TOWARDS SOLVING ESSENCE WITH LOCAL SEARCH: A PROOF OF CONCEPT USING SETS AND MULTISETS

Saad Attieh, Christopher Jefferson, Ian Miguel, Peter Nightingale
TOWARDS SOLVING ESSENCE WITH LOCAL SEARCH: A PROOF OF CONCEPT USING SETS AND MULTISETS

Saad Attieh, Christopher Jefferson, Ian Miguel, Peter Nightingale
LOCAL SEARCH

➤ Incomplete search, focus on finding good solutions fast.

➤ Use a set of moves (heuristics, neighbourhoods) to iteratively improve on the active solution.

➤ Choice of moves is most critical for performance, hence moves are usually very problem specific.

➤ Meta heuristics are used to select from set of moves, determines whether or not the new solution should be accepted.
ATHANOR

➤ Automated local search.
➤ Deduction of high quality neighbourhoods.
➤ Operates over very high-level specification.
ARE WE THE FIRST?

➤ Oscar CBLS

➤ Propagation guided, large neighbourhood search

➤ Explanation guided, large neighbourhood search

➤ All of these solvers derive their moves or neighbourhoods from analysis of the constraints in a problem.
THE PROBLEM OF AMBIGUOUS TYPES

find $S : \text{set (size 3)} \text{ of } \text{int}(1..5)$
**OCCURRENCE**

**find** $S : \text{set (size 3) of int}(1..5)$

$$S = \{1, 3, 4\}$$

$$\sum(X[1..5]) = 3$$
find $S : \text{set (size 3) of int}(1..5)$

$S = \{1,3,4\}$
NEIGHBOURHOODS

\[ S = \{1, 3, 4\} \]

\[ S = \{1, 2, 4\} \]
CONSTRAINTS DON’T HELP MUCH

➤ alldiff([a,b,c,d,e])
➤ Is it a set,
➤ or an injective function,
➤ or part of a partition,
➤ etc.
ESSENCE: AN ABSTRACT CONSTRAINT SPECIFICATION LANGUAGE

- Distinguished by its support for variables with high level, arbitrarily nested types
- set, sequence, partition, set of sequence of tuple, multi set of partition….
- Models can be automatically refined for input to low level solvers.
  - SAT, CP, ILP
- Athanor needs no refinement, operates directly on the structures available in Essence
TWO KEY BENEFITS
EASY TO DERIVE SEMANTICALLY MEANINGFUL NEIGHBOURHOODS

➤ find S : set of int(1..5)
  ➤ Add to s
  ➤ remove from s
  ➤ exchange one element for another
➤ find m : mset (maxSize 10) of set of int(1..5)
  ➤ Select a single element of m and apply any of the above
  ➤ Exchange elements between sets in m
  ➤ Add sets to or remove sets from m
A FRAMEWORK FOR
CONSTRAINT BASED LOCAL
SEARCH USING ESSENCE

Özgür Akgün, Saad Attieh, Ian P. Gent,
Christopher Jefferson, Ian Miguel,
Peter Nightingale, András Z. Salamon,
Patrick Spracklen, James Wetter
DYNAMIC SCALING DURING SEARCH

- find t : set of set of int(1..25)
- Usually represented as 2d matrix, requires $2^{25}$ rows.
- Optimal value of set might be very small in comparison.
- Athanor understands that sets have a variable size and will dynamically allocate or deallocate memory accordingly.
- Athanor will also dynamically add and remove constraints as new elements are added or removed.
SONET, MULTISET OF SET

![Graph showing time vs. objective with different algorithms]

- **Algorithms**
  - black: Athanor
  - blue: LNS_EB
  - red: LNS_PG
  - orange: Oscar
  - green: SNS

- **Graph Details**
  - X-axis: Time (in seconds)
  - Y-axis: Objective
CONCLUSIONS

➢ Enhanced CBLS using abstract Essence types,

➢ Automatic construction of semantically meaningful neighbourhoods,

➢ Dynamic scaling,

➢ Strong performance shown with problems using sets and multisets.

➢ Read the paper and speak to me for more details:
  ➢ Incremental evaluation, dynamic unrolling, more problem classes…