Optimized Staff Scheduling at Swissport

Prof. Dr. Andreas Klinkert
Dr. Peter Fusek
Dipl. Ing. Roman Berner
Rita Thalmann
Simona Segessenmann

Zurich University of Applied Sciences
Zurich University of Applied Sciences
Swissport, Optimization Specialist
Swissport, Application Project Manager
Swissport, Application Specialist
Project Overview

Project "Auto-Rostering"

- Joint Development of Swissport and Zurich University of Applied Sciences (ZHAW)
- Initially: Swiss National Research Project (CTI)
- Now: Strategic R&D Cooperation Swissport - ZHAW
Project Stakeholders

Swissport International Ltd.

• Largest international ground handling company, 61'000 employees
• 290 airports worldwide, 48 countries, 835 customer airlines
• 230 million passengers, 4.1 million tonnes cargo (per year)
• Pilot site: Zurich Airport (80+% by Swissport)
Project Stakeholders

Airport Ground Handling: Business Areas

• **Passenger Services:**
  – Check-In
  – Gate Handling
  – Transfer Services
  – Surface Transports, Special Assistance, VIP Lounges, ...

• **Ramp Services:**
  – Baggage Handling
  – Aircraft Handling: Push-back tractors, Ground power units, Stairs, ...
  – Aircraft Servicing and Cleaning
  – Unit Load Device (ULD) Control and Management
  – Aircraft maintenance
  – Executive aviation handling, ...
Institute of Data Analysis & Process Design (IDP)

• Zurich University of Applied Sciences (ZHAW), School of Engineering
  – Founded 1874, 9 bachelor programs, national MSE master, 2000 students, 13 R&D institutes and centers

• IDP Activities:
  – Business projects (R&D, Consulting, …)
  – Teaching: "Wirtschaftsingenieurwesen", Aviation, Traffic Systems, IT, …

• IDP Competencies:
  – Design, analysis and optimization of complex business processes
  – Business Analytics: Data Analysis, Statistics, …
  – **Optimization Group**: LP, ILP, NLP, Polyhedral Combinatorics, Graph Theory, Heuristics, Metaheuristics, Simulation, Stochastic Process Modeling, …
Overview: Staff Scheduling at Swissport

General Personnel Planning Functions:

- Task Generation & Shift Construction
  - Translate demand from flight schedules into shifts

  - Rostering
    - Days-Off Planning
    - Shift Assignment

  - Real-Time Dispatching
    - Control of real-time situation

Software at Swissport:

- Inform - GroundStar
  - "GS Planning"

- Axedo: "Web-Roster"

- SWP: Manual Rostering

- Interflex: "GS Rostering"

- Inform - GroundStar
  - "GS RealTime"

Opportunity: Auto-Rostering
Project Context

Tool: Task Generation and Shift Construction

["GS Planning", from GroundStar Suite, INFORM, Aachen, Germany]
**Project Context**

**Tool: Rostering**

["GS Rostering", SP-Expert, INTERFLEX, Stuttgart, Germany]

- Manual planning board
- No support for automated, optimized planning

<table>
<thead>
<tr>
<th></th>
<th>MON</th>
<th>TUE</th>
<th>WED</th>
<th>THU</th>
<th>FRI</th>
<th>SAT</th>
<th>SUN</th>
</tr>
</thead>
<tbody>
<tr>
<td>JU</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TH</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SA</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SU</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

*Please note that the image contains a Gantt chart with times and days marked.*
Motivation for Automatic Rostering

Initial Rostering at Swissport

- Manual rostering (via GS Rostering)
- 15 - 20 planners
  - special education, long-term experience
- **230+ days per month** planning effort
- High importance of employee preferences
- Various informal planning aspects
  - planners implicitly know preferences of their employees
- Different individual planning policies of planners
- Different quality of plans
- Different opinions on fairness and quality
- **Expensive, laborious, time consuming, subjective**
  → Opportunity for Automatic Rostering
Motivation for In-House Development

Need For Development of New Rostering Tool

• Rostering at SWP (in particular Zurich Airport):
  – Complex, station specific, monthly planning

• **Employee preferences** of crucial importance ("Shift Bidding")
  – Employee satisfaction critical for success

• **Numerous** types of regulations and preferences
  – Labor law, unions, company regulations, personal wishes, etc.

• **Complex informal** framework for individual preference handling

• Preference fulfillment: about **95%** with manual planning

• **Large-scale** planning groups:
  – Total 2000+ employees at Zurich Airport
  – Planning groups up to 1000 employees, hundreds of shifts per day
Motivation for In-House Development

Need For Development of New Rostering Tool (cont.)

• **Demand-driven** rostering, no repetitive shift patterns ("wheels")
  – Take into account dynamic variation of demand

• **Evaluated** commercial rostering tools were **inadequate** or produced **unsatisfactory** results, e.g.
  – Solutions only **partially feasible**, to be fixed manually
  – Variety of constraints and goals too complex or not representable
  – Problem dimensions too large
  – Computational "instability": small input changes yield large output changes ("planner's nightmare")
  – No information about solution quality ("optimality gap")
  – Lack of bottleneck analysis, rapid rough-cut planning, and decision support features
Methodological Aspects

Exact Methods vs. Heuristics

• Most large-scale rostering tools mainly rely on meta-heuristics based on stochastic search:
  – Trajectory based: Hill Climbing, Tabu Search, Simulated Annealing, (Variable) Neighborhood Search, …
  – Population based (evolutionary): Genetic Algorithms, Scatter Search, …

• Typically (far) sub-optimal solutions
• No information about solution quality (distance from optimum)
• Inherent high degree of randomness
• Little exploitation of mathematical problem structure
Auto-Rostering: Methodology

**Exact Methods vs. Heuristics (cont.)**

- Auto-Rostering substantially relies on **exact** methods, in particular **Integer Linear Programming**

- **Explicit** mathematical description of solution space
  - Intensive exploitation of mathematical (polyhedral) **structure**
  - Elaboration of good polyhedral **formulations** (if possible)

- Solution by means of high-performance ILP solver:
  - **Gurobi** is the only solver able to solve our problems in reasonable time and quality (THANKS, GUROBI !!!)

- Reduced computation time through smart B&B truncation

- **Combination** with various other large-scale optimization techniques:
  - decomposition
  - relaxation
  - pre- and post-processing
  - heuristic procedures
Handling of Hard Constraints

- **Intrinsic issue** with meta-heuristics used in most rostering tools:
  - Difficulty to explicitly handle hard constraints
  - Common approach: constraint relaxation with penalty ("dualization")

- High risk to produce **infeasible** (partially feasible) solutions:
  - Not all hard constraints satisfied
  - To be fixed manually by human planners

- **Auto-Rostering** guarantees **strict feasibility** of solutions:
  - Possible due to underlying MIP methodology
  - If no feasible solution exists (mathematically proven): explanation and hints for recovery (crucial)
Auto-Rostering: Methodology

Feasibility Issues

- Tools based on *(meta-)*heuristics and constraint relaxation always produce "solution"
  - "Solution" often has unsatisfied hard constraints, i.e. "infeasible solution"

- *In contrast:* Tools based on MIP may produce no solution at all!
  - Because of explicit formulation of hard constraints

- For users, getting no solution is inacceptable
  - Psychologically, users typically prefer "rubbish" over "nothing".

- Best approach to handle infeasibilities:
  - Provide *hints* about causes of infeasibilities, bottlenecks and recovery

- Computing infeasibility hints is challenging
  - With regard to both, methodology and computational complexity
Feasibility Issues (cont.)

- Possible approaches for infeasibility hints:
  - Search minimal inconsistent constraint subsystems
    - Supported by Gurobi, but i.g. computationally hard
    - Maybe no answer in time, or not interpretable
  - Solve entire "brute force" relaxation and interpret slacks
    - Supported by Gurobi and LPL modelling language
    - Maybe not tractable, since no answer in time
    - Maybe tractable, but slacks not interpretable
  - Heuristically devise sophisticated partial relaxations/decompositions
    - Then interpret slacks and/or partial solutions
    - In our project, most successful approach
    - But high effort for development and programming

- Regarding overall time and financial budget of this project:
  - Infeasibility handling consumed approx. **40%** of total budget
  - Reason: Very tightly constrained problems, most instances initially infeasible
Implementation, Results

Computational Results

• **MIP Model:**
  – Implemented in LPL (algebraic modeling language, virtual-optima.com)
  – About 15'000 lines LPL MIP code

• **Java Framework** (core data model, controllers, adapters, scripting, etc.):
  – About 20'000 lines of Java code

• Large MIP instances, e.g.:
  – 48'041 rows, 2'189'126 cols, 5'785'274 nonzeros (before Gurobi presolve)
  – 34'176 rows, 216'962 cols (binary), 955'861 nonzeros (after Gurobi presolve)

• **Computation time** (until sufficient gap, typically << 1%):
  – About 15 - 30 hours for most difficult instances

• Computation time is permanent issue and challenge:
  – results must be delivered within strict operational deadlines
  – computation times at limit of deadlines
  – fluctuation over instances of same group
  – fluctuation for different computational random seeds
Final Remarks

Conclusion

• Swissport: Complex large-scale rostering problems
• No satisfactory commercial software
• Approach based on MIP (instead of metaheuristics)

• Advantages:
  – High quality results, lower computation times, complex instances solvable
  – Explicit handling of hard constraints, higher computational stability, …

• Issues:
  – Computation times at limit of deadlines, unpredictable fluctuations
  – Sophisticated and expensive handling of infeasibility

• Deployment at Zurich Airport completed 2015 (9 planning departments)
  – Substantial planning improvements and financial benefits

• Deployment and commercialization continues with other airports and customers