

APLICACIONES REALES DE MODELOS BIOINSPIRADOS

Tema 9: Software para la simulación de modelos computacionales en Membrane Computing.

David Orellana Martín
Mario de J. Pérez Jiménez
Luis Valencia Cabrera

Grupo de investigación en Computación Natural
Dpto. Ciencias de la Computación e Inteligencia Artificial
Universidad de Sevilla

Máster Universitario en Lógica, Computación e Inteligencia Artificial

Curso 2025-26



Motivación

Punto de partida

- ▶ (MPJ) *No hay nada más práctico que una buena teoría*
- ▶ (LVC) *Nada mejor para la teoría que funcionar en la práctica*

¿Cuál es la situación en Membrane Computing?

- ▶ **Teoría**, fundamentos sólidos
- ▶ **¿Práctica?**
 - ▶ *Modelización* diseñando soluciones a problemas reales
 - ▶ *Simulación* para manejar a nivel práctico los modelos

Idea inicial

Metodología para el manejo práctico de modelos

- ▶ **Problema** → **Solución**

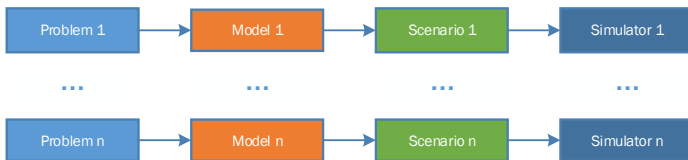
Herramientas software para soportar la metodología

- ▶ Ayudar a *diseñadores* de los modelos
- ▶ Proporcionar apps para *usuarios finales*
- ▶ Aplicar a **problemas relevantes** de la **vida real**

Tendencias de simulación en Membrane Computing

Specific purpose. Un problema o instancia del mismo

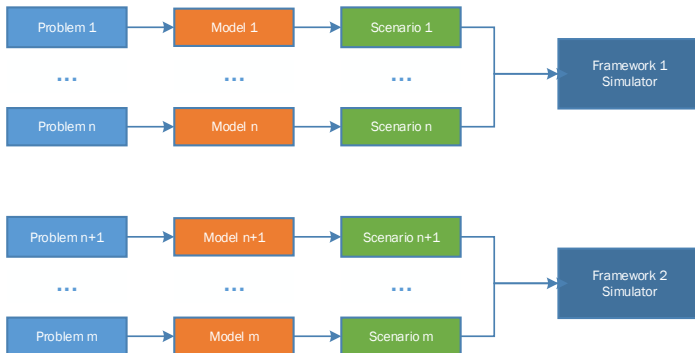
- ▶ Simuladores *ad-hoc*.
- ▶ Enfoque: solucionar un **problema/instancia específica**.



Tendencias de simulación en Membrane Computing

Framework oriented. Soluciones para un tipo de sistemas P

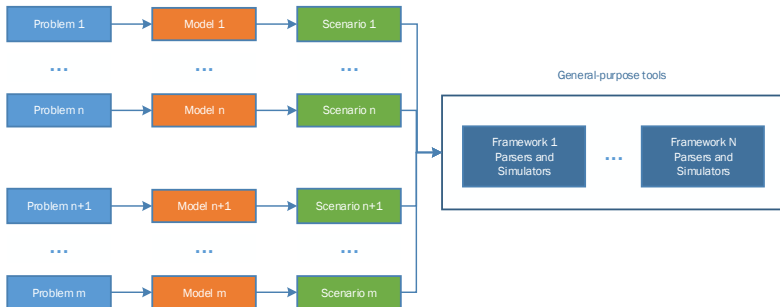
- ▶ Rango más amplio, dentro de un **marco específico**.
- ▶ **MetaPlab, Infobiotics Workbench, kPWorkbench**, etc.



Tendencias de simulación en Membrane Computing

General purpose. Genérico

- ▶ **Alcance Global** (muchos marcos en Membrane Computing).
- ▶ **P-Lingua framework¹, UPSimulator, Formal Framework.**



¹Grupo de Investigación en Computación Natural (RGNC, US) .

P-Lingua framework

Concepto general

- ▶ Un **lenguaje estándar**, P-Lingua, para especificar, definir sistemas P y familias de los mismos, permitiendo parametrizar los modelos y emplear programación estructurada.
- ▶ Un conjunto de **herramientas software open source**, pLinguaCore, para trabajar con estos sistemas:
 - ▶ **Parsers** para reconocer el lenguaje, identificar errores sintácticos o semánticos, distinguir reglas válidas según los tipos de sistemas P, etc.
 - ▶ Utilidades para generar **representaciones de salida** de los modelos anteriores, capacidad para **instanciar** los sistemas P generados, y **simularlos** mediante distintos motores.
 - ▶ Opción de llamar a la **biblioteca** pLinguaCore desde **Java**.
 - ▶ Herramientas de **línea de comandos** para *depuración*, *parsing de salida* y *simulación*.
- ▶ Capas de *software adicionales*, empleando pLinguaCore:
 - ▶ Un entorno para la **experimentación virtual**: MeCoSim.

Parsers y simuladores

- ▶ Cell-like P systems.
- ▶ Tissue-like P systems.
- ▶ Spiking Neural P systems.
- ▶ PDP systems.
- ▶ Simple kernel P systems.

Múltiples variantes de cada uno de estos tipos

Estado actual

- ▶ **pLinguaCore 4.0** (2007-2013), muchos **modelos incorporados**
- ▶ **pLinguaCore en MeCoSim** (2010-*hoy*), continuación de pLinguaCore 4.0, incorporando sucesivos parsers y simuladores.
- ▶ pLinguaCore 5.0 (2019-*hoy*), en desarrollo, rediseño de los anteriores generalizando los modelos soportados y permitiendo la meta-definición de tipos y variantes con un nuevo lenguaje de especificación de los marcos.

P-Lingua

Referencias

- ▶ [Sitio web de P-Lingua](#), en forma de wiki.
- ▶ [Página de software en sitio web de Ignacio Pérez](#), desarrollador de P-Lingua
- ▶ Material del [curso matvida](#)
- ▶ [Presentación introductoria sobre P-Lingua](#)

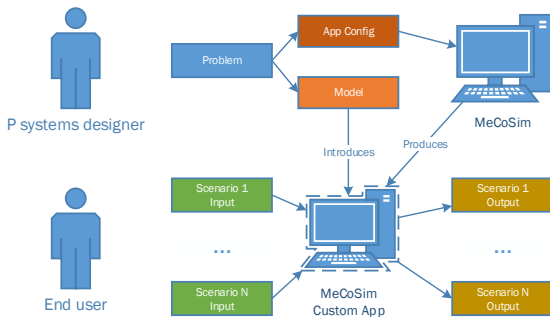
Limitaciones

- ▶ Alto acoplamiento instancia - solución en mismo archivo.
- ▶ Sin interfaz de alto nivel para diseñadores de sistemas P.
- ▶ Orientado a expertos en sistemas P, no a usuarios finales.

MeCoSim (Membrane Computing Simulator)

Objetivos

- ▶ Para **diseñadores de sistemas P**: modelización, depuración, simulación, análisis, visualización, etc.; provisión de *end-user applications* personalizadas.
- ▶ Para **usuarios finales**: apps a medida (**caja negra**), para introducir datos, lanzar experimentos virtuales y tomar resultados.
- ▶ Requerido: **flexibilidad, extensibilidad**.



Referencias

- ▶ **An environment for virtual experimentation with computational models based on P systems**, donde se detalla la filosofía y funcionalidades incluidas en MeCoSim.
- ▶ **Sitio web de MeCoSim.**, para descarga, primeros pasos, videos, casos de estudio, etc.
- ▶ Material del **curso matvida**

Simulación de sistemas P con P-Lingua y MeCoSim

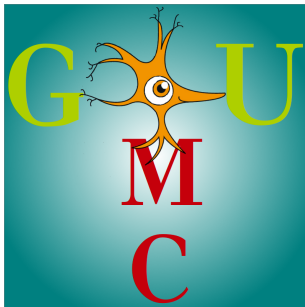
Aplicación a la modelización de problemas de la vida real

Referencias

- ▶ Presentación: *Software for modeling ecosystems by means of P systems*, por Ignacio Pérez.
- ▶ Tema: *Modelos en Dinámica de Poblaciones*, del *curso matvida*.
- ▶ Artículo: modelización de un caso real - *el mejillón cebra en el embalse de Ribarroja*, como se presentó en Ecological Complexity. (descargar ejemplo [aquí](#))

Aceleración de simuladores: GPU

El proyecto PMCGPU



Nace en **Aceleración de Simuladores de Sistemas de Membranas Mediante Computación de Altas Prestaciones con GPU**, por Miguel Á. Martínez del Amor.

PMCGPU

¿Para qué?

- ▶ Problemas **muy** difíciles de resolver (problemas **NP**-completos).
- ▶ Instancias de modelos extremadamente grandes.

Referencias

- ▶ **Aceleración de Simuladores de Sistemas de Membranas Mediante Computación de Altas Prestaciones con GPU**, por Miguel Á. Martínez del Amor. Donde se indican las particularidades de cada uno de los modelos implementados.
- ▶ **Sitio web de PMCGPU**, para descarga, instalación, modelos, etc.