Simulación de Sistemas Continuos y a Tramos

Prof. Dr. François E. Cellier
Institut für Computational Science
ETH Zürich

28 de junio 2007

Introducción

Todos los métodos de la simulación que encontramos hasta ahora trabajan de una forma u otra con *extrapolaciones polinomiales* usando *series de Taylor*.

Introducción

Todos los métodos de la simulación que encontramos hasta ahora trabajan de una forma u otra con *extrapolaciones polinomiales* usando *series de Taylor*.

Una propiedad fundamental de los *polinomios* es que *no tienen discontinuidades*.

Introducción

Todos los métodos de la simulación que encontramos hasta ahora trabajan de una forma u otra con *extrapolaciones polinomiales* usando *series de Taylor*.

Una propiedad fundamental de los *polinomios* es que *no tienen discontinuidades*.

En consecuencia, si un modelo contiene discontinuidades, no puede simularse a través de dichas discontinuidades usando extrapolaciones polinomiales.

Introducción

Todos los métodos de la simulación que encontramos hasta ahora trabajan de una forma u otra con *extrapolaciones polinomiales* usando *series de Taylor*.

Una propiedad fundamental de los *polinomios* es que *no tienen discontinuidades*.

En consecuencia, si un modelo contiene discontinuidades, no puede simularse a través de dichas discontinuidades usando extrapolaciones polinomiales.

La gran mayoría de los *sistemas de la ingeniería tienen muchas discontinuidades* que tenemos que incluir en sus modelos.

Introducción

Todos los métodos de la simulación que encontramos hasta ahora trabajan de una forma u otra con *extrapolaciones polinomiales* usando *series de Taylor*.

Una propiedad fundamental de los *polinomios* es que *no tienen discontinuidades*.

En consecuencia, si un modelo contiene discontinuidades, no puede simularse a través de dichas discontinuidades usando extrapolaciones polinomiales.

La gran mayoría de los *sistemas de la ingeniería tienen muchas discontinuidades* que tenemos que incluir en sus modelos.

Los métodos de integración numérica de sistemas dinámicos que desarrollamos hasta ahora no sirven para la simulación de *sistemas híbridos*. Se necesita algo mejor.

Abusando del Control del Paso

¿Qué pasa si cerramos los ojos e integramos a través de una discontinuidad del modelo usando unos de los algoritmos con control del paso introducidos antes?

Abusando del Control del Paso

¿Qué pasa si cerramos los ojos e integramos a través de una discontinuidad del modelo usando unos de los algoritmos con control del paso introducidos antes?

El algoritmo no sabe que existe una discontinuidad. Lo que nota es un *cambio rápido en la trayectoria*.

Abusando del Control del Paso

¿Qué pasa si cerramos los ojos e integramos a través de una discontinuidad del modelo usando unos de los algoritmos con control del paso introducidos antes?

El algoritmo no sabe que existe una discontinuidad. Lo que nota es un *cambio rápido en la trayectoria*.

El algoritmo piensa que *apareció un nuevo autovalor* muy a la izquierda en el plano complejo.

Abusando del Control del Paso

¿Qué pasa si cerramos los ojos e integramos a través de una discontinuidad del modelo usando unos de los algoritmos con control del paso introducidos antes?

El algoritmo no sabe que existe una discontinuidad. Lo que nota es un *cambio rápido en la trayectoria*.

El algoritmo piensa que *apareció un nuevo autovalor* muy a la izquierda en el plano complejo.

En consecuencia, el algoritmo *reduce el paso* para apresar este autovalor dentro de su *dominio de la precisión*.

Abusando del Control del Paso

¿Qué pasa si cerramos los ojos e integramos a través de una discontinuidad del modelo usando unos de los algoritmos con control del paso introducidos antes?

El algoritmo no sabe que existe una discontinuidad. Lo que nota es un *cambio rápido en la trayectoria*.

El algoritmo piensa que *apareció un nuevo autovalor* muy a la izquierda en el plano complejo.

En consecuencia, el algoritmo *reduce el paso* para apresar este autovalor dentro de su *dominio de la precisión*.

Sin embargo, el autovalor es un guasón. No se deja apresar. Aunque el paso se reduzca tanto como se quiera, el autovalor se queda fuera del dominio.

Abusando del Control del Paso II

¿Cuándo acaba la iteración?

Abusando del Control del Paso II

¿Cuándo acaba la iteración?

La reducción del paso continúa hasta que suceda una de dos cosas:

Abusando del Control del Paso II

¿Cuándo acaba la iteración?

La reducción del paso continúa hasta que suceda una de dos cosas:

- El tamaño del paso llega a su *valor mínimo* especificado en el algoritmo de control.

Abusando del Control del Paso II

¿Cuándo acaba la iteración?

La reducción del paso continúa hasta que suceda una de dos cosas:

- El tamaño del paso llega a su *valor mínimo* especificado en el algoritmo de control.
- El control del paso decide que la *precisión de la integración es aceptable*. Eso puede ocurrir porque con la reducción del paso, los términos no lineales en la expansión de Taylor pierden importancia. Con un paso suficientemente pequeño *cada algoritmo se comporta como el algoritmo de Euler*.

Abusando del Control del Paso II

¿Cuándo acaba la iteración?

La reducción del paso continúa hasta que suceda una de dos cosas:

- El tamaño del paso llega a su *valor mínimo* especificado en el algoritmo de control.
- El control del paso decide que la *precisión de la integración es aceptable*. Eso puede ocurrir porque con la reducción del paso, los términos no lineales en la expansión de Taylor pierden importancia. Con un paso suficientemente pequeño *cada algoritmo se comporta como el algoritmo de Euler*.

Una vez que la discontinuidad pertenece al pasado, el autovalor bromista desaparece nuevamente.

Abusando del Control del Paso II

¿Cuándo acaba la iteración?

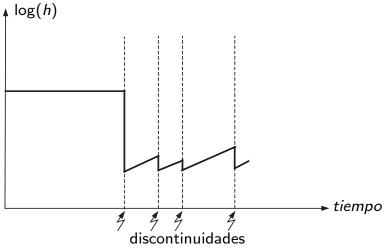
La reducción del paso continúa hasta que suceda una de dos cosas:

- El tamańo del paso llega a su *valor mínimo* especificado en el algoritmo de control.
- El control del paso decide que la *precisión de la integración es aceptable*. Eso puede ocurrir porque con la reducción del paso, los términos no lineales en la expansión de Taylor pierden importancia. Con un paso suficientemente pequeño *cada algoritmo se comporta como el algoritmo de Euler*.

Una vez que la discontinuidad pertenece al pasado, el autovalor bromista desaparece nuevamente.

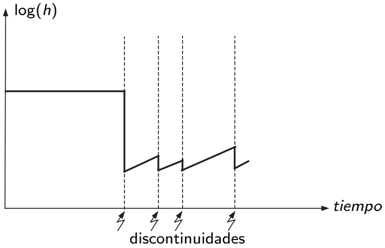
El algoritmo de control del paso lentamente empieza aumentar el tamaño del paso hasta que llega al valor óptimo . . . o hasta que se encuentra la próxima discontinuidad.

Abusando del Control del Paso III



El abuso del control del paso a menudo funciona bastante bien y es por esa razón que algunos de los entornos de modelado y simulación de sistemas dinámicos no ofrecen ningún algoritmo especial para el tratamiento de discontinuidades.

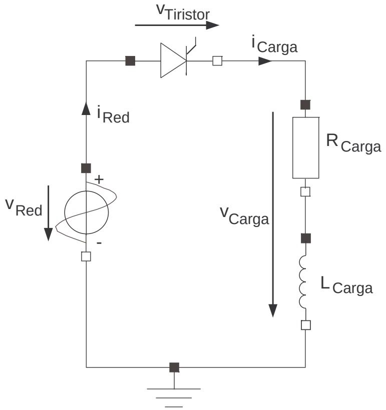
Abusando del Control del Paso III



El abuso del control del paso a menudo funciona bastante bien y es por esa razón que algunos de los entornos de modelado y simulación de sistemas dinámicos no ofrecen ningún algoritmo especial para el tratamiento de discontinuidades.

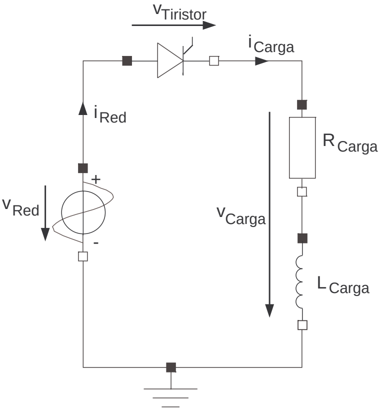
Sin embargo, el método no es eficiente y a veces no funciona correctamente.

El Control de la Velocidad de Trenes



En Suiza, los trenes operan en una *red de corriente alterna* de $16\frac{2}{3}\,\mathrm{Hz}$.

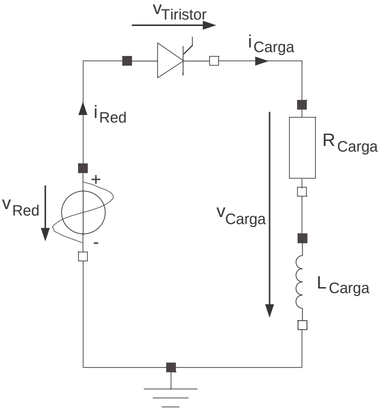
El Control de la Velocidad de Trenes



En Suiza, los trenes operan en una *red de corriente alterna* de $16\frac{2}{3}\,\mathrm{Hz}$.

Para el *control de la velocidad* se usan circuitos eléctricos con tiristores (rectificadores controlados).

El Control de la Velocidad de Trenes

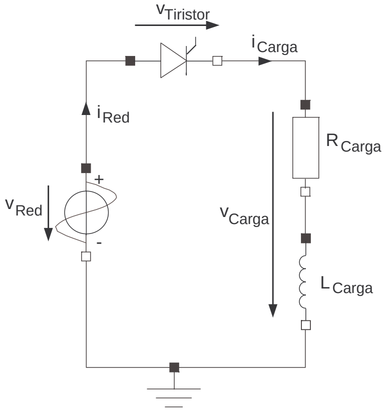


En Suiza, los trenes operan en una *red de corriente alterna* de $16\frac{2}{3}\,\mathrm{Hz}$.

Para el *control de la velocidad* se usan circuitos eléctricos con tiristores (rectificadores controlados).

El circuito más simple se muestra a la izquierda. El resistor con un inductor en serie representa la carga (el tren). El tiristor bloquea la corriente negativa (funcionando como un diodo) y bloquea una parte adicional de cada período.

El Control de la Velocidad de Trenes



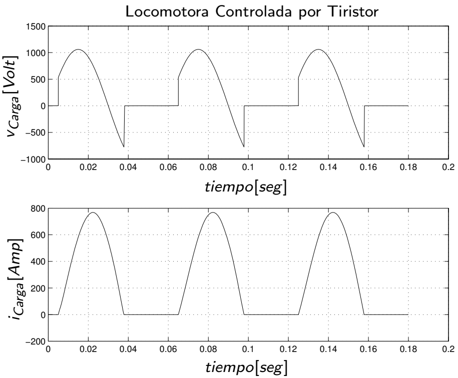
En Suiza, los trenes operan en una *red de corriente alterna* de $16\frac{2}{3}\mathrm{Hz}$.

Para el *control de la velocidad* se usan circuitos eléctricos con tiristores (rectificadores controlados).

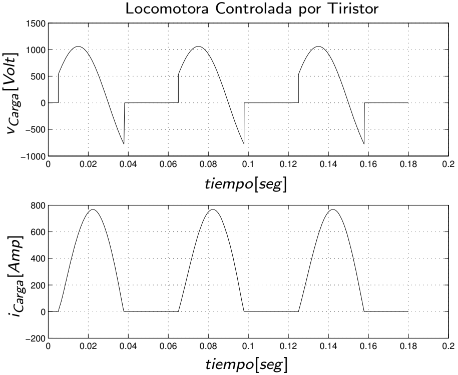
El circuito más simple se muestra a la izquierda. El resistor con un inductor en serie representa la carga (el tren). El tiristor bloquea la corriente negativa (funcionando como un diodo) y bloquea una parte adicional de cada período.

El porcentaje del período no bloqueado se controla por un *ángulo de encendido*.

El Control de la Velocidad de Trenes II

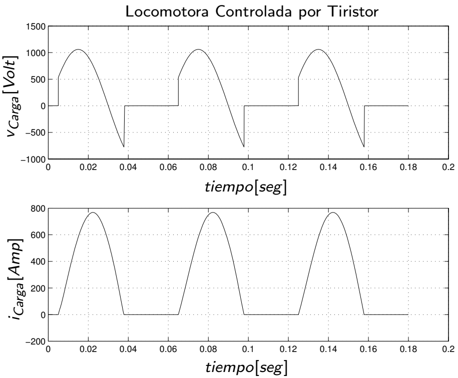


El Control de la Velocidad de Trenes II



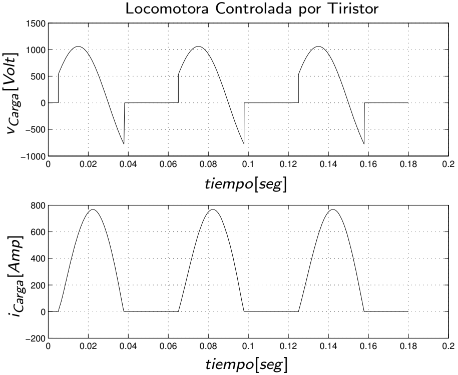
- Los resultados de la simulación muestran discontinuidades.

El Control de la Velocidad de Trenes II



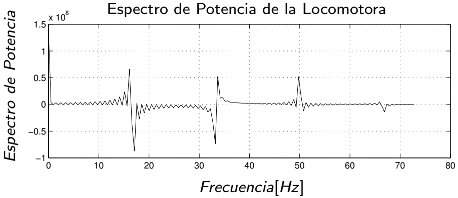
- Los resultados de la simulación muestran discontinuidades.
- La potencia eléctrica $P = v_{\text{Carga}} \cdot i_{\text{Carga}}$ depende del *ángulo de encendido*.

El Control de la Velocidad de Trenes II

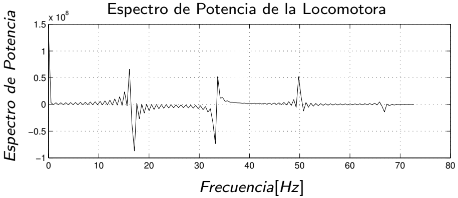


- Los resultados de la simulación muestran discontinuidades.
- La potencia eléctrica $P = v_{\mathrm{Carga}} \cdot i_{\mathrm{Carga}}$ depende del *ángulo de encendido*.
- En la simulación se usó un ángulo de encendido de 30^{o}. La velocidad del tren disminuye con ángulos más grandes.

El Control de la Velocidad de Trenes III

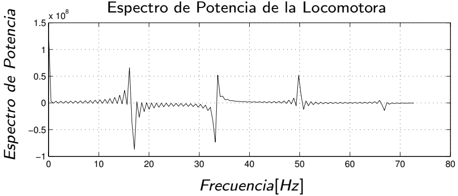


El Control de la Velocidad de Trenes III



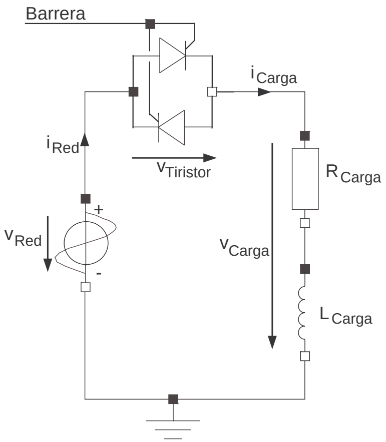
- Hay mucha potencia en el *tercer armónico* que se encuentra a 50Hz.

El Control de la Velocidad de Trenes III



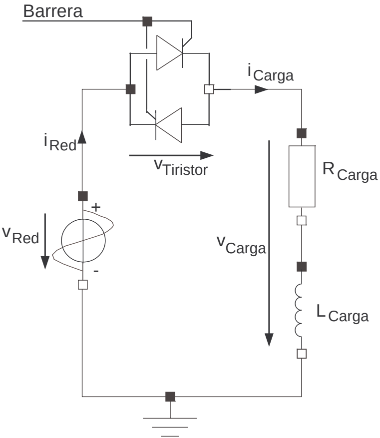
- Hay mucha potencia en el *tercer armónico* que se encuentra a $50\mathrm{Hz}$.
- **Cuando los trenes con tres locomotoras subían lentamente al Gotardo, los contadores de electricidad de las casas próximas a las vías se volvían a cero, lo que no divertía a la compañía eléctrica del Cantón de Uri.**

El Control de la Velocidad de Trenes IV



Durante unos años se usó otro circuito de control de velocidad en los trenes de la línea Zurich-Meilen-Rapperswil.

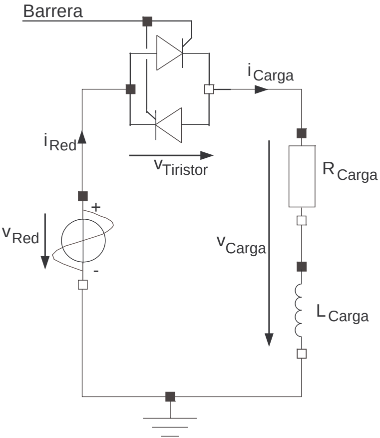
El Control de la Velocidad de Trenes IV



Durante unos años se usó otro circuito de control de velocidad en los trenes de la línea Zurich-Meilen-Rapperswil.

La barrera de tiristores se controlaba de tal manera que cierto número de períodos podía pasar mientras que los otros eran bloqueados.

El Control de la Velocidad de Trenes IV

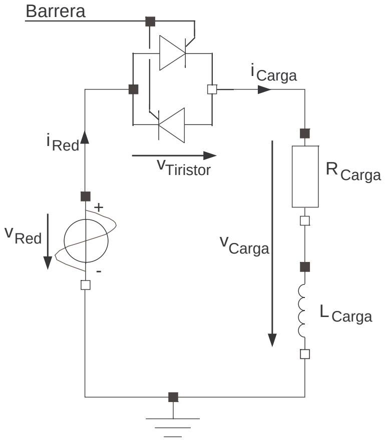


Durante unos años se usó otro circuito de control de velocidad en los trenes de la línea Zurich-Meilen-Rapperswil.

La barrera de tiristores se controlaba de tal manera que cierto número de períodos podía pasar mientras que los otros eran bloqueados.

Se habla de una *estrategia de control por ráfaga*.

El Control de la Velocidad de Trenes IV



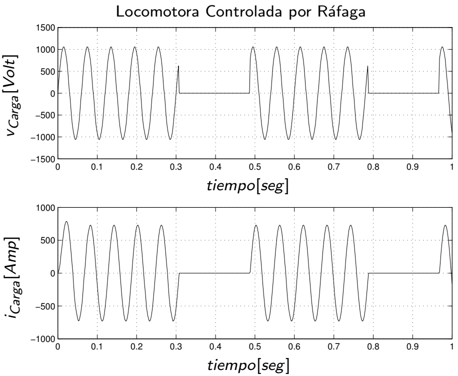
Durante unos años se usó otro circuito de control de velocidad en los trenes de la línea Zurich-Meilen-Rapperswil.

La barrera de tiristores se controlaba de tal manera que cierto número de períodos podía pasar mientras que los otros eran bloqueados.

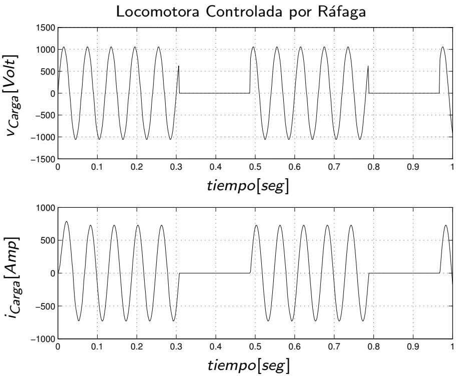
Se habla de una *estrategia de control por ráfaga*.

Se trabajaba con *paquetes de ocho períodos*.

El Control de la Velocidad de Trenes V

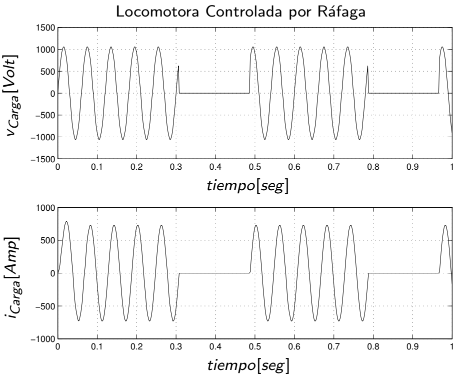


El Control de la Velocidad de Trenes V



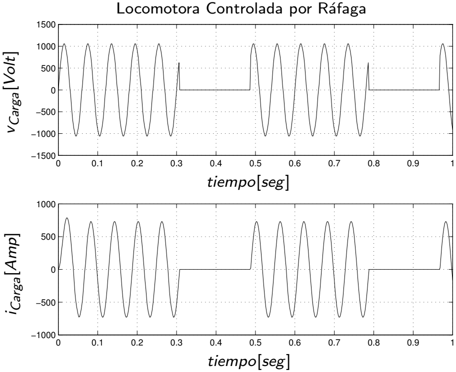
- Los resultados de la simulación muestran discontinuidades menos serias que en el control por tiristor usado antes.

El Control de la Velocidad de Trenes V



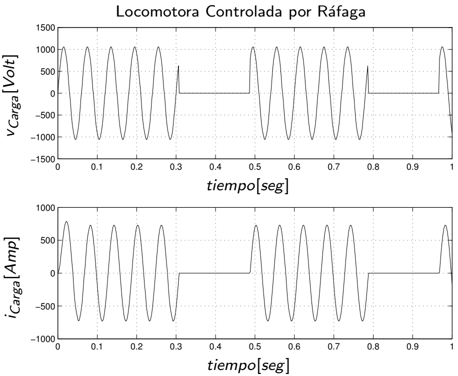
- Los resultados de la simulación muestran discontinuidades menos serias que en el control por tiristor usado antes.
- En consecuencia, esta estrategia de control no afecta la red eléctrica operada a 50Hz.

El Control de la Velocidad de Trenes V



- Los resultados de la simulación muestran discontinuidades menos serias que en el control por tiristor usado antes.
- En consecuencia, esta estrategia de control no afecta la red eléctrica operada a 50Hz.
- Sin embargo, este tipo de control solamente permite ocho velocidades distintas.

El Control de la Velocidad de Trenes V

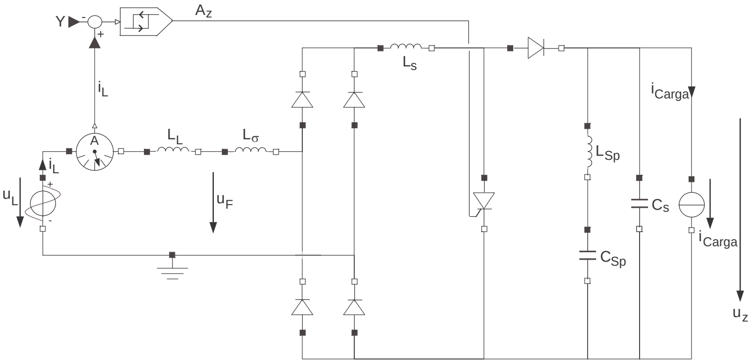


- Los resultados de la simulación muestran discontinuidades menos serias que en el control por tiristor usado antes.
- En consecuencia, esta estrategia de control no afecta la red eléctrica operada a $50\mathrm{Hz}$.
- Sin embargo, este tipo de control solamente permite ocho velocidades distintas.

- **Cuando los trenes salían de las estaciones, se sentían los cambios abruptos de la velocidad, que resultaban en masajes gratuitos en la musculatura del estómago de los pasajeros.**

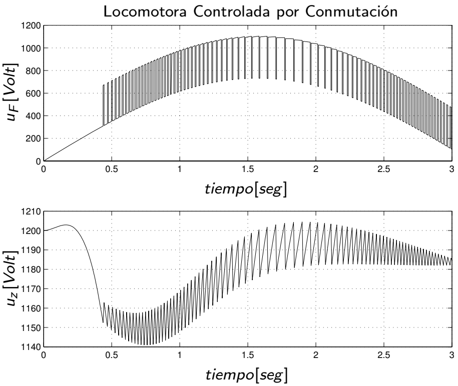
El Control de la Velocidad de Trenes VI

Para evitar los problemas encontrados antes, se desarrolló un circuito de control bastante más complejo con un *rectificador en los cuatro cuadrantes* y con *conmutación*.



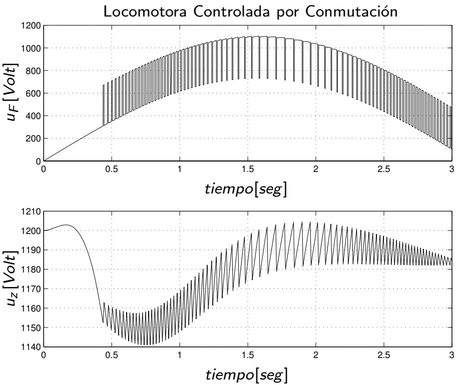
El Control de la Velocidad de Trenes VII

Lo que deberíamos obtener son los resultados siguientes:



El Control de la Velocidad de Trenes VII

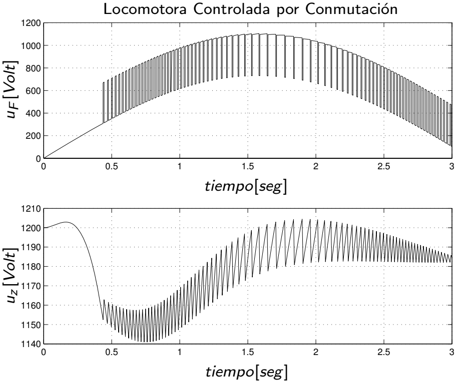
Lo que deberíamos obtener son los resultados siguientes:



- Los resultados de la simulación muestran discontinuidades muy serias. Por eso se necesita un *control del paso muy eficaz*.

El Control de la Velocidad de Trenes VII

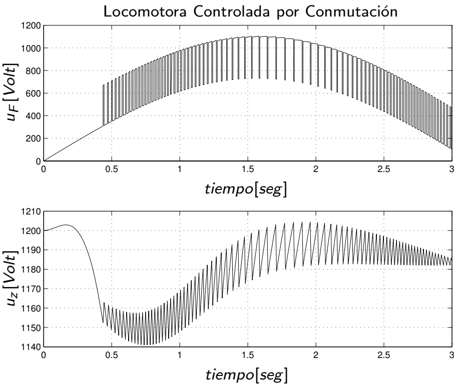
Lo que deberíamos obtener son los resultados siguientes:



- Los resultados de la simulación muestran discontinuidades muy serias. Por eso se necesita un *control del paso muy eficaz*.
- Estas señales tienen mucho contenido de altas frecuencias en sus espectros de potencia. Sin embargo, no tienen mucha potencia en el tercer armónico.

El Control de la Velocidad de Trenes VII

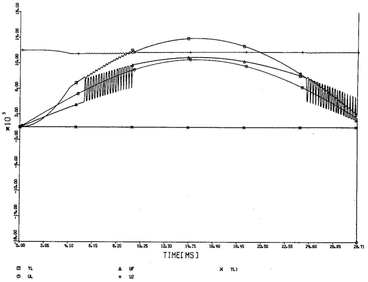
Lo que deberíamos obtener son los resultados siguientes:

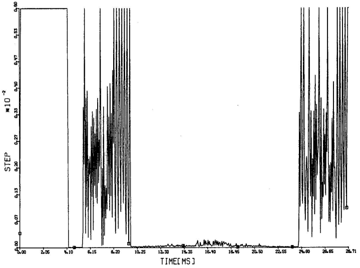


- Los resultados de la simulación muestran discontinuidades muy serias. Por eso se necesita un *control del paso muy eficaz*.
- Estas señales tienen mucho contenido de altas frecuencias en sus espectros de potencia. Sin embargo, no tienen mucha potencia en el tercer armónico.
- Entonces no tocan a la red eléctrica comercial de $50\mathrm{Hz}$.

El Control de la Velocidad de Trenes VIII

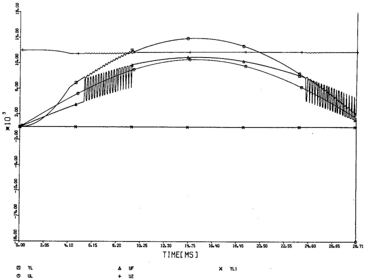
Originalmente simulamos ese modelo usando un algoritmo RK4 con abuso del control del paso para el tratamiento de las discontinuidades. Los resultados que obtuvimos son:

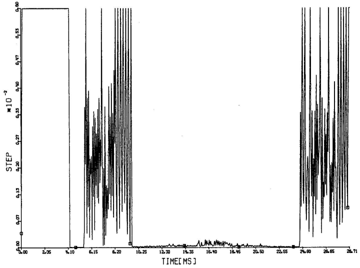




El Control de la Velocidad de Trenes VIII

Originalmente simulamos ese modelo usando un algoritmo RK4 con abuso del control del paso para el tratamiento de las discontinuidades. Los resultados que obtuvimos son:

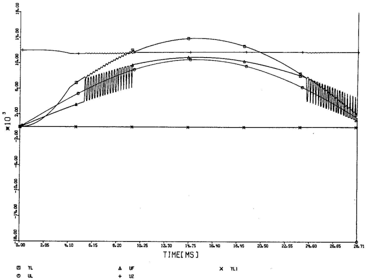


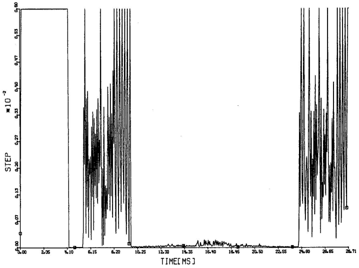


Durante cierto intervalo, los *resultados de la simulación son incorrectos*, mientras que el *paso de integración se reduce a su valor mínimo*.

El Control de la Velocidad de Trenes VIII

Originalmente simulamos ese modelo usando un algoritmo RK4 con abuso del control del paso para el tratamiento de las discontinuidades. Los resultados que obtuvimos son:





Durante cierto intervalo, los *resultados de la simulación son incorrectos*, mientras que el *paso de integración se reduce a su valor mínimo*.

La simulación está arrastrándose.

El Control de la Velocidad de Trenes IX

¿Qué pasó?

El Control de la Velocidad de Trenes IX

¿Qué pasó?

- Para la simulación usamos un *algoritmo RK4 explícito con abuso del control del paso* para el *tratamiento de las discontinuidades*.

El Control de la Velocidad de Trenes IX

¿Qué pasó?

- Para la simulación usamos un *algoritmo RK4 explícito con abuso del control del paso* para el *tratamiento de las discontinuidades*.
- Se trató de un algoritmo en cuatro etapas.

El Control de la Velocidad de Trenes IX

¿Qué pasó?

- Para la simulación usamos un *algoritmo RK4 explícito con abuso del control del paso* para el *tratamiento de las discontinuidades*.
- Se trató de un algoritmo en cuatro etapas.
- Los efectos de una conmutación son tan graves en este modelo que a veces occurió más de una conmutación en un mismo paso.

El Control de la Velocidad de Trenes IX

¿Qué pasó?

- Para la simulación usamos un *algoritmo RK4 explícito con abuso del control del paso* para el *tratamiento de las discontinuidades*.
- Se trató de un algoritmo en cuatro etapas.
- Los efectos de una conmutación son tan graves en este modelo que a veces occurió más de una conmutación en un mismo paso.
- Si el número de conmutaciones dentro de un paso fue *par*, nos quedamos al final del paso del mismo lado de la valla, y por eso, tuvimos que tratar otra vez de superarla durante el próximo paso.

El Control de la Velocidad de Trenes IX

¿Qué pasó?

- Para la simulación usamos un *algoritmo RK4 explícito con abuso del control del paso* para el *tratamiento de las discontinuidades*.
- Se trató de un algoritmo en cuatro etapas.
- Los efectos de una conmutación son tan graves en este modelo que a veces occurió más de una conmutación en un mismo paso.
- Si el número de conmutaciones dentro de un paso fue *par*, nos quedamos al final del paso del mismo lado de la valla, y por eso, tuvimos que tratar otra vez de superarla durante el próximo paso.
- La simulación procedió durante muchísimo tiempo con el paso mínimo permitido, incapaz de superar la valla.

El Control de la Velocidad de Trenes IX

¿Qué pasó?

- Para la simulación usamos un *algoritmo RK4 explícito con abuso del control del paso* para el *tratamiento de las discontinuidades*.
- Se trató de un algoritmo en cuatro etapas.
- Los efectos de una conmutación son tan graves en este modelo que a veces occurió más de una conmutación en un mismo paso.
- Si el número de conmutaciones dentro de un paso fue *par*, nos quedamos al final del paso del mismo lado de la valla, y por eso, tuvimos que tratar otra vez de superarla durante el próximo paso.
- La simulación procedió durante muchísimo tiempo con el paso mínimo permitido, incapaz de superar la valla.

Abusar del control del paso de integración para el tratamiento de discontinuidades no es seguro. A veces funciona bastante bien, pero no funciona siempre.

Eventos en el Tiempo

El problema es que no le *avisamos* al integrador que había una discontinuidad. El integrador no sabe *leer* modelos. Solamente sabe *ejecutarlos*.

Eventos en el Tiempo

El problema es que no le *avisamos* al integrador que había una discontinuidad. El integrador no sabe *leer* modelos. Solamente sabe *ejecutarlos*.

Lo que se necesita es un *elemento sintáctico* en el *lenguaje de la descripción del modelo* para poder informar explícitamente al integrador de la ocurrencia de discontinuidades.

Eventos en el Tiempo

El problema es que no le *avisamos* al integrador que había una discontinuidad. El integrador no sabe *leer* modelos. Solamente sabe *ejecutarlos*.

Lo que se necesita es un *elemento sintáctico* en el *lenguaje de la descripción del modelo* para poder informar explícitamente al integrador de la ocurrencia de discontinuidades.

Llamamos *eventos discretos* a las discontinuidades. Lo que se necesita son mecanismos explícitos para la descripción y el tratamiento de eventos.

Eventos en el Tiempo

El problema es que no le *avisamos* al integrador que había una discontinuidad. El integrador no sabe *leer* modelos. Solamente sabe *ejecutarlos*.

Lo que se necesita es un *elemento sintáctico* en el *lenguaje de la descripción del modelo* para poder informar explícitamente al integrador de la ocurrencia de discontinuidades.

Llamamos *eventos discretos* a las discontinuidades. Lo que se necesita son mecanismos explícitos para la descripción y el tratamiento de eventos.

En algunos casos, se conoce con antelación el tiempo de ocurrencia de un evento. En este caso hablamos de *eventos en el tiempo*.

Eventos en el Tiempo II

Consideramos otra vez el control de los trenes por un tiristor. El tiempo del cierre de la barrera se conoce con antelación. La barrera se cierra por primera vez α grados después del comienzo del período.

Eventos en el Tiempo II

Consideramos otra vez el control de los trenes por un tiristor. El tiempo del cierre de la barrera se conoce con antelación. La barrera se cierra por primera vez α grados después del comienzo del período.

Podemos calcular:

$$\begin{aligned} t_{\text{período}} &= \frac{1}{2\pi f} \\ t_{\text{evento}} &= \frac{\alpha}{360} \cdot t_{\text{período}} \end{aligned}$$

Eventos en el Tiempo II

Consideramos otra vez el control de los trenes por un tiristor. El tiempo del cierre de la barrera se conoce con antelación. La barrera se cierra por primera vez α grados después del comienzo del período.

Podemos calcular:

$$
\begin{aligned}
t_{\text{período}} &= \frac{1}{2\pi f} \\
t_{\text{evento}} &= \frac{\alpha}{360} \cdot t_{\text{período}}
\end{aligned}
$$

Se puede *planificar* la primera ocurrencia de un evento de cierre de la barrera para el instante de tiempo t_{evento}.

Eventos en el Tiempo II

Consideramos otra vez el control de los trenes por un tiristor. El tiempo del cierre de la barrera se conoce con antelación. La barrera se cierra por primera vez α grados después del comienzo del período.

Podemos calcular:

$$\begin{aligned} t_{\text{período}} &= \frac{1}{2\pi f} \\ t_{\text{evento}} &= \frac{\alpha}{360} \cdot t_{\text{período}} \end{aligned}$$

Se puede *planificar* la primera ocurrencia de un evento de cierre de la barrera para el instante de tiempo t_{evento}.

Una de las acciones asociadas con el evento del cierre es planificar la próxima ocurrencia del mismo evento al instante $t + t_{\text{período}}$.

Eventos en el Tiempo II

Consideramos otra vez el control de los trenes por un tiristor. El tiempo del cierre de la barrera se conoce con antelación. La barrera se cierra por primera vez α grados después del comienzo del período.

Podemos calcular:

$$\begin{aligned} t_{\text{período}} &= \frac{1}{2\pi f} \\ t_{\text{evento}} &= \frac{\alpha}{360} \cdot t_{\text{período}} \end{aligned}$$

Se puede *planificar* la primera ocurrencia de un evento de cierre de la barrera para el instante de tiempo t_{evento}.

Una de las acciones asociadas con el evento del cierre es planificar la próxima ocurrencia del mismo evento al instante $t + t_{\text{período}}$.

El evento de la apertura de la barrera es otro tipo de evento del cual hablaremos más tarde.

Eventos en el Tiempo III

Vale mencionar que la *discontinuidad asociada con el evento* no se encuentra en el modelo continuo. Todas las trayectorias son perfectamente continuas. Solamente la *condición bajo la cual ocurre el evento* se conoce durante la simulación continua.

Eventos en el Tiempo III

Vale mencionar que la *discontinuidad asociada con el evento* no se encuentra en el modelo continuo. Todas las trayectorias son perfectamente continuas. Solamente la *condición bajo la cual ocurre el evento* se conoce durante la simulación continua.

Por esa razón, el integrador no tendrá problemas. No tratará nunca de simular a través de una discontinuidad en el modelo.

Eventos en el Tiempo III

Vale mencionar que la *discontinuidad asociada con el evento* no se encuentra en el modelo continuo. Todas las trayectorias son perfectamente continuas. Solamente la *condición bajo la cual ocurre el evento* se conoce durante la simulación continua.

Por esa razón, el integrador no tendrá problemas. No tratará nunca de simular a través de una discontinuidad en el modelo.

La *localización de un evento en el tiempo* es trivial. Es un *problema de salidas densas*. Hacemos exactamente lo mismo que hacemos para localizar un instante en el tiempo en el cual queremos *registrar los valores de las salidas*.

Eventos en el Tiempo III

Vale mencionar que la *discontinuidad asociada con el evento* no se encuentra en el modelo continuo. Todas las trayectorias son perfectamente continuas. Solamente la *condición bajo la cual ocurre el evento* se conoce durante la simulación continua.

Por esa razón, el integrador no tendrá problemas. No tratará nunca de simular a través de una discontinuidad en el modelo.

La *localización de un evento en el tiempo* es trivial. Es un *problema de salidas densas*. Hacemos exactamente lo mismo que hacemos para localizar un instante en el tiempo en el cual queremos *registrar los valores de las salidas*.

Si el algoritmo de integración reduce el paso para encontrar los *instantes de comunicación* (lo hacemos así normalmente con los integradores en un solo paso), haremos lo mismo para la localización de eventos en el tiempo.

Eventos en el Tiempo III

Vale mencionar que la *discontinuidad asociada con el evento* no se encuentra en el modelo continuo. Todas las trayectorias son perfectamente continuas. Solamente la *condición bajo la cual ocurre el evento* se conoce durante la simulación continua.

Por esa razón, el integrador no tendrá problemas. No tratará nunca de simular a través de una discontinuidad en el modelo.

La *localización de un evento en el tiempo* es trivial. Es un *problema de salidas densas*. Hacemos exactamente lo mismo que hacemos para localizar un instante en el tiempo en el cual queremos *registrar los valores de las salidas*.

Si el algoritmo de integración reduce el paso para encontrar los *instantes de comunicación* (lo hacemos así normalmente con los integradores en un solo paso), haremos lo mismo para la localización de eventos en el tiempo.

Por otro lado, si el algoritmo de integración interpola trayectorias para encontrar los *instantes de comunicación* (lo hacemos así normalmente con los integradores en múltiples pasos), haremos lo mismo para la localización de eventos en el tiempo.

Eventos en el Tiempo IV

Una vez que se localizó el evento, las acciones asociadas con el evento se ejecutan.

Eventos en el Tiempo IV

Una vez que se localizó el evento, las acciones asociadas con el evento se ejecutan.

Después se trata de una nueva simulación continua con nuevas condiciones iniciales.

Eventos en el Tiempo IV

Una vez que se localizó el evento, las acciones asociadas con el evento se ejecutan.

Después se trata de una nueva simulación continua con nuevas condiciones iniciales.

Los *eventos en el tiempo* se guardan en un *calendario de eventos*, una lista lineal enlazada hacia adelante y hacia atrás, de tal manera, que el *próximo evento en el tiempo* se encuentra siempre en el principio de la lista.

Eventos en el Tiempo IV

Una vez que se localizó el evento, las acciones asociadas con el evento se ejecutan.

Después se trata de una nueva simulación continua con nuevas condiciones iniciales.

Los *eventos en el tiempo* se guardan en un *calendario de eventos*, una lista lineal enlazada hacia adelante y hacia atrás, de tal manera, que el *próximo evento en el tiempo* se encuentra siempre en el principio de la lista.

La simulación continua solamente tiene que conocer el *tiempo de ocurrencia del próximo evento en el tiempo*.

Eventos en el Tiempo IV

Una vez que se localizó el evento, las acciones asociadas con el evento se ejecutan.

Después se trata de una nueva simulación continua con nuevas condiciones iniciales.

Los *eventos en el tiempo* se guardan en un *calendario de eventos*, una lista lineal enlazada hacia adelante y hacia atrás, de tal manera, que el *próximo evento en el tiempo* se encuentra siempre en el principio de la lista.

La simulación continua solamente tiene que conocer el *tiempo de ocurrencia del próximo evento en el tiempo*.

La simulación continua procede hasta el tiempo de ocurrencia del próximo evento. En ese instante, la simulación continua termina, las acciones asociadas con el evento (la discontinuidad) se procesan, y una nueva simulación continua empieza con las nuevas condiciones iniciales.

Eventos en el Tiempo V

En consecuencia, las simulaciones continuas se ejecutan ahora a tramos. Se interrumpen por los tratamientos de los eventos discretos.

Eventos en el Tiempo V

En consecuencia, las simulaciones continuas se ejecutan ahora a tramos. Se interrumpen por los tratamientos de los eventos discretos.

Un modelo que especifica explícitamente los eventos discretos se llama *modelo híbrido*.

Eventos en el Tiempo V

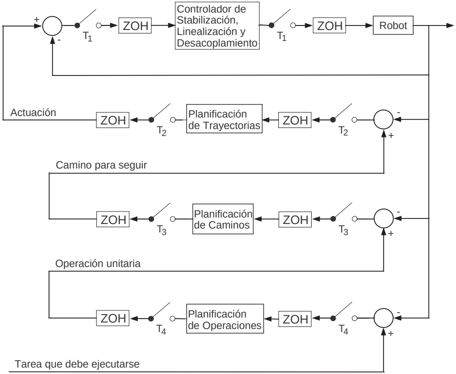
En consecuencia, las simulaciones continuas se ejecutan ahora a tramos. Se interrumpen por los tratamientos de los eventos discretos.

Un modelo que especifica explícitamente los eventos discretos se llama *modelo híbrido*.

La simulación de un modelo híbrido se llama *simulación a tramos*.

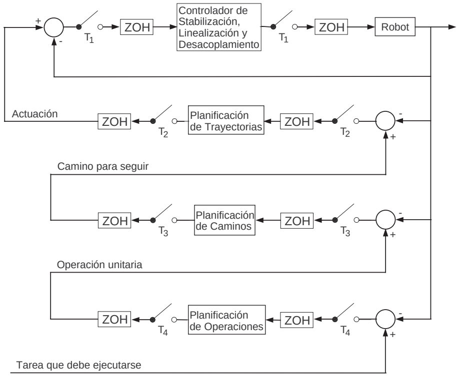
La Simulación de Sistemas de Control Digital

Los sistemas de control digital pueden modelarse usando modelos híbridos. En consecuencia pueden simularse a tramos.



La Simulación de Sistemas de Control Digital

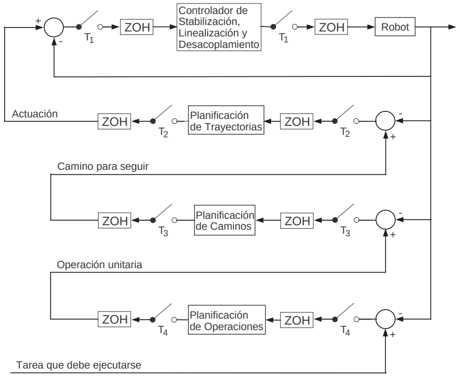
Los sistemas de control digital pueden modelarse usando modelos híbridos. En consecuencia pueden simularse a tramos.



- Hay cuatro bucles cerrados usando muestreos con diferentes intervalos en el tiempo $T_1 \leq T_2 \leq T_3 \ll T_4$.

La Simulación de Sistemas de Control Digital

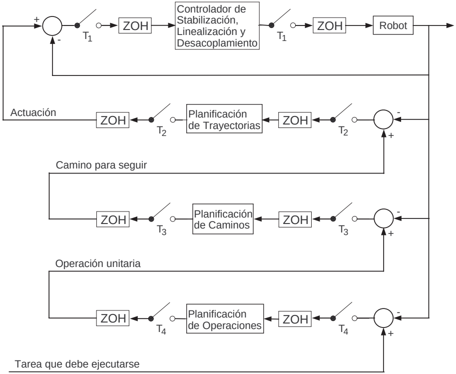
Los sistemas de control digital pueden modelarse usando modelos híbridos. En consecuencia pueden simularse a tramos.



- Hay cuatro bucles cerrados usando muestreos con diferentes intervalos en el tiempo $T_1 \leq T_2 \leq T_3 \ll T_4$.
- Cada muestreador representa una serie abierta de eventos en el tiempo.

La Simulación de Sistemas de Control Digital

Los sistemas de control digital pueden modelarse usando modelos híbridos. En consecuencia pueden simularse a tramos.



- Hay cuatro bucles cerrados usando muestreos con diferentes intervalos en el tiempo $T_1 \leq T_2 \leq T_3 \ll T_4$.
- Cada muestreador representa una serie abierta de eventos en el tiempo.
- El calendario de los eventos mantiene los cuatro tipos de eventos y decide en cada momento cual es el próximo evento que debe ejecutarse.

Eventos en los Estados

A menudo no se conoce con antelación el *tiempo de ocurrencia* de un evento, sino solamente la *condición bajo la cual ocurre*.

Eventos en los Estados

A menudo no se conoce con antelación el *tiempo de ocurrencia* de un evento, sino solamente la *condición bajo la cual ocurre*.

Estos eventos se llaman *eventos en los estados*.

Eventos en los Estados

A menudo no se conoce con antelación el *tiempo de ocurrencia* de un evento, sino solamente la *condición bajo la cual ocurre*.

Estos eventos se llaman *eventos en los estados*.

Los eventos en los estados *no pueden planificarse*.

Eventos en los Estados

A menudo no se conoce con antelación el *tiempo de ocurrencia* de un evento, sino solamente la *condición bajo la cual ocurre*.

Estos eventos se llaman *eventos en los estados*.

Los eventos en los estados *no pueden planificarse*.

En el ejemplo del control de la velocidad de los trenes usando un tiristor, el evento de la apertura de la barrera es un evento en un estado. No se sabe con antelación cuando ocurre. Solamente se sabe que ocurre cuando la corriente pasa por cero.

Eventos en los Estados II

Un evento en los estados se especifica por una *condición de evento*, formulada normalmente por una *función de variables continuas de cruce por cero*.

Eventos en los Estados II

Un evento en los estados se especifica por una *condición de evento*, formulada normalmente por una *función de variables continuas de cruce por cero*.

Tal función se llama *función de detección de un evento* en los estados.

Eventos en los Estados II

Un evento en los estados se especifica por una *condición de evento*, formulada normalmente por una *función de variables continuas de cruce por cero*.

Tal función se llama *función de detección de un evento* en los estados.

Puede haber múltiples funciones de detección de eventos simultáneamente activas.

Eventos en los Estados II

Un evento en los estados se especifica por una *condición de evento*, formulada normalmente por una *función de variables continuas de cruce por cero*.

Tal función se llama *función de detección de un evento* en los estados.

Puede haber múltiples funciones de detección de eventos simultáneamente activas.

Durante la simulación continua, las funciones activas de detección de eventos tienen que vigilarse para detectar si una de ellas cruza por cero.

Eventos en los Estados II

Un evento en los estados se especifica por una *condición de evento*, formulada normalmente por una *función de variables continuas de cruce por cero*.

Tal función se llama *función de detección de un evento* en los estados.

Puede haber múltiples funciones de detección de eventos simultáneamente activas.

Durante la simulación continua, las funciones activas de detección de eventos tienen que vigilarse para detectar si una de ellas cruza por cero.

Una vez que una de las funciones activas de detección de eventos ha cruzado por cero, un algoritmo de *localización de eventos* en los estados se activa.

Eventos en los Estados II

Un evento en los estados se especifica por una *condición de evento*, formulada normalmente por una *función de variables continuas de cruce por cero*.

Tal función se llama *función de detección de un evento* en los estados.

Puede haber múltiples funciones de detección de eventos simultáneamente activas.

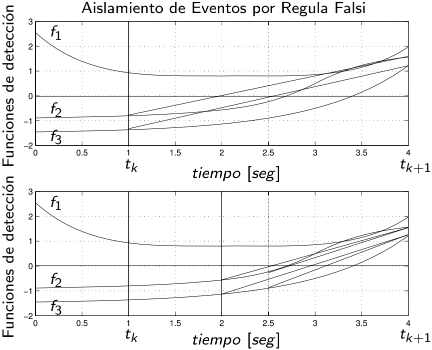
Durante la simulación continua, las funciones activas de detección de eventos tienen que vigilarse para detectar si una de ellas cruza por cero.

Una vez que una de las funciones activas de detección de eventos ha cruzado por cero, un algoritmo de *localización de eventos* en los estados se activa.

En la literatura matemática, el algoritmo de localización de eventos también se llama *algoritmo de solución de raíces*.

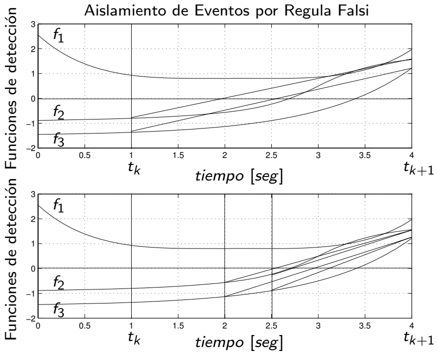
Múltiples Cruces Simultáneos

Si más de una función de detección de eventos encuentra un cruce por cero durante el mismo paso de integración, se usa un algoritmo para detectar cual de estos cruces sucede primero.



Múltiples Cruces Simultáneos

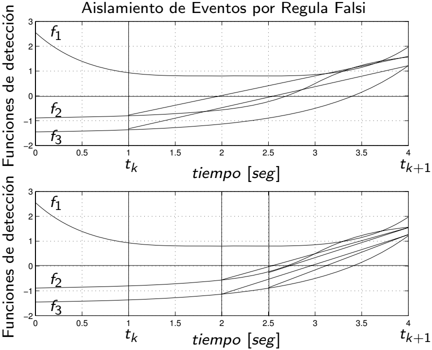
Si más de una función de detección de eventos encuentra un cruce por cero durante el mismo paso de integración, se usa un algoritmo para detectar cual de estos cruces sucede primero.



- En el método de *Regula Falsi* se conectan los puntos finales de cada función de detección de cruces por cero.

Múltiples Cruces Simultáneos

Si más de una función de detección de eventos encuentra un cruce por cero durante el mismo paso de integración, se usa un algoritmo para detectar cual de estos cruces sucede primero.

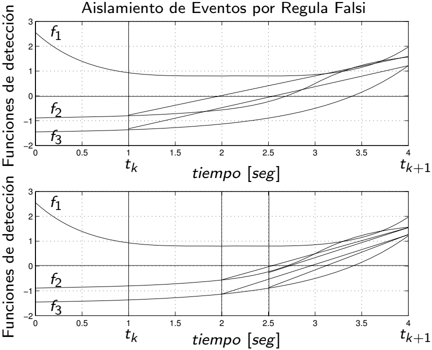


- En el método de *Regula Falsi* se conectan los puntos finales de cada función de detección de cruces por cero.
- El próximo instante de la iteración se calcula usando la fórmula:

$$t_{\text{próximo}} = \min_{\forall i} \left[\frac{f_i(t_{k+1}) \cdot t_k - f_i(t_k) \cdot t_{k+1}}{f_i(t_{k+1}) - f_i(t_k)} \right]$$

Múltiples Cruces Simultáneos

Si más de una función de detección de eventos encuentra un cruce por cero durante el mismo paso de integración, se usa un algoritmo para detectar cual de estos cruces sucede primero.



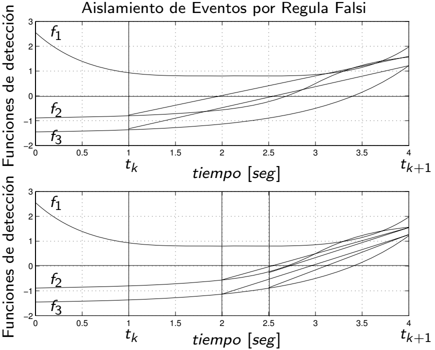
- En el método de *Regula Falsi* se conectan los puntos finales de cada función de detección de cruces por cero.
- El próximo instante de la iteración se calcula usando la fórmula:

$$t_{\text{próximo}} = \min_{\forall i} \left[\frac{f_i(t_{k+1}) \cdot t_k - f_i(t_k) \cdot t_{k+1}}{f_i(t_{k+1}) - f_i(t_k)} \right]$$

- Si no hay ningún cruce durante $t \in [t_k, t_{\text{próximo}}]$, se pone $t_k = t_{\text{próximo}}$ y se repite el algoritmo.

Múltiples Cruces Simultáneos

Si más de una función de detección de eventos encuentra un cruce por cero durante el mismo paso de integración, se usa un algoritmo para detectar cual de estos cruces sucede primero.



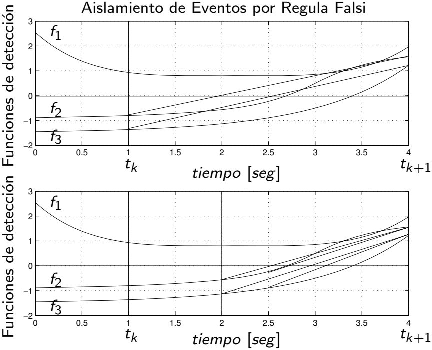
- En el método de *Regula Falsi* se conectan los puntos finales de cada función de detección de cruces por cero.
- El próximo instante de la iteración se calcula usando la fórmula:

$$t_{\text{próximo}} = \min_{\forall i} \left[\frac{f_i(t_{k+1}) \cdot t_k - f_i(t_k) \cdot t_{k+1}}{f_i(t_{k+1}) - f_i(t_k)} \right]$$

- Si no hay ningún cruce durante $t \in [t_k, t_{\text{próximo}}]$, se pone $t_k = t_{\text{próximo}}$ y se repite el algoritmo.
- Si todavía hay múltiples cruces durante $t \in [t_k, t_{\text{próximo}}]$, se pone $t_{k+1} = t_{\text{próximo}}$ y se repite el algoritmo.

Múltiples Cruces Simultáneos

Si más de una función de detección de eventos encuentra un cruce por cero durante el mismo paso de integración, se usa un algoritmo para detectar cual de estos cruces sucede primero.



- En el método de *Regula Falsi* se conectan los puntos finales de cada función de detección de cruces por cero.
- El próximo instante de la iteración se calcula usando la fórmula:

$$t_{\text{próximo}} = \min_{\forall i} \left[\frac{f_i(t_{k+1}) \cdot t_k - f_i(t_k) \cdot t_{k+1}}{f_i(t_{k+1}) - f_i(t_k)} \right]$$

- Si no hay ningún cruce durante $t \in [t_k, t_{\text{próximo}}]$, se pone $t_k = t_{\text{próximo}}$ y se repite el algoritmo.
- Si todavía hay múltiples cruces durante $t \in [t_k, t_{\text{próximo}}]$, se pone $t_{k+1} = t_{\text{próximo}}$ y se repite el algoritmo.
- *Si hay exactamente un cruce durante* $t \in [t_k, t_{\text{próximo}}]$*, se ha simplificado el problema.*

Múltiples Cruces Simultáneos II

- El *aislamiento de eventos* usando el método de *Regula Falsi* converge siempre, el intervalo se reduce durante cada iteración, pero no puede decirse con antelación, cuantas iteraciones se necesitan para obtener convergencia.

Múltiples Cruces Simultáneos II

- El *aislamiento de eventos* usando el método de *Regula Falsi* converge siempre, el intervalo se reduce durante cada iteración, pero no puede decirse con antelación, cuantas iteraciones se necesitan para obtener convergencia.
- Existe otro método que también converge siempre, pero que tiene la ventaja que el intervalo se reduce en cada iteración un 37%.

Múltiples Cruces Simultáneos II

- El *aislamiento de eventos* usando el método de *Regula Falsi* converge siempre, el intervalo se reduce durante cada iteración, pero no puede decirse con antelación, cuantas iteraciones se necesitan para obtener convergencia.
- Existe otro método que también converge siempre, pero que tiene la ventaja que el intervalo se reduce en cada iteración un 37%.

  Se trata de la *sección de oro*.

Múltiples Cruces Simultáneos II

- El *aislamiento de eventos* usando el método de *Regula Falsi* converge siempre, el intervalo se reduce durante cada iteración, pero no puede decirse con antelación, cuantas iteraciones se necesitan para obtener convergencia.
- Existe otro método que también converge siempre, pero que tiene la ventaja que el intervalo se reduce en cada iteración un 37%.

  Se trata de la *sección de oro*.

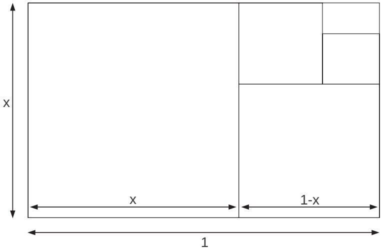


Figure: La sección de oro.

Múltiples Cruces Simultáneos II

- El *aislamiento de eventos* usando el método de *Regula Falsi* converge siempre, el intervalo se reduce durante cada iteración, pero no puede decirse con antelación, cuantas iteraciones se necesitan para obtener convergencia.
- Existe otro método que también converge siempre, pero que tiene la ventaja que el intervalo se reduce en cada iteración un 37%.

  Se trata de la *sección de oro*.

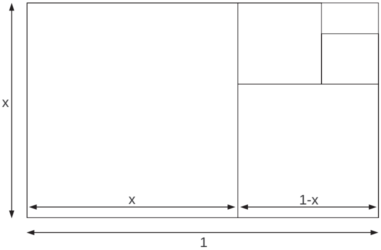
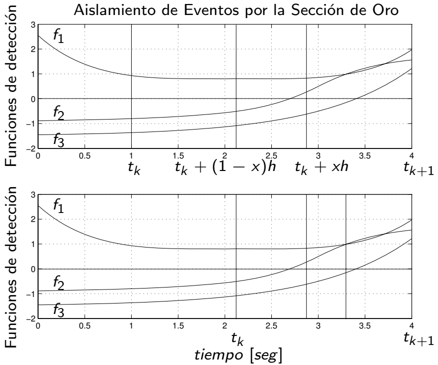


Figure: La sección de oro.

$$\frac{x}{1} = \frac{1-x}{x} \quad \Rightarrow \quad x = 0.618$$

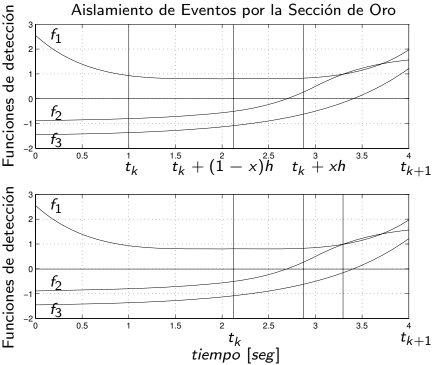
Múltiples Cruces Simultáneos III

¿Como puede usarse la *sección de oro* para el *aislamiento de eventos*?



Múltiples Cruces Simultáneos III

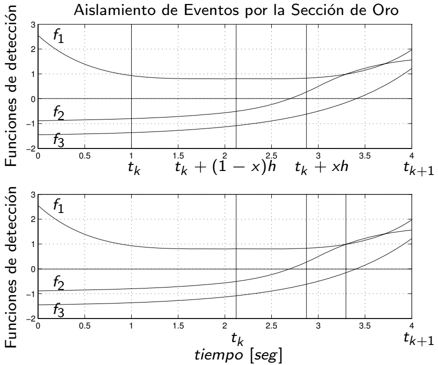
¿Como puede usarse la *sección de oro* para el *aislamiento de eventos*?



- En el método de la *sección de oro* se calculan dos puntos intermedios en el intervalo $t \in [t_k, t_{k+1}]$ relacionados con el intervalo por la sección de oro.

Múltiples Cruces Simultáneos III

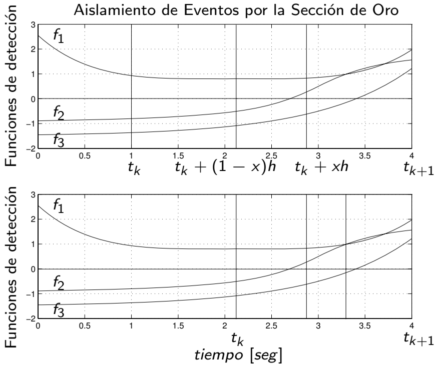
¿Como puede usarse la *sección de oro* para el *aislamiento de eventos*?



- En el método de la *sección de oro* se calculan dos puntos intermedios en el intervalo $t \in [t_k, t_{k+1}]$ relacionados con el intervalo por la sección de oro.
- Si no hay ningún cruce durante $t \in [t_k, t_k + (1-x) \cdot h]$, se pone $t_k = t_k + (1-x) \cdot h$ y se repite el algoritmo con un punto intermedio nuevo.

Múltiples Cruces Simultáneos III

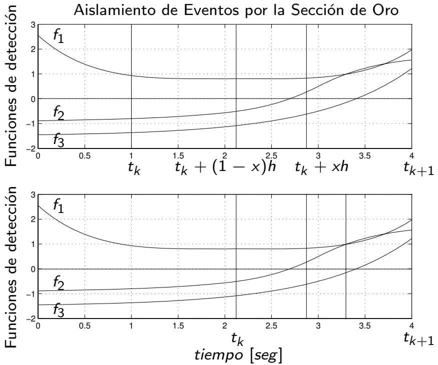
¿Como puede usarse la *sección de oro* para el *aislamiento de eventos*?



- En el método de la *sección de oro* se calculan dos puntos intermedios en el intervalo $t \in [t_k, t_{k+1}]$ relacionados con el intervalo por la sección de oro.
- Si no hay ningún cruce durante $t \in [t_k, t_k + (1-x) \cdot h]$, se pone $t_k = t_k + (1-x) \cdot h$ y se repite el algoritmo con un punto intermedio nuevo.
- Si todavía hay múltiples cruces durante $t \in [t_k, t_k + (1-x) \cdot h]$, se pone $t_{k+1} = t_k + x \cdot h$ y se repite el algoritmo con un punto intermedio nuevo.

Múltiples Cruces Simultáneos III

¿Como puede usarse la *sección de oro* para el *aislamiento de eventos*?



- En el método de la *sección de oro* se calculan dos puntos intermedios en el intervalo $t \in [t_k, t_{k+1}]$ relacionados con el intervalo por la sección de oro.
- Si no hay ningún cruce durante $t \in [t_k, t_k + (1-x) \cdot h]$, se pone $t_k = t_k + (1-x) \cdot h$ y se repite el algoritmo con un punto intermedio nuevo.
- Si todavía hay múltiples cruces durante $t \in [t_k, t_k + (1-x) \cdot h]$, se pone $t_{k+1} = t_k + x \cdot h$ y se repite el algoritmo con un punto intermedio nuevo.
- *Si hay exactamente un cruce durante* $t \in [t_k, t_k + (1-x) \cdot h]$*, se ha simplificado el problema.*

Múltiples Cruces Simultáneos IV

Múltiples Cruces Simultáneos IV

- Entre tanto hemos resuelto el problema del *aislamiento de eventos*.

Múltiples Cruces Simultáneos IV

- Entre tanto hemos resuelto el problema del *aislamiento de eventos*.

- Si se encuentra más de un cruce por cero dentro de un paso de la integración, reducimos el intervalo $t \in [t_k, t_{k+1}]$ usando uno de dos métodos propuestos: *Regula Falsi* y la *sección de oro*, hasta que nos quede exactamente un *cruce por cero* de una *función de detección de eventos* dentro del intervalo.

Múltiples Cruces Simultáneos IV

- Entre tanto hemos resuelto el problema del *aislamiento de eventos*.
- Si se encuentra más de un cruce por cero dentro de un paso de la integración, reducimos el intervalo $t \in [t_k, t_{k+1}]$ usando uno de dos métodos propuestos: *Regula Falsi* y la *sección de oro*, hasta que nos quede exactamente un *cruce por cero* de una *función de detección de eventos* dentro del intervalo.
- Entonces el problema se ha reducido ahora al problema de la *localización de un solo evento ya capturado de los dos lados dentro de un intervalo fijo*.

La Localización de Eventos

Métodos de la Integración Numérica a un Paso

- Es cierto que los dos métodos de *aislamiento de eventos* también funcionarán para la *localización de eventos*.

La Localización de Eventos

Métodos de la Integración Numérica a un Paso

- Es cierto que los dos métodos de *aislamiento de eventos* también funcionarán para la *localización de eventos*.
- El problema es que tienen *velocidad de convergencia* lineal.

La Localización de Eventos

Métodos de la Integración Numérica a un Paso

- Es cierto que los dos métodos de *aislamiento de eventos* también funcionarán para la *localización de eventos*.
- El problema es que tienen *velocidad de convergencia* lineal.

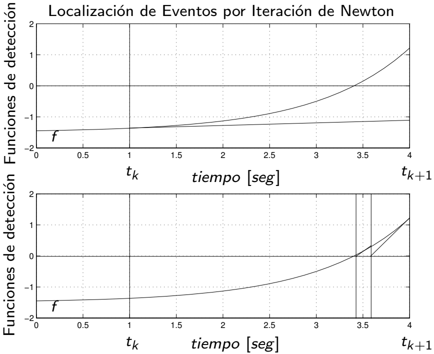
  Se puede pensar que la *iteración de Newton* funcionará mejor, porque tiene velocidad de convergencia cuadrada.

La Localización de Eventos

Métodos de la Integración Numérica a un Paso

- Es cierto que los dos métodos de *aislamiento de eventos* también funcionarán para la *localización de eventos*.
- El problema es que tienen *velocidad de convergencia* lineal.

  Se puede pensar que la *iteración de Newton* funcionará mejor, porque tiene velocidad de convergencia cuadrada.



La Localización de Eventos II

Métodos de la Integración Numérica a un Paso

La Localización de Eventos II

Métodos de la Integración Numérica a un Paso

- Desafortunadamente, la iteración de Newton no tiene convergencia siempre. En el ejemplo, si empezamos por la derecha obtenemos convergencia rápidamente, mientras que si empezamos por la izquierda, el siguiente paso de iteración nos lleva fuera del intervalo.

La Localización de Eventos II

Métodos de la Integración Numérica a un Paso

- Desafortunadamente, la iteración de Newton no tiene convergencia siempre. En el ejemplo, si empezamos por la derecha obtenemos convergencia rápidamente, mientras que si empezamos por la izquierda, el siguiente paso de iteración nos lleva fuera del intervalo.
- Como ya capturamos el cruce dentro de un intervalo conocido, no nos gusta buscar soluciones fuera del intervalo.

La Localización de Eventos II

Métodos de la Integración Numérica a un Paso

- Desafortunadamente, la iteración de Newton no tiene convergencia siempre. En el ejemplo, si empezamos por la derecha obtenemos convergencia rápidamente, mientras que si empezamos por la izquierda, el siguiente paso de iteración nos lleva fuera del intervalo.
- Como ya capturamos el cruce dentro de un intervalo conocido, no nos gusta buscar soluciones fuera del intervalo.
- Si el próximo paso de la iteración de Newton nos llevara fuera del intervalo empezando en ambos lados, sería mejor reducir el intervalo un poco más usando uno de los dos algoritmos introducidos antes y después tratar con Newton de nuevo.

La Localización de Eventos II

Métodos de la Integración Numérica a un Paso

- Desafortunadamente, la iteración de Newton no tiene convergencia siempre. En el ejemplo, si empezamos por la derecha obtenemos convergencia rápidamente, mientras que si empezamos por la izquierda, el siguiente paso de iteración nos lleva fuera del intervalo.
- Como ya capturamos el cruce dentro de un intervalo conocido, no nos gusta buscar soluciones fuera del intervalo.
- Si el próximo paso de la iteración de Newton nos llevara fuera del intervalo empezando en ambos lados, sería mejor reducir el intervalo un poco más usando uno de los dos algoritmos introducidos antes y después tratar con Newton de nuevo.
- Una vez que se ha reducido el intervalo bastante, Newton nos dará convergencia y lo hará más rápidamente que cualquier de los dos algoritmos introducidos antes.

La Localización de Eventos III

Métodos de la Integración Numérica a un Paso

La Localización de Eventos III

Métodos de la Integración Numérica a un Paso

- Otro algoritmo quizás aún más eficiente que la *iteración de Newton* para la localización de eventos es la *interpolación polinomial*.

La Localización de Eventos III

Métodos de la Integración Numérica a un Paso

- Otro algoritmo quizás aún más eficiente que la *iteración de Newton* para la localización de eventos es la *interpolación polinomial*.
- Como cada función de detección de eventos puede expresarse en función de variables del estado, conocemos no sólo el valor de las funciones de detección en ambos lados del intervalo, sino también el valor de sus derivadas.

La Localización de Eventos III

Métodos de la Integración Numérica a un Paso

- Otro algoritmo quizás aún más eficiente que la *iteración de Newton* para la localización de eventos es la *interpolación polinomial*.
- Como cada función de detección de eventos puede expresarse en función de variables del estado, conocemos no sólo el valor de las funciones de detección en ambos lados del intervalo, sino también el valor de sus derivadas.
- Entonces puede proponerse un polinomio de interpolación:

$$p(t) = a \cdot t^3 + b \cdot t^2 + c \cdot t + d$$

con la derivada:

$$\dot{p}(t) = 3a \cdot t^2 + 2b \cdot t + c$$

La Localización de Eventos III

Métodos de la Integración Numérica a un Paso

- Otro algoritmo quizás aún más eficiente que la *iteración de Newton* para la localización de eventos es la *interpolación polinomial*.
- Como cada función de detección de eventos puede expresarse en función de variables del estado, conocemos no sólo el valor de las funciones de detección en ambos lados del intervalo, sino también el valor de sus derivadas.
- Entonces puede proponerse un polinomio de interpolación:

$$p(t) = a \cdot t^3 + b \cdot t^2 + c \cdot t + d$$

con la derivada:

$$\dot{p}(t) = 3a \cdot t^2 + 2b \cdot t + c$$

- En consecuencia sabemos que:

$$\begin{aligned}
p(t_k) &= a \cdot t_k^3 + b \cdot t_k^2 + c \cdot t_k + d = f_k \\
p(t_{k+1}) &= a \cdot t_{k+1}^3 + b \cdot t_{k+1}^2 + c \cdot t_{k+1} + d = f_{k+1} \\
\dot{p}(t_k) &= 3a \cdot t_k^2 + 2b \cdot t_k + c = \dot{f}_k = h_k \\
\dot{p}(t_{k+1}) &= 3a \cdot t_{k+1}^2 + 2b \cdot t_{k+1} + c = \dot{f}_{k+1} = h_{k+1}
\end{aligned}$$

La Localización de Eventos IV

Métodos de la Integración Numérica a un Paso

La Localización de Eventos IV

Métodos de la Integración Numérica a un Paso

- Concluimos:

$$\begin{pmatrix} f_k \\ f_{k+1} \\ h_k \\ h_{k+1} \end{pmatrix} = \begin{pmatrix} t_k^3 & t_k^2 & t_k & 1 \\ t_{k+1}^3 & t_{k+1}^2 & t_{k+1} & 1 \\ 3t_k^2 & 2t_k & 1 & 0 \\ 3t_{k+1}^2 & 2t_{k+1} & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}$$

La Localización de Eventos IV

Métodos de la Integración Numérica a un Paso

- Concluimos:

$$\begin{pmatrix} f_k \\ f_{k+1} \\ h_k \\ h_{k+1} \end{pmatrix} = \begin{pmatrix} t_k^3 & t_k^2 & t_k & 1 \\ t_{k+1}^3 & t_{k+1}^2 & t_{k+1} & 1 \\ 3t_k^2 & 2t_k & 1 & 0 \\ 3t_{k+1}^2 & 2t_{k+1} & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}$$

- De esta representación pueden calcularse los valores de los cuatro coeficientes del polinomio de interpolación fácilmente.

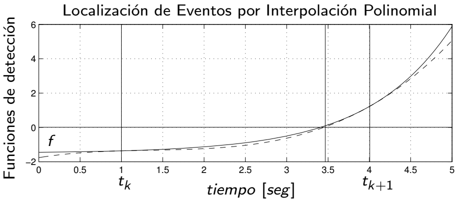
La Localización de Eventos IV

Métodos de la Integración Numérica a un Paso

- Concluimos:

$$\begin{pmatrix} f_k \\ f_{k+1} \\ h_k \\ h_{k+1} \end{pmatrix} = \begin{pmatrix} t_k^3 & t_k^2 & t_k & 1 \\ t_{k+1}^3 & t_{k+1}^2 & t_{k+1} & 1 \\ 3t_k^2 & 2t_k & 1 & 0 \\ 3t_{k+1}^2 & 2t_{k+1} & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}$$

- De esta representación pueden calcularse los valores de los cuatro coeficientes del polinomio de interpolación fácilmente.



La Localización de Eventos V

Métodos de la Integración Numérica a un Paso

La Localización de Eventos V

Métodos de la Integración Numérica a un Paso

- La *interpolación polinomial* tiene una *velocidad de convergencia cúbica* y por eso converge aún más rápidamente que la *iteración de Newton*.

La Localización de Eventos V

Métodos de la Integración Numérica a un Paso

- La *interpolación polinomial* tiene una *velocidad de convergencia cúbica* y por eso converge aún más rápidamente que la *iteración de Newton*.
- Además el algoritmo encuentra al menos una solución dentro del intervalo. En consecuencia, la *convergencia puede garantizarse* igual que en los casos de los métodos de la *Regula Falsi* y de la *sección de oro*.

La Localización de Eventos VI

Métodos de la Integración Numérica a Múltiples Pasos

La Localización de Eventos VI

Métodos de la Integración Numérica a Múltiples Pasos

- Si se trata de un *algoritmo de integración en múltiples pasos*, el algoritmo de la localización de eventos puede mejorarse aún más, porque tenemos acceso al *vector de Nordsieck*.

La Localización de Eventos VI

Métodos de la Integración Numérica a Múltiples Pasos

- Si se trata de un *algoritmo de integración en múltiples pasos*, el algoritmo de la localización de eventos puede mejorarse aún más, porque tenemos acceso al *vector de Nordsieck*.
- Podemos escribir:

$$\mathcal{F}(\hat{h}) = \mathcal{F}_i(t_{\text{próximo}}) = \mathcal{F}_i(t_{k+1}) + \hat{h}\,\frac{d\mathcal{F}_i(t_{k+1})}{dt} + \frac{\hat{h}^2}{2}\,\frac{d^2\mathcal{F}_i(t_{k+1})}{dt^2} + \frac{\hat{h}^3}{6}\,\frac{d^3\mathcal{F}_i(t_{k+1})}{dt^3} + \cdots = 0.0$$

La Localización de Eventos VI

Métodos de la Integración Numérica a Múltiples Pasos

- Si se trata de un *algoritmo de integración en múltiples pasos*, el algoritmo de la localización de eventos puede mejorarse aún más, porque tenemos acceso al *vector de Nordsieck*.
- Podemos escribir:

$$\mathcal{F}(\hat{h}) = \mathcal{F}_i(t_{\text{próximo}}) = \mathcal{F}_i(t_{k+1}) + \hat{h}\frac{d\mathcal{F}_i(t_{k+1})}{dt} + \frac{\hat{h}^2}{2}\frac{d^2\mathcal{F}_i(t_{k+1})}{dt^2} + \frac{\hat{h}^3}{6}\frac{d^3\mathcal{F}_i(t_{k+1})}{dt^3} + \cdots = 0.0$$

- Es una función en la incógnita $\hat{h}$ que puede resolverse por una iteración de Newton.

La Localización de Eventos VI

Métodos de la Integración Numérica a Múltiples Pasos

- Si se trata de un *algoritmo de integración en múltiples pasos*, el algoritmo de la localización de eventos puede mejorarse aún más, porque tenemos acceso al *vector de Nordsieck*.
- Podemos escribir:

$$\mathcal{F}(\hat{h}) = \mathcal{F}_i(t_{\text{próximo}}) = \mathcal{F}_i(t_{k+1}) + \hat{h}\,\frac{d\mathcal{F}_i(t_{k+1})}{dt} + \frac{\hat{h}^2}{2}\,\frac{d^2\mathcal{F}_i(t_{k+1})}{dt^2} + \frac{\hat{h}^3}{6}\,\frac{d^3\mathcal{F}_i(t_{k+1})}{dt^3} + \cdots = 0.0$$

- Es una función en la incógnita $\hat{h}$ que puede resolverse por una iteración de Newton.
- Empezamos con $\hat{h}^0 = 0.5 \cdot (t_k - t_{k+1})$ e iteramos:

$$\hat{h}^{\ell+1} = \hat{h}^{\ell} - \frac{\mathcal{F}(\hat{h}^{\ell})}{\mathcal{H}(\hat{h}^{\ell})}$$

donde:

$$\mathcal{H}(\hat{h}) = \frac{d\mathcal{F}(\hat{h})}{d\hat{h}} = \frac{d\mathcal{F}_i(t_{k+1})}{dt} + \hat{h}\,\frac{d^2\mathcal{F}_i(t_{k+1})}{dt^2} + \frac{\hat{h}^2}{2}\,\frac{d^3\mathcal{F}_i(t_{k+1})}{dt^3} + \ldots$$

La Localización de Eventos VII

Métodos de la Integración Numérica a Múltiples Pasos

La Localización de Eventos VII

Métodos de la Integración Numérica a Múltiples Pasos

- Como tenemos acceso al *vector de Nordsieck*, esta iteración converge con la misma precisión que la integración misma en un sólo paso. No puede esperarse más.

La Localización de Eventos VII

Métodos de la Integración Numérica a Múltiples Pasos

- Como tenemos acceso al *vector de Nordsieck*, esta iteración converge con la misma precisión que la integración misma en un sólo paso. No puede esperarse más.

- Solamente tenemos acceso al vector de Nordsieck para variables del estado. Entonces, si la función de la detección de eventos no coincide con una de las variables del estado, vale la pena, aumentar el sistema de ecuaciones diferenciales con:

$$\dot{x}_{n+i} = \frac{d\mathcal{F}_i(\mathbf{x})}{dt}$$

La Localización de Eventos VII

Métodos de la Integración Numérica a Múltiples Pasos

- Como tenemos acceso al *vector de Nordsieck*, esta iteración converge con la misma precisión que la integración misma en un sólo paso. No puede esperarse más.

- Solamente tenemos acceso al vector de Nordsieck para variables del estado. Entonces, si la función de la detección de eventos no coincide con una de las variables del estado, vale la pena, aumentar el sistema de ecuaciones diferenciales con:

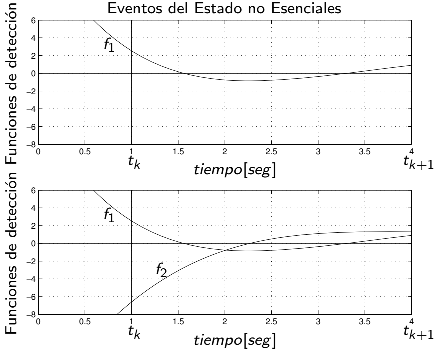
$$\dot{x}_{n+i} = \frac{d\mathcal{F}_i(\mathbf{x})}{dt}$$

- De esta manera tendremos que integrar una variable de estado adicional, pero se nos simplifica y acelera la localización de eventos.

Eventos del Estado no Esenciales

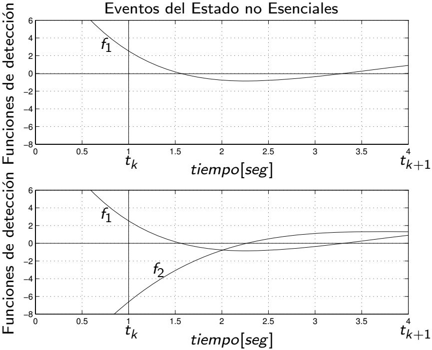
Eventos del Estado no Esenciales

A veces es buena idea, aumentar el conjunto de funciones de detección de eventos con una función adicional que es la derivada de otra.



Eventos del Estado no Esenciales

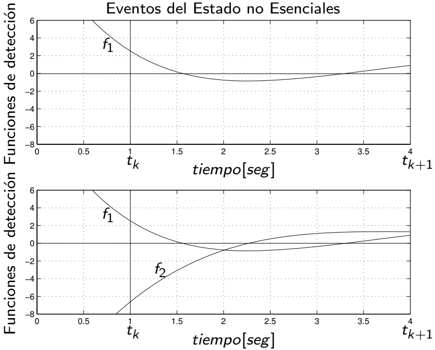
A veces es buena idea, aumentar el conjunto de funciones de detección de eventos con una función adicional que es la derivada de otra.



- Inicialmente se detecta el cruce de la función f_2.

Eventos del Estado no Esenciales

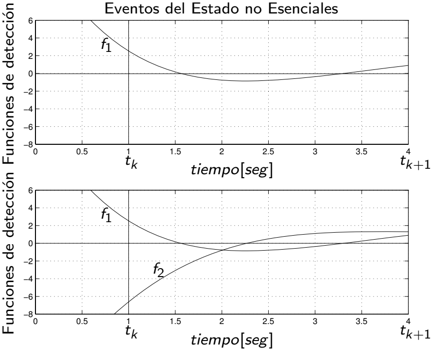
A veces es buena idea, aumentar el conjunto de funciones de detección de eventos con una función adicional que es la derivada de otra.



- Inicialmente se detecta el cruce de la función f_2.
- Iterando sobre ello se descubrirá un segundo cruce de la función f_1 más temprano.

Eventos del Estado no Esenciales

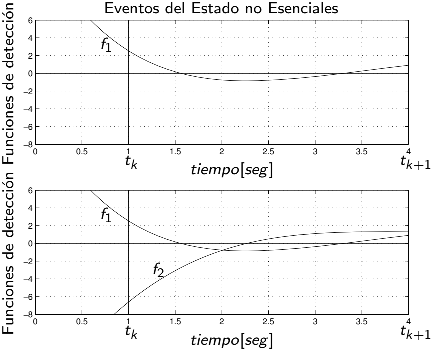
A veces es buena idea, aumentar el conjunto de funciones de detección de eventos con una función adicional que es la derivada de otra.



- Inicialmente se detecta el cruce de la función f_2.
- Iterando sobre ello se descubrirá un segundo cruce de la función f_1 más temprano.
- El evento asociado con la función f_2 se llama *no esencial*, porque no hay ninguna acción asociada con este evento.

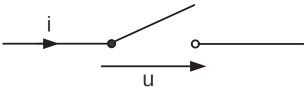
Eventos del Estado no Esenciales

A veces es buena idea, aumentar el conjunto de funciones de detección de eventos con una función adicional que es la derivada de otra.



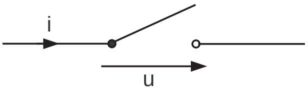
- Inicialmente se detecta el cruce de la función f_2.
- Iterando sobre ello se descubrirá un segundo cruce de la función f_1 más temprano.
- El evento asociado con la función f_2 se llama *no esencial*, porque no hay ninguna acción asociada con este evento.
- Sin embargo, sin la función de la detección adicional, f_2, hubiéramos pasado por alto los dos cruces esenciales de la función f_1.

El Conmutador Eléctrico



- El conmutador eléctrico está caracterizado por dos variables, el voltaje u y la corriente i.

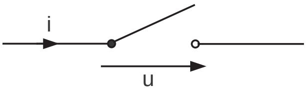
El Conmutador Eléctrico



- El conmutador eléctrico está caracterizado por dos variables, el voltaje *u* y la corriente *i*.

- Ya sabemos que las ecuaciones que se obtienen de sistemas físicos son *acausales* inicialmente.

El Conmutador Eléctrico

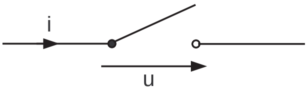


- El conmutador eléctrico está caracterizado por dos variables, el voltaje u y la corriente i.

- Ya sabemos que las ecuaciones que se obtienen de sistemas físicos son *acausales* inicialmente.
- En Dymola, la ecuación del conmutador eléctrico puede formularse de la forma siguiente:

$$0 = \textbf{ if } conmutador == abierto \textbf{ then } i \textbf{ else } u;$$

El Conmutador Eléctrico



- El conmutador eléctrico está caracterizado por dos variables, el voltaje u y la corriente i.

- Ya sabemos que las ecuaciones que se obtienen de sistemas físicos son *acausales* inicialmente.
- En Dymola, la ecuación del conmutador eléctrico puede formularse de la forma siguiente:

$$0 = \textbf{if } conmutador == abierto \textbf{ then } i \textbf{ else } u;$$

- El conmutador eléctrico también puede describirse por la ecuación algebraica:

conmutador	m_o
abierto	1
cerrado	0

$$0 = m_o \cdot i + (1 - m_o) \cdot u$$

El Conmutador Eléctrico II

Para la *ecuación de la conmutación algebraica*:

$$0 = m_o \cdot i + (1 - m_o) \cdot u$$

existen dos posibles causalizaciones:

$$i = \frac{m_o - 1}{m_o} \cdot u$$
$$u = \frac{m_o}{m_o - 1} \cdot i$$

El Conmutador Eléctrico II

Para la *ecuación de la conmutación algebraica*:

$$0 = m_o \cdot i + (1 - m_o) \cdot u$$

existen dos posibles causalizaciones:

$$i = \frac{m_o - 1}{m_o} \cdot u$$
$$u = \frac{m_o}{m_o - 1} \cdot i$$

Desafortunadamente, cada una de ellas resulta en una *división por cero* en una de las dos posiciones del conmutador.

El Conmutador Eléctrico II

Para la *ecuación de la conmutación algebraica*:

$$0 = m_o \cdot i + (1 - m_o) \cdot u$$

existen dos posibles causalizaciones:

$$i = \frac{m_o - 1}{m_o} \cdot u$$
$$u = \frac{m_o}{m_o - 1} \cdot i$$

Desafortunadamente, cada una de ellas resulta en una *división por cero* en una de las dos posiciones del conmutador.

La causalidad computacional de la ecuación de la conmutación depende de la posición del conmutador.

El Conmutador Eléctrico II

Para la *ecuación de la conmutación algebraica*:

$$0 = m_o \cdot i + (1 - m_o) \cdot u$$

existen dos posibles causalizaciones:

$$i = \frac{m_o - 1}{m_o} \cdot u$$
$$u = \frac{m_o}{m_o - 1} \cdot i$$

Desafortunadamente, cada una de ellas resulta en una *división por cero* en una de las dos posiciones del conmutador.

La causalidad computacional de la ecuación de la conmutación depende de la posición del conmutador.

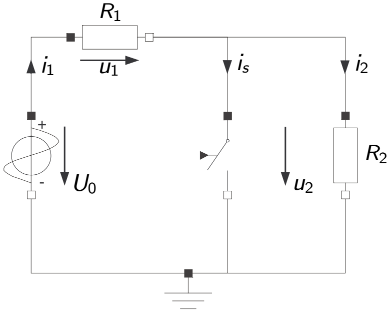
La única manera de lograr una *causalidad computacional libre* es incluyendo la ecuación de la conmutación dentro de un *bucle algebraico*.

El Conmutador Eléctrico III

Empezamos con un ejemplo. Se simula un pequeño circuito eléctrico.

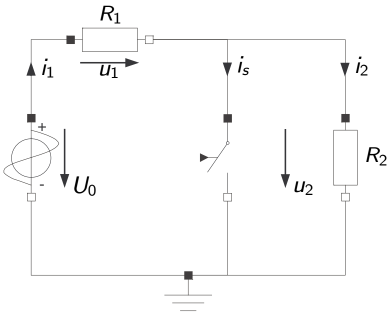
El Conmutador Eléctrico III

Empezamos con un ejemplo. Se simula un pequeño circuito eléctrico.



El Conmutador Eléctrico III

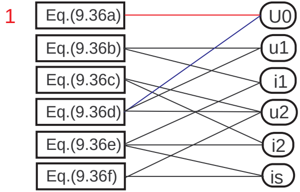
Empezamos con un ejemplo. Se simula un pequeño circuito eléctrico.



$$
\begin{aligned}
U_0 &= f(t) \\
u_1 &= R_1 \cdot i_1 \\
u_2 &= R_2 \cdot i_2 \\
U_0 &= u_1 + u_2 \\
i_1 &= i_s + i_2 \\
0 &= m_o \cdot i_s + (1 - m_o) \cdot u_2
\end{aligned}
$$

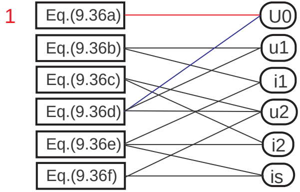
El Conmutador Eléctrico IV

$$
\begin{aligned}
U_0 &= f(t) \\
u_1 &= R_1 \cdot i_1 \\
u_2 &= R_2 \cdot i_2 \\
U_0 &= u_1 + u_2 \\
i_1 &= i_s + i_2 \\
0 &= m_o \cdot i_s + (1 - m_o) \cdot u_2
\end{aligned}
$$

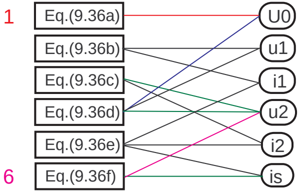


El Conmutador Eléctrico IV

$$
\begin{aligned}
U_0 &= f(t) \\
u_1 &= R_1 \cdot i_1 \\
u_2 &= R_2 \cdot i_2 \\
U_0 &= u_1 + u_2 \\
i_1 &= i_s + i_2 \\
0 &= m_o \cdot i_s + (1 - m_o) \cdot u_2
\end{aligned}
$$

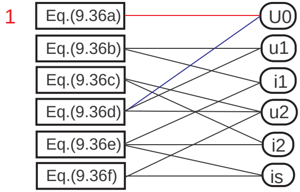


Las *ecuaciones de la conmutación* tienen que incluirse en la lista de las *ecuaciones residuales*.

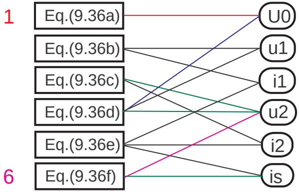


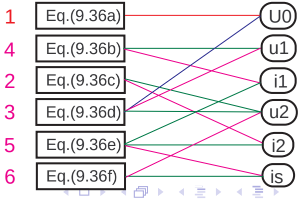
El Conmutador Eléctrico IV

$$
\begin{aligned}
U_0 &= f(t) \\
u_1 &= R_1 \cdot i_1 \\
u_2 &= R_2 \cdot i_2 \\
U_0 &= u_1 + u_2 \\
i_1 &= i_s + i_2 \\
0 &= m_o \cdot i_s + (1 - m_o) \cdot u_2
\end{aligned}
$$

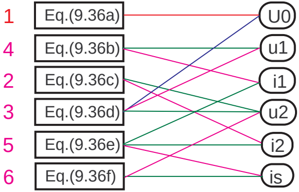


Las *ecuaciones de la conmutación* tienen que incluirse en la lista de las *ecuaciones residuales*.





El Conmutador Eléctrico V



$$U_0 = f(t)$$

$$i_2 = \frac{1}{R_2} \cdot u_2$$

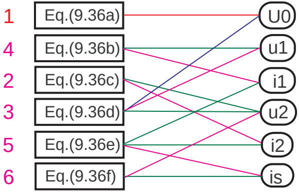
$$u_1 = U_0 - u_2$$

$$i_1 = \frac{1}{R_1} \cdot u_1$$

$$i_s = i_1 - i_2$$

$$u_2 = \frac{m_o}{m_o - 1} \cdot i_s$$

El Conmutador Eléctrico V



$$
\begin{aligned}
U_0 &= f(t) \\
i_2 &= \frac{1}{R_2} \cdot u_2 \\
u_1 &= U_0 - u_2 \\
i_1 &= \frac{1}{R_1} \cdot u_1 \\
i_s &= i_1 - i_2 \\
u_2 &= \frac{m_o}{m_o - 1} \cdot i_s
\end{aligned}
$$

Usamos substitución:

$$
\begin{aligned}
u_2 &= \frac{m_o}{m_o - 1} \cdot i_s \\
&= \frac{m_o}{m_o - 1} \cdot (i_1 - i_2) \\
&= \frac{m_o}{(m_o - 1) \cdot R_1} \cdot u_1 - \frac{m_o}{(m_o - 1) \cdot R_2} \cdot u_2 \\
&= \frac{m_o}{(m_o - 1) \cdot R_1} \cdot U_0 - \frac{m_o}{(m_o - 1) \cdot R_1} \cdot u_2 - \frac{m_o}{(m_o - 1) \cdot R_2} \cdot u_2 \\
&= \frac{m_o}{(m_o - 1) \cdot R_1} \cdot U_0 - \frac{m_o \cdot (R_1 + R_2)}{(m_o - 1) \cdot R_1 \cdot R_2} \cdot u_2
\end{aligned}
$$

El Conmutador Eléctrico VI

Resolviendo para u_2 obtenemos:

$$u_2 = \frac{m_o \cdot R_2}{m_o \cdot (R_1 + R_2) + (m_o - 1) \cdot R_1 \cdot R_2} \cdot U_0$$

Esta ecuación no tiene división por cero en ninguna de las dos posiciones del conmutador.

El Conmutador Eléctrico VI

Resolviendo para u_2 obtenemos:

$$u_2 = \frac{m_o \cdot R_2}{m_o \cdot (R_1 + R_2) + (m_o - 1) \cdot R_1 \cdot R_2} \cdot U_0$$

Esta ecuación no tiene división por cero en ninguna de las dos posiciones del conmutador.

Las ecuaciones pueden escribirse entonces de la forma siguiente:

$$\begin{aligned}
U_0 &= f(t) \\
u_2 &= \frac{m_o \cdot R_2}{m_o \cdot (R_1 + R_2) + (m_o - 1) \cdot R_1 \cdot R_2} \cdot U_0 \\
i_2 &= \frac{1}{R_2} \cdot u_2 \\
u_1 &= U_0 - u_2 \\
i_1 &= \frac{1}{R_1} \cdot u_1 \\
i_s &= i_1 - i_2
\end{aligned}$$

El Conmutador Eléctrico VI

Resolviendo para u_2 obtenemos:

$$u_2 = \frac{m_o \cdot R_2}{m_o \cdot (R_1 + R_2) + (m_o - 1) \cdot R_1 \cdot R_2} \cdot U_0$$

Esta ecuación no tiene división por cero en ninguna de las dos posiciones del conmutador.

Las ecuaciones pueden escribirse entonces de la forma siguiente:

$$\begin{aligned}
U_0 &= f(t) \\
u_2 &= \frac{m_o \cdot R_2}{m_o \cdot (R_1 + R_2) + (m_o - 1) \cdot R_1 \cdot R_2} \cdot U_0 \\
i_2 &= \frac{1}{R_2} \cdot u_2 \\
u_1 &= U_0 - u_2 \\
i_1 &= \frac{1}{R_1} \cdot u_1 \\
i_s &= i_1 - i_2
\end{aligned}$$

Estas ecuaciones no tienen ni un bucle algebraico ni una división por cero. Entonces pueden simularse sin problemas.

Conclusiones

Mostramos que modelos de la ingeniería a menudo tienen discontinuidades.

Conclusiones

Mostramos que modelos de la ingeniería a menudo tienen discontinuidades.

Si las consecuencias de las discontinuidades son graves, el método del abuso del control del paso de integración para su localización no funciona correctamente.

Conclusiones

Mostramos que modelos de la ingeniería a menudo tienen discontinuidades.

Si las consecuencias de las discontinuidades son graves, el método del abuso del control del paso de integración para su localización no funciona correctamente.

Se mostró que una simulación correcta necesita de una descripción híbrida del modelo acompañada por una simulación a tramos.

Conclusiones

Mostramos que modelos de la ingeniería a menudo tienen discontinuidades.

Si las consecuencias de las discontinuidades son graves, el método del abuso del control del paso de integración para su localización no funciona correctamente.

Se mostró que una simulación correcta necesita de una descripción híbrida del modelo acompañada por una simulación a tramos.

Hablamos de las consideraciones numéricas asociadas con el aislamiento y la localización de eventos.

Conclusiones

Mostramos que modelos de la ingeniería a menudo tienen discontinuidades.

Si las consecuencias de las discontinuidades son graves, el método del abuso del control del paso de integración para su localización no funciona correctamente.

Se mostró que una simulación correcta necesita de una descripción híbrida del modelo acompañada por una simulación a tramos.

Hablamos de las consideraciones numéricas asociadas con el aislamiento y la localización de eventos.

Terminamos con la discusión sobre la ecuación de conmutación que necesita un preprocesamiento simbólico previo a que la simulación a tramos pueda funcionar.