Tug of War Problem with *Answer Set Programming – clingo* An Encoding

Claudio Cesar de Sá

Independent Researcher

Road map of this presentation:

1. About the ASP (clingo) – previous presentation (done)
2. Requisites – previous presentation (done)
3. Tug of War (well knowed problem from competitive programming sites and contests)
4. A modelling in ASP
5. A solution in clingo
6. Conclusions

Road map of this presentation:

1. About the ASP (clingo) – previous presentation (done)
2. Requisites – previous presentation (done)
3. Tug of War (well knowed problem from competitive programming sites and contests)
4. A modelling in ASP
5. A solution in clingo
6. Conclusions

Attention: some background in logic and declarative language is recommended!

Tug of War Problem

Figura: Practical application of this problem

Tug of War

From: `https://www.geeksforgeeks.org/tug-of-war/`
and/or `https://www.codechef.com/problems/C0319TSH`

Given a set of n integers, divide the set in two subsets of $n/2$ sizes each such that the difference of the sum of two subsets is as minimum as possible. If n is even, then sizes of two subsets must be strictly $n/2$ and if n is odd, then size of one subset must be $(n-1)/2$ and size of other subset must be $(n+1)/2$.

Examples

From: `https://www.geeksforgeeks.org/tug-of-war/`

- Example 1: let given set be {3, 4, 5, -3, 100, 1, 89, 54, 23, 20}, the size of set is 10. Output for this set should be {4, 100, 1, 23, 20} and {3, 5, -3, 89, 54}. Both output subsets are of size 5 and sum of elements in both subsets is same (148 and 148).
- Let us consider another example where n is odd. Let given set be {23, 45, -34, 12, 0, 98, -99, 4, 189, -1, 4}. The output subsets should be {45, -34, 12, 98, -1} and {23, 0, -99, 4, 189, 4}. The sums of elements in two subsets are 120 and 121 respectively.
- This problem is beauty: *easy to understand, hard to solve it*!

Comments

- Again: all the combinations must be found!
- Input: a set of numbers, in our implementation an array, aiming a possible repetitions of these numbers.
- Output: two sets with the same size/cardinality, or with difference of one number for set A or B.
- Complexity: NP-Complete (all the combinations must be examined) – Set partition problem is NP complete – `https://www.geeksforgeeks.org/set-partition-is-np-complete/`
- Optimization: NP-Hard, due the minimum value of the absolute difference between the sum of two sets.

Some comments:

- I did it in Minizinc
- Some approaches for this problem can be taken: Simulated Annealing, Ant Colony, Depth-First Search, ..., meta-heuristics in general presents a good efficiency
- Dynamic Programming is the most suitable for contest programming
- The full code discussed here is found in: `https://github.com/claudiosa/CCS/tree/master/asp_Answer_Set_Programming/tug_of_war.lp`
- We are commenting the modelling in parts of its code

Modelling:

Ground terms, exactly written like Prolog syntax.

The map, countries and their relations with neighbours:

Another representation is possible, but until now, everything was reused – kept simple as possible!

Modelling the problem under its requisites:

Basically, that's all!

Preparing for a optimization:

That's all!

An output:

```
clingo ../map_coloring.lp 0 --out-ifs='\n' --out-atomf=%s.
clingo version 5.3.0
```

Conclusions:

- ASP is strongly declarative (roots from the logic to attack the problems representation)
- Generate and test methodology
- ASP's workflow, modeling, grounding, solving (and optimizing)
- clingo = gringo+clasp + ...
- Allows you to embed a Python coding in order to minimize the difficulties (☹) of input and output data
- An encoding in ASP is excellent exercise to keep your mind very active!
- Finally, a huge gratitude for the **potassco-users list**, always reactive for my silly doubts, where I had been learning much.

Contact and comments (are must welcome ☺):

- `https://claudiocesar.wordpress.com/`
- This presentation and the code discussed: **`https://github.com/claudiosa/CCS/tree/master/asp_Answer_Set_Programming`**
- There is a directory to Youtube!
- ✉: `ccs1664@gmail.com`
- This material has a partial support from WhatsTV Inc. `https://en.whatstv.com.br/`, here our gratitude!
- *Thank you so much*!