

# Integration Schemas for Constraint Answer Set Programming: a Case Study

---

Marcello Balduccini  
Drexel University

Yulia Lierler  
University of Nebraska at Omaha

# Introduction

---

- Different KR&R techniques for different domains
- Planning: ASP
- Scheduling: CP
- Scheduling + Planning: ?
  
- Possible solution:
  - Hybrid approaches
    - Satisfiability modulo theories (SMT)
    - Constraint answer set programming (CASP)

# CASP

---

- Modeling capabilities of ASP
- SAT-like solving technology
- Constraint processing for non-Boolean constructs
  
- Multiple CASP solvers:
  - ACSolver
  - Clingcon
  - EZCSP
  - IDP
  
- Applications:
  - Commercial printing
  - Robotics

# Motivation

---

- Similarities and differences between CASP solvers are unclear
- Implications of integration schemas used are unclear
  - EZCSP: black-box architecture
  - ACSolver, clingcon: tighter integration
- Development of CASP solvers is hard
  - Requires expertise in SAT, ASP, CSP areas
- Principled and general study of development methods needed
- Need for standardized techniques to integrate computational methods from multiple research areas

# In This Paper

---

- Case study of CASP integration schemas and their performance
- 3 integration schemas
  - Black-box
  - Grey-box
  - Clear-box
- 2 domains from ASPCOMP 2011
  - Weighted Sequence
  - Incremental Scheduling
- 3 types of encodings
  - Pure ASP
  - True CASP
  - Pure CSP

# CASP

---

- Generally, rules are of the form:

$$l_0 \leftarrow l_1, \dots, l_k, \text{not } l_{k+1}, \dots, \text{not } l_m$$

–  $l_i$ : regular or constraint atoms

- Example:  $p \leftarrow q, x > 2.$

- Semantics (intuition):

1. Constraint atom  $c$  treated as regular atom;  
 $\Pi'$  is  $\Pi +$  choice rules  $\{c\}$
2.  $A$  is answer set of  $\Pi$  if  $A$  is answer set of  $\Pi'$   
and  $A$  satisfies all relevant constraint atoms

# Integration Schemas Considered

---

- Black-box
  - ASP solver finds an answer set,  $A$ , and terminates
  - CP solver checks if constraint atoms of  $A$  have a solution
  - No a solution: denials added to  $\Pi$  and ASP solver is called again
  - **Pros**: solver-independent implementation; CP solver called rarely
  - **Cons**: full answer set must be computed; ASP search space discarded
- Grey-box
  - ASP solver finds an answer set,  $A$ , and is suspended
  - CP solver checks if constraint atoms of  $A$  have a solution
  - No a solution: ASP solver is resumed and looks for another answer set
  - **Pros**: CP solver called rarely; ASP search space re-used
  - **Cons**: slight dependence on ASP solver API; full answer set must be computed
- Clear-box
  - ASP and CP solver are interleaved:
    - While ASP solver computes an answer set, CP solver checks partial answer set,  $P$
    - If constraint atoms of  $P$  have no solution, CP triggers backtracking in ASP
  - **Pros**: early pruning; ASP and CP search space are re-used
  - **Cons**: complete dependence on ASP and CP solver APIs; CP solver called often

# Encodings Considered

---

- Encodings can be written to rely on solving capabilities of either side
- Pure ASP
  - Constraint atoms not used
- Pure CSP
  - ASP component is trivial (e.g. no loops through negation, no choice rules)
- True CASP
  - Both ASP and CP component are non-trivial

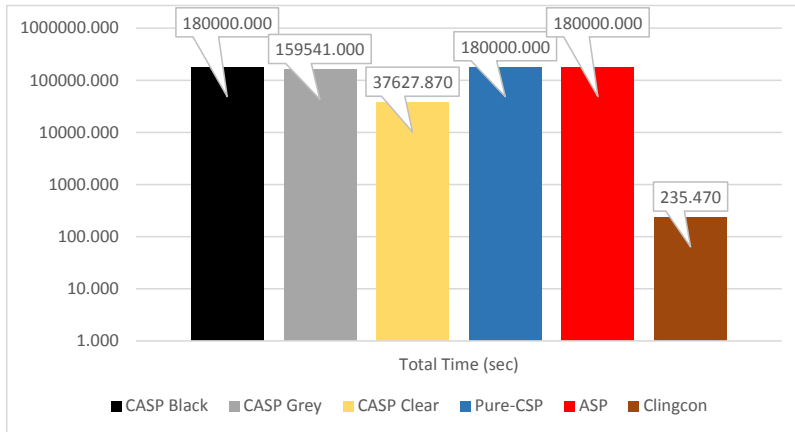


# Experiments

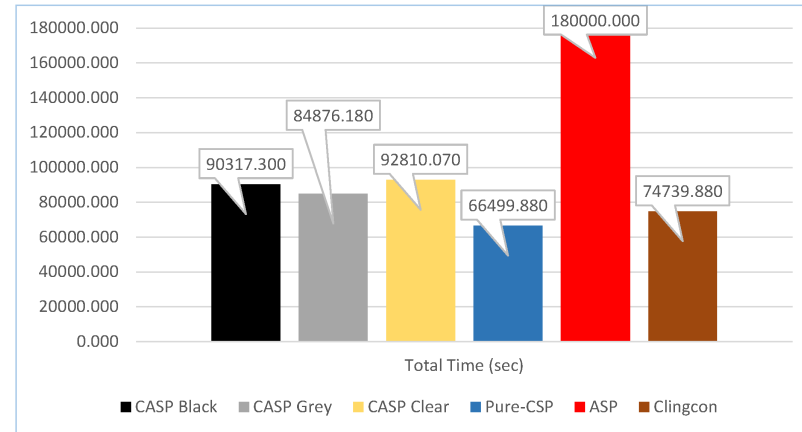
---

- EZCSP used as testbed
  - Extended to support black-box, grey-box and clear-box integration schemas
  - MiniSAT API used for grey-box and clear-box
- Also compared with cmodels and clingcon
- ASPCOMP11 domains:
  - Weighted-sequence
  - Incremental scheduling

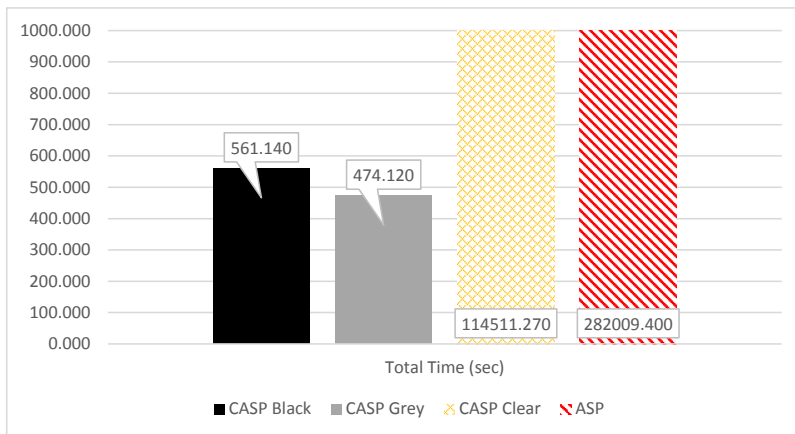
# Experimental Results



Weighted-sequence: total times; logarithmic scale



Incremental scheduling: total times



Reverse folding: total times

## Instances

- Weighted sequence:
  - 30, ASPCOMP11
- Incremental scheduling:
  - 50, ASPCOMP11: easy, in paper
  - 30, manual: hard, shown here

Timeout: 6,000 seconds

# Conclusions

---

- Best integration schema is domain-dependent
- Finer-grained schemas may be useful
  - Between grey-box and clear-box
- Hybrid systems should support various integration schemas
  - Standardized, flexible APIs for ASP and CP solvers are necessary
- Solvers and problem instances available online

<http://www.mbalduccini.tk/ezcsp/aspocp2013/ezcsp-binaries.tgz>

<http://www.mbalduccini.tk/ezcsp/aspocp2013/experiments.tgz>