

ALGORITMO GENÉTICO PARA DECISIONES INTELIGENTES EN CRUCES DE VEHÍCULOS CON FUNCIÓN DE COSTE DINÁMICA

E. Onieva¹, J. Villagrà¹, V. Milanés¹, J. Godoy¹

¹Programa AUTOPIA, Centro de Automática y Robótica (UPM-CSIC) La Poveda-Arganda del Rey, 28500, Madrid,
{enrique.onieva,jorge.villagra,vicente.milanes,jorge.godoy}@csic.es

Resumen

Este trabajo analiza el caso donde un vehículo autónomo debe cooperar con otro manual para llevar a cabo una maniobra de cruce. La dificultad del escenario es la unidireccionalidad de la cooperación a llevar a cabo por el vehículo autónomo, dado que no puede asumir que el manual respetará las normas básicas de circulación en cruces.

La velocidad a seguir por el vehículo autónomo es calculada por un sistema basado en reglas difusas (SBRD); que será optimizado por medio de un algoritmo genético. La función de coste será calculada mediante la simulación incremental de escenarios de cruce, y evaluando si las decisiones tomadas son adecuadas para el escenario. Una vez calculada la velocidad a seguir, ésta se manda a una capa de bajo nivel, encargada de accionar los pedales del vehículo.

Palabras Clave: Algoritmos Genéticos, Lógica Difusa, Control de Vehículos, Cruces Inteligentes.

1 INTRODUCCIÓN

Los accidentes de tráfico en intersecciones suponen un alto coste en términos de muertes, lesiones y daño a la propiedad. La seguridad en intersecciones es vista como un área de investigación con alta prioridad, dado que los accidentes en intersecciones suponen entre un 25 % y 45 % del total de accidentes [7] y, dado que las intersecciones representan una proporción pequeña del trazado viario, esto supone una cantidad altamente desproporcionada.

Los Sistemas Inteligentes de Transporte aplican técnicas de robótica e inteligencia artificial para lograr una conducción

segura y eficiente [2]. Los estudios en conducción autónoma pueden dividirse en: (i) control de velocidad [9], sobre el acelerador y freno (y en algunos casos el cambio de marcha); (ii) control de dirección, sobre el volante [6].

En carretera abierta, los vehículos pueden ser completamente autónomos mediante una combinación apropiada de los sistemas mencionados. No obstante, hay una ligera necesidad de incluir comportamientos más avanzados cuando tratamos con entornos urbanos [3]. En estos entornos, puede darse el caso en que, aún actuando de manera *óptima* el vehículo se vea involucrado en una situación en la que un accidente es inevitable.

El presente trabajo está motivado por la futura necesidad de convivencia entre vehículos manuales y autónomos, y de cómo estos últimos deberán actuar bajo cooperación incompleta. Para ilustrar esto, proponemos un escenario en el que dos vehículos se aproximan a una intersección sin señalizar; uno de ellos conducido por una persona y otro completamente autónomo. Las acciones a llevar a cabo por el primero son desconocidas para el autónomo, que únicamente tendrá la visión del estado actual del otro y deberá actuar unilateralmente para evitar riesgos. En el caso de dos vehículos autónomos, ambos llevarían a cabo una cooperación, cediendo uno de ellos el paso, o adaptando sus velocidades [5]. Pero, dado que uno de ellos es conducido manualmente, esta cooperación se hace imposible.

Por ello proponemos un sistema de dos capas capaz de: (i) calcular la velocidad a seguir por el vehículo autónomo para realizar la maniobra de forma segura y eficaz y (ii) actuar sobre los pedales de éste para seguir la velocidad dada por la capa anterior.

La primera estará implementada por medio de un sistema basado en reglas difusas (SBRD) [10] de cuatro entradas referentes al estado del escenario.

Dichos controladores serán optimizados por medio de un algoritmo genético [4], que los evaluará en un conjunto de escenarios de cruce y calculará un coste parcial de acuerdo con resultados de simulación. Por ejemplo: si el con-

trolador provoca una colisión, éste será penalizado; en caso contrario, será recompensado en función de su efectividad. En total se optimizarán 81 estructuras de controladores, con diferentes granularidades para las entradas; por otra parte, la velocidad a seguir será dada en términos relativos (incremento de velocidad) o absolutos (velocidad a seguir); obteniéndose así 162 tipos de controladores.

La principal novedad en el diseño del sistema presentado radica en la elección de la función de coste. Ésta utiliza un conjunto de escenarios creciente y cambiante generación tras generación, con lo que se consigue reducir el tiempo de ejecución, así como maximizar los casos a considerar por el controlador. La función de coste evaluará al seguridad y eficacia de los controladores en cada uno de los escenarios.

2 ARQUITECTURA DE CONTROL PROPUESTA

Supondremos que el sistema de control dispone de las posiciones, direcciones y velocidades de ambos vehículos. En caso de no disponer de un mapa que nos indique dónde se encuentra la intersección, las distancias de ambos vehículos al punto de cruce deben ser calculadas. Las siguientes subsecciones detallan las dos capas de la arquitectura de control.

2.1 Determinación de la Velocidad Adecuada

Ambas distancias al punto de cruce ($D_{\{A|M\}}$) y velocidades ($S_{\{A|M\}}$) son utilizadas como entradas de un SBRD, éstas están representadas gráficamente en la figura 1.

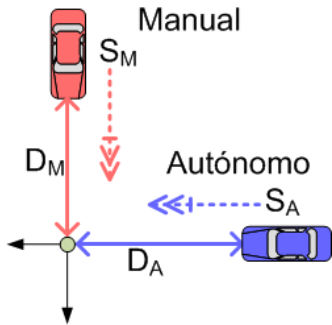


Figura 1: Cálculo del punto de cruce de ambos vehículos.

Todas las variables de entrada se codifican por medio de etiquetas triangulares uniformemente distribuidas en los intervalos $[-10, 100]m$ para las distancias y $[0, 50]km/h$ para las velocidades. Cada variable tiene su propio número de etiquetas a usar en la fuzzificación ($\#D_M, \#D_A, \#S_M, \#S_A$).

La base de reglas ($\#D_M \cdot \#D_A \cdot \#S_M \cdot \#S_A$) reglas formadas mediante la composición (operador Y) de todas las variables. Así pues, la estructura del conjunto de antecedentes quedaría de la forma:

SI ($D_M = D_M^i$) Y ($D_A = D_A^j$) Y ($S_M = S_M^k$) Y ($S_A = S_A^l$) donde $i = \{1 \dots \#D_M\}$, $j = \{1 \dots \#D_A\}$, $k = \{1 \dots \#S_M\}$ y $l = \{1 \dots \#S_A\}$.

La salida del sistema, llamada *out*, podrá representar: (i) La modificación requerida sobre la velocidad actual del vehículo. $S_{ref} = S_A + out$. (ii) La velocidad absoluta necesaria. $S_{ref} = out$. La salida será codificada por medio de *#out* singletons.

Finalmente, la asignación de singletons de salida a antecedentes de la base de reglas es codificada por medio de un vector, para así poder ser manejado por el algoritmo genético: $Ind_i \in \{1, \dots, \#out\}$, $\forall i \in \{1, \dots, \#D_M \cdot \#D_A \cdot \#S_M \cdot \#S_A\}$. Donde el valor asignado a Ind_i significa usar dicho singleton como consecuente de la regla número i .

2.2 Comportamiento Longitudinal del Vehículo

El comportamiento longitudinal de un vehículo autónomo puede ser aproximado (para dinámicas lentamente cambiantes y superficie plana) por la función de transferencia de segundo orden:

$$G(s) = \frac{K\omega_n^2}{s^2 + 2\eta\omega_n s + \omega_n^2}$$

donde $K = 25.14$, $\eta = 160$, y $\omega_n = 55.87$ (véase [8]).

Para alcanzar las velocidades de referencia dadas por el SBRD, se implementa un controlador PI que, aparte de ser una técnica sencilla y bien conocida [1], permite describir el comportamiento en el dominio de Laplace como $C(s) = K_P E(s) + K_I \frac{E(s)}{s}$, siendo $E(s)$ el error de velocidad y usando $K_P = 0.3$ y $K_I = 0.1$ como ganancias.

Finalmente, la dinámica del sistema en lazo cerrado es discretizada y reescrita como la función lineal:

$$S_A(t_k) = a_2 S_{ref}(t_{k-1}) + a_1 S_{ref}(t_{k-2}) + a_0 S_{ref}(t_{k-3}) - b_2 S_A(t_{k-1}) - b_1 S_A(t_{k-2}) - b_0 S_A(t_{k-3})$$

con $a_0 = -5.467 \cdot 10^{-5}$, $a_1 = -0.2041$, $a_2 = 0.2495$, and $b_0 = 0$, $b_1 = 0.7421$, $b_2 = -1.697$, y donde $S_{ref}(t_k)$ y $S_A(t_k)$ representan las velocidades de referencia y actuales del vehículo en el instante t_k .

3 PROCESO DE OPTIMIZACIÓN GENÉTICO

El algoritmo genético optimiza la asignación de consecuentes al conjunto completo de antecedentes del SBRD (sección 2.1).

Se utiliza un esquema clásico de algoritmo genético estacionario, el cual, en cada generación, selecciona dos padres, aplica sobre ellos los operadores de cruce y mutación para generar dos hijos y reemplaza (en caso) dos individuos de la generación actual con ellos.

El proceso comienza con una población de 100 individuos generados aleatoriamente y se ejecuta durante 1000 generaciones. Los operadores implementados son:

- *Selección estocástica*: proporcional al valor de la función objetivo de un individuo.
- *Cruce uniforme*: para cada posición del vector hijo, se hace una selección aleatoria sobre de cuál de los padres el valor debe ser heredado. El segundo hijo es generado usando las elecciones inversas.
- *Mutación simple*: cada gen se cambia por un valor permitido con $p_m = 2/(\#D_M \cdot \#D_A \cdot \#S_M \cdot \#S_A)$.
- *Reemplazamiento*: cada uno de los hijos generados reemplazará a uno de sus padres en caso de que el valor de la función objetivo mejore.

3.1 Función Objetivo

Para evaluar el comportamiento de un controlador, éste será ejecutado en un número creciente de escenarios:

$$N_{sc} = 1 + 19 \cdot \lfloor g/G \rfloor$$

, representando g y G , respectivamente, la generación actual y el número total de generaciones. Dado que el algoritmo se ejecuta durante 1000 generaciones, se generarán un total de 10510 escenarios diferentes, a lo largo de todo el proceso.

En los escenarios, ambos vehículos se aproximan perpendicularmente al punto de cruce comenzando ambos a distancias y con velocidades aleatorias:

$$\{D_M, D_A\} = U(50, 80) \quad \{S_M, S_A\} = U(10, 40)$$

Cada escenario evolucionará dos veces:

1. Sin considerar las referencias dadas por el SBRD (ejecución libre o E_l), donde el vehículo mantiene constante su velocidad.
2. Siguiendo las referencias (ejecución controlada o E_c).

El propósito de E_l es el de determinar si se produciría una colisión si el vehículo autónomo no variara su velocidad; mientras que el de E_c es determinar si dicha colisión es evitada gracias a las indicaciones del SBRD, y su eficacia.

Cada una de las evoluciones del escenario inicial puede llevar a tres resultados diferentes según el momento en que el vehículo autónomo entra/sale de una *zona segura* definida 5 metros antes y después de punto de cruce:

1. Los vehículos no coinciden en la zona segura (no hay colisión, C_0). En este caso se espera que el SBRD haga que el vehículo no altere la velocidad.

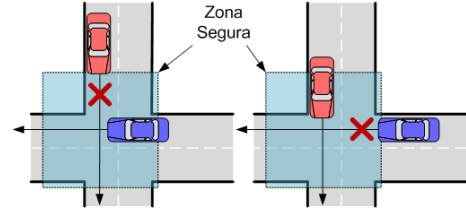


Figura 2: Situación con colisión lateral (izquierda) y frontal (derecha).

2. Se produce una colisión lateral, es decir ambos coinciden en la zona segura pero el autónomo ha entrado antes, C_L . Aquí se espera que el SBRD aumente ligeramente la velocidad, para salir de la zona segura antes de que el manual entre.
3. Se produce una colisión frontal ambos coinciden en la zona segura, habiendo entrado el manual antes, C_F . En este caso el SBRD deberá reducir la velocidad, para entrar en la zona segura una vez el manual haya salido.

Los casos que implican colisión se muestran gráficamente en la figura 2.

Dado que el esquema de generación de situaciones iniciales hace que sea más probable (sobre el 75% de las veces) que se genere un escenario sin colisión en ejecución libre ($E_l = C_0$), dicha generación es restringida a que se generen situaciones que evolucionen a cada uno de los tres posibles casos con la misma probabilidad:

$$P(E_l \rightarrow C_0) = P(E_l \rightarrow C_L) = P(E_l \rightarrow C_F) = 1/3$$

La función objetivo será calculada por medio de la suma de objetivos parciales, representando cada uno de ellos la penalización obtenida en cada escenario inicial:

$$Fitness = \sum_{i=1, \dots, N_{sc}} F(E_l^i, E_c^i)$$

donde el resultado parcial se calcula en función de las nueve posibles combinaciones de resultados ($E_{l|c}^i = C_{O|L|F}$) de la siguiente manera:

- No se produce colisión alguna ($E_l^i = E_c^i = C_0$); entonces la penalización será menor cuanto menos se haya alterado la velocidad inicial, usando $F = |\int S_A^l - \int S_A^c|$. En la figura 3 se ilustra este criterio.
- El SBRD evitan una colisión lateral ($E_l^i = C_L$ Y $E_c^i = C_0$); la penalización será de $F = |\int S_A^l - \int S_A^c|$ si se ha aumentado la velocidad inicial ($\int S_A^c > \int S_A^l$) y de $F = 2500$ en caso contrario.
- La situación especular se produce cuando ($E_l^i = C_F$ Y $E_c^i = C_0$). En este caso, la penalización será de $F = |\int S_A^l - \int S_A^c|$ si se ha reducido la velocidad ($\int S_A^c < \int S_A^l$) y de $F = 2500$ en caso contrario.

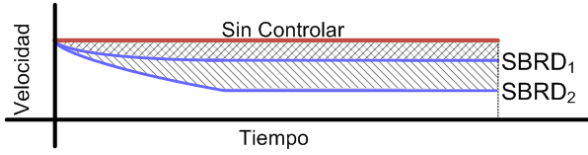


Figura 3: Ejecución de dos controladores: $SBRD_1$ frena menos que $SBRD_2$, por tanto recibe menor $|\int S_A^f - \int S_A^c|$.

- El SBRD no evita una colisión ($E_l^i = C_{L|F}$ Y $E_c^i = C_{L|F}$); en este caso se penaliza con $F = 5000$.
- El SBRD produce una colisión ($E_l^i = C_0$ Y $E_c^i = C_{L|F}$); en este caso se aplica la máxima penalización de $F = 10000$, dado que la colisión se ha producido por seguir las consignas del SBRD.

Las penalizaciones constantes $F = \{2500, 5000, 10000\}$ se establecen a partir del hecho que, en una ejecución promedio en la que el SBRD detiene inmediatamente el vehículo, se obtendría $F = |\int S_A^f - \int S_A^c| \approx 2000$ por eso, 2500 representa ligeramente más penalización que dicha ejecución, 5000 representa algo más del doble de penalización y 10000 representa unas cinco veces más penalización que un SBRD promedio con política de *detenerse siempre*. Estos valores se establecieron empíricamente.

El objetivo del proceso de ajuste será el de minimizar la suma de las funciones objetivo parciales. Finalmente resaltar que, dado que la función objetivo no es determinística (el número y situación inicial de escenarios cambia en cada generación), cada hijo es evaluado en los mismos escenarios que sus padres. Por otra parte el proceso de evaluación se repite en cada generación sobre toda la población, dado que un individuo con buen comportamiento en los escenarios de una generación no tiene por qué ser bueno en los de la siguiente.

4 EXPERIMENTACIÓN Y RESULTADOS

El número de etiquetas triangulares para codificar cada una de las variables de entrada utilizado fue 2, 3, y 4, o, de manera formal:

$$\{\#D_M, \#D_A, \#S_M, \#S_A\} \in \{2, 3, 4\}^4$$

, generándose así, 81 configuraciones de granularidad.

El número de consecuentes usados, así como sus rangos dependen del tipo de controlador a implementar:

- Para controladores que infieren la velocidad de manera relativa tenemos que $\#out = 3$, llamados: *frenar* en -10, *mantener* en 0 y *acelerar* en 10.
- Para controladores que devuelven la velocidad absoluta a seguir tenemos que $\#out = 4$, llamados: *para* en 0, *despacio* en 10, *medio* en 20 y *rápido* en 30.

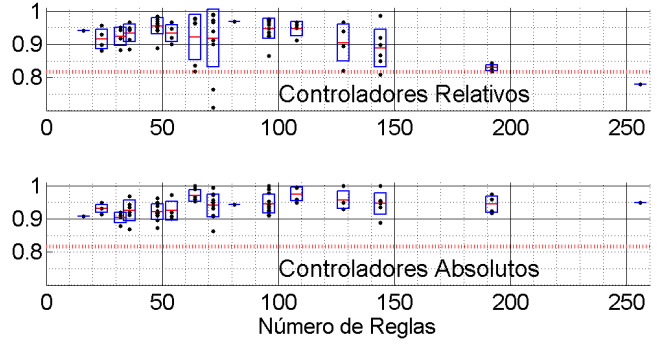


Figura 4: Seguridad de los Controladores. Las líneas rojas representan la media para un determinado número de reglas, las azules la desviación estándar. La línea roja punteada muestra el valor en ejecución sin controlar.

Cada ejecución del algoritmo genético fue repetida 10 veces. De ahora en adelante, usaremos $(REL|ABS)_{(\#D_M, \#D_A, \#S_M, \#S_A)}$ como resultado promedio de controladores de relativos y absolutos, respectivamente.

En total, el proceso se ejecuta 162 veces usando las 2 clases de SBRD (REL y ABS) y las 81 granularidades.

Finalmente, los controladores resultantes fueron probados en escenarios generados mediante combinación de distancias al punto de cruce y velocidades iniciales, de la forma:

$$\begin{aligned} [D_A, D_M]_0 &\in \{50, 55, 60, 65, 70, 75, 80\}^2 \\ [S_M, S_A]_0 &\in \{10, 15, 20, 25\}^2 \end{aligned}$$

haciendo así, un total de $7 \cdot 7 \cdot 4 \cdot 4 = 784$ escenarios de prueba. Los resultados serán comparados en términos de seguridad y eficacia en la realización de la maniobra.

4.1 Seguridad de los Controladores

En la figura 4 se muestra el porcentaje de escenarios finalizados sin colisión (sobre los 784), con los SBRD agrupados por número de reglas.

A primera vista puede decirse que con los controladores absolutos se obtienen mejores resultados; por otra parte, algunos controladores relativos obtienen peores resultados que no actuar sobre la velocidad inicial del vehículo. Cuatro de ellos obtienen una seguridad del 100%, que son: ABS_{4423} , ABS_{3433} , ABS_{3344} y ABS_{2442} .

Esta aparente dependencia entre seguridad y *tipo de controlador* puede llevar a confusión. Por ello, la figura 5 muestra la seguridad de los controladores en un plano, donde los ejes X-Y representan la seguridad de una determinada granularidad para controladores del tipo ABS y REL, respectivamente. No existen diferencias significativas entre granularidades que funcionan mejor con un esquema que con otro (46% frente a 54%); por ello se deduce que algunas configuraciones de granularidad se comportan mejor

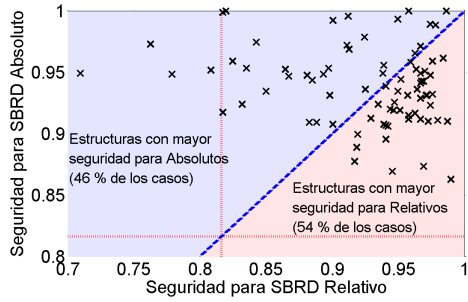


Figura 5: Correlación entre seguridad obtenida para una granularidad. La línea azul limita la región de no diferencia y las rojas, la seguridad en ejecución libre.

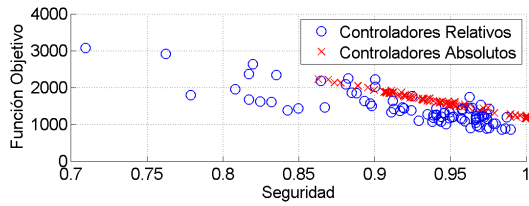


Figura 6: Correlación entre seguridad y función objetivo.

que otras, independientemente del tipo de salida implementada.

4.2 Eficacia de los Controladores

Esta sección pretende analizar los resultados para garantizar que los controladores, aparte de seguros, son eficaces. Por ejemplo: *detener siempre* el vehículo tendría máxima seguridad (nunca colisionaría), pero sería ineficaz.

En la figura 6 se muestra la correlación entre seguridad y eficacia, medida en términos de función objetivo (sección 3.1), ya que ésta recompensa individuos que llevan a cabo la maniobra sin provocar colisión, en el menor tiempo posible y respetando siempre el orden en que deben cruzar los vehículos. Se puede observar que el valor de función objetivo decrece casi linealmente conforme aumenta la seguridad; lo que implica que los individuos con bajo nivel de riesgo están actuando de manera *inteligente* para evitar colisiones.

4.3 Evaluación Cualitativa

En esta sección se pretende evaluar la manera en que ciertos SBRD evitan situaciones de riesgo. Se analizarán los sistemas $\{ABS|REL\}_{\{3344|4242\}}$ dado que la seguridad de $ABS_{\{3344|4242\}}$ es del 100%, mientras que la de $REL_{\{3344|4242\}}$ es del 98% y 99%; por lo que parece que usar las granularidades $\{3344|4242\}$ es lo más acertado para el diseño de SBRD para llevar a cabo la maniobra.

Estos cuatro SBRD se prueban en cuatro situaciones, donde ambos vehículos comienzan a 50m del punto de cruce

($D_{M|A} = 50$), con diferentes velocidades. Las evoluciones se muestran en las figuras 7 a 10, que pasamos a analizar en adelante. En dichas figuras, la línea azul representa la evolución de la distancia en ejecución libre; las líneas rojas representan valores del vehículo manual; la franja gris representa la zona segura ($\pm 5m$) y, finalmente, el área rosada representa el intervalo durante el cual la zona segura es ocupada por el vehículo manual.

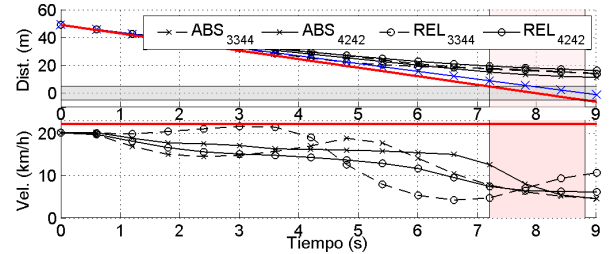


Figura 7: Evolución con el vehículo autónomo comenzando ligeramente más despacio ($E_I = C_F$).

La figura 7 muestra un experimento que llevaría a una colisión frontal, ya que el vehículo manual entra primero al área de cruce. No se encuentran diferencias mayores entre los comportamientos mostrados por los controladores. Todos ellos reducen la velocidad para evitar la colisión (sin detenerse), siendo REL_{3344} el único que acelera una vez pasado el riesgo.

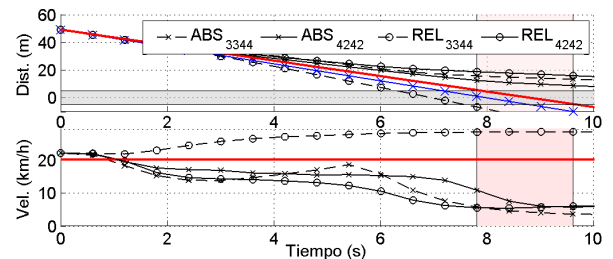


Figura 8: Evolución con el vehículo autónomo comenzando ligeramente más rápido ($E_I = C_L$).

En el segundo caso (figura 8) se produciría una colisión lateral, por lo que se espera que el vehículo autónomo acelere para abandonar la zona de riesgo antes de que entre el manual; únicamente REL_{3344} muestra este comportamiento mientras el resto, aunque consiguen evitar la colisión, lo hacen por medio de reducir la velocidad del vehículo.

Finalmente, las figuras 9 y 10 representan situaciones donde no existe riesgo de colisión, dado que el vehículo autónomo circula a mayor o menor velocidad que el manual de manera considerable y ambos no coinciden en la zona de riesgo. En estos casos se espera que los controladores varíen en la menor medida de lo posible la velocidad; lo cual es obtenido en mayor medida por REL_{3344} . Por tanto es éste controlador el que parece implementar el comportamiento que consideramos más inteligente, de una manera global.

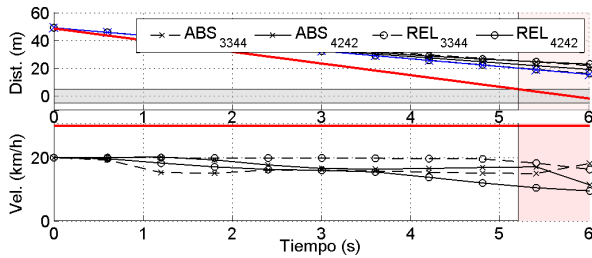


Figura 9: Evolución con el vehículo autónomo comenzando más despacio; sin riesgo de colisión ($E_l = C_0$).

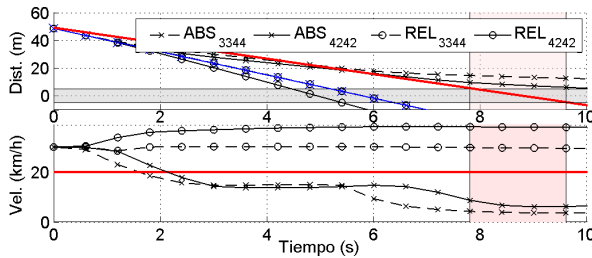


Figura 10: Evolución con el vehículo autónomo comenzando más rápido; sin riesgo de colisión ($E_l = C_0$).

5 Conclusiones y Trabajos Futuros

Este trabajo presenta un proceso de optimización genético de controladores difusos. Su objetivo es guiar un vehículo al aproximarse a un cruce donde otro vehículo no coopera para hacer la maniobra de manera óptima.

Está estructurado en dos capas. La primera de ellas fue un SBRD encargado de calcular la velocidad apropiada para circular sin riesgo, el cual fue objeto de optimización. La segunda representa la capa de bajo nivel del vehículo; la evolución de la dinámica de éste de acuerdo con las consignas recibidas.

Se han optimizado un total de 162 SBRD, con 81 granularidades de las variables de entrada y 2 significados de la salida (velocidad absoluta o relativa).

Para evaluar individuos, se implementa una función que simula la evolución del vehículo sin controlarlo y controlado en un número creciente de escenarios, para así considerar si: (i) el controlador evita una colisión que se hubiera producido sin él; (ii) el controlador no puede evitar dicha colisión; (iii) el controlador provoca una colisión que no se hubiera producido sin él. Con esto, la función objetivo valora la seguridad y eficacia del controlador durante la maniobra, recompensándolo o penalizándolo en consecuencia.

Los controladores obtenidos son sometidos a un amplio estudio, evaluándolos en 784 escenarios y comparándolos en términos de seguridad y eficacia, para así asegurarnos que dichos controladores no implementan una política de *parar siempre*, la cual sería segura, pero poco inteligente.

Trabajos futuros irán orientados en la línea de la implementación de controladores más complejos para, por ejemplo, controlar también el volante del vehículo en caso de colisión inevitable, y realizar un giro para evitarlo (o minimizar consecuencias). Finalmente, se espera coordinar el presente sistema con otros propios de vehículos autónomos, como control de distancia con un vehículo precedente.

Agradecimientos

Este trabajo ha sido realizado gracias al CYCIT y al Plan Nacional a través de sus proyectos GUIADE (P9/08) y TRANSITO (TRA2008-06602-C03-01).

Referencias

- [1] K. Aström and T. Häggglund: Advanced PID Controllers. *International Society of Automation*, 2006.
- [2] R. Bishop: Intelligent vehicle technology and trends. *Artech House*, 2005.
- [3] K. Dresner and P. Stone: A multiagent approach to autonomous intersection management. *Journal of Artificial Intelligence Research*, 31(1), pp. 591–656, 2008.
- [4] D. Goldberg: Genetic algorithms in search, optimization, and machine learning. *Addison-wesley*, 1989.
- [5] V. Milanés, J. Pérez, E. Onieva, and C. González: Controller for urban intersections based on wireless communications and fuzzy logic. *IEEE Transactions on Intelligent Transportation Systems*, 11(1), pp. 234–248, 2010.
- [6] J. Pérez, V. Milanés, and E. Onieva: Cascade architecture for lateral control in autonomous vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 12(1), pp. 73-82, 2011.
- [7] M. Staubach: Factors correlated with traffic accidents as a basis for evaluating advanced driver assistance systems. *Accident Analysis and Prevention*, 41(5), pp. 1025–1033, 2009.
- [8] I. Tejado, V. Milanés, J. Villagra, H. Godoy, J. HosseinNia, and B. M. Vinagre: Low speed control of an autonomous vehicle by using a fractional pi. *2011 IFAC World Congress*. 2009.
- [9] B. Yang, L. Keqiang, and F. Nenglian: Feedback Linearization Tracking Control of Vehicle Longitudinal Acceleration Under Low-Speed Conditions. *Journal of Dynamic Systems, Measurement, and Control*, 130(5), pp. 51–62, 2008.
- [10] L. Zadeh: Fuzzy sets. *Information and Control*, 8(3), pp. 338-353, 1965.