Sevilla Carpets: new extensions for new problems

David Orellana-Martín and Sevilla Team

Research Group of Natural Computing, Department of Computer Science and Artificial Intelligence, University of Sevilla

February 2015

1. Introduction

2. Sevilla Carpets
   - Parameters for the Descriptive Complexity

3. Tool description
   - Rules format

4. Future work, ideas...

1. Introduction

2. Sevilla Carpets
   - Parameters for the Descriptive Complexity

3. Tool description
   - Rules format

4. Future work, ideas...

Introduction

P systems:

- Massively parallel computing devices
- Great number of symbol-objects, membranes and rules
- Powerful tools for solving **NP**-complete problems
- Solutions to this kind of problems in polynomial time

But...

- If we work with a increasing number of membranes, how can we describe the complexity of the computational process?

Introduction

G. Ciobanu, Gh. Păun and Gh. Ştefănescu presented a new way to describe the complexity of a computation in a P system, the so-called *Sevilla Carpet*.

1. Introduction

2. Sevilla Carpets
    - Parameters for the Descriptive Complexity

3. Tool description
    - Rules format

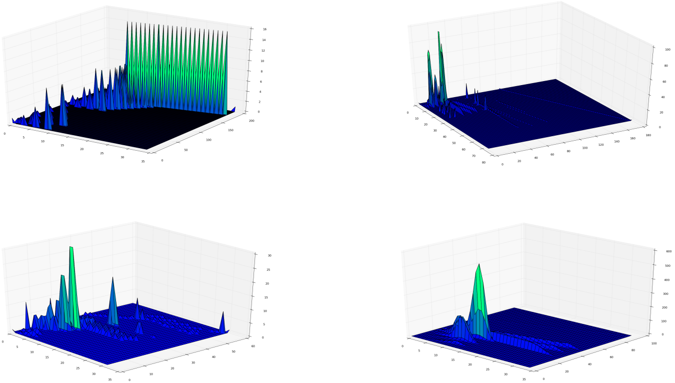
4. Future work, ideas...

Sevilla Carpets

Initial variants

1. Yes or no, rules in membranes in a step;
2. Yes or no, rules in a step;
3. Number of times of the rule in a step;
4. Cannot, would, is;
5. Multiply number of times of the rule in a step with it cost.

Sevilla Carpets



SAT, Knapsack, Partition and Subset Sum single computation carpets

Sevilla Carpets

- A three-dimensional picture may not provide significant information by itself
- It's why we have to define parameters related with Sevilla Carpets

Parameters for the Descriptive Complexity

- **Weight:** the sum of all the elements in the carpet.
- **Surface:** is the multiplication of the number of steps by the total number of the rules used.
- **Height:** is defined as the peak of the computation.
- **Average Weight:** division of *weight* to the *surface*.
- **Variance:** the sum of the squared differences between the elements of the carpet and the average weight, divided by the surface.

Parameters for the Descriptive Complexity

In GPU-based simulators for P systems, it has been observed that the speed-up obtained by such parallel simulations highly depends on how distributed the rule applications are during the simulated computation.

Parameters for the Descriptive Complexity

Some specific parameters related to the performance of GPU-simulations on GP-systems are:

- **Density of objects per membrane:** themore different objects are in the membrane, the higher thread usage.
- **Rule intensity:** rules that cannot be paralelized.
- **Communication among membranes:** how we have PDP-systems defined, the skin is executed on the CPU, this process slows down all the process.

1. Introduction
2. Sevilla Carpets
   - Parameters for the Descriptive Complexity
3. Tool description
   - Rules format
4. Future work, ideas...

Tool description

We have done a script in *python* that, receiving a PLinguaCore output, and returns a Sevilla carpet that represents the computation of the output file.

Rules format

```
STEP k:
Rules selected for MEMBRANE ID: x, Label:  y, Charge:  z
n * #r q
```

Where:

- k represents the current step
- r is the index of the applied rule
- q is the rule itself
- n represents the number of times that rule r is applied in step k

1. Introduction

2. Sevilla Carpets
   - Parameters for the Descriptive Complexity

3. Tool description
   - Rules format

4. Future work, ideas...

Future work, ideas...

- Before that, it was done manually.
- After processing the output of simulators for P systems with active membranes written in Prolog, we had to run the `gnuplot` script.
- Now all of this is done faster and simpler.
- We can cover all the models of P systems included in the P-Lingua framework.

Future work, ideas...

- Add parameters to the script to get another kind of carpet.
- Have an on-the-fly implementation of the algorithm, so we can obtain partial results.
- MeCoSim plugin... Done!
- In probabilistic P systems, it might be interesting to extract several samples of computations and then use the average values in order to generate the Sevilla Carpet and the associated parameters.
- Add a “projections option” to the script.
- And now, is your turn to power up the carpets!

Thank you!