A Pyramidal Evolutionary Algorithm with Different Inter-Agent Partnering Strategies for Scheduling Problems

Annual Operational Research Conference 43, Bath, UK, 2001.

This paper combines the idea of a hierarchical distributed genetic algorithm with different inter-agent partnering strategies. Cascading clusters of sub-populations are built from bottom up, with higher-level sub-populations optimising larger parts of the problem. Hence, higher-level sub-populations search a larger search space with a lower resolution whilst lower-level sub-populations search a smaller search space with a higher resolution. The effects of different partner selection schemes amongst the agents on solution quality are examined for two multiple-choice optimisation problems. It is shown that partnering strategies that exploit problem-specific knowledge are superior and can counter inappropriate (sub-) fitness measurements.

Dr Uwe Aickelin

School of Computer Science
University of Nottingham
NG8 1BB UK
uxa@cs.nott.ac.uk

The Nurse Scheduling Problem

Objective:

- To create weekly schedules on ward basis.
- To satisfy working contracts and to have fair schedules.
- To take as many nurses' requests into account as possible.

Decomposition:

1. Ensuring that nurses present can cover the overall demand.

2. Scheduling the days and/or nights a nurse works.

3. Splitting the day shifts into early and late shifts.

Typical Dimensions of Data:

30 nurses, 3 grade bands, 7 part time options, 411 different shift patterns, varying demand levels.

The Nurse Model

$$x_{ij} = \begin{cases} 1 \text{ nurse } i \text{ works pattern } j \\ 0 \text{ else} \end{cases}$$

$$a_{jk} = \begin{cases} 1 \text{ pattem } j \text{ covers day } k \\ 0 \text{ else} \end{cases}$$

$$q_{is} = \begin{cases} 1 \text{ nurse } i \text{ is of grade } s \text{ or higher} \\ 0 \text{ else} \end{cases}$$

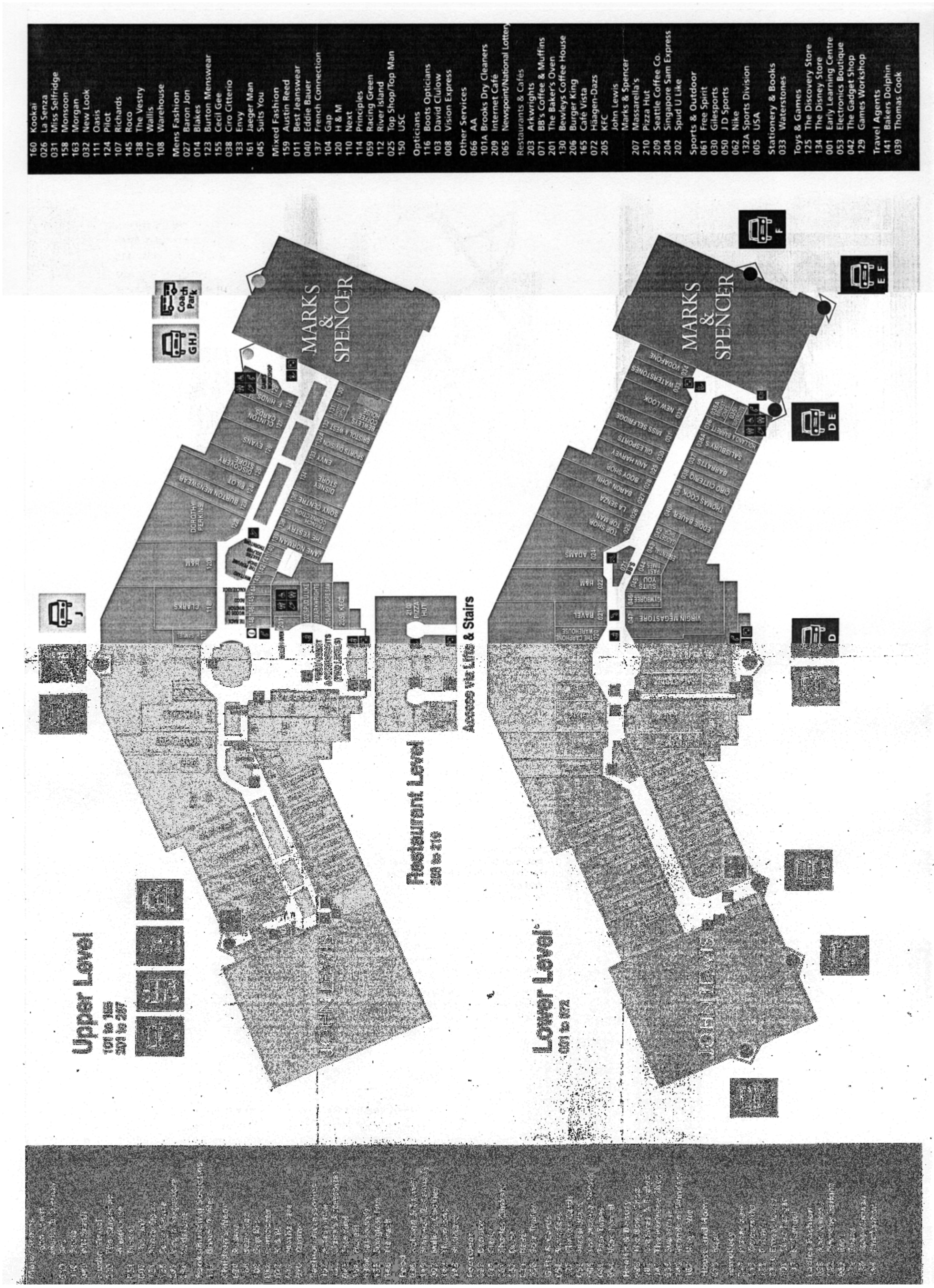
p_{ij} = penalty cost of nurse i working pattern j
R_{ks} = demand of nurses with grade s on day k
F(i) = set of feasible shift patterns for nurse i

$$\sum_{i=1}^{n} \sum_{j=1}^{m} p_{ij} x_{ij} \quad \rightarrow \quad \text{min!}$$

$$\sum_{j \in F(i)} x_{ij} = 1 \qquad \forall i$$

$$\sum_{i \in F(i)} \sum_{j=1}^{m} q_{is} a_{jk} x_{ij} \geq R_{ks} \quad \forall k, s$$

The Mall Problem



The Mall Model

Shop Income Factors:

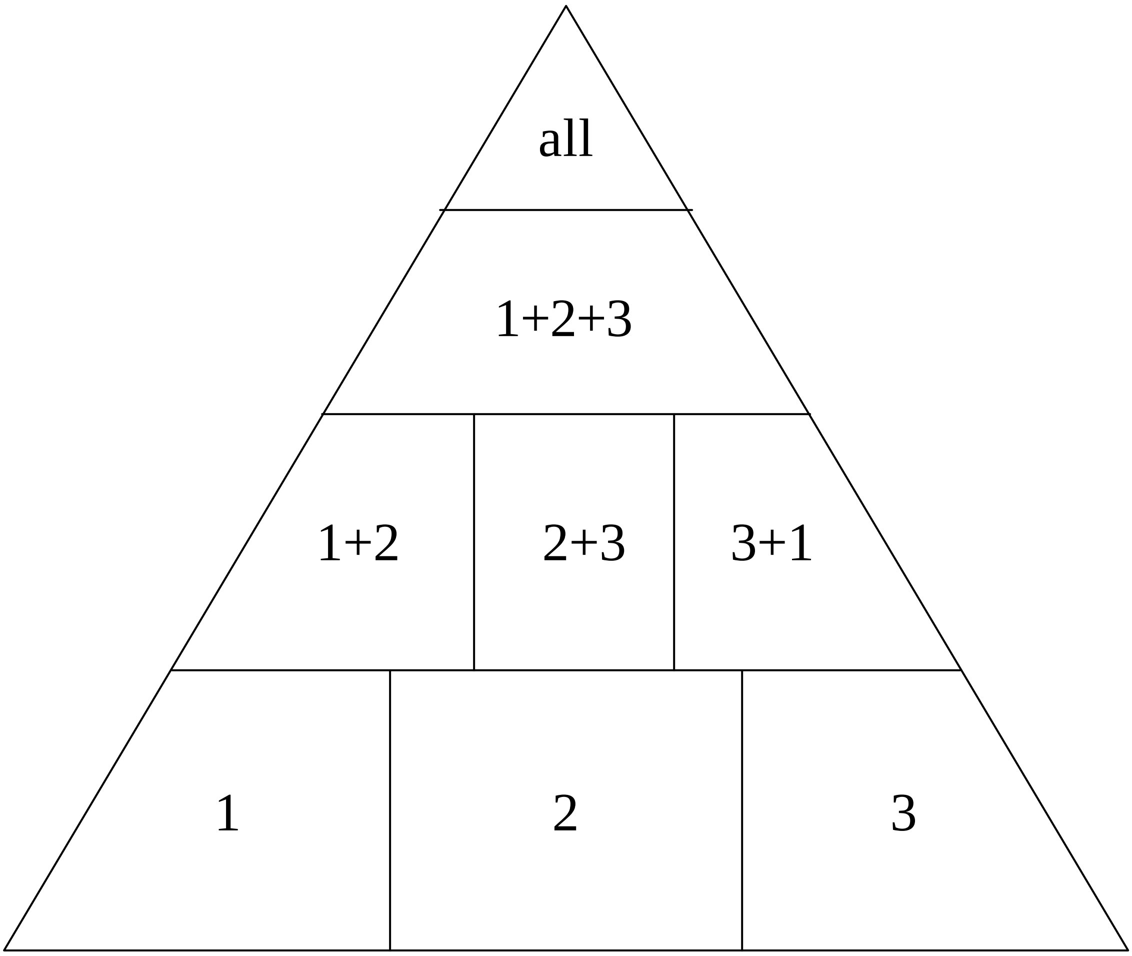
- The attractiveness of the area in which the shop is located.
- The total number of shops of the same type in the mall.
- The size of the shop.
- Synergy effects with neighbouring shops.

Constraints:

- Maximum number of shops per shop type.
- Maximum number of small / medium / large shops.
- One shop unit per location.

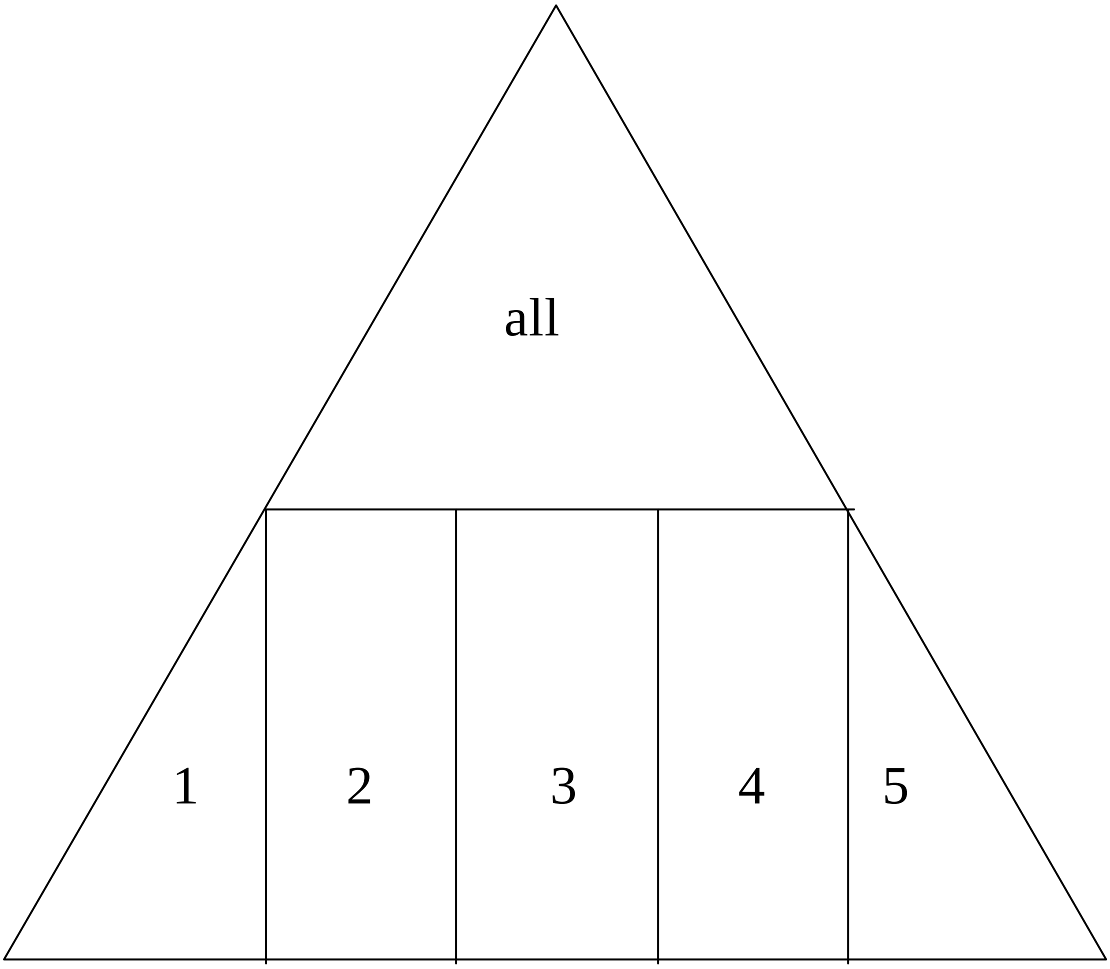
Pyramidal Structure for the Nurses

Split according to grades:



Pyramidal Structure for the Mall

Split according to areas:



Partnering Strategies

- Rank-Selection (S) based on sub-fitness score
- Random (R)
- Best (B) based on sub-fitness
- Distributed (D) on a toroidal grid
- Joined (J)
- Attractiveness (A): rank-based & probabilistic depending on created fitness
- Partner Choice (C): Select Best Partner out of 10.

Nurse Scheduling Results

Basic GA:

- Some instances solved satisfactorily but many infeasible solutions.

Pyramidal GA:

- Marked improvement in performance but not yet as good as other methods.

Partnering Strategies:

- The more important the sub-fitness scores, the better they worked: R & D did poorly, A & C best

Mall Problem Results

Basic GA:

- Good results close to theoretic bounds.

Pyramidal GA:

- Far poorer results than with standard GA.

Partnering Strategies:

- All apart from B improve results.
- A & C better than standard GA.

Conclusions

- Partnering strategies improve results (crossover-hillclimber).
- Local search hillclimber still required.
- If sub-fitness measure is good, selection-based methods work well.
- If sub-fitness measure is poor then random works as well as others.
- Try Partnering strategies for obtaining sub-fitness scores?