ISI International School for Informatics HAGENBERG

JOHANNES KEPLER UNIVERSITÄT LINZ | JKU

GRASP with Path Relinking for a Production Planning Problem

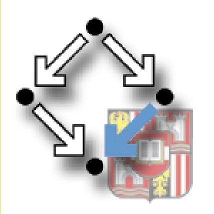
Master Thesis

Plamen Alexandrov, ISI Hagenberg '09

Contents

Master's Project Info

- International School for Informatics (ISI Hagenberg)
- Johannes Kepler University
  - **Academic Advisor:**
    Prof. (FH) Priv.-Doz. DI Dr. Michael Affenzeller
- RISC Software GmbH Company
  - **Industrial Advisors**:
    DI Dr Peter Stadelmeyer, DI Roman Stainko
- Two **ISI students**:
  - Alexandra Jimborean implements Tabu search
  - Plamen Alexandrov implements GRASP



Production Planning

- The purpose of production planning is:
  - To minimize **production time and costs**
  - To efficiently organize the use of **resources**
  - To maximize **efficiency** in the workplace
- It should be achieved under changing **selling conditions**.
- Optimal use of **production capacities** and **resources** is needed.
  - Solution: effective **production planning**.

Production Planning
A Problem from Industry

- Production consists of:
  - **21 machines** with different production capacities
  - **5 days** planning horizon
  - **272 product types**, called items
  - **Settings** for producing a different product (3 hours)
  - **Customer demands** as jobs in daily numbers
  - **Future demands** for the following week
  - **Backlog costs** for unmet demands
  - No limiting storage costs

Production Planning
a "Toy" Example

- The first plan is worse:

	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
M1	2	34	34	34	34	34	34	34	34	34	34	34	34	34	34
M2	37	37	37	37	37	37	16	16	16	16	16	16	16	16	50
M3	34	34	34	34	34	14	14	14	41	41	16	16	12	12	12
M4	12	12	12	12	12	34	34	34	34	12	12	50	50	50	50
M5	50	16	16	16	16	16	41	41	41	34	34	34	34	34	34
M6	14	14	14	14	14	14	14	18	18	18	18	18	18	18	2

- Than the second:

	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
M1	34	34	34	34	34	34	34	34	34	34	34	34	34	34	34
M2	37	16	16	16	16	16	16	16	16	16	16	16	16	12	12
M3	34	34	34	34	34	34	34	34	34	34	34	34	34	34	34
M4	12	12	12	37	37	37	37	37	37	16	16	16	16	16	16
M5	50	50	41	41	41	41	41	41	41	41	50	50	50	50	2
M6	5	14	14	14	14	14	14	14	14	18	18	18	18	18	18

Production Planning Objective

- The objective of our problem is to minimize **setup costs** and **backlog costs**.
- Moreover, if there is available capacity, **future demands** should be processed.
- Production schedules exceeding the **maximum possible production** of an item at the end should be penalized.

Production Planning Modeling

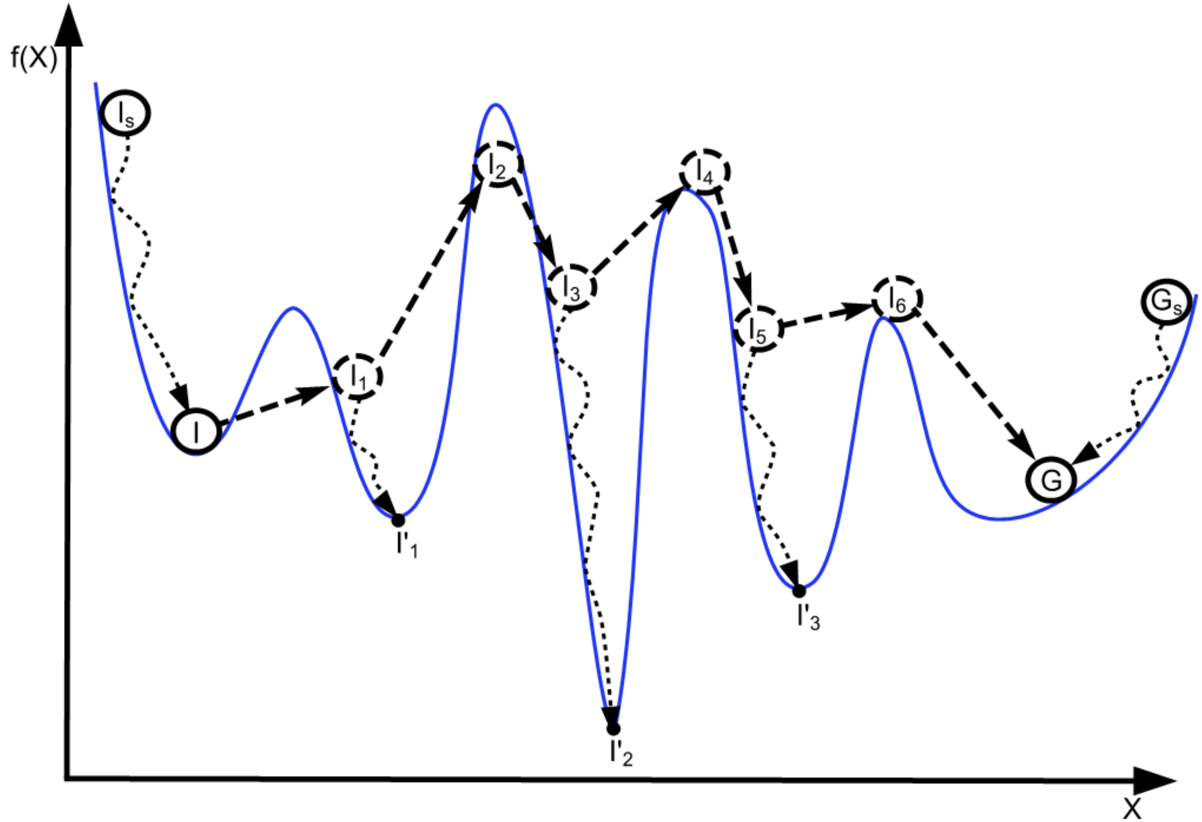
- Production processes are described by **lotsizing and scheduling models**
- We use a **Discrete Lot Sizing and Scheduling Model**
  - The problem is **NP-hard**!
  - If backloging is not allowed, even obtaining a feasible solution is NP-complete!
- Additional constraints reduce the problem size:
  - Maximum number of settings per period
  - Limitation of parallel production

GRASP Metaheuristic

- Greedy Randomized Adaptive Search Procedure (GRASP)
- GRASP is a metaheuristic which combines:
  - A semi-greedy construction heuristic and
  - A local search improvement heuristic
- GRASP is a multi-start iterative strategy. At each iteration:
  - An initial solution is constructed by a semi-greedy construction procedure (one element at a time)
  - Local search applies iterative improvement of the initial solution until a local optimum is found

GRASP with Path Relinking Solution Combination

- Uses **adaptive memory** during the search process: pool.
- Exploits this memory by **combining solutions**



GRASP with Path Relinking
Basic Structure

- $P = \emptyset$; // an empty pool with fixed maximum pool size
- **for** i = 1 ... max_iter **do**
  - X_s = *GreedyRandomizedConstruction*(rand);
  - X_l = *LocalSearch*(X_s, rand); // with randomized nhood
  - P→*Accept*(X_l);
  - **if** i % relink_interval == 0 **then**
    - X_e = P→*SelectEdge*(X_l);
    - X_l = P→*Relink*(X_l, X_e);
- **repeat** P = P→*RelinkAll*(); // post-optimization phase
- **until** No Further Improvement;
- **return** P → Best();

Computational Comparison (1 hour runtime)

Probl.	Result	*MIP* Value	*MIP* Time	*TS* Value	*TS* Time	*GRASP-MR* Value	*GRASP-MR* Time	*GRASP-PAR* Value	*GRASP-PAR* Time
*big*1	avg.	42.539	60	38.625	65	39.327	47	38.954	25
	best					38.591	47	36.625	10
*big*2	avg.	96.183	60	96.784	60	98.413	44	96.201	23
	best					95.386	42	92.47	14
*big*3	avg.	240.61	60	292.483	69	246.71	42	247.54	20
	best					242.32	27	241.12	9
*toy*1	avg.	439.94	60	438.75	20	438.75	5.76	438.75	3.68
	best	438.75	120			438.75	1	438.75	0.17
*toy*2	avg.	24.909	60	21.922	10	23.525	11	22.208	6.12
	best					21.922	13	21.14	1.3
*toy*3	avg.	37.139	60	37.119	10	29.679	13	29.822	7
	best					28.011	15	28.011	2

Achievements and Contributions

- Achievements:
  - Sequential implementation of ***GRASP*** for the initial problem (in C++).
  - Design and implementation of ***Path Relinking***.
  - **Parallel implementation** of the algorithm (in Open-MP).
  - **Comparison** with other approaches: *MIP* and *Tabu Search*
- Contributions:
  - Obtaining **competitive results**(even better, short runtime)
  - **Parametrization** of simple heuristics
  - Defining **selection pressure** for randomized local search
  - Designing a new **path exploration strategy**

Future Research Directions

- Self-adaptive Parameter Optimization
  - **Randomization parameters** of construction heuristic (*Reactive GRASP*)
  - **Selection pressure** (relinking suboptimal solutions)
  - **Diversity in pool** (regular relinking, mutation)
  - **Improvement frequency** (longer runtimes, diversity)
- Distributed Parallel Strategies
  - **Independent strategies** (TTT value)
  - **Collaborative strategies** (migration, diversity)
  - **Effect of PRNGs** (parallel schemes, *Monte Carlo*)