Results of the 5th International Competition on Computational Models of Argumentation

Matti Järvisalo Tuomo Lehtonen Andreas Niskanen

University of Helsinki, Finland

September 3, 2023

ICCMA

"The competition aims at nurturing research and development of implementations for computational models of argumentation."

`http://argumentationcompetition.org/`

- Biennial competition series for argumentation reasoning systems organized since 2015
- Calls for submitting both **solvers** and challenging **benchmarks**
- Focus on **abstract argumentation**, recent interest in **dynamic** and **structured** formalisms

Semantics and Reasoning Tasks

Argumentation **semantics**:

- Complete (CO)
- Stable (ST)
- Preferred (PR)
- Semi-stable (SST)
- Stage (STG)
- Ideal (ID)

Note: no grounded (GR) semantics since 2021.

Reasoning **tasks**:

- Credulous acceptance (DC)
- Skeptical acceptance (DS)
- Single extension (SE)

Note: we do not consider enumeration (EE) nor counting (CE).

Subtracks

A combination of semantics and reasoning problem is a **subtrack**

- Exclude DC-PR ($=$ DC-CO), DS-CO ($=$ GR), SE-CO ($=$ GR), DC-ID and DS-ID ($\approx$ SE-ID)
- Solvers can be submitted to **any choice of subtracks**
- **No requirement** for solvers to support, e.g., all reasoning problems for a specific semantics

Tracks

- **Main track:** exact solvers for abstract argumentation
    - sequential and open-source, no portfolios
    - parallel, portfolio-based, and closed-source solvers invited to participate in the **No-limits track**
    - subtracks: all combinations of semantics and problems
- **Approximate track:** inexact solvers for abstract argumentation
    - solvers are not required to provide correct answers to acceptance queries
    - incorrect solutions discarded and correct solutions taken into account
    - subtracks: all combinations of semantics and acceptance problems (DC, DS)

Tracks

- **Dynamic track:** solvers for queries over sequences of related AFs
  - changes to an initial AF and acceptance queries issued via an API
  - subtracks: DC-CO, DC-ST, DS-ST, ~~DS-PR~~
- **ABA track:** exact solvers for assumption-based argumentation
  - focus flat ABA frameworks over the logic programming fragment
  - subtracks: DC-CO, DC-ST, DS-PR, DS-ST, SE-PR, SE-ST

Resource Limits and Scoring

Main, Dynamic, and ABA tracks:

- Time limit: 1200 seconds (wall time for No-limits track, CPU time for other tracks)
- Memory limit: 32 GB
- **PAR-2 scoring**
  - score of a solver on an instance is $2 \cdot 1200$ if resource limits reached, and solving time otherwise
  - score of a solver on a subtrack is the sum of scores over every instance
  - solver with lowest score wins a subtrack

Approximate track:

- Time limit: 60 seconds CPU time
- Memory limit: 32 GB
- Solver with the **largest number of correctly solved instances** wins a subtrack, CPU time as a tiebreaker
  - at least one Main track solver reports the same result

Input Format

More compact AF file format similar to DIMACS formats:

```
p af 5
# this is a comment
1 2
2 4
4 5
5 4
5 5
```

For ABA frameworks a similar format is used.

Output Format

Solvers participating to Main and No-limits tracks are required to produce **witnesses for acceptance queries**.

- DC and "YES" answers: extension containing query argument
- DS and "NO" answers: extension excluding query argument

Output on w-line, as on the SE task:

```
user$ solver -p DC-CO -f instance_file -a 1
YES
w 1 4
```

For the Approximate and ABA tracks, a YES or NO suffices for all acceptance queries.

IPAFAIR: New API for Dynamic Track

On the Dynamic track, solver I/O is performed via IPAFAIR:

`https://bitbucket.org/coreo-group/ipafair`

Includes functions for **adding/removing arguments/attacks**, as well as performing **credulous/skeptical acceptance queries**

Both C and Python versions: solvers must implement the Python version, and a C-to-Python example wrapper is available

- C version inspired by IPASIR: incremental API for SAT solving

IPAFAIR

Demo!

Benchmark Selection

Main, No-Limits, and Approximate Tracks

Benchmark domains:

- All domains in ICCMA 2017 (11 domains) and ICCMA 2019 (2 domains)
  - GroundedGenerator, SccGenerator, StableGenerator: new AFs generated with similar parameters
- New generator submitted to ICCMA 2023: **crusti_g2io**

Selection procedure:

- Generated AFs using crusti_g2io with suggested parameters
- Sampled 32 AFs from the new crusti_g2io domain and at most 25 AFs from the rest of the domains, resulting in 329 instances
- Acceptance query: sample from the set of arguments which are not roots nor self-attackers

Benchmark Selection

ABA Track

- Used a simple generator which allows for varying the following parameters:
  - number of atoms (25, 100, 500, 2000, 5000)
  - proportion of axioms (10%, 30%)
  - maximum number of rules deriving each sentence (5, 10)
  - maximum size of each rule body (5, 10)
- Generated 10 instances for each combination, resulting in 400 instances
- Acceptance query: sample from the set of atoms for which there is at least one derivation

Benchmark Selection

Dynamic Track

- Benchmark AFs used in Main, No-limits, and Approximate tracks as input
- For each of the AFs:
  - Perform BFS from query argument, fixing 33.3% of the first arguments encountered as static, and add another 33.3% of the arguments to the initial AF
  - From the fixed arguments, generate 15 additional query arguments
  - Repeat 64 times:
    - Check acceptance status of each query argument
    - Perform 32 changes to the AF: add or delete arguments along with incident attacks

Witness Verification

Witness extensions were verified using the following procedure:

- Check that witness exists in required scenarios (SE except for "NO" on SE-ST, "YES" on DC, "NO" on DS)
- Check that query is contained (DC) or excluded (DS)
- Verify the witness using standard SAT-based techniques:
    - CO, ST: construct standard SAT encoding and check that witness extends to a satisfiable assignment
    - PR, SST, STG: additional UNSAT check in addition to verifying satisfiability
    - ID as exception: simply check that extensions reported by different solvers are the same

SAT solver calls performed using GLUCOSE (v4.1) via PYSAT

- PR, SST, STG: UNSAT proofs recorded and verified using DRAT-TRIM

Witness Verification

All witnesses successfully verified with the following exceptions:

- Timeouts: 1 on SE-PR and DS-PR (verified with a higher timeout), 2 on SE-SST (memory-out with a higher timeout)
- Memory-outs due to constructing UNSAT proof: 10 on SE-PR, 1 on DC-SST, DC-STG, DS-SST, DS-STG

Participants

Remarks:

- Fuzz testing applied to each of the participating solvers: in case bugs were encountered, solver developers were contacted and bug fixes were allowed to the extent feasible
- All solvers involving any of the organizers of ICCMA 2023 were made known to the ICCMA steering committee and published online before the submission deadline
  - Benchmark generation and selection done using a random seed provided by the ICCMA steering committee members

Participants

Main and No-Limits Tracks

Solver	Authors
CRUSTABRI	Jean-Marie Lagniez, Emmanuel Lonca and Jean-Guy Mailly
FUDGE	Matthias Thimm, Federico Cerutti and Mauro Vallati
μ-TOKSIA	Andreas Niskanen and Matti Järvisalo
PORTSAT	Sylvain Declercq, Quentin Januel Capellini, Christophe Yang, Jérôme Delobelle and Jean-Guy Mailly

- μ-TOKSIA with 2 different versions: GLUCOSE and CMSAT

Participants

Approximate Track

Solver	Authors
AFGCNv2	Lars Malmqvist
ARIPOTER-DEGREES	Jérôme Delobelle, Jean-Guy Mailly and Julien Rossit
ARIPOTER-HCAT	Jérôme Delobelle, Jean-Guy Mailly and Julien Rossit
FARGO-LIMITED	Matthias Thimm
HARPER++	Matthias Thimm

Participants

Dynamic Track

Solver	Authors
CRUSTABRI	Jean-Marie Lagniez, Emmanuel Lonca and Jean-Guy Mailly
μ-TOKSIA	Andreas Niskanen and Matti Järvisalo
κ-SOLUTIONS	Christian Pasero and Johannes P. Wallner

- μ-TOKSIA with 2 versions: STATIC and DYNAMIC

Participants

ABA Track

Solver	Authors
ACBAR	Tuomo Lehtonen, Anna Rapberger, Markus Ulbricht and Johannes P. Wallner
ASPFORABA	Tuomo Lehtonen, Matti Järvisalo and Johannes P. Wallner
ASTRA	Andrei Popescu and Johannes P. Wallner
CRUSTABRI	Jean-Marie Lagniez, Emmanuel Lonca and Jean-Guy Mailly
FLEXABLE	Martin Diller, Sarah Alice Gaggl and Piotr Gorczynca

Results

Main Track

	Rank (average PAR-2 score)			
Solver	DC-CO	DC-SST	DC-ST	DC-STG
Crustabri	2 (172.92)	3 (411.80)	**1 (139.29)**	**1 (444.33)**
Fudge	- (147.31)	- (311.79)	- **(132.86)**	- (507.53)
μ-toksia (cmsat)	3 (202.88)	2 (268.39)	3 (224.83)	2 (459.92)
μ-toksia (glucose)	**1 (143.56)**	**1 (263.32)**	2 (154.56)	3 (504.51)
PORTSAT	- (152.20)	-	- (166.32)	-

	Rank (average PAR-2 score)			
Solver	DS-PR	DS-SST	DS-ST	DS-STG
Crustabri	2 (279.27)	**1 (357.38)**	**1 (223.34)**	**1 (360.12)**
Fudge	- (435.91)	- (501.33)	- (236.00)	- (429.91)
μ-toksia (cmsat)	3 (325.07)	3 (401.54)	3 (317.58)	2 (438.09)
μ-toksia (glucose)	**1 (242.69)**	2 (362.83)	2 (271.21)	3 (497.12)
PORTSAT	- (1151.41)	-	- **(219.11)**	-

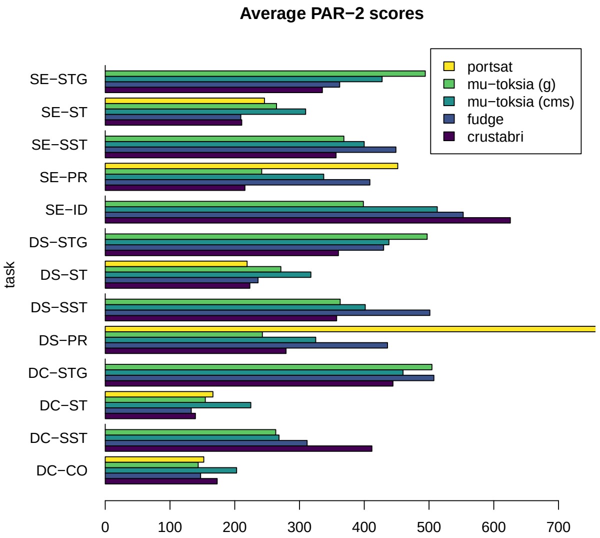
Results

Main Track

	Rank (average PAR-2 score)				
Solver	SE-ID	SE-PR	SE-SST	SE-ST	SE-STG
Crustabri	3 (625.59)	**1 (215.76)**	**1 (356.48)**	**1 (210.83)**	**1 (335.33)**
Fudge	- (552.81)	- (408.70)	- (448.90)	- **(209.39)**	- (362.04)
μ-toksia (cmsat)	2 (512.76)	3 (337.43)	3 (399.93)	3 (309.49)	2 (427.59)
μ-toksia (glucose)	**1 (398.65)**	2 (241.65)	2 (368.52)	2 (264.52)	3 (494.24)
PORTSAT	-	- (451.73)	-	- (245.92)	-

Results

Main Track



Results

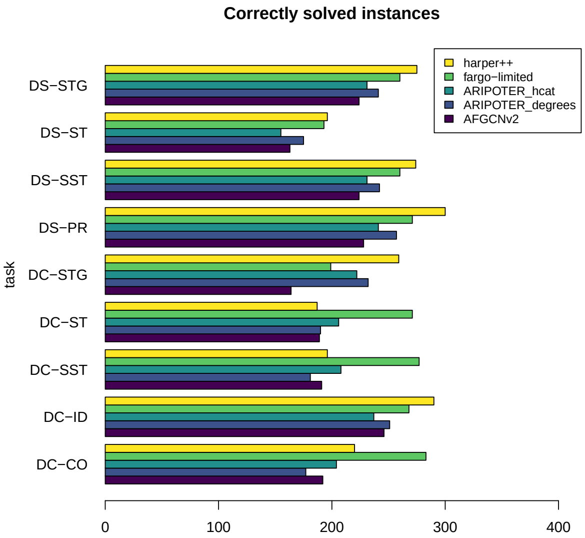
Approximate Track

	Rank (# correctly solved)				
Solver	DC-CO	DC-ID	DC-SST	DC-ST	DC-STG
AFGCNv2	4 (192)	4 (246)	4 (191)	4 (189)	5 (164)
ARIPOTER (DEGREES)	5 (177)	3 (251)	5 (181)	3 (190)	2 (232)
ARIPOTER (HCAT)	3 (204)	5 (237)	2 (208)	2 (206)	3 (222)
FARGO-LIMITED	**1 (283)**	2 (268)	**1 (277)**	**1 (271)**	4 (199)
HARPER++	2 (220)	**1 (290)**	3 (196)	5 (187)	**1 (259)**

	Rank (# correctly solved)			
Solver	DS-PR	DS-SST	DS-ST	DS-STG
AFGCNv2	5 (228)	5 (224)	4 (163)	5 (224)
ARIPOTER (DEGREES)	3 (257)	3 (242)	3 (175)	3 (241)
ARIPOTER (HCAT)	4 (241)	4 (231)	5 (155)	4 (231)
FARGO-LIMITED	2 (271)	2 (260)	2 (193)	2 (260)
HARPER++	**1 (300)**	**1 (274)**	**1 (196)**	**1 (275)**

Results

Approximate Track



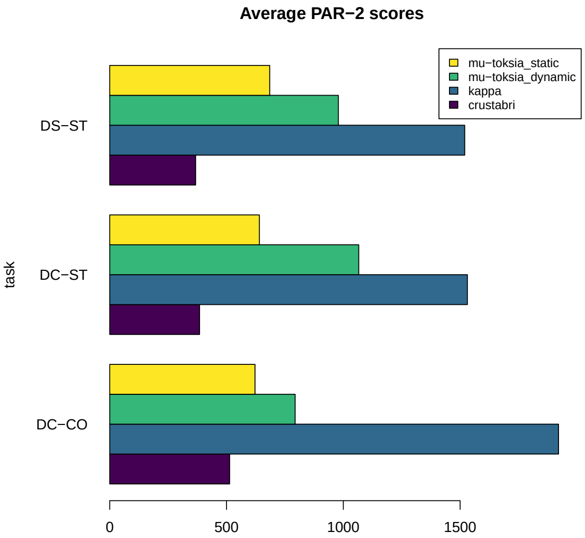
Results

Dynamic Track

	Rank (average PAR-2 score)		
Solver	DC-CO	DC-ST	DS-ST
Crustabri	**1 (513.37)**	**1 (384.68)**	**1 (367.82)**
μ-toksia (static)	2 (622.01)	2 (640.56)	2 (684.92)
μ-toksia (dynamic)	3 (793.80)	3 (1066.09)	3 (978.76)
κ-solutions	4 (1921.25)	4 (1531.09)	4 (1519.69)

Results

Dynamic Track



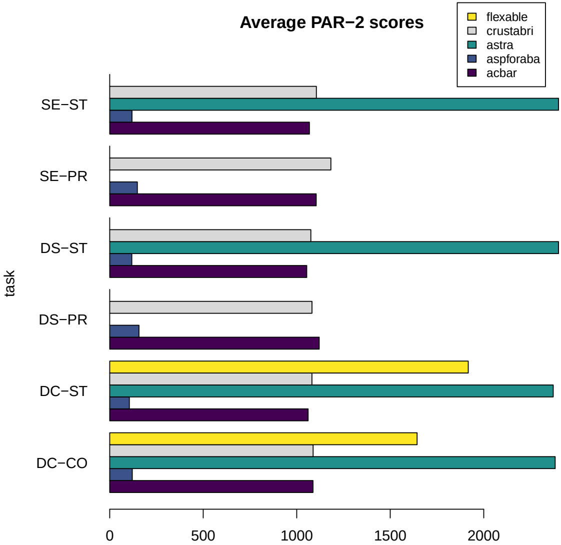
Results

ABA Track

	Rank (average PAR-2 score)					
Solver	SE-ST	SE-PR	DS-ST	DS-PR	DC-ST	DC-CO
ASPforABA	**1 (119.1)**	**1 (147.8)**	**1 (118.2)**	**1 (156.5)**	**1 (105.1)**	**1 (120.6)**
AcbAr	2 (1067.9)	2 (1104.2)	2 (1053.6)	2 (1120.3)	2 (1060.9)	2 (1087.0)
Crustabri	- (1105.1)	- (1182.7)	- (1075.5)	- (1081.6)	- (1081.4)	- (1087.7)
flexABle	-	-	-	-	3 (1917.3)	3 (1643.7)
ASTRA	3 (2400.0)	-	3 (2400.0)	-	4 (2371.7)	4 (2382.0)

Results

ABA Track



Statistics

- Main and No-limits tracks solving: 68.9 CPU days
    - Witness verification: 1.94 CPU days
    - UNSAT proof checking: 1.39 CPU days
- Approximate track: 1.81 CPU days
- Dynamic track: 23.3 CPU days
- ABA track: 43.8 CPU days

Grand total: 141.14 CPU days $\approx$ 0.39 CPU years!

Summary

Main track winners:

- Crustabri (DC-ST, DC-STG, DS-SST, DS-ST, DS-STG, SE-PR, SE-SST, SE-ST, SE-STG)
- μ-toksia (glucose) (DC-CO, DC-SST, DS-PR, SE-ID)

Approximate track winners:

- fargo-limited (DC-CO, DC-SST, DC-ST)
- harper++ (DC-ID, DC-STG, DS-PR, DS-SST, DS-ST, DS-STG)

Dynamic track winner: Crustabri (all subtracks)
ABA track winner: ASPforABA (all subtracks)

Discussion

New features:

- Solvers can choose to participate **in any subtrack**
- Main track: **witnesses** required for YES/NO answers on DC/DS
  - Verification using standard SAT solving methods
- No-limits track: **parallel** and **portfolio** solvers
- Dynamic track: **IPAFAIR** interface for dynamic AF solvers

Considerations:

- All participants in Main, No-limits, and Dynamic tracks based on (iterative) SAT solving
- Few new benchmarks: one generator submitted
- No verification of DC/NO and DS/YES answers

Remarks on ABA Track

- First time structured argumentation included in ICCMA
- Systems with differing objectives and architecture
- E.g. FLEXABLE is mainly aimed at providing dialectical justifications instead of (efficiently) deciding acceptance or computing extensions
- Range of benchmarks limited for now
  - E.g., some solvers known to work best when less circularity in rules (ASTRA, ACBAR)
  - Benchmark accumulation for future competitions important for this new track

Thanks to all participants of ICCMA 2023!

Thanks to the Finnish Computing Competence Infrastructure (FCCI) for supporting this project with computational and data storage resources!

Congratulations to all the winners!

`https://iccma2023.github.io/`