What can Constraint Programming contribute to an existing Optimisation application?

Patrik Eveborn, Jonas Fager

OptimalSolutions

Outline

- Optimal Solutions AB
- Home Care in Sweden
- Laps Care
- The optimisation engine in Laps
- What can CP contribute to Laps

OptimalSolutions

Field Service

Human Logistics

Home Care

Networks

Logistics

Optimization

Components

Financials

Production

Science

Areas

OptimalSolutions

Application sectors

Home Care in Sweden

- By law, the local authorities have to provide visiting services to allow older people to continue living independently at home
- Wide range of services, from cleaning to medical care
- Sector employs 80,000 people, about 2% of Sweden’s total workforce
- Fast growing sector due to ageing population

OptimalSolutions

Visit attributes

- Type of visit:
  - Cleaning, washing, medical,….
- Time windows
  - 45 minute visit between 8.30-9.45
  - 1.5 hour visit between 13.00-15.00
- Skills
  - Medical, language, gender
- Preferences
  - Same staff member
- Geographical location
  - Own house, apartment or retirement home

OptimalSolutions

Staff member attributes

- Skills
- Preferred geographical areas
- Working hours
- Target workload
- Planned breaks (eg lunch)
- Home base
- Travel times between visits (by car, bike or foot)

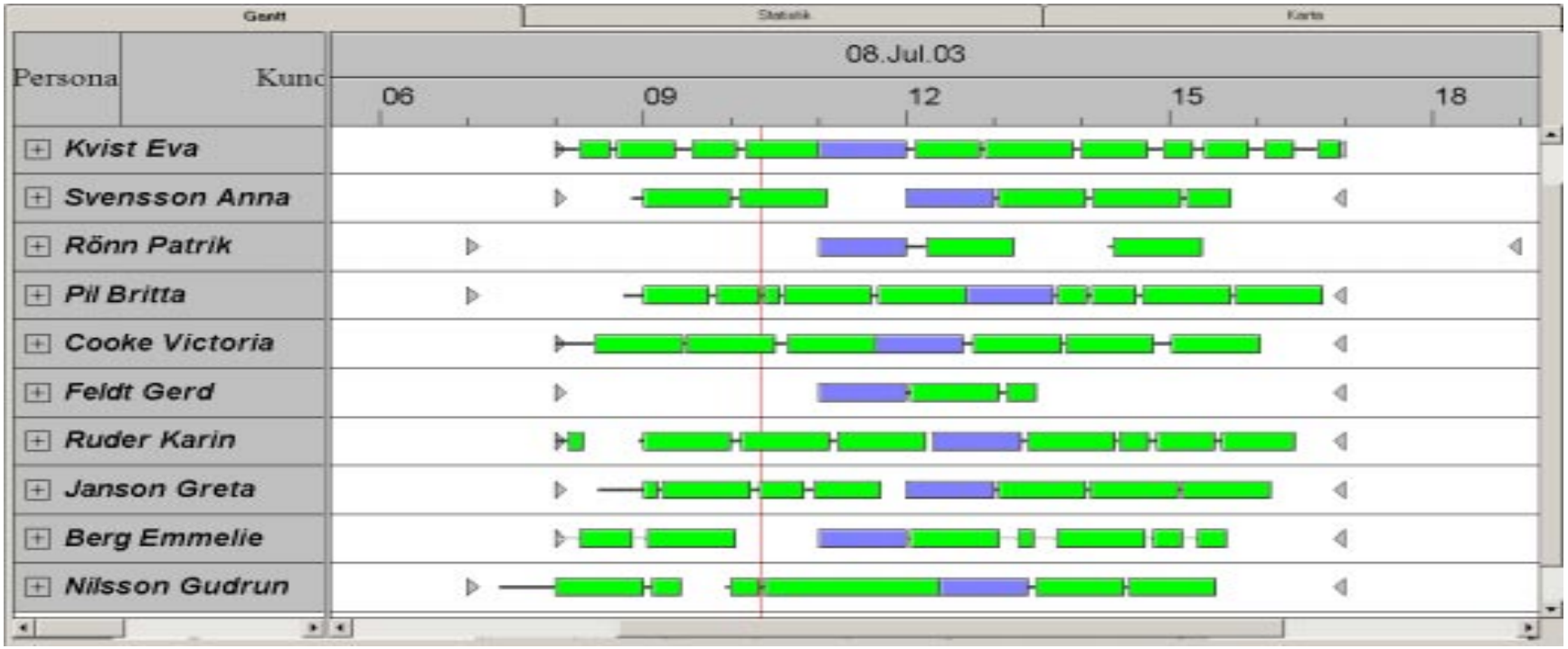
OptimalSolutions

Aims

- Allocate maximum number of visits to staff
- Maximize Customer Quality
  - measured through continuity (short / long term)
- Maximize Staff Quality
  - measured through preferences for areas, even workload etc.
- Minimize travel time

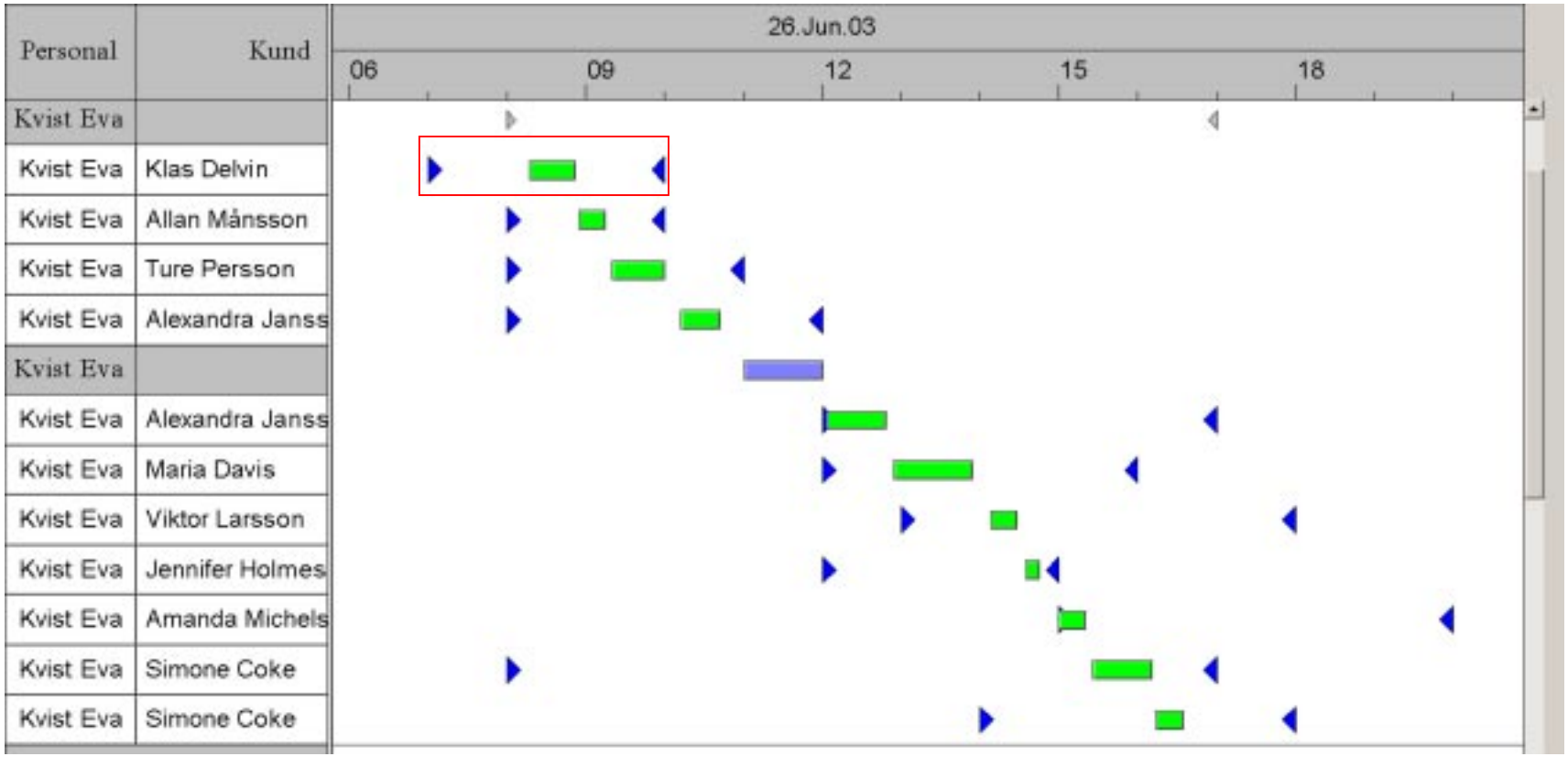
OptimalSolutions

Gantt-chart of visits and staff



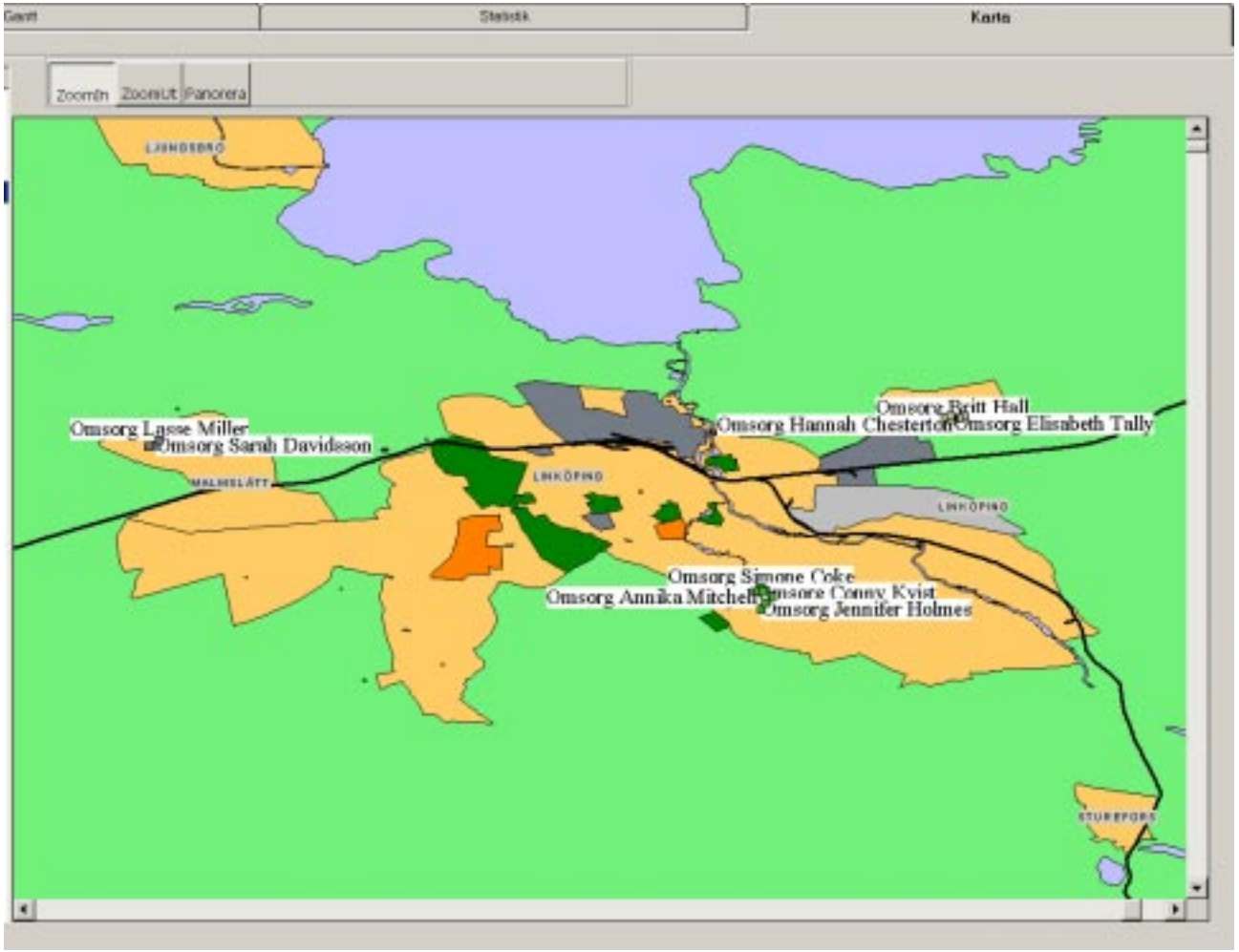
OptimalSolutions

Detailed plan showing time windows



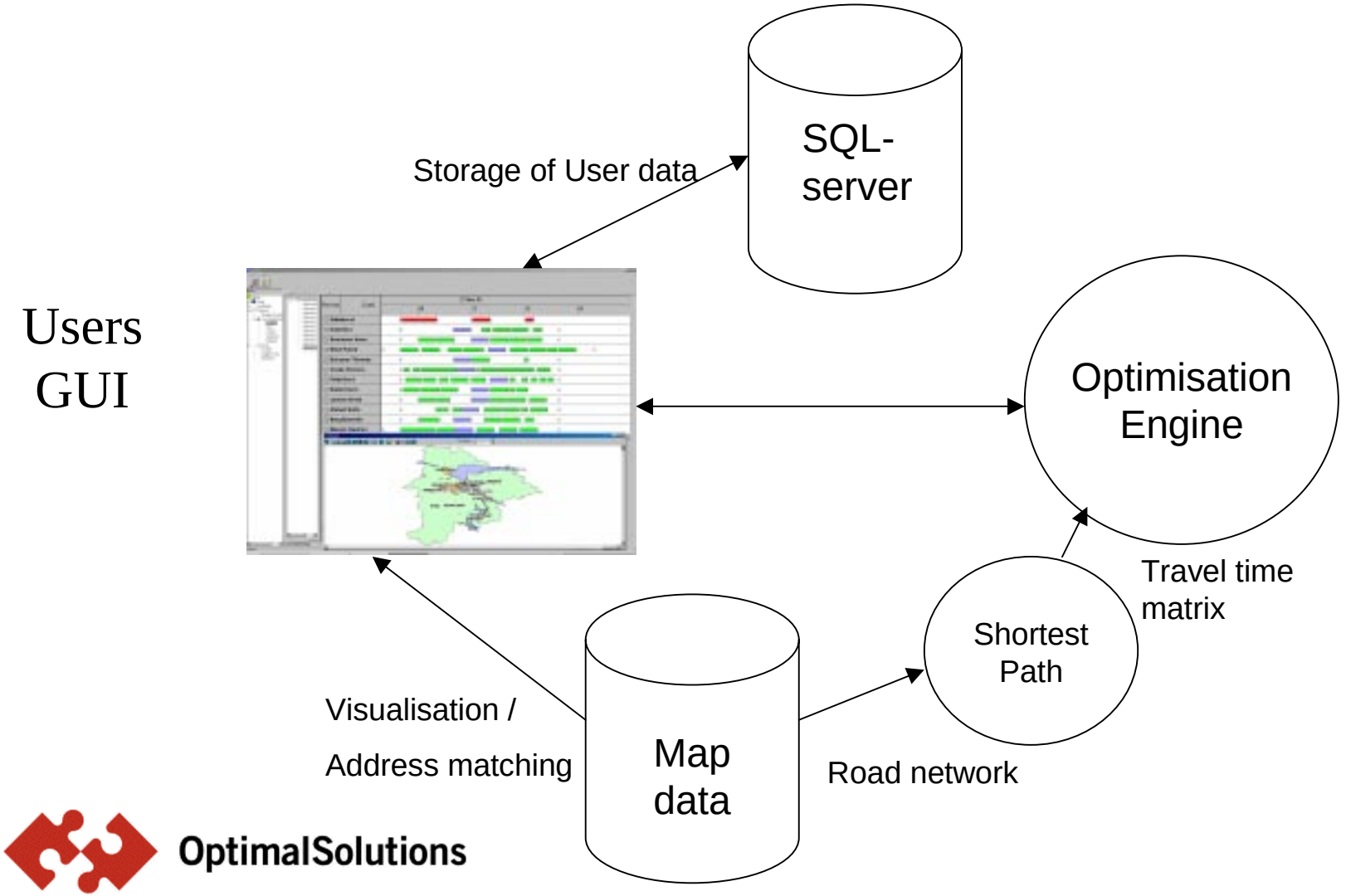
OptimalSolutions

Map View



OptimalSolutions

Laps Care – System overview



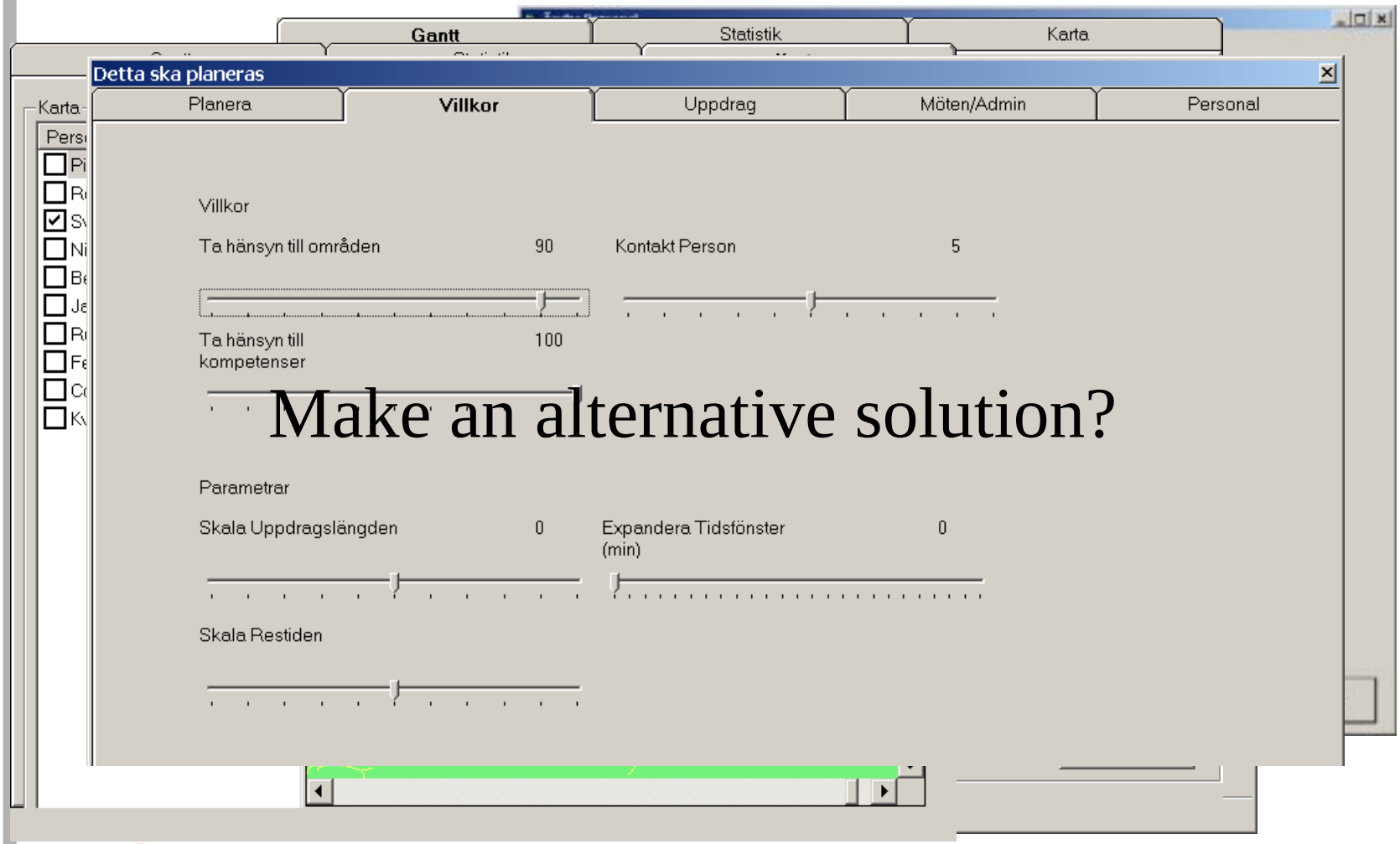
OptimalSolutions

Estimating travel times

A complete travel time matrix is required for all pairs of customers for each transportation mode

OptimalSolutions

Planning in Laps Care



Make an alternative solution?

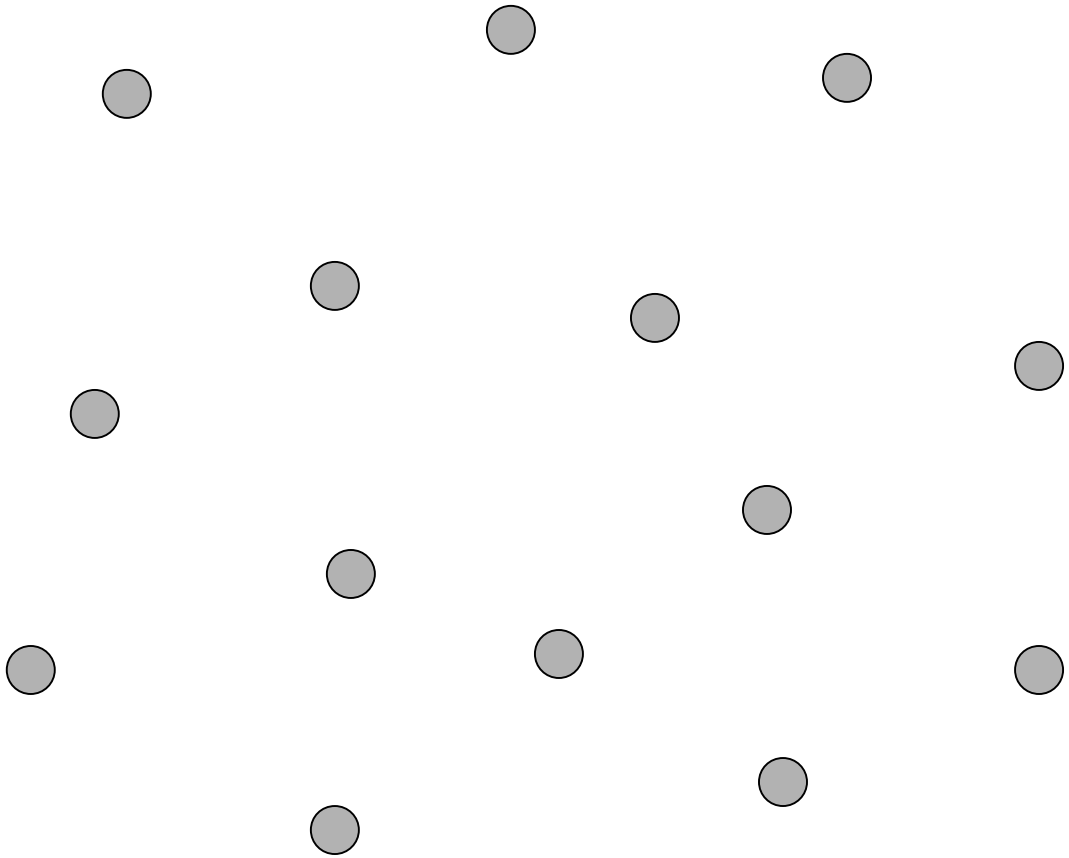
OptimalSolutions

Solution approach

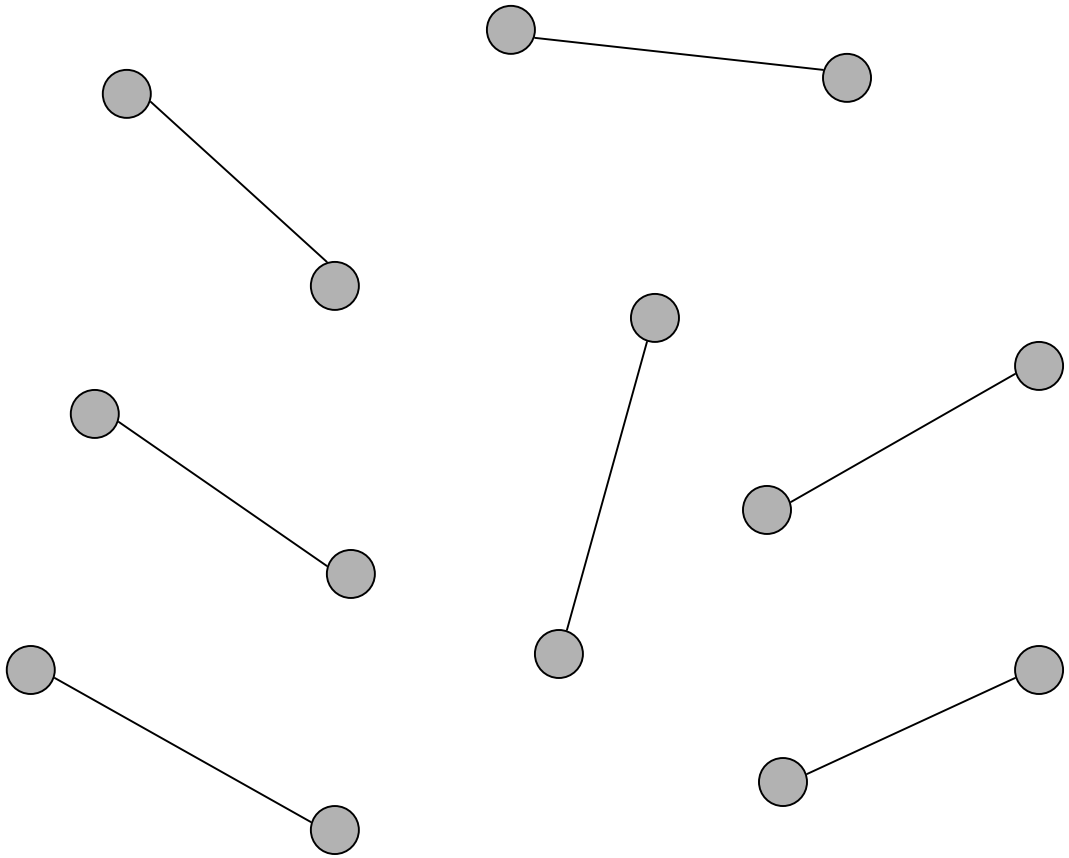
- Quick response times required by the clients
- Experience with Branch & Price
  - Too slow
- Repeated Matching
  - Vehicle routing problems
  - Facility location problems
  - Airline scheduling

OptimalSolutions

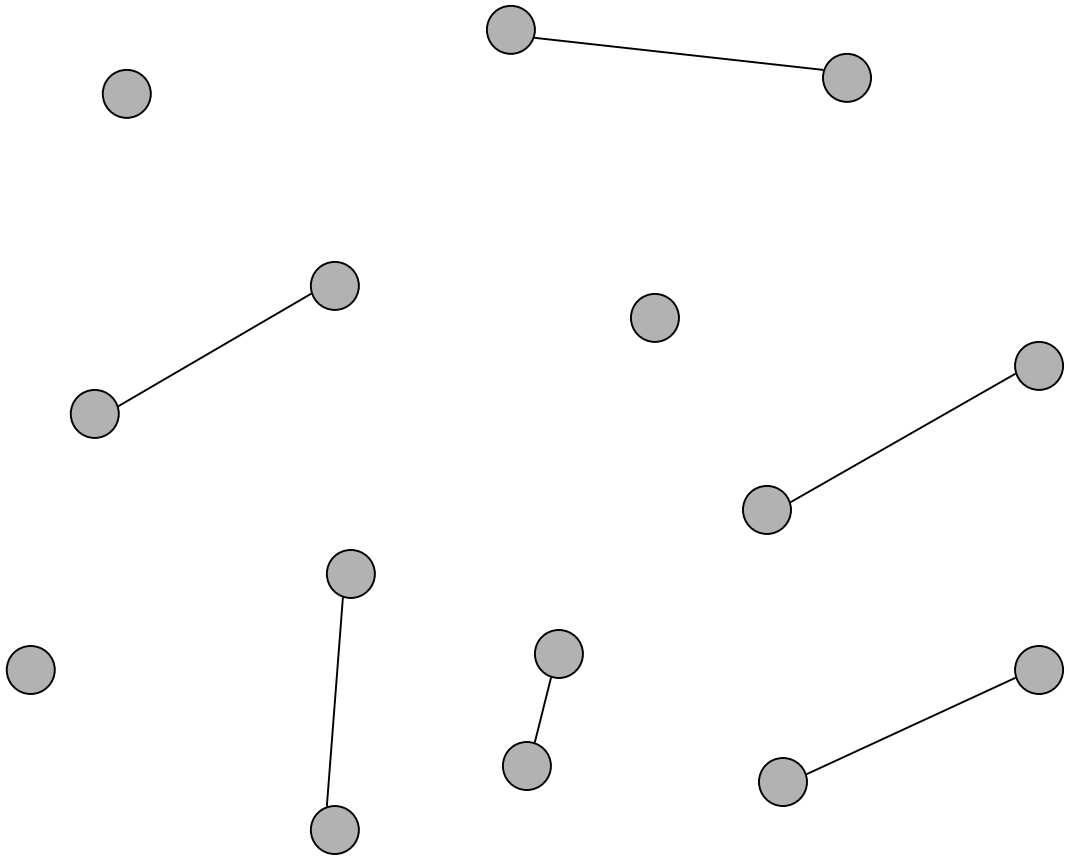
Given a set of objects



Perfect matching



Matching with self matching



Matching problem

$$z_{ij} = \begin{cases} 1, \text{if object } a_i \text{ is matched with object } a_j \\ 0, \text{otherwise} \end{cases}$$

$$\min w = \sum_{i \in I} \sum_{j \in J_i} d_{ij} z_{ij}$$

$$st \quad \sum_{j \in I} z_{ij} = 1, \quad i \in I$$

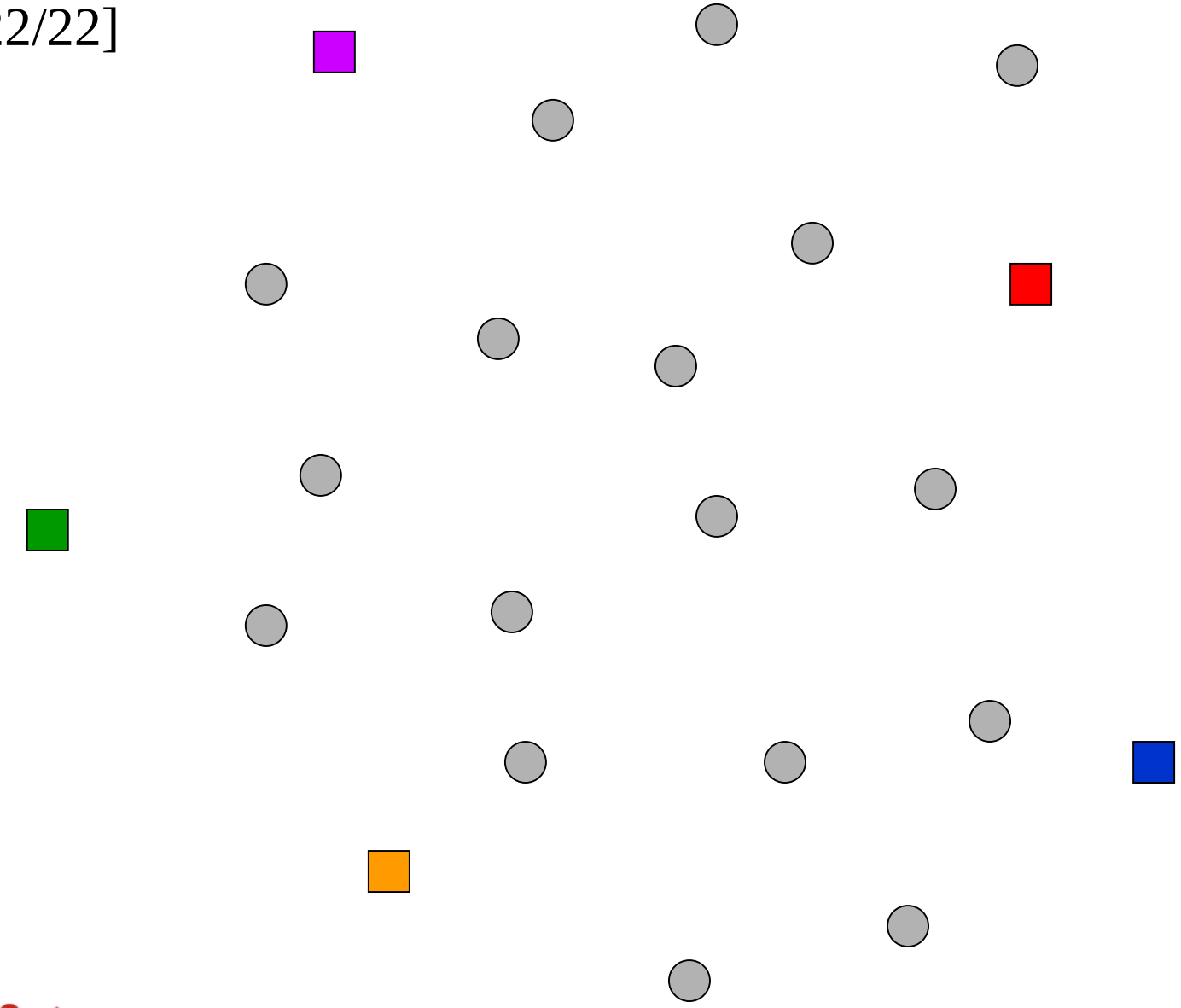
$$\sum_{i \in I} z_{ij} = 1, \quad j \in I$$

$$z_{ij} = z_{ji}, \quad i,j \in I$$

$$z_{ij} \in \{0,1\}, \quad i, j \in I$$

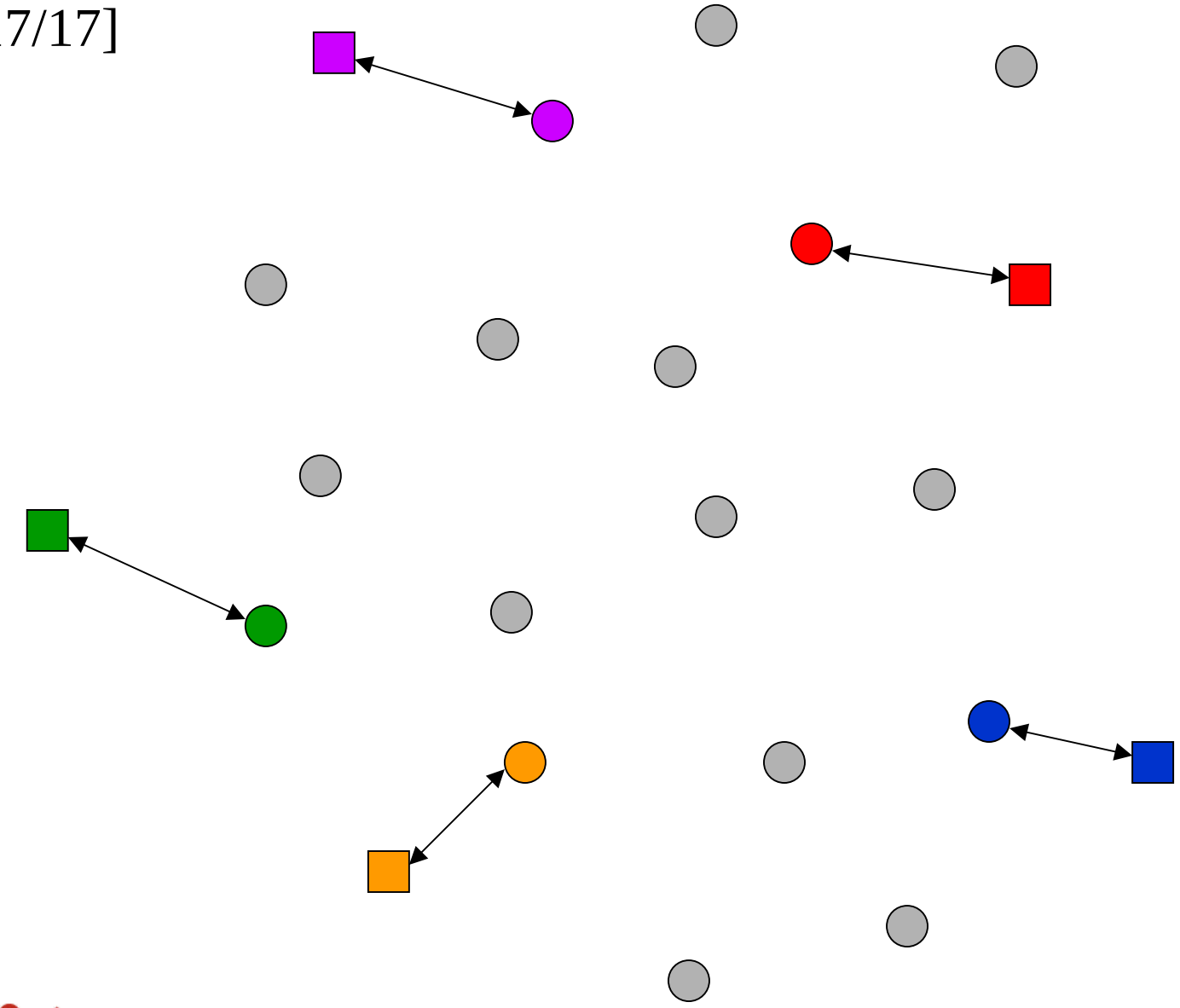
OptimalSolutions

D:[22/22]



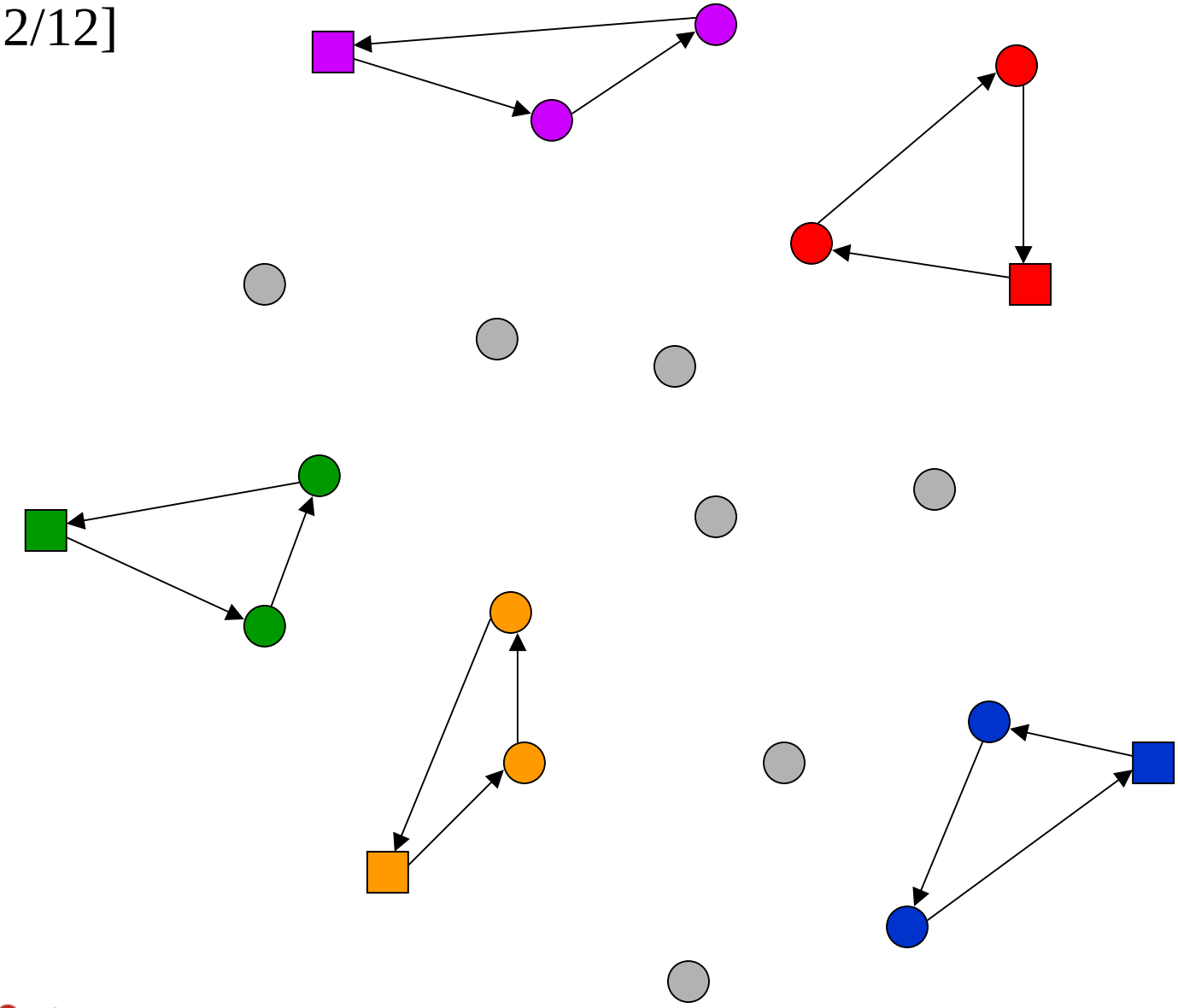
OptimalSolutions

D:[17/17]



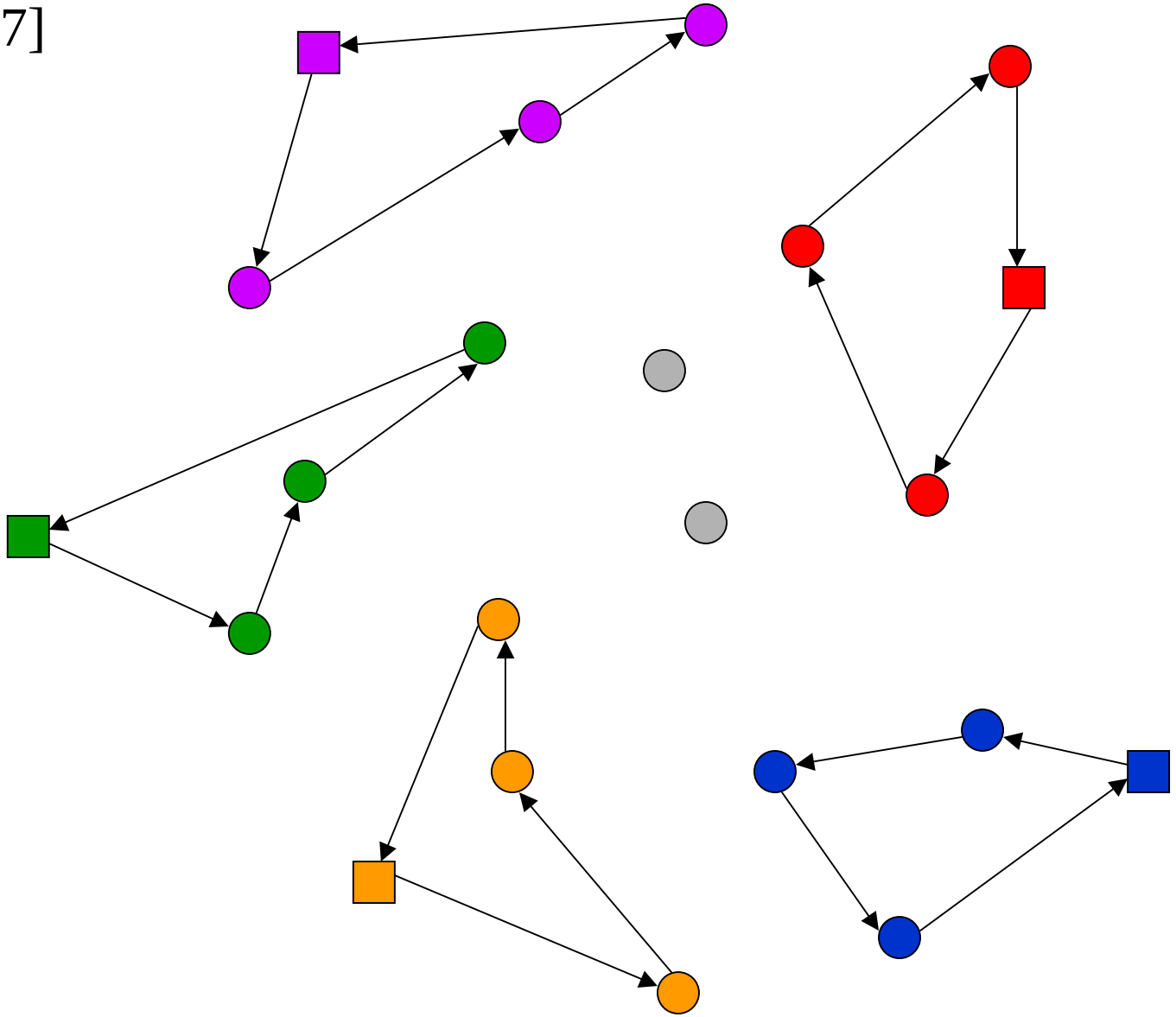
OptimalSolutions

D:[12/12]



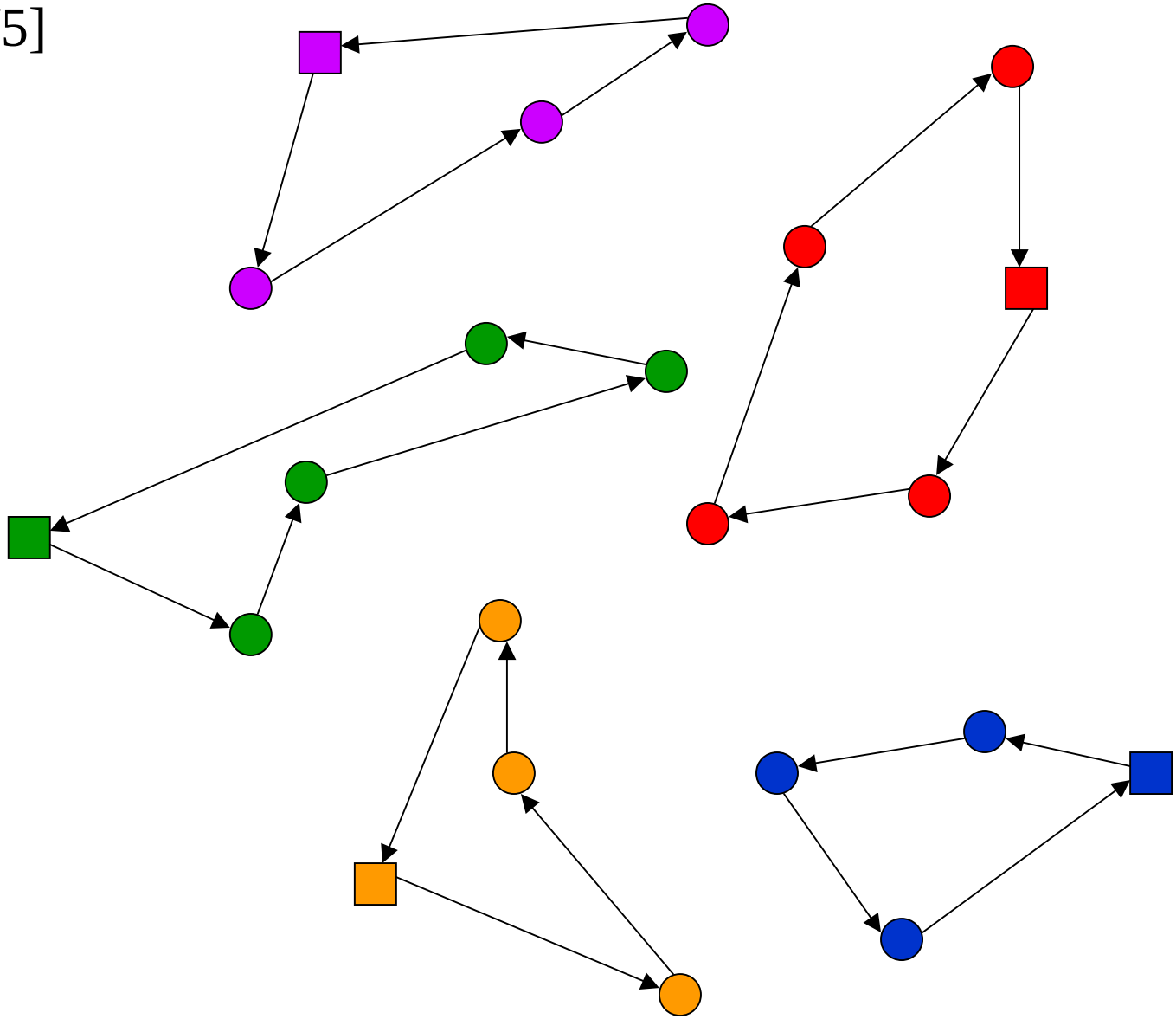
OptimalSolutions

D:[7/7]



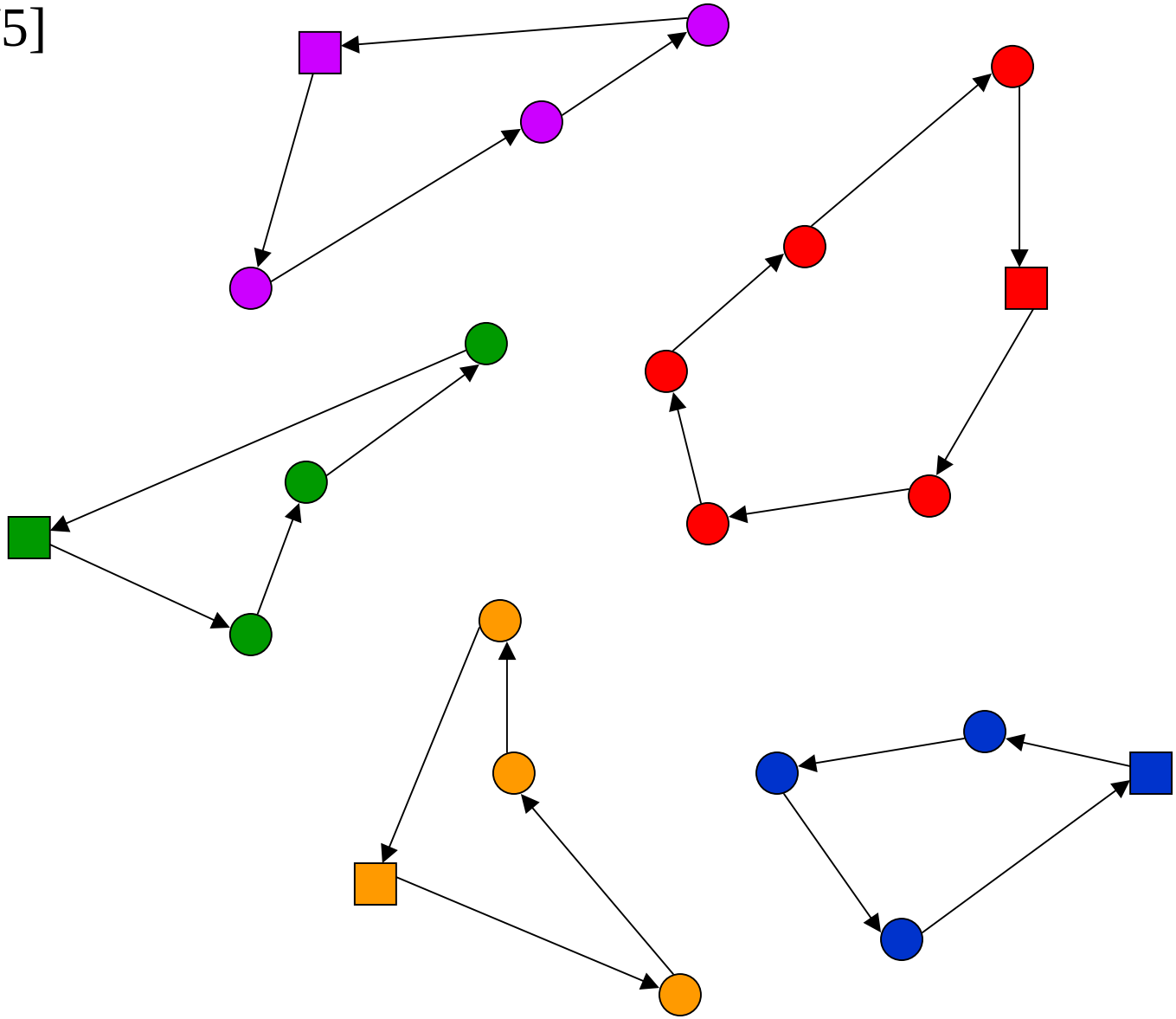
OptimalSolutions

D:[5/5]



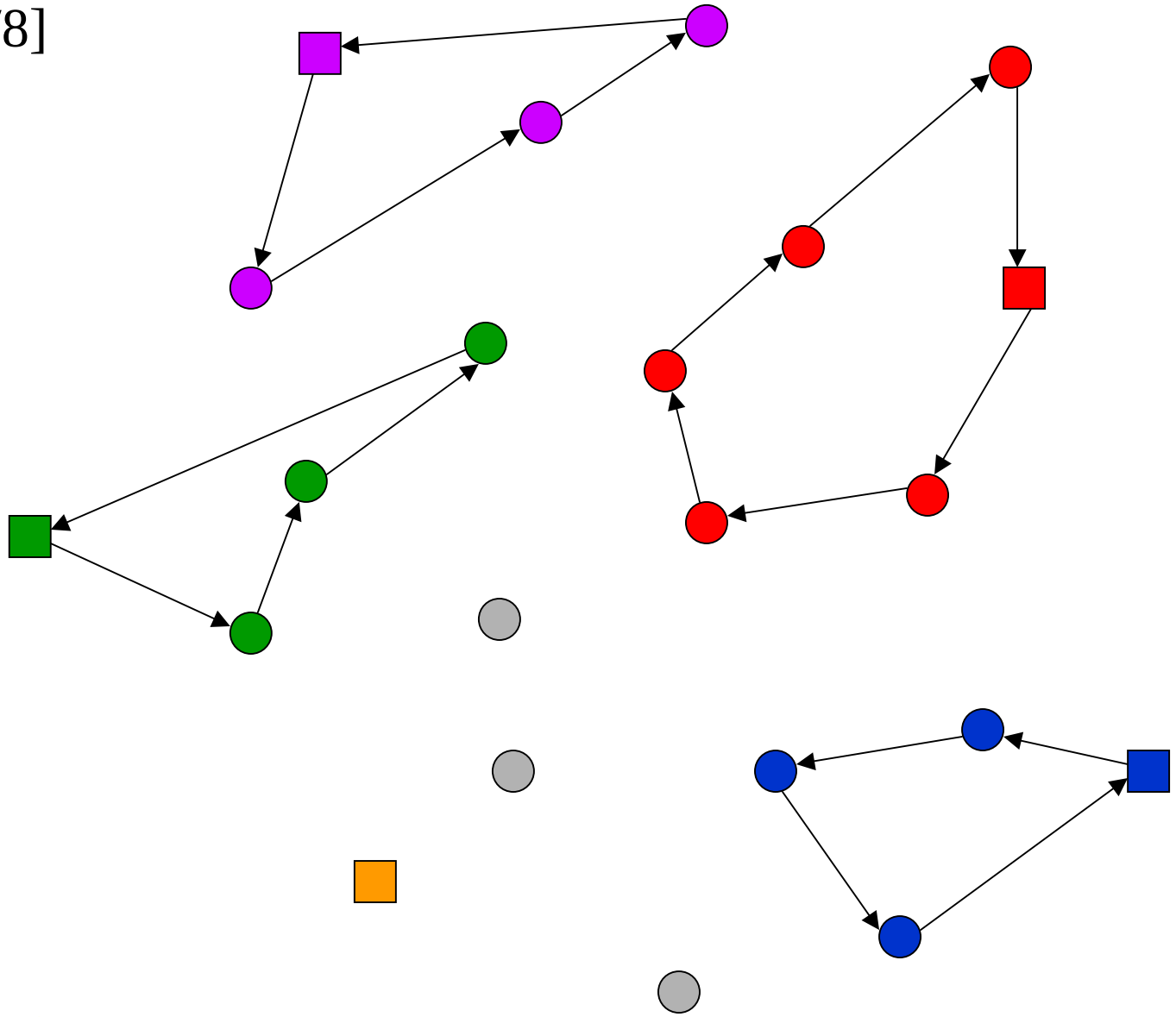
OptimalSolutions

D:[5/5]



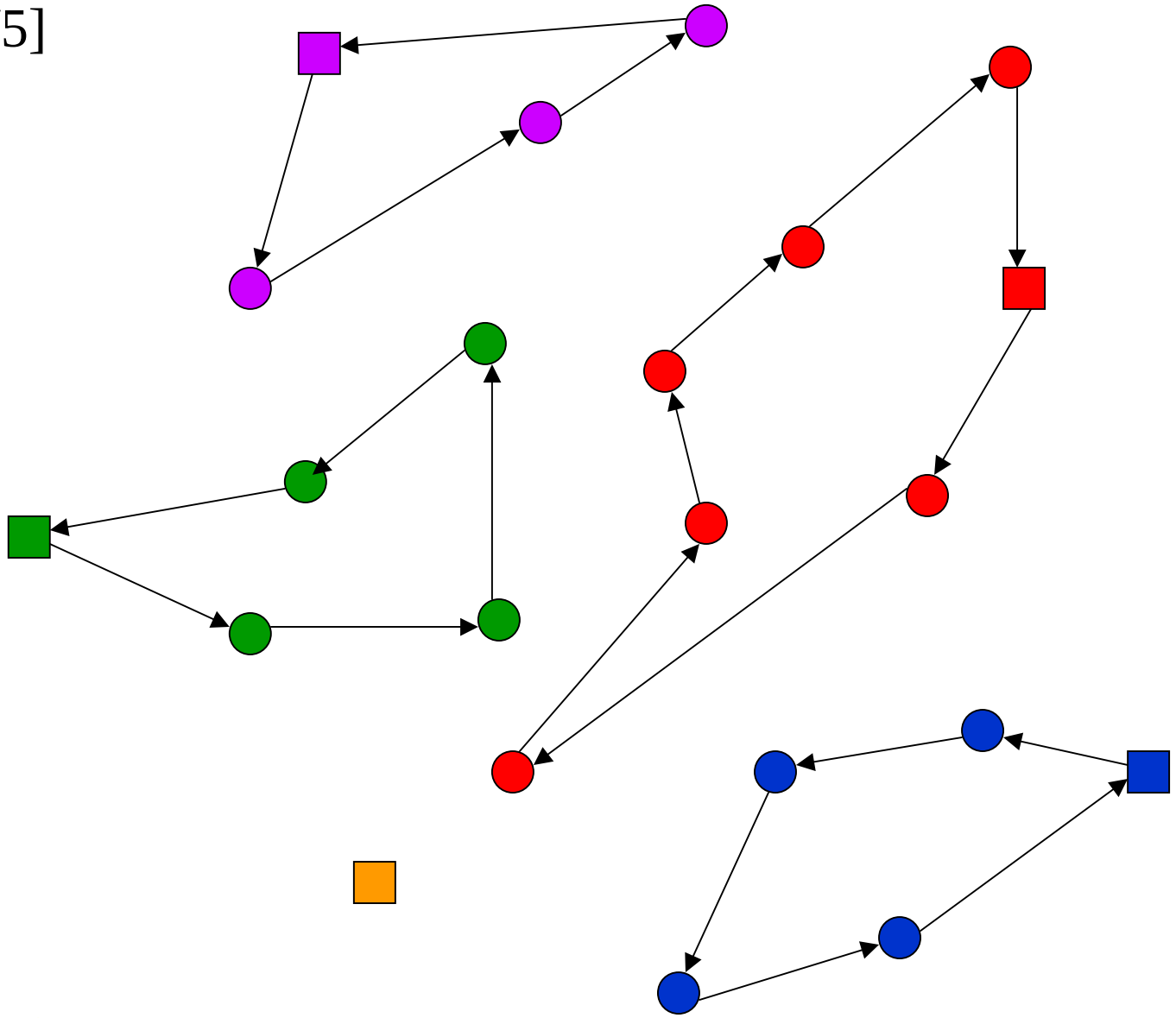
OptimalSolutions

D:[8/8]



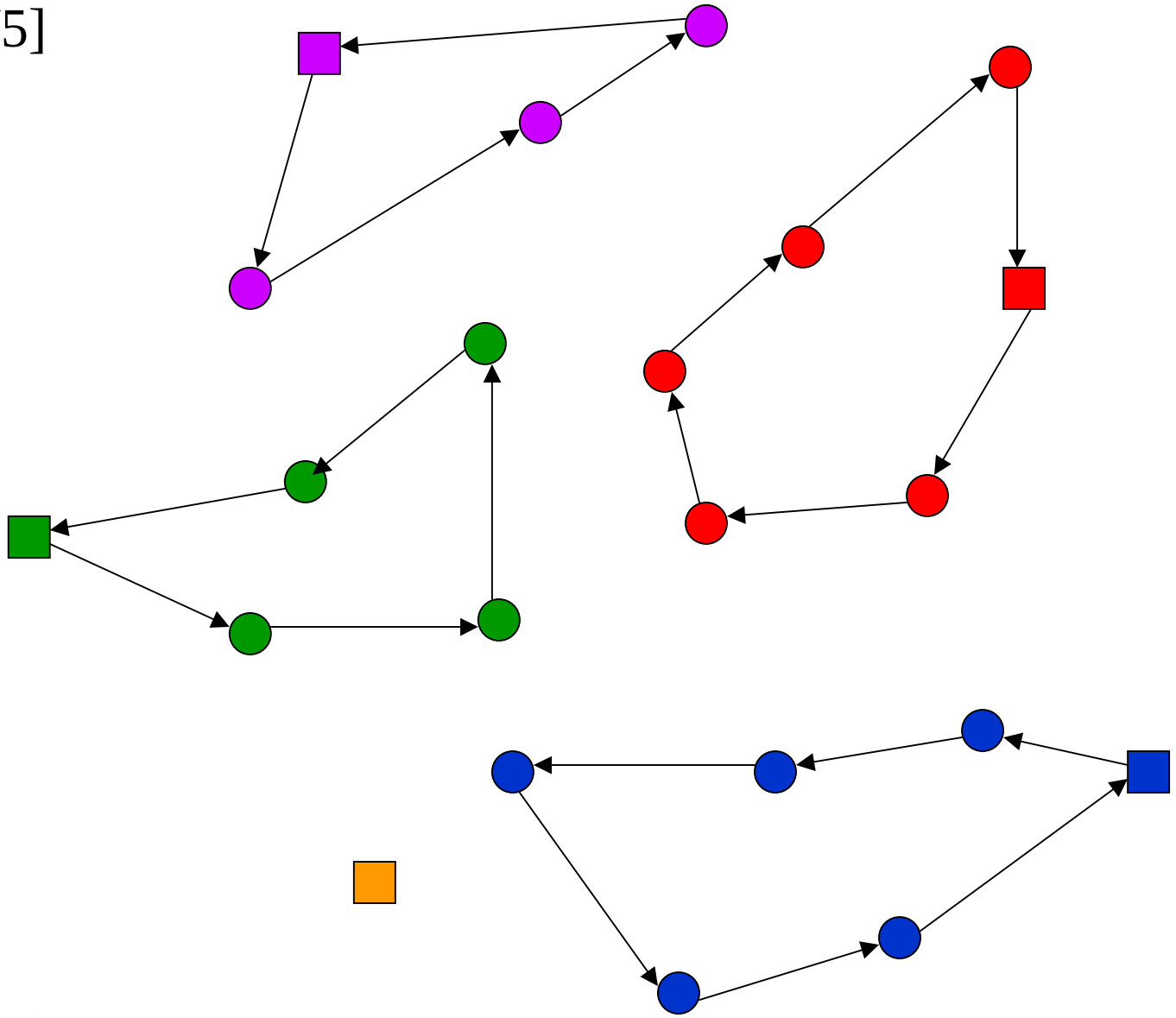
OptimalSolutions

D:[5/5]



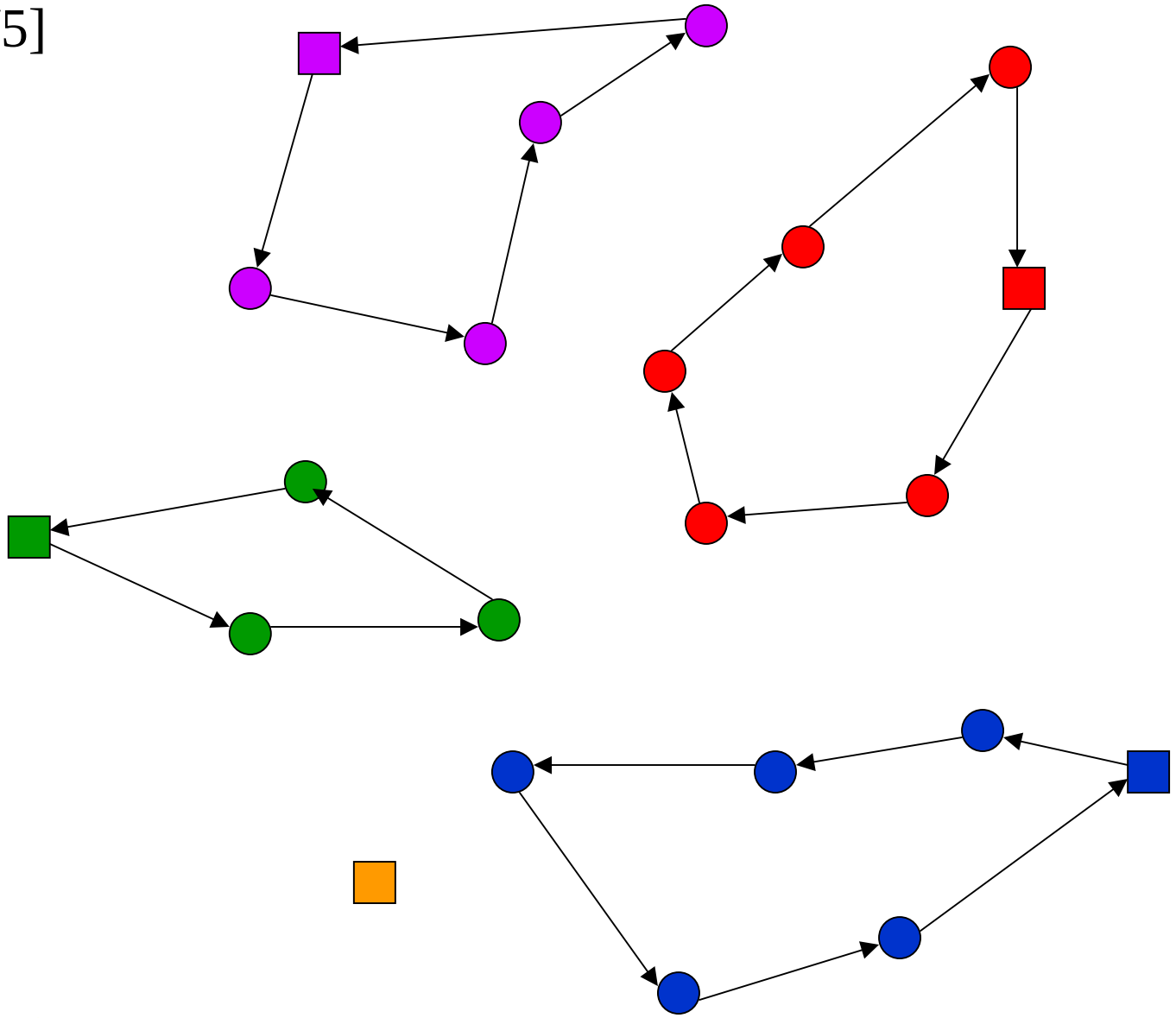
OptimalSolutions

D:[5/5]



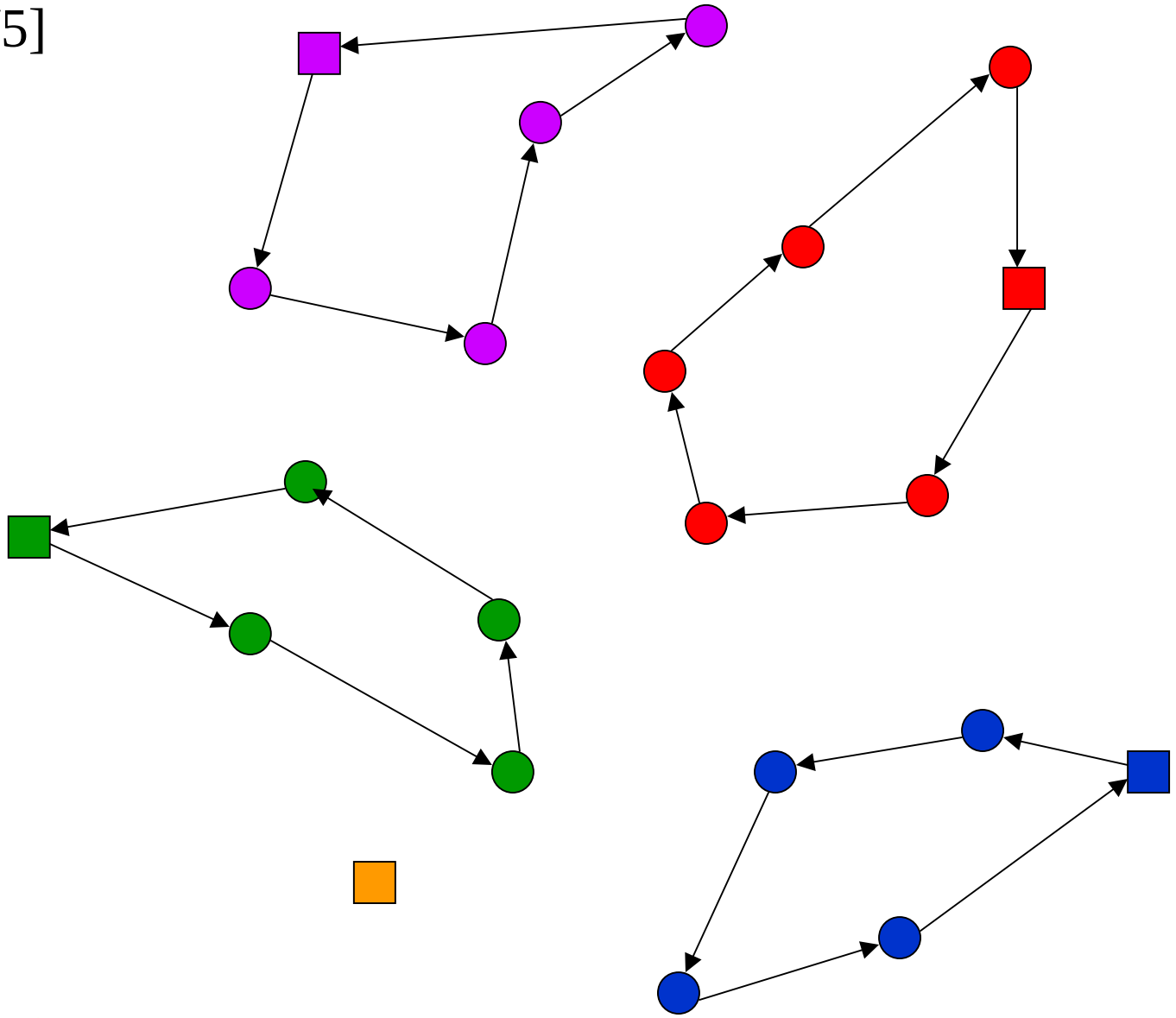
OptimalSolutions

D:[5/5]



OptimalSolutions

D:[5/5]



OptimalSolutions

Experiences / Results

- Quick solution times (within a couple of minutes)
- Large savings in operational planning time
  - Instead of all staff staying 30-45 minutes; only a few minutes are required → 7% saving of total time
- Travel times
  - Savings are in the order of 20%
- Quality
  - Better combination of customer and staff member preferences
- Decreased sick leave

OptimalSolutions

Modelling

CP provide a different way of describing the world into a model.

OptimalSolutions

Some features of commercial OR development

- Limited resources
- Often incomplete problem description at start
- Impossible to know all constraints before you're done

OptimalSolutions

Reality is the murder of a beautiful theory by hard facts

In practice it is easy to use CP to express different user demands of what is a *good* solution.

For a user it is often easier to reason in terms of CP than a pure mathematical model.

OptimalSolutions

What are our goals with using CP?

- Quick modeling
- Quick prototyping
- Cheap

OptimalSolutions

Mozart-Oz to the rescue.

- Ideas can easily be tested.
- The ideas can be compared.
- Small fragments can be implemented in C/C++/VB.

OptimalSolutions

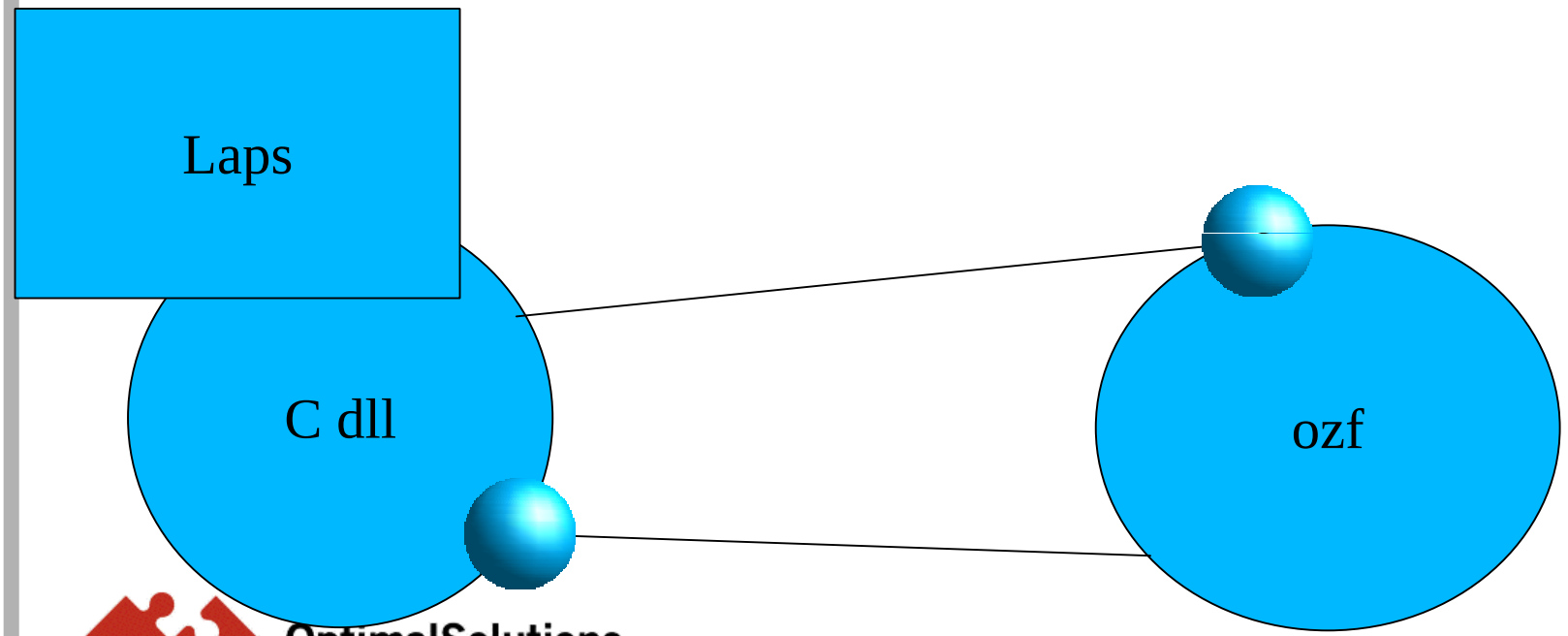
Not only CSP

- GUI is one of the most important areas.
- Guided help for manual changes. Hard to visualize a solution to a complex problem.
- Need an integrated solver.

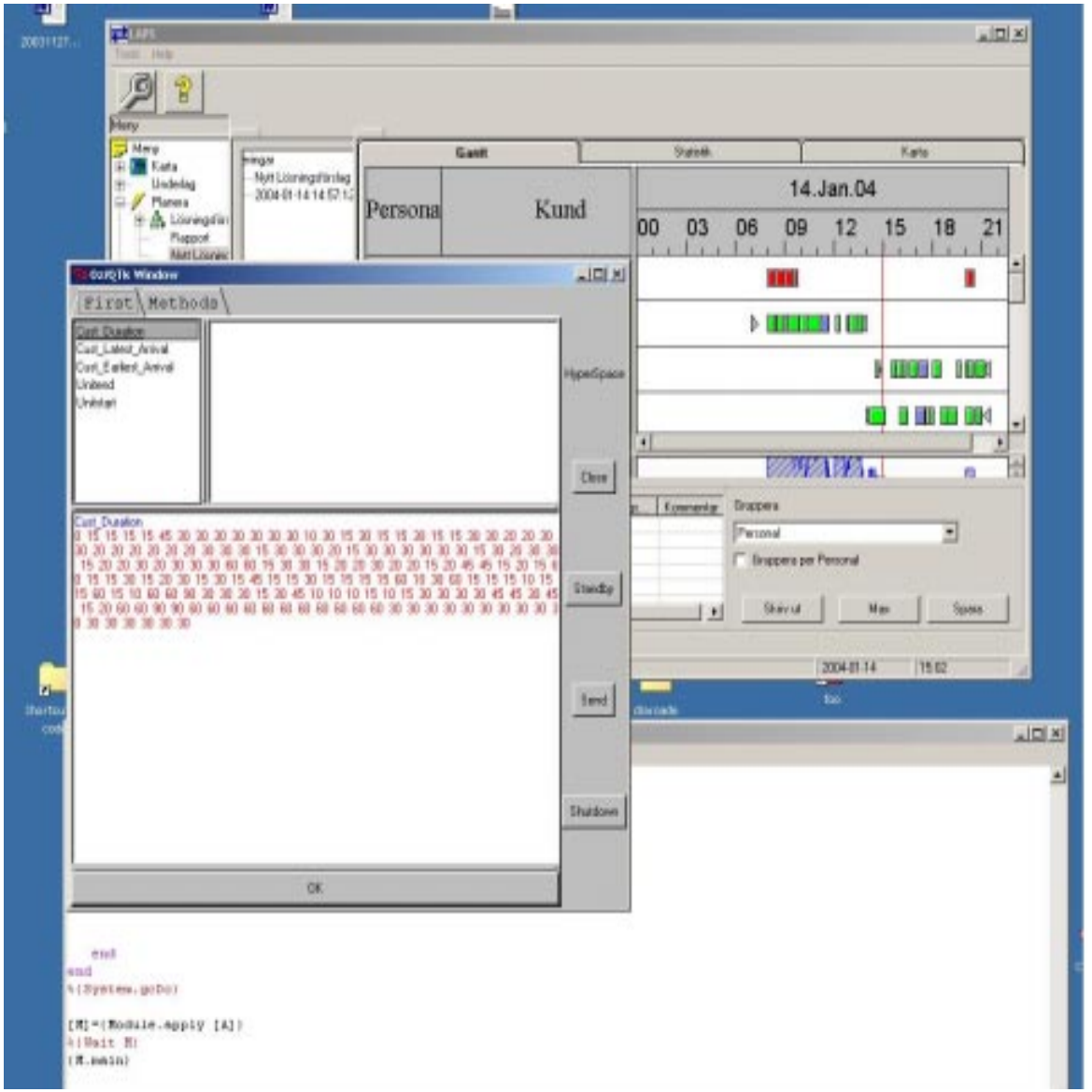
OptimalSolutions

Max Guevara

- Prototype for prototyping.
- Focused for win32.



OptimalSolutions



OptimalSolutions

Matching Problem, Again

- {Record.forAllInd MatchWith proc {$ SI S}
  - {Record.forAllInd MatchWith proc {$ QI Q}
    - S =: QI = Q =: SI
  - end}
  - {FD.element S MatchCost.SI Cost.SI}
  - end }
- TotCost = {FD.sum Cost '=:'}
- {FD.distinct MatchWith}

- {Record.forAllInd Pairs proc{$ I O}
  - {FD.element MatchWith.I WindowL TL.I}
  - {FD.element MatchWith.I WindowU TU.I}
  - T.I >=: TL.I
  - T.I =<: TU.I
  - T.I =: T.O
  - end }

- Root = sol(object:MatchWith taskstart:T cost:Cost totcost:TotCost)
- {FD.distribute ff MatchWith}

OptimalSolutions