

---

# **Centre for Artificial Intelligence**

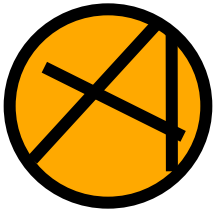
## **Universidade Nova de Lisboa**

**CENTRIA**

<http://centria.fct.unl.pt/>

Pedro Barahona

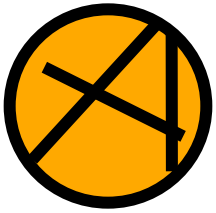
(contact [pb@di.fct.unl.pt](mailto:pb@di.fct.unl.pt))



# CENTRIA

---

- A Research Centre hosted by the Computer Science Dep. of UNL
- Structured into different research areas
- With special Interest for EMCL
  - **KRR** - Knowledge Representation and Reasoning
    - João Leite / Alfredo Gabaldon
  - **SCC** - Soft Computing and **Constraints**
    - Pedro Barahona / Marco Correia
    - Focus on **Constraint (Logic) Programming**

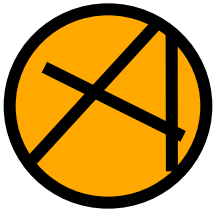


# Constraint Logic Programming

---

## ■ Main Insight

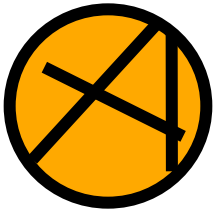
- **Constraint Propagation (to reduce the search space)**
- Extends unification of Herbrand terms (Logic Programming) to unification of other domains, most importantly
  - Finite Domains
  - (Finite) Sets
  - Continuous Domains
  - Linear constraints over Real Domains,
  - Other specialised Domains
- Maintains the **backtracking search** of Logic Programming, with some adaptations



# Constraint Logic Programming

---

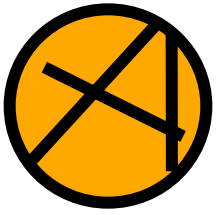
- Constraint Programming at UNL
  - Covers both Foundations (Models and Execution) and Applications
  - A group with 4 Lecturers\*, 1 Researcher\*\*, 1 Post-Doc\*\*, 2 Ph.D students\*\*\*, 4 MSc students.
  - Recent EMCL dissertations (one *best EMCL award*)
  - 3 new projects recently approved (European Space Agency, Carnegie Mellon, Portuguese Foundation for Science and Technology) and 2 about to be submitted
  - Some funding for M.Sc. / Ph.D. students.
    - \* Pedro Barahona, Francisco Azevedo, Jorge Cruz, Ludwig Krippahl
    - \*\* Davide d'Alimonte, Marco Correia
    - \*\*\* Luís Baptista, Elsa Carvalho



# Constraint Logic Programming

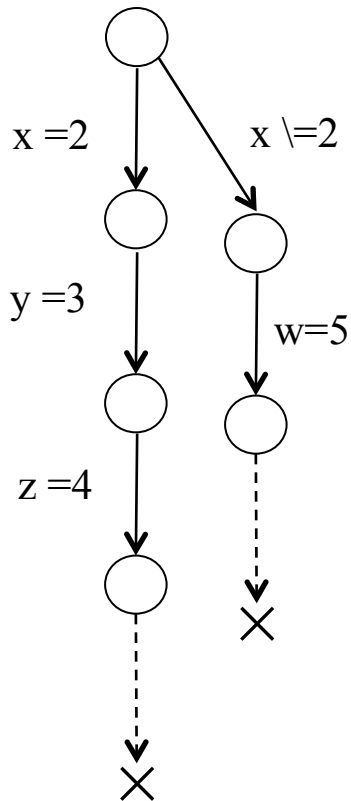
---

- Topics for Research (MSc and PhD)
  - **Foundations - Models and Execution**
  - Efficient **Models**
    - No-goods, Restarts, Symmetry Breaking
    - Counting Solutions, Heuristics and Backtracking
    - Hybrid Solvers
    - Continuous and Probabilistic Models
  - **Applications**
    - Bioinformatics
    - Oceanography
    - Model-Based Software verification



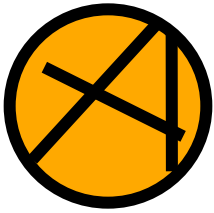
# Constraint Logic Programming

## ■ Restarts, No-Goods and Symmetry Breaking



- Record sets of conflict variables, to avoid repeating failures
  - Useful in restarts, but also without restarts if ...
  - Reduced (minimal) sets of conflicting variables are found
    - Ex:  $x = 2 \ \& \ y = 3 \ \& \ z = 4$  explain the conflict
  - Symmetry Breaking
    - Ex: For every solution with  $x = 2, y = 3 \ \& \ z = 4$ 
      - there is a solution with  $x = 3, y = 2 \ \& \ z = 6$
  - No-good composition (a la SAT)
    - $x \neq 2 \ \vee \ z \neq 4$
    - $x = 2 \ \vee \ w \neq 5 \implies z \neq 4 \ \vee \ w \neq 5$

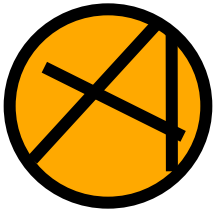
(A PhD going on here)



# Constraint Logic Programming

---

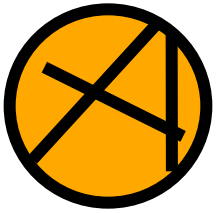
- Counting Solutions and Heuristics
  - Counting solutions is an important problem
    - Robustness of a model
    - How hard is it to satisfy it ?
      - E.g. A partially enumerated constraint model.
  - Provide information to lookahead heuristics
    - A non trivial implementation of first fail
    - Knowledge compilation approach (**Best EMCL thesis**)
    - Preliminary integration with **CaSPER** (more below)



# Constraint Logic Programming

---

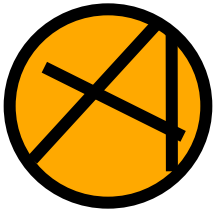
- Global Constraints & Hybrid Solvers
  - Constraint Propagation is local to each constraint
  - Global Constraints allow more propagation
  - Relaxation of linear models to reals are quite efficient
    - SIMPLEX
  - They can provide bounds for variables in Hybrid Solvers
  - Good performance in some examples (Benders decomposition)
  - Hybrid solving with CaSPER (post-doc)
    - Joint work with Operations Research colleagues
    - On going activity



# Constraint Logic Programming

---

- Constraints in Continuous Domains & Probabilistic Models
  - Constraint Propagation in continuous domains require different methods
    - Box-consistency, potentially infinite fix-points
    - Bound Approximations - Interval Newton Method
- Probabilistic Models
  - Pure constraints lead to safe reasoning (a value between 2 and 10 is possible)
  - ... but it does not distinguish likely outcomes (it is more likely around 4)
  - Currently laying out the foundations of Joint Constraint and Probabilistic reasoning (**ongoing PhD**)
  - Collaboration with University of Nantes (F)

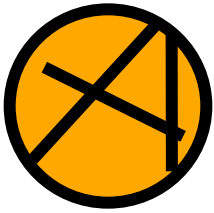


# Constraint Logic Programming

---

- Applications to Bioinformatics
  - 1 Phd and several MSc thesis already finished (2 EMCL thesis last year), with related publications
  - Most previous work in Protein Structure Determination and Docking
  - Recently focussing on Multiple Sequence Alignment
    - Just starting project with funds for an MSc / **Phd** student
  - Innovative Domains and Constraints



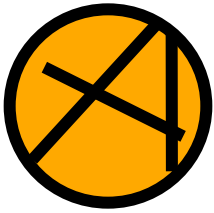


# Constraint Logic Programming

---

## ■ Applications to Oceanography

- A reverse Engineering problem: find parameters for a model that fit the available data.
- In this case, levels of algae or pollutants in ocean water collected by satellite pictures.
- Models have uncertainty, and have probability distributions associated
- Results should reflect these issues, and correlate the multiple parameter values.
- **A PhD being finished, an ESA project** (being extended until 2015?), and two national project proposals, to fund more scholarships.



# Constraint Logic Programming

## ■ Sea Surface Reflectance Model

- Continuous
- Highly non-linear

$$a_{Chla}(\lambda) = A(\lambda) \times Chla^{(1-B(\lambda))}$$

$$b_{Chla}(\lambda) = 0.3 \times Chla^{0.62} \times (550/\lambda)$$

$$a_{CDOM}(\lambda) = CDOM \times \exp(-0.017 \times (\lambda - 440))$$

$$a_{NPPM}(\lambda) = NPPM \times 0.04 \times \exp(-0.0123 \times (\lambda - 440))$$

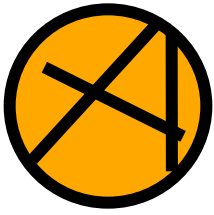
$$b_{NPPM}(\lambda) = NPPM \times 0.51 \times (\lambda/555)^{-0.5}$$

$$a(\lambda) = a_w(\lambda) + a_{Chla}(\lambda) + a_{NPPM}(\lambda) + a_{CDOM}(\lambda)$$

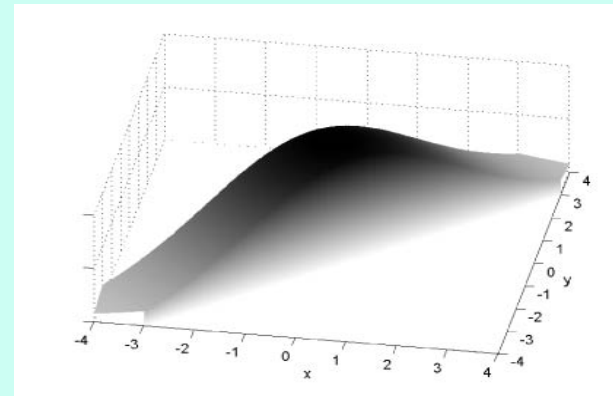
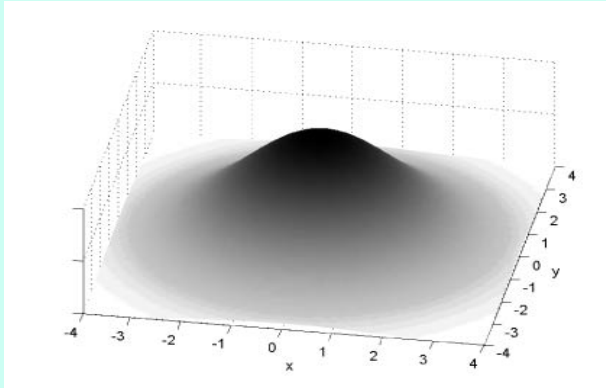
$$b_b(\lambda) = 0.0183 \times (b_w(\lambda) + b_{Chla}(\lambda) + b_{NPPM}(\lambda))$$

$$R_{rs}(\lambda) = 0.044 \times \frac{b_b(\lambda)}{a(\lambda) + b_b(\lambda)}$$

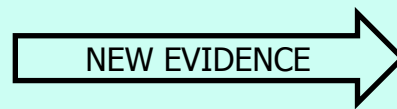
| $\lambda$          | 412      | 443      | 490      | 510      | 555      | 670      |
|--------------------|----------|----------|----------|----------|----------|----------|
| $R_{obs}(\lambda)$ | 0.005638 | 0.005588 | 0.006337 | 0.005571 | 0.004364 | 0.000733 |



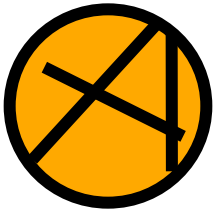
# Constraint Logic Programming



probability  
space

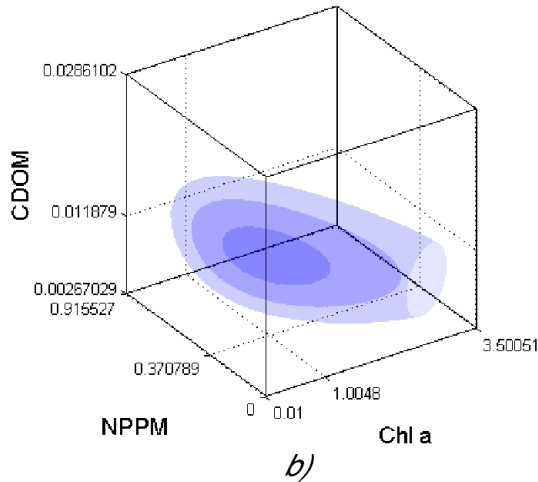


*conditional*  
probability space



# Constraint Logic Programming

## ■ Sea Surface Reflectance Initial results



$$a_{Chla}(\lambda) = A(\lambda) \times Chla^{(1-B(\lambda))}$$

$$b_{Chla}(\lambda) = 0.3 \times Chla^{0.62} \times (550/\lambda)$$

$$a_{CDOM}(\lambda) = CDOM \times \exp(-0.017 \times (\lambda - 440))$$

$$a_{NPPM}(\lambda) = NPPM \times 0.04 \times \exp(-0.0123 \times (\lambda - 440))$$

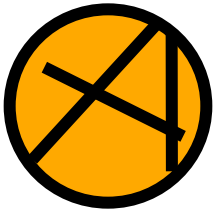
$$b_{NPPM}(\lambda) = NPPM \times 0.51 \times (\lambda/555)^{-0.5}$$

$$a(\lambda) = a_w(\lambda) + a_{Chla}(\lambda) + a_{NPPM}(\lambda) + a_{CDOM}(\lambda)$$

$$b_b(\lambda) = 0.0183 \times (b_w(\lambda) + b_{Chla}(\lambda) + b_{NPPM}(\lambda))$$

$$R_{rs}(\lambda) = 0.044 \times \frac{b_b(\lambda)}{a(\lambda) + b_b(\lambda)}$$

| $\lambda$          | 412      | 443      | 490      | 510      | 555      | 670      |
|--------------------|----------|----------|----------|----------|----------|----------|
| $R_{obs}(\lambda)$ | 0.005638 | 0.005588 | 0.006337 | 0.005571 | 0.004364 | 0.000733 |



# Constraint Logic Programming

---

- Applications to Software Certification
  - Software Design by Contract – append pre and post conditions to chunks of code (e.g. Java Modelling Language JML)
  - Model-based diagnosis: Code, pre-conditions and the **negation** of post-conditions are modelled as a constraint satisfaction problem.
    - Identified solutions are bugs!
  - A joint project recently started with IST (Portugal) and Carnegie Mellon University – CSP vs. other approaches (SAT- and/or SMT-based).
  - An EMCL thesis going on



# Constraint Logic Programming

---

- All these projects will benefit from having a development platform available
  - To test models
  - To handle different various domains,
  - To implement search strategies
  
- Such platform, **CaSPER**, has been developed at the UNL by **Marco Correia** who has recently finished his PhD (**next talk**).



# Constraint Logic Programming

---

Thank you