Scene Interpretation as Configuration

What is a configuration problem?
Construct an aggregate (a configuration) given
- generic descriptions of parts
- compatibility constraints between parts
- a concrete task description

Scene interpretation may be viewed as constructing a "scene aggregate" which
• meets generic constraints and
• incorporates parts prescribed by the concrete task

Both, scene interpretation and configuration, can be described as logical model construction.

Methods and tools of configuration technology may be exploited
Illustration of Configuration

- boxes (frames) specify aggregate and component properties
- has-part relations bind components to aggregates
- is-a relations describe variants of entities
- constraints between entities (not shown) restrict choices and parameter combinations

A Real Configuration Task

Placement of cabin equipment (seats, kitchens, toilets, etc.) in view of
- customer wishes
- technical constraints
- legal constraints
- optimality criteria
Example of Concept Definition in KONWERK

KONWERK is a configuration system prototype developed at the AI Lab (LKI) of Hamburg University in 1986 - 1994. The commercial system Engcon has been developed based on KONWERK.

Concept "galley" describes service station in Airbus A340

```lisp
(defun def-concept
  :name galley
  :super-concept \{cabin-interior-component rectangle\}
  :parameters
    ref-nr [integer 2531000 2533999]
    door \{1 2 4\}
    trolleys \{0 2 3 4 5 6 7 8 9 10\}
    half-size-trolleys \{0 1 2 3 4 5\}
    meals [integer 28 140]
    type \{longitudinal transversal\}
    height \{full half\} (default 'full)
  :relations
    part-of \{passenger-class\}
```

Object Descriptors

Object descriptors define object classes (concepts) by specifying possible instances (compare with concept expressions in a DL).

Specific values: red, 35t, car37
Choice sets: \{red yellow green black blue\}
Intervals: \[10km/h 300km/h\]
Predicates: (:satisfies evenp)
Concepts: (a car)
  (a chassis (axle_load \[10t 40t\]))
Atomic concepts: (a symbol (self \{red yellow green black blue\}))
  (a number (self \[0 inf\]))
Logical operators: (:and \[50 100\] (:satisfies evenp))
Set Descriptors

Set: (:set (a motor) (a body) (a chassis))
short form for
(:set #[(a car_part) 3 3] #[(a motor) 1 1] #[(a body) 1 1] #[(a chassis) 1 1])
Subset: (:some (a motor) 2 4) ≈ #[(a motor) 2 4]

Decomposition Relations

(is (a car)
  (an object
    (has-parts (:set (a motor)
               (a body)
               (a chassis))))))

(is (a car)
  (an object
    (has-motor (a motor))
    (has-body (a body))
    (has-chassis (a chassis))))
Example: Concept for Building Recognition

Concept definition for the aggregate "wall"

```clojure
(def-do
  :name Wall
  :oberkonzept Scene-Part
  :relationen ((has-elements (:spezialisierte-menge
                              (:einige (ein Image-Object) :min 0 :max inf)
                              :spezialisierungstyp :=
                              :spezialisierung
                              #{[ein Balcony] 0 inf}
                              #{[ein Window] 0 inf}
                              #{[ein Gate] 0 inf}
                              #{[ein Entrance] 0 inf}))
  (element-of
   (:spezialisierte-menge
    (:einige (ein Scene-Part) :min 1 :max 1)
    :spezialisierungstyp >:
    :spezialisierung
    #{[ein Building] 0 1}
    #{[ein Entrance] 0 1}
    #{[ein Balcony] 0 1])))
```

Representation Language of KONWERK

Language constructs can be mapped to logical constructs of a description logic by using:

- Conjunction
- Negation and disjunction with atomic concepts
- Value restrictions
- Qualifying number restrictions
- Inverse roles
- Sets
- Concrete domains over R
Constraint-based Configuration

- Constraints represent relations between parameters or concepts
- Constraints are multi-directional
- Generating a constraint network (system of equations)
- Consistency check for value settings
- Restricting value ranges by propagation
- Computing all solutions by using constraint-satisfaction technologies
- Incrementally increasing constraint net

Constraint Propagation (1)

KONWERK supports constraint propagation for local consistency
KONWERK also supports constraint propagation for global consistency

**Constraint Propagation (2)**

Constraints in KONWERK

- **Constraint classes**
  Predominantly domain-independent computational procedures
  *Examples: adder, multiplier, sum, equal*

- **Conceptual constraints**
  Description of a domain-specific constraint type, instantiation rules
  *Example: motor displacement = sum of cylinder displacements*

- **Constraint instances**
  Dynamically generated at configuration time

- **Constraint net**
  Propagates values through all constraint instances, recognizes conflicts
Example: Constraint for Building Recognition

Constraint ensures that the bounding-boxes of parts of a wall are contained in the bounding-box of a wall.

(def-konzeptuelles-constraint
 :name All-Parts-of-Wall-are-in-Wall
 :variablen-pattern-paare (
   (?w :name Wall)
   (?p :name Image-Object
    :relationen ((element-of
                   ((globale-bedingung
                     '(is-in-set ?w *it*)))))
    :constraint-aufrufe ((bb-inside-fuzzy (?w 'Pos-X-1) (?p 'Pos-X-1)
                          (?w 'Pos-Y-1) (?p 'Pos-Y-1)
                          (?w 'Pos-X-2) (?p 'Pos-X-2)
                          (?w 'Pos-Y-2) (?p 'Pos-Y-2)))))

Central Configuration Cycle

Repeat
Check for goal completion
Determine current strategy
Determine possible configuration steps
Select from agenda and execute one of
  { aggregate instantiation,
   aggregate expansion,
   instance specialisation,
   parameterisation,
   instance merging }
Propagate constraints
Check for conflict

For value selection one can use
- default value assignment
- computational procedure
- user interaction
- library solution
- local breadth-first search
In case of a conflict, backtracking occurs. One may select one of 3 backtracking strategies:

- **Chronological backtracking**
- "Intelligent" backtracking
- Backtracking with preservation of non-conflicting data

The Scene Interpretation System

SCENIC
Structure of Scene-Interpretation System SCENIC

Scene Interpretation
- High-level Interpretation System
  - Primitive Symbolic Scene Description
    - Geometric Scene Description
    - Segmentation and Tracking Unit
  - Metric-Symbolic Interface

realised by configuration system KONWERK

Structure of Conceptual Knowledge Base of SCENIC

upper model

domain-specific model (not all concepts shown)
Example of Table-laying Scene

Stationary cameras observe living room scene and recognize meaningful occurrences, e.g. placing a cover onto the table.

In the following experiment: laying a table for a dinner-for-2

Bounding-Box Abstractions

- object shapes are represented as 2D boxes
- aggregates hide internal structure
- box locations and distances are interval-valued
- value ranges and their correlations may be described by joint probability distributions
### Initial Bottom-up Instantiation of Concepts

<table>
<thead>
<tr>
<th>Concept</th>
<th>Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>moving-saucer-3</td>
<td>71</td>
</tr>
<tr>
<td>moving-hand-2</td>
<td>90</td>
</tr>
<tr>
<td>moving-plate-2</td>
<td>85</td>
</tr>
<tr>
<td>moving-saucer-2</td>
<td>92</td>
</tr>
<tr>
<td>moving-hand-5</td>
<td>88</td>
</tr>
<tr>
<td>moving-touch-1 (hand-4 dish-2)</td>
<td>78</td>
</tr>
<tr>
<td>transport-1 (hand-2 plate-3)</td>
<td>46</td>
</tr>
<tr>
<td>transport-2 (hand-5 plate-2)</td>
<td>92</td>
</tr>
<tr>
<td>transport-3 (hand-4 dish-2)</td>
<td>229</td>
</tr>
<tr>
<td>transport-4 (hand-3 saucer-2)</td>
<td>268</td>
</tr>
<tr>
<td>transport-5 (hand-5)</td>
<td>270</td>
</tr>
<tr>
<td>transport-3 (hand-4 dish-2)</td>
<td>270</td>
</tr>
<tr>
<td>transport-5 (hand-5)</td>
<td>270</td>
</tr>
</tbody>
</table>

**Legend:**
- **primitive stationary concepts**
- **primitive motion concepts**
- **aggregate concepts**

### Experimental Results (1)

**Intermediate state of interpretation after 51 interpretation steps:**
- "lay-dinner-for-2" hypothesis based on partial evidence
- predictions about future actions and locations
- high-level disambiguation of low-level classification
- influence of context
Experimental Results (2)

- alternative interpretation in terms of top-down choices “dinner-for-one” and “cluttered-table” (after backtracking)