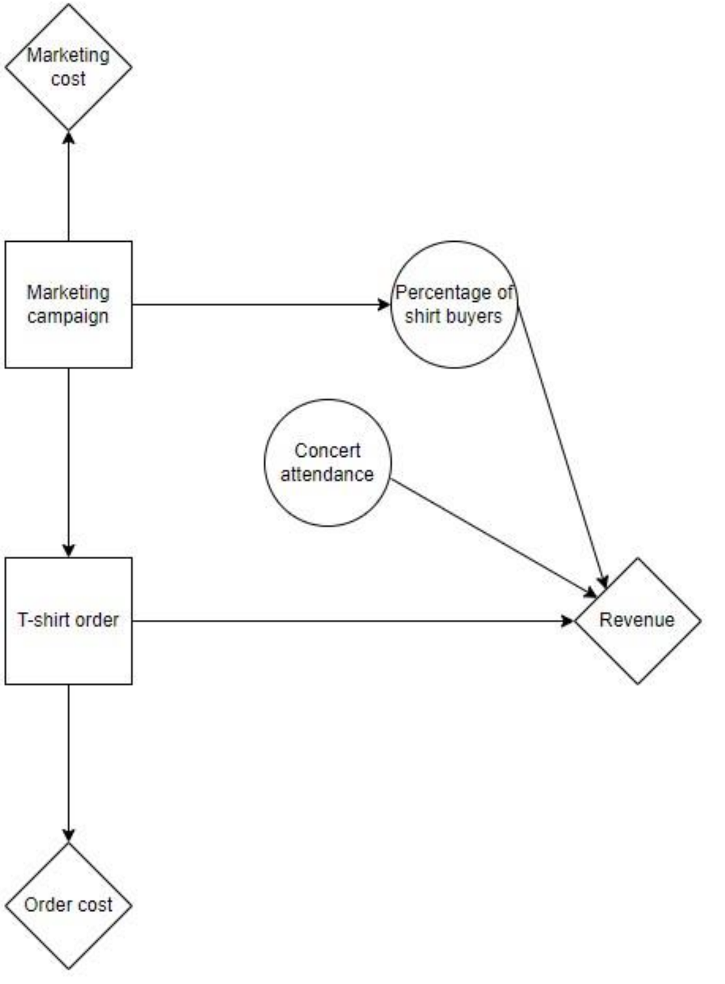
- Solving an influence diagram means finding the decision strategy that results in the best objective value
  - Often maximizing the expected utility
- The framework turns problems represented with influence diagrams into MILP problems
  - Solved with solvers like Gurobi
- Has a limitation that the decisions and chance events must have finite set of discrete states

Aim of the thesis

- Extend the Decision Programming framework to solve a simple problem having a continuous decision variable
- Solve a discrete and continuous version of the problem using Julia package DecisionProgramming.jl
- Compare the continuous and discrete formulations of the problem

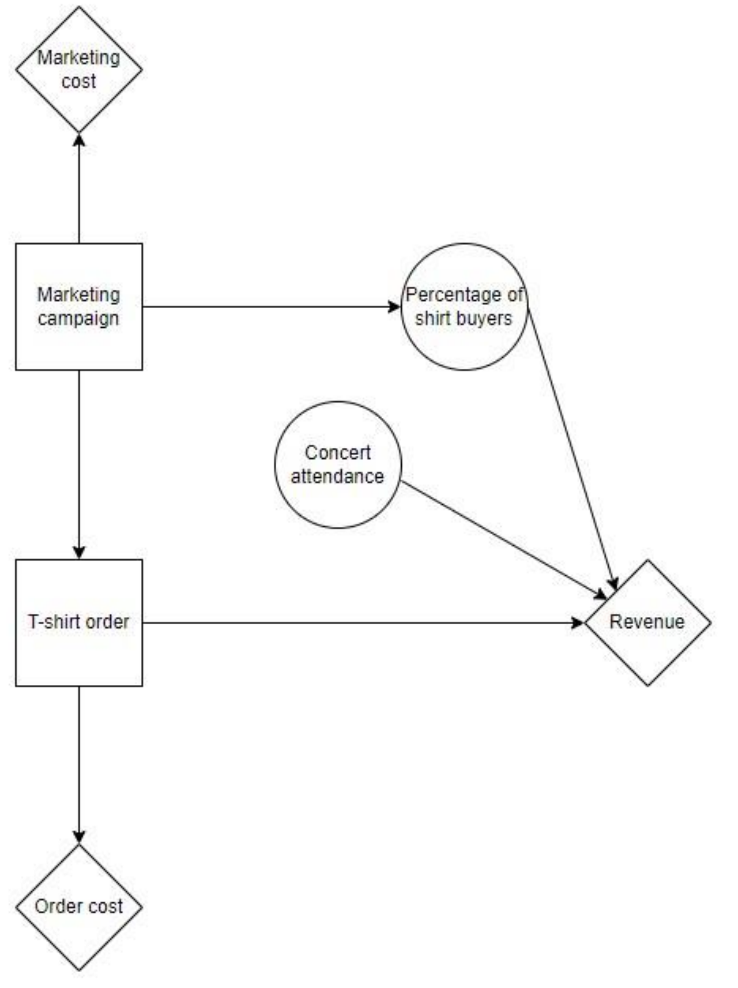
Example problem: T-shirt sales

- T-shirts sold in front of a concert venue
- Concert attendance and percentage of attendance who will buy the shirt are unknown
- T-shirt order size is modeled as a continuous decision
- A decision (yes/no) for arranging a marketing campaign
  – Increases the popularity of t-shirts among the attendees



Information states and paths

- An immediate predecessor of a node **j** is a node **v** that has an arc to the node **j**
- An information state $s_{I(j)}$ of a node **j** is the realizations of its immediate predecessor nodes
- A path ***s*** in a diagram is a sequence of realizations in the decision and chance nodes
  - (Yes, 5000_{shirts}, $70000_{\text{attendees}}$, 10%)



The continuous extensions

- Continuous decision variables $y_{s_{I(i)}}$ are added, one for each set of paths that have a different realization for the information state $s_{I(i)}$ of the continuous decision ***i***
- In our problem node ***i*** is the order size decision
  - ***i*** has two information states corresponding to the states of the marketing decision
- For each path ***s***, a function U(***s***, $y_{s_{I(i)}}$) maps $y_{s_{I(i)}}$ and discrete chance and decision realizations to a utility value

Results

- In the discretized version the order size was broken into k discrete states
- With 5 or more states the discrete model was larger than the continuous on all measured scales

→ Continuous version is faster to solve

	Discrete version with k states	Continuous version
All variables	20k + 2	58
Binary variables	2k + 2	2
Constraints	20k + 6	98

Conclusions

- Continuous version has a smaller model size and gives an exact solution
  - Discrete version sufficient when approximations are enough
  - With a small problem it is also possible to use small number of states and iteratively update them to find optimum
- Similar reasoning could be applied to larger problems
- Restrictions:
  - The example is rather simple, and it gives not much insight on the computational performance on larger problems
  - The extended formulation is restricted to problems where there are no decision or chance nodes depending on the continuous decision

References

- Ahti Salo, Juho Andelmin, and Fabricio Oliveira. Decision programming for mixed-integer multi-stage optimization under uncertainty. European Journal of Opera-tional Research, 299(2):550–565, 2022.
- Fabricio Oliveira, Olli Herrala, Jaan Tollander de Balsch, Helmi Hankimaa, ja Juho Andelmin. DecisionProgramming.jl, October 2021. URL https://github.com/gamma-opt/DecisionProgramming.jl
- Robert T Clemen and Terence Reilly. Making hard decisions with DecisionTools. Cengage Learning, 2013.