

IV Jornadas de Automática Marítima
21-22 de Octubre de 2010

Plataforma de simulación para cooperación de vehículos marinos

Alejandro Moreno, David Moreno, Diccino Chaos y Joaquín
Aranda
Departamento de Informática y Automática
(UNED)

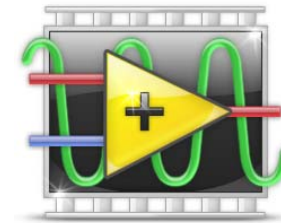
UNED

The logo for 'automar' features the word in a stylized, lowercase font with a blue wave graphic above it. Below the main text, there is a small line of text: 'COOPERACIÓN E INVESTIGACIÓN EN AUTOMÁTICA Y ROBOTICA PARA LA INGENIERIA MARITIMA'.

automar
COOPERACIÓN E INVESTIGACIÓN EN AUTOMÁTICA Y ROBOTICA PARA LA INGENIERIA MARITIMA

Plataforma de simulación para cooperación de vehículos marinos

- * DEVS
 - * (Formalismo de M&S)
- * LabVIEW
 - * (Plataforma de Desarrollo)
- * UMV's
 - * (Objetivo y Caso Práctico)



DEVS

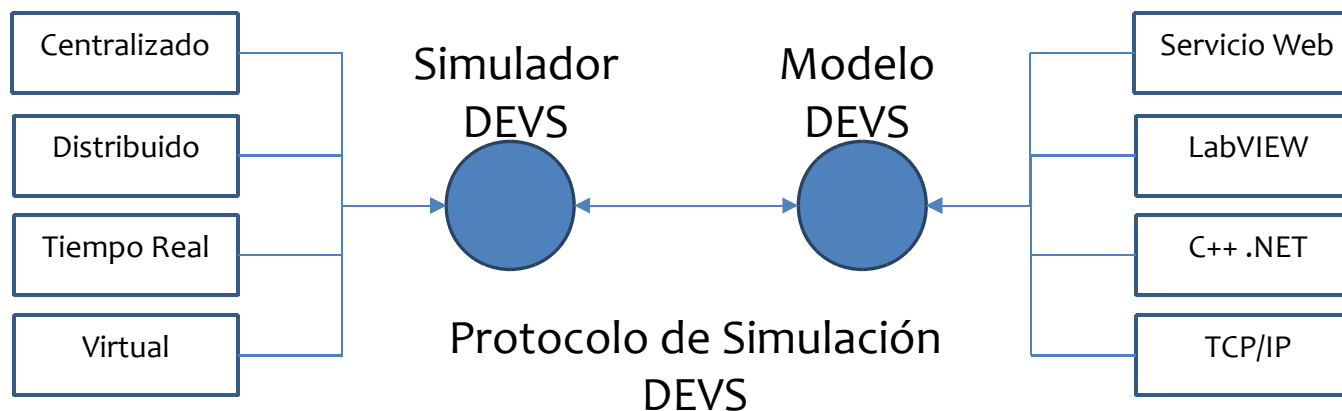
(Discrete Event System Specification)

- * Proporciona ventajas para analizar y diseñar sistemas complejos: completitud, verificabilidad, extensibilidad, y mantenibilidad.
- * Reproduce Especificaciones de Sistemas de Tiempo Discreto (DTSS) y aproxima paradigmas de modelado continuo (DESS).
- * Describe sistemas de eventos discretos y continuos.

DEVS

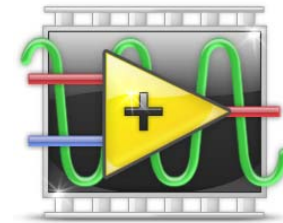
(Discrete Event System Specification)

- * Separa conceptualmente los modelos del simulador haciendo posible simular el mismo modelo utilizando distintos simuladores trabajando en modos de ejecución centralizado, paralelo y distribuido.



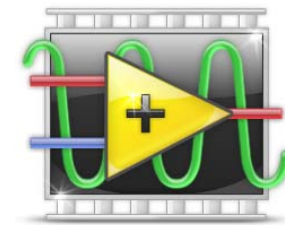
LabVIEW

- * Multitud de disciplinas de ingeniería y científicas, tanto profesionalmente como en el mundo académico, han adoptado LabVIEW.
- * Herramientas gráficas y textuales para el procesamiento digital de señales.
- * **Visualización y manejo de gráficas con datos dinámicos.**
- * Tiempo Real.
- * **Manejo de Interfaces de Comunicaciones:**
 - * USB, Bluetooth, TCP/IP, Servicios Web, etc...



LabVIEW

- * **Integración de Hardware (HIL).**
- * **Análisis y Procesado de Señales Avanzado.**
- * **Programación en MultiCore o Distribuida.**
- * **Almacenamiento de Datos e Informes**
- * **Múltiples enfoques de Programación**
 - * **Código C, MathScript, Flujo de Datos, Simulación y Diagramas de Estados.**
- * **Capacidad para interactuar con otros lenguajes, aplicaciones y paradigmas (Interoperabilidad)**



Simulador

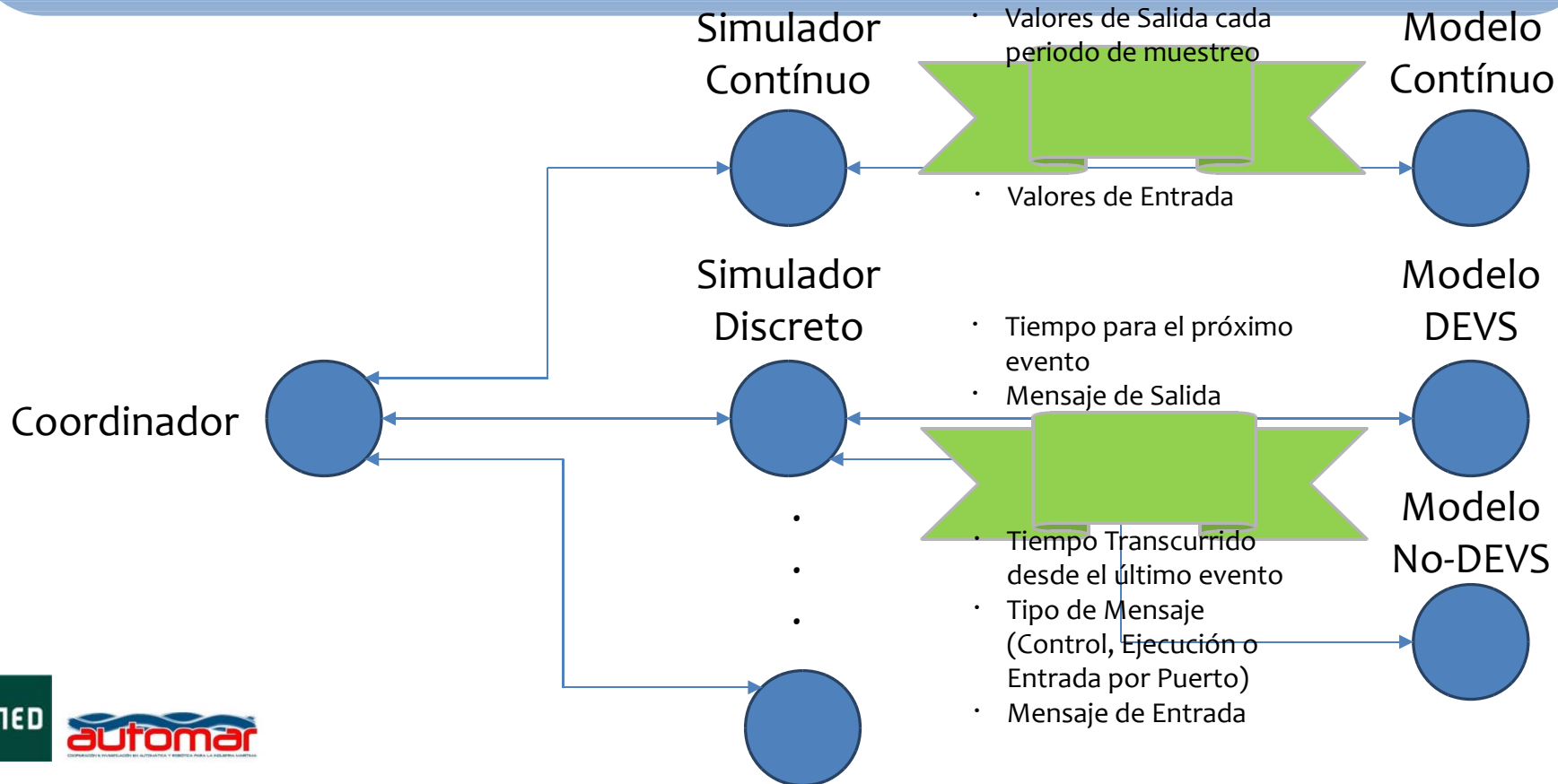
- * **El simulador permite:**
 - * Realizar simulaciones basadas en eventos discretos (DEVS y No-DEVS) junto con simulaciones continuas.
 - * De forma centralizada, distribuida, remota en tiempo real o en tiempo virtual.
- * **Índice:**
 - * Arquitectura del Simulador.
 - * Protocolo de Comunicación.
 - * Flujo de Datos.

Arquitectura

Arquitectura

- * El coordinador sincroniza y comunica los simuladores que pueden simular modelos discretos, muestrear un modelo continuo u otra acción.
- * Un simulador de eventos discretos puede simular un modelo DEVS o un modelo No-DEVS.
 - * Un modelo No-DEVS puede realizar cualquier acción de forma discreta, como la comunicación con Hardware o remota mediante WS, TCP/IP, etc.
- * Un simulador continuo muestrea un modelo continuo a intervalos discretos.

Protocolo de Comunicación



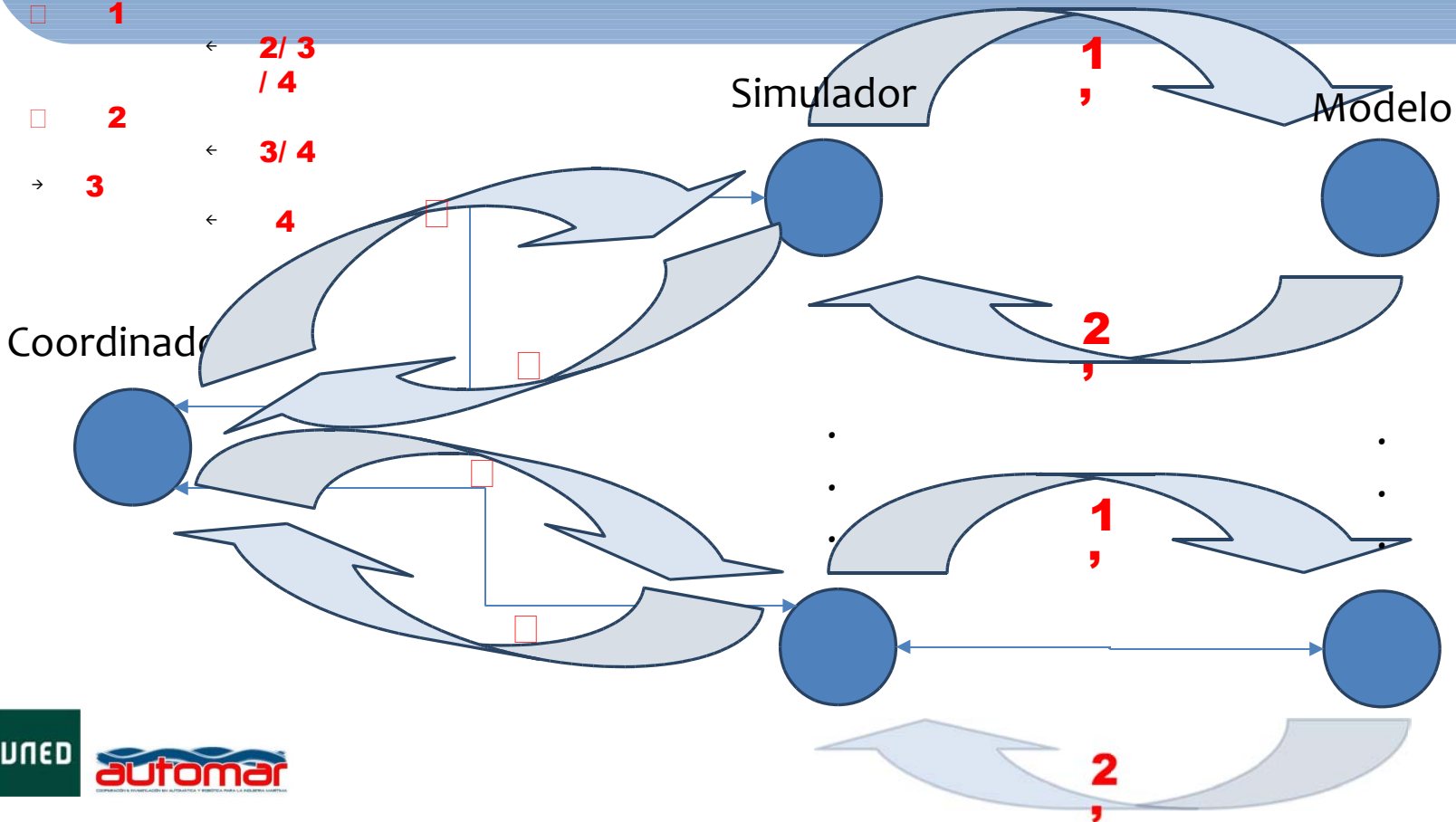
Protocolo de Comunicación

- * El coordinador se comunica con los simuladores para enviarles mensajes de entrada, de ejecución por tiempo o de control por medio de un mensaje y tiempo de evento asociado.
- * Los simuladores se comunican con el coordinador para enviarle mensajes de salida con el tiempo del próximo evento.

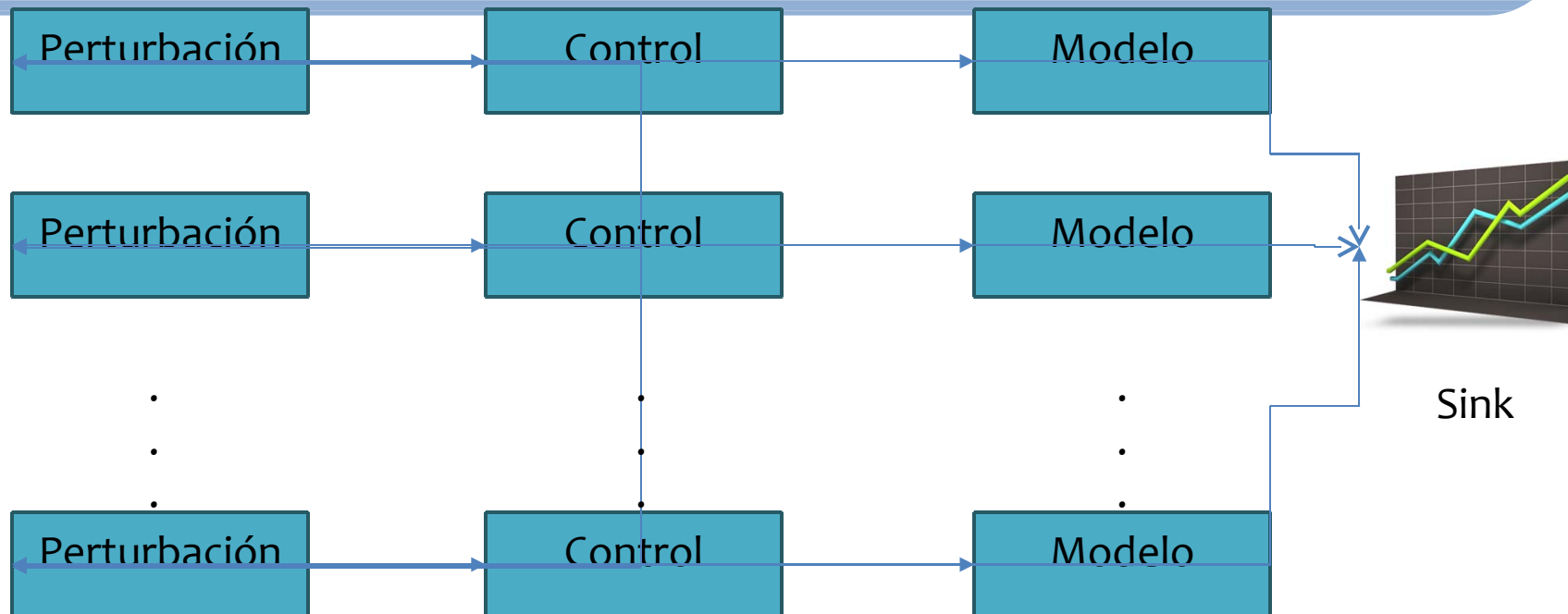
Diagrama de Flujo

Posibles Ordenes de Ejecución

- **1** ← **2/3**
/ **4**
- **2** ← **3/4**
- **3** ← **4**



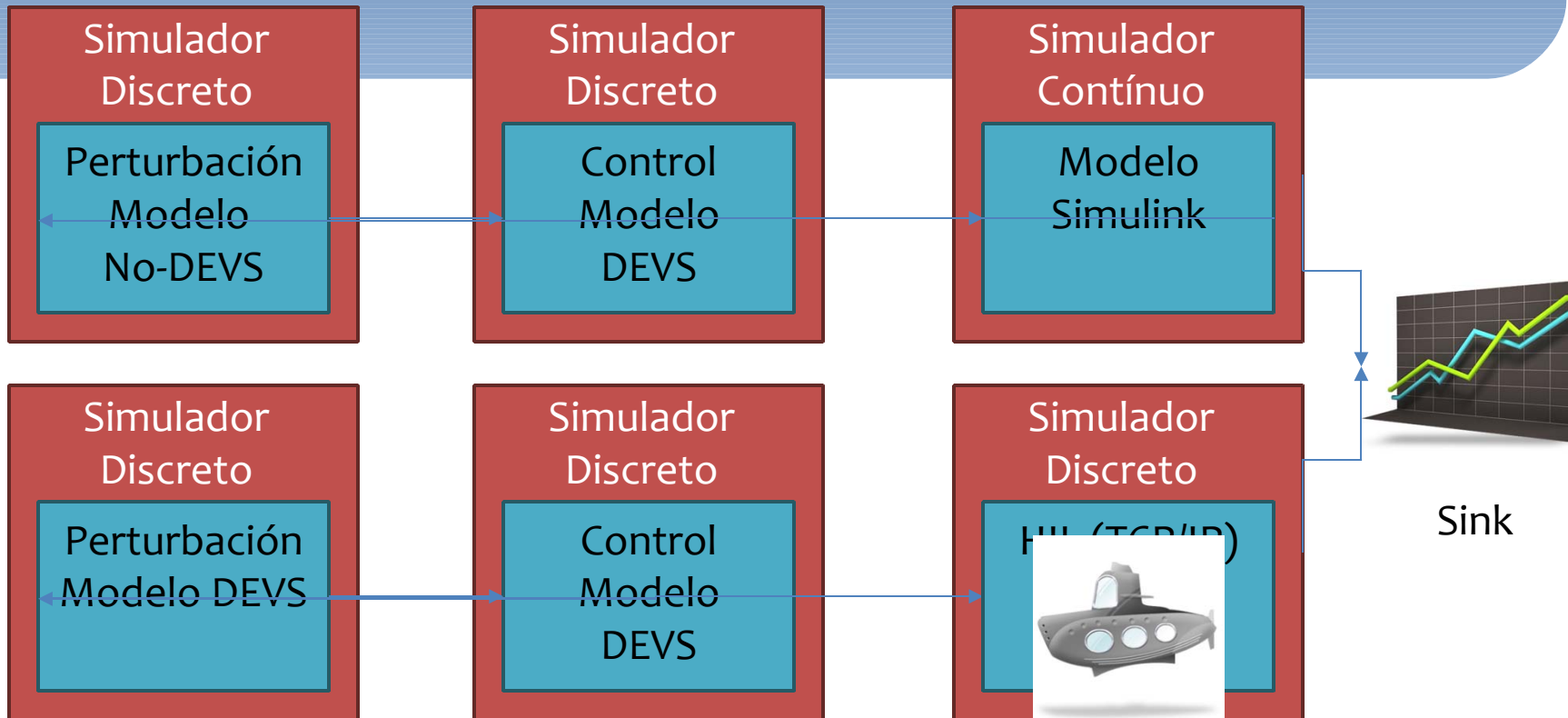
Modelado de Sistemas de Cooperación



Modelado de Sistemas de Cooperación

- * Perturbación:
 - * Introduce perturbaciones en las comunicaciones y limita el campo de visibilidad del modelo.
- * Control:
 - * Realiza un control basado en la cooperación.
- * Modelo:
 - * Modela la entidad a simular.

Posibles Casos



Posibles Casos

- * Tanto la perturbación, como el control o el modelo se pueden definir mediante un modelo DEVS, No-DEVS o Continuo.
- * De esta forma podemos incluir en la simulación Hardware, comunicaciones externas o con otros simuladores ajenos a la plataforma.

Objetivos

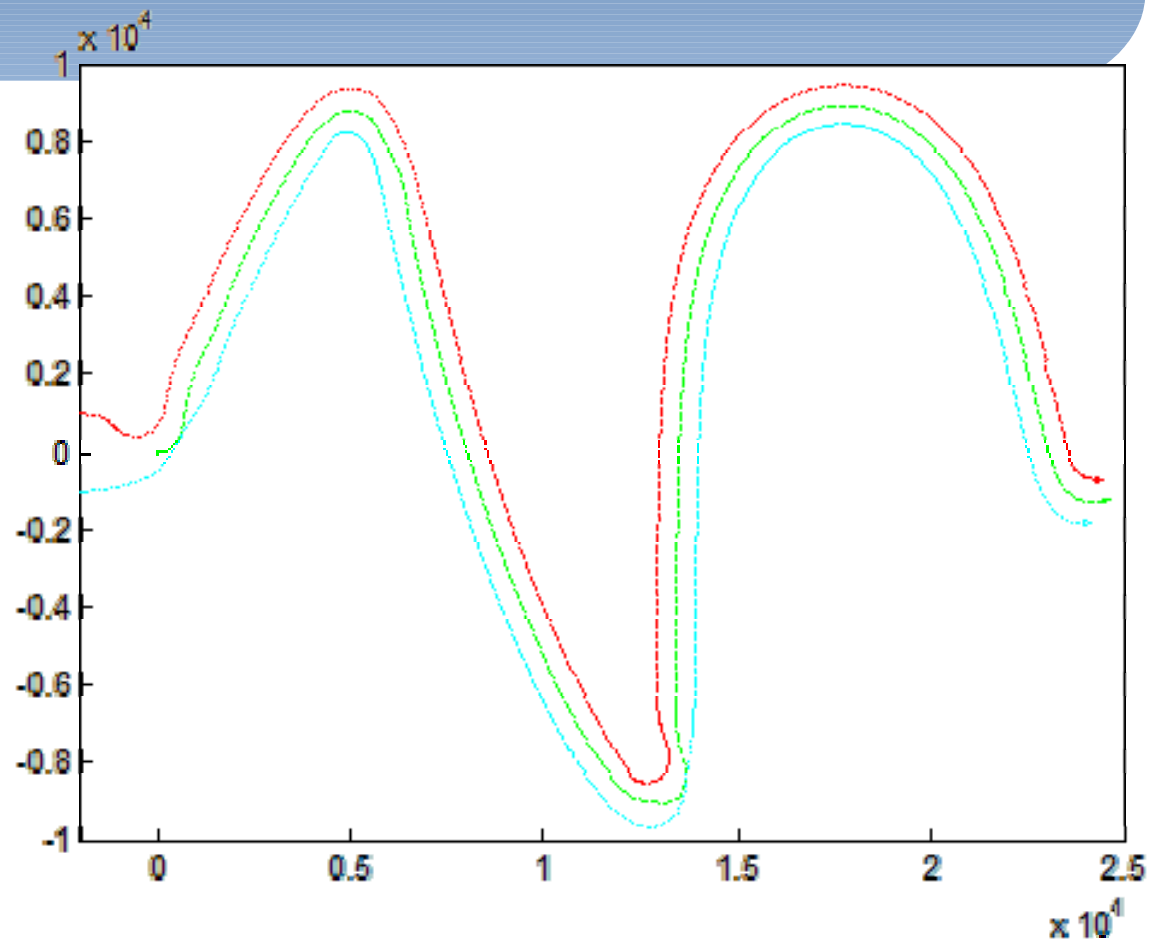
- * El usuario pueda:
 - * Realizar simulaciones en tiempo real con modelos simulados y reales (HIL).
 - * Seleccionar, parametrizar cada componente o implementarlo mediante el paradigma que desee (Simulación, HIL, Simulink, etc...).
 - * Aplicar al mismo modelo distintos controles y perturbaciones o vice-versa.
 - * Visualizar y almacenar los resultados dinámicamente.

Objetivos

- * Interfaz gráfica usable para la edición rápida y dinámica de escenarios.
- * Banco de Pruebas para experimentos de cooperación de vehículos.

Ejemplo Cooperación

- * Path Following Coordinado
- * Un Lider.
- * Dos Seguidores.



¿Preguntas?

Gracias.....