

# Evaluación de un prototipo de simulador simil FIRA, desarrollado con la herramienta Phi

Gustavo Armagno  
Facundo Benavides  
Claudia Rostagnol

8 de noviembre de 2005



Facultad de Ingeniería  
Universidad de la República Oriental del Uruguay

## Resumen

El propósito de este documento es evaluar un prototipo de simulador simil FIRA, desarrollado con una herramienta de simulación denominada Phi. El resultado de esta evaluación podría determinar la utilización de este simulador como alternativa al simulador de la *Middle League Simurosot* proporcionado por la FIRA.

# Índice

<b>1. Introducción</b>	<b>4</b>
<b>2. Análisis</b>	<b>4</b>
<b>3. Prototipo</b>	<b>7</b>
<b>4. Conclusiones</b>	<b>10</b>

## 1. Introducción

En la búsqueda de un simulador que extienda las funcionalidades del proporcionado por la FIRA, fue evaluado un *framework* para desarrollar simuladores denominado Phi. La elección de un simulador alternativo se basa en que el mismo cuente con cierto número de características que faciliten y potencien el aprendizaje automático de los equipos. Entre ellas se destacan:

- Que la simulación se ajuste lo más posible al modelo de la FIRA.
- La posibilidad de generar un archivo de registro con el estado completo de la simulación.
- La capacidad de controlar el paso de la simulación, con la finalidad de potenciar el aprendizaje.
- La interoperabilidad entre el simulador y las estrategias desarrolladas en código Java por el equipo FRUTo.
- La posibilidad de controlar el simulador de manera programática (sin que intermedie un operador).

Phi (disponible en <http://www.basso.inf.br/phi/>) es una herramienta de código abierto (licenciada bajo GPL) que permite construir simuladores de robots. A partir de los ejemplos proporcionados por la aplicación y gracias a la colaboración de Eduardo W. Basso (estudiante de maestría en inteligencia artificial en la Universidade Federal do Rio Grande do Sul), fue posible prototipar un simulador simil FIRA, parametrizable, con la finalidad de evaluar su potencial aplicación en el proyecto.

El presente documento está dividido en tres secciones:

- **Análisis.** Analiza las razones por las cuales se tomó la decisión de construir un simulador utilizando la herramienta Phi, así como también las fortalezas y debilidades de la misma frente a los motores de simulación del *Middle League Simurosot* de la FIRA y GSim del equipo FRUTo.
- **Prototipo.** Resume los desafíos que debieron ser sorteados para construir el prototipo, a partir de los ejemplos existentes.
- **Conclusiones.** A partir del prototipo desarrollado se evalúa la viabilidad de utilizarlo en el proyecto.

## 2. Análisis

### Fortalezas

- La simulación se realiza utilizando el modelo físico proporcionado por la librería ODE (disponible en <http://www.ode.org>). Esta librería está desarrollada bajo la licencia GPL y cuenta con buena documentación.
- Proporciona una interfaz gráfica en 3D y permite posicionar el punto de vista en cualquier punto del mundo. Esto permite realizar observaciones precisas sobre cómo se están comportando los cuerpos simulados, y así poder corregir deficiencias en el modelo.

- Habilita la interoperabilidad entre aplicaciones externas y el simulador, utilizando un mecanismo de comunicación a través de *sockets* y basado en eventos.
- Los simuladores ejecutados por esta herramienta están escritos en un lenguaje de *scripting* denominado Lua. La ventaja de esto es que los errores en el modelo pueden corregirse rápidamente y sin necesidad de recompilar el simulador.
- Es posible controlar el paso de la simulación.
- Es posible generar un archivo de registro de la simulación.
- Es distribuida junto con algunos ejemplos; entre ellos, una primera aproximación al simulador de la FIRA.
- Se cuenta con el código fuente de la herramienta lo cual, en caso de ser necesario, permitiría ampliar la funcionalidad de la misma. Existe, además, una fluida comunicación con los desarrolladores.
- El sitio Web del proyecto Phi cuenta con un foro cuya finalidad es evacuar interrogantes, así como compartir experiencias y aportes de los usuarios.

## Debilidades

- La versión actual no permite sincronizar el simulador con las aplicaciones externas. La sincronización es un elemento particularmente importante a tener en cuenta, ya que el simulador de la FIRA espera a que cada estrategia haya terminado de procesar el estado del juego para continuar con la simulación.
- De acuerdo al sitio Web de la herramienta, la versión actual del protocolo de comunicación dista de ser robusta y eficiente.
- Bajo el sistema operativo Windows, no está garantizado que la herramienta funcione correctamente.
- Por el momento, la documentación existente es pobre. Sólo se cuenta con la lista de algunas funciones del lenguaje Lua.

## Comparación entre los distintos simuladores

La figura 1 compara ciertas características entre los simuladores FIRA, GSim y la herramienta Phi (es importante recordar que Phi no es un simulador, sino que es un *framework* para desarrollar simuladores). Estas características resultan importantes para la evaluación del simulador, ya que pueden determinar la elección del mismo.

	FIRA	GSim	Phi
Modelo basado en leyes físicas	V	X	V
Modelo completo	V	X	V
Interfaz 3D	V	X	V
Logging	X	X	V
Estrategias Java-based	V (*)	V	V (*)
Control programático	X	V	V
Eficiencia en la interacción	V	V	X
Adaptabilidad	X	V (**)	V
Control del paso de simulación	X	V	V
Código fuente	X	V	V
Documentación	X	V	V (***)
Multi-plataforma	X	V	V
Simulación sincrónica	V	V	X
Simulación asincrónica	X	V	V

- (\*) La comunicación entre las estrategias y el simulador se realiza a través de sockets.
- (\*\*) Permite configurar fácilmente algunos parámetros del simulador (datos del campo de juego, posición de los robots, posición de la pelota, etc.). Sin embargo, no es posible modificar las características de los cuerpos simulados sin recompilar el código.
- (\*\*\*) Están documentadas las funciones del lenguaje Lua y se proporcionan algunos ejemplos. El sitio Web de Phi cuenta con un foro de discusión en actividad.

Figura 1: Comparación de algunas características (filas) entre los simuladores FIRA, GSim y la herramienta Phi (columnas). La “X” indican la ausencia y la “V” la presencia de una determinada característica en el correspondiente simulador.

Como se mencionó anteriormente, uno de los hitos importantes del proyecto es contar con un simulador que mejore las prestaciones del que proporciona la FIRA. El simulador GSim posee características interesantes, entre las que se destacan la posibilidad de configurar algunos parámetros de manera programática, como el estado inicial del juego (posición y velocidad iniciales de los robots y de la pelota), el paso de simulación y la duración del juego. Sin embargo, la versión actual del motor del simulador GSim es incompleta, ya que no contempla colisiones entre robots. Esto representa su principal desventaja, ya que el conocimiento adquirido en el GSim por algunas técnicas de aprendizaje (por citar un ejemplo, una técnica que permita reconocer las jugadas del oponente) podría resultar inútil al trasladarlo al simulador de la FIRA.

La principal debilidad de la versión actual del motor de simulación de Phi es la imposibilidad hacer una pausa en el mismo (lo que se ha dado a llamar *simulación sincrónica*) para esperar que las estrategias terminen de procesar el estado del juego y retornar las velocidades de los robots. Sin embargo, más allá de las características compartidas con el simulador GSim (ver figura 1), las siguientes, presentes en el motor de Phi, avalaron la construcción de un prototipo para evaluar el impacto de desarrollar un simulador simil FIRA completo:

**Modelo basado en leyes físicas** Proporciona una simulación más realista que el modelo del

simulador GSim, basado en reglas geométricas.

**Modelo completo** Contempla todo tipo de colisiones (entre ellas, las colisiones entre robots).

**Interfaz 3D** La proyección en pantalla del modelo tridimensional del mundo facilita la puesta a punto del simulador, ya sea para detectar fallas en el modelo, como para mejorar la similitud con el simulador de la FIRA.

### 3. Prototipo

Para desarrollar un prototipo de simulador simil FIRA utilizando la herramienta Phi, se partió de la base de un *script* de ejemplo que especifica el campo de juego, el cuerpo del robot y la pelota, una función a través de la cual se le asigna velocidad a cada rueda y un ejemplo sobre cómo permitir que una aplicación externa se comunique con el simulador.

Las primeras pruebas fueron realizadas tomando en cuenta algunas observaciones sobre el simulador de ejemplo. A partir de estas observaciones se comenzó a construir un prototipo de simulador que visualmente se aproxime más al proporcionado por la FIRA.

#### Observaciones sobre el ejemplo

##### Dimensiones

Como muestra la figura 2, las dimensiones de la cancha son distintas a las que están definidas para la *Middle League Simurosot* de la FIRA. El croquis muestra en línea continua las dimensiones especificadas en el ejemplo, y en línea punteada las dimensiones del simulador de la FIRA.

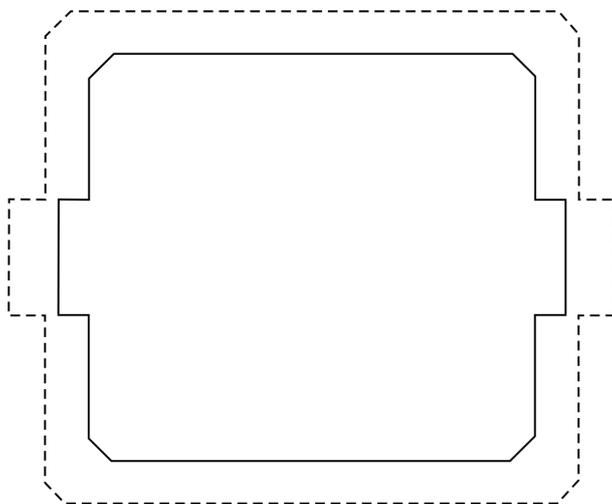


Figura 2: La línea punteada muestra las dimensiones especificadas en las reglas de la *Middle League Simurosot*, mientras que la línea continua corresponde a las dimensiones del ejemplo.

##### Tambaleo

A mayor velocidad, la estabilidad del robot comienza a verse comprometida. En los arranques y las frenadas, el robot se tambalea sobre el eje de las ruedas, llegando a golpearse la parte anterior

y posterior contra el suelo (ver figura 3).

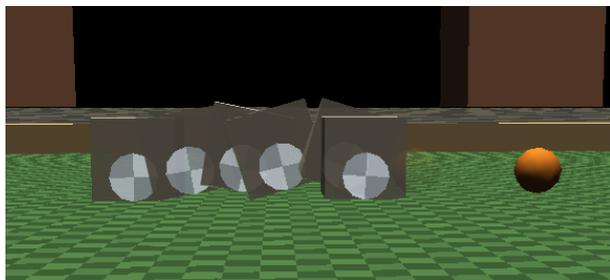


Figura 3: La foto estroboscópica muestra el tambaleo del robot a velocidades relativamente altas.

Como muestra la figura 4 este efecto suele provocar vuelcos, generalmente, cuando el robot choca contra una pared.

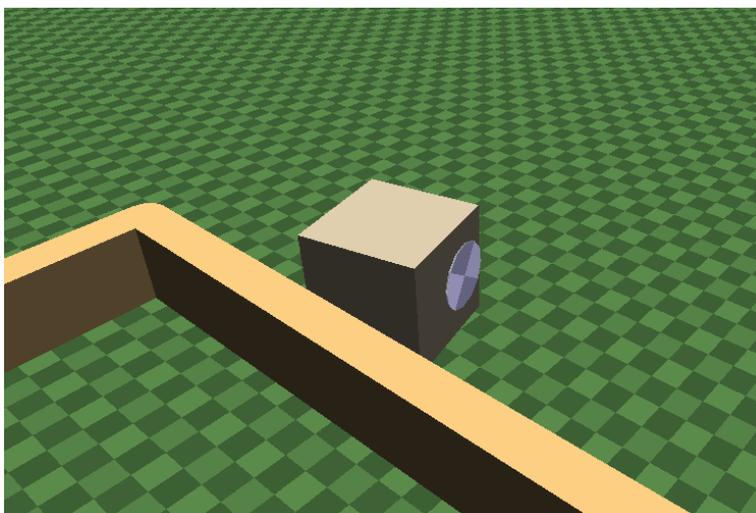


Figura 4: El tambaleo puede provocar el vuelco del robot.

Vale la pena recalcar la importancia de que el simulador genere una proyección tridimensional de lo que está ocurriendo: probar un modelo físico en un entorno bidimensional resultaría muy costoso.

### **Elevación de la pelota**

Se ha observado que el robot puede hacer despegar la pelota del suelo al encerrarla contra una pared. Esto puede deberse a que la presión entre el robot y la pared sobre la pelota haga que la misma escape a gran velocidad en dirección vertical. En ocasiones, la pelota se escapa de los límites de la cancha.

## Cantidad de jugadores

En el ejemplo interactúa un único robot con la pelota, lo cual podría estar impidiendo detectar problemas de desempeño del simulador.

## Características del prototipo

- Permite parametrizar algunos componentes del juego, como por ejemplo las dimensiones de la cancha, el tamaño y el peso de los robots y la pelota, la posición inicial de éstos, etc.
- Se corrigieron las dimensiones de la cancha, para que coincidan con las del simulador de la FIRA.
- Se le agregaron cuatro puntos de apoyo de libre movilidad (esferas unidas por una articulación en forma de bola) en los extremos de la cara inferior del robot (ver figura 5). Es posible desactivar este soporte extra, modificando un archivo de configuración.
- Se elevaron un poco las paredes para evitar que la pelota escape de la cancha.
- Se agregaron dos equipos de cinco robots cada uno, cuyas posiciones iniciales coinciden con las del simulador GSim (ver figura 6).
- Se desarrollaron las clases necesarias para comunicar una aplicación Java con el simulador.

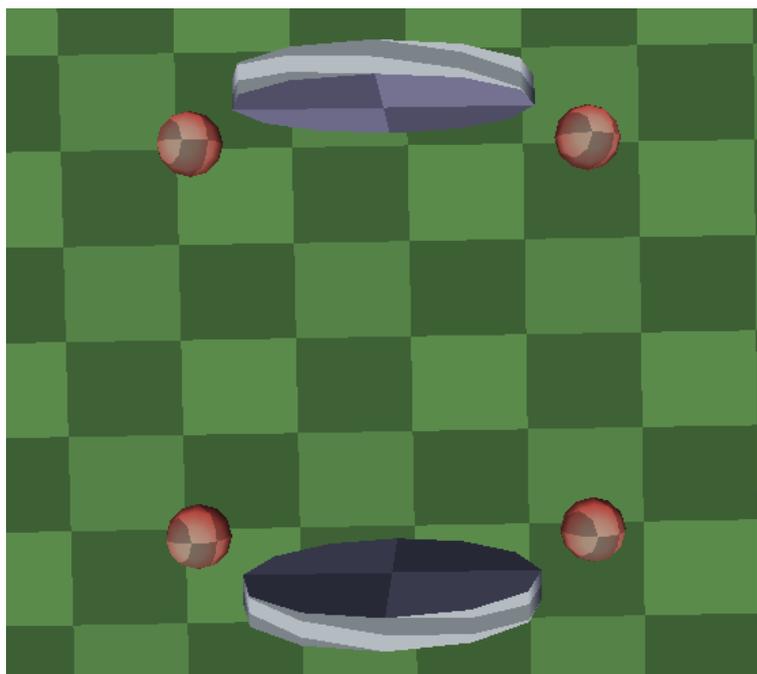


Figura 5: Corte horizontal del robot sin el chasis, muestra los cuatro soportes esféricos de libre movilidad.

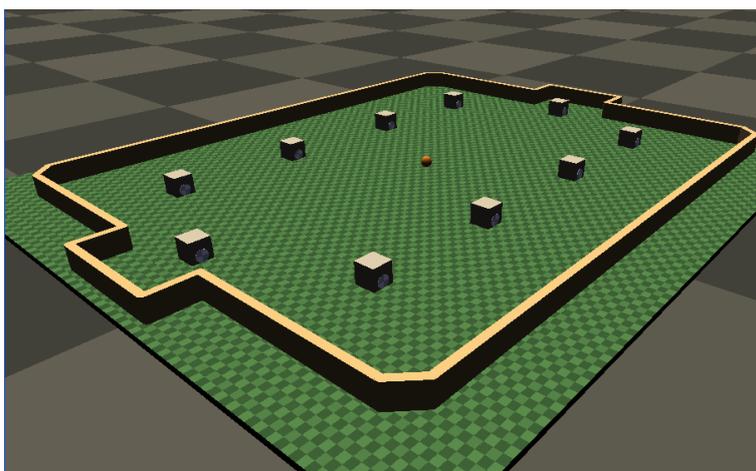


Figura 6: Equipos izquierdo y derecho completos, ubicados en las posiciones reglamentarias.

### Observaciones al prototipo

- La nueva configuración del cuerpo del robot reduce considerablemente los tambaleos, aunque aún puede apreciarse una pequeña vibración en los arranques y frenadas.
- No se observaron vuelcos.
- Rara vez la pelota se eleva. Sin embargo, al ser presionada contra la pared puede llegar a ser despedida fuera de los límites del campo de juego.
- La visualización del juego con diez jugadores en la cancha empeora considerablemente el desempeño del simulador (no se realizaron pruebas de generación de un archivo de registro como única salida).

## 4. Conclusiones

La principal desventaja de la versión actual de la herramienta Phi es la imposibilidad de sincronizar el envío y la recepción de información. Esta característica es importante, ya que el simulador de la FIRA espera a que ambas estrategias procesen el estado del juego y retornen las velocidades de los robots para continuar con la simulación. Debe evaluarse el impacto de modificar la herramienta Phi para que permita sincronizar el simulador con las estrategias.

A diferencia del simulador GSim, Phi cuenta con un modelo de simulación basado en leyes físicas el cual, por ejemplo, contempla choques entre robots. Esto resulta interesante, ya que implicaría un ahorro de tiempo considerable ante la necesidad de modificar el modelo de simulación del GSim. Además, la posibilidad de visualización tridimensional permite detectar fácilmente errores en el modelo.

Por último, algo que no puede perderse de vista es que el proyecto Phi está en plena actividad, y existe interés por parte de los desarrolladores de que los usuarios envíen sus comentarios para corregir fallas y extender las funcionalidades.

Las oportunidades de utilizar el simulador simil FIRA ejecutado por Phi son auspiciosas. Sin embargo, aún debe desarrollarse el *middleware* necesario para poder conectar efectivamente las estrategias desarrolladas por el equipo FRUTo con el simulador.