

Control de intersecciones semaforizadas aplicando aprendizaje por refuerzo multiagente

Carolina Higuera Arias

Asesor: Ph.D Fernando Enrique Lozano Martinez

Universidad de los Andes

Diciembre 16, 2016



Congestión vehicular en Bogotá

Causas

- Incremento del parque automotor
- Atraso en la infraestructura vial
- Semáforos mal programados

Consecuencias

- Altos tiempos de espera
- Problemas económicos
- Problemas ambientales

Pregunta de investigación

¿Cómo mejorar la infraestructura actual, de tal manera que se haga un uso inteligente y óptimo de los semáforos de la malla vial?

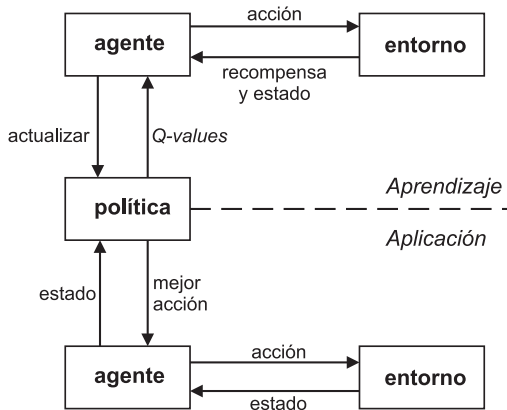
Obtener una estrategia de control con las siguientes características:

- Accionado por el tránsito
- Independiente del modelo matemático del sistema
- Que busque minimizar objetivos específicos para el sistema

} Aplicar aprendizaje por refuerzo multiagente (MARL)

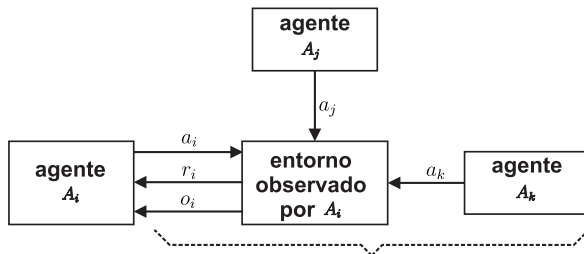
Aprendizaje por refuerzo

Caso de un solo agente:



Aprendizaje por refuerzo

Caso multiagente:



No cumple propiedad de Markov

a_x Acción del agente A_x

r_i Recompensa recibida por el agente A_i

o_i Estado observado por el agente A_i

- Se puede describir todo el sistema como un MDP multiagente colaborativo.

Aprendizaje por refuerzo

Caso multiagente:

Sistema multiagente descrito principalmente por:

- Un tiempo discreto k
- Un grupo de n agentes A_1, A_2, \dots, A_n
- Un conjunto finito de estados $\mathbf{s}^k \in \mathcal{S}$
- Un conjunto de acciones conjuntas $\mathbf{a}^k \in \mathcal{A}$
- Una función de recompensa $R_i : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ que entrega a cada agente i una recompensa numérica r_i^k

$$\text{Donde } \mathbf{R}(\mathbf{s}^k, \mathbf{a}^k) = \sum_{i=1}^n R_i(\mathbf{s}^k, \mathbf{a}^k)$$

Aprendizaje por refuerzo

Caso multiagente:

Teniendo en cuenta $Q(\mathbf{s}, \mathbf{a}) = E \{ R^k | \mathbf{s}^k = \mathbf{s}, \mathbf{a}^k = \mathbf{a} \}$

Es posible descomponer la función Q del sistema en una combinación lineal de funciones por agente:

$$Q(\mathbf{s}, \mathbf{a}) = \sum_{i=1}^n Q_i(\mathbf{s}_i, a_i)$$

Regla de actualización para el caso multiagente:

$$Q_i^{k+1}(\mathbf{s}_i^k, a_i^k) = (1 - \alpha^{k+1}) Q_i^k(\mathbf{s}_i^k, a_i^k) + \alpha^{k+1} \left[r_i^{k+1} + \gamma \max_{\mathbf{a}' \in \mathcal{A}} Q^k(\mathbf{s}^{k+1}, \mathbf{a}') \right]$$

Aprendizaje por refuerzo

Caso multiagente:

Teniendo en cuenta $Q(\mathbf{s}, \mathbf{a}) = E \{ R^k | \mathbf{s}^k = \mathbf{s}, \mathbf{a}^k = \mathbf{a} \}$

Es posible descomponer la función Q del sistema en una combinación lineal de funciones por agente:

$$Q(\mathbf{s}, \mathbf{a}) = \sum_{i=1}^n Q_i(\mathbf{s}_i, a_i)$$

Regla de actualización para el caso multiagente:

$$Q_i^{k+1}(\mathbf{s}_i^k, a_i^k) = (1 - \alpha^{k+1}) Q_i^k(\mathbf{s}_i^k, a_i^k) + \alpha^{k+1} \left[r_i^{k+1} + \gamma \max_{\mathbf{a}' \in \mathcal{A}} Q^k(\mathbf{s}^{k+1}, \mathbf{a}') \right]$$

Emerge la necesidad de coordinación de acciones entre agentes para maximizar la recompensa global a largo plazo

Problema de coordinación

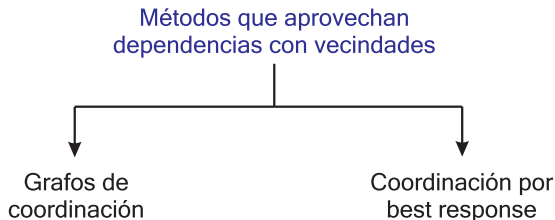
Encontrar $\mathbf{a}' = \operatorname{argmax}_{\mathbf{a}' \in \mathcal{A}} Q(\mathbf{s}, \mathbf{a}')$

Enfoques para establecer coordinación

Principio de localidad

Dos objetos suficientemente alejados uno de otro no pueden influirse mutuamente de manera instantánea.

En el sistema de tránsito: la acción de cada agente influye en mayor medida en el estado percibido alrededor de su vecindad.



Método 1: Q-Learning y grafos de coordinación

- Representa, por medio de un grafo $G = (V, E)$, problemas en los cuales el agente i solo exhibe necesidad de coordinación con una vecindad $\Gamma(i)$.
- Permite la descomposición por arco de la función Q global.

$$Q(\mathbf{s}, \mathbf{a}) = \sum_{(i,j) \in E} Q_{ij}(\mathbf{s}_{ij}, a_i, a_j)$$

- Regla de actualización con Q-Learning¹:

$$Q_{ij}^{k+1}(\mathbf{s}_{ij}^k, a_i^k, a_j^k) = (1 - \alpha) Q_{ij}^k(\mathbf{s}_{ij}^k, a_i^k, a_j^k) + \alpha \left[\frac{r_i^{k+1}}{|\Gamma(i)|} + \frac{r_j^{k+1}}{|\Gamma(j)|} + \gamma Q_{ij}^k(\mathbf{s}_{ij}^{k+1}, a_i^*, a_j^*) \right]$$

¹Propuesto por: J. Kok en *Cooperation and Learning in Cooperative Multiagent Systems*. Ph.D thesis, University of Amsterdam, 2006.

Método 1: Q-Learning y grafos de coordinación

- Representa, por medio de un grafo $G = (V, E)$, problemas en los cuales el agente i solo exhibe necesidad de coordinación con una vecindad $\Gamma(i)$.
- Permite la descomposición por arco de la función Q global.

$$Q(\mathbf{s}, \mathbf{a}) = \sum_{(i,j) \in E} Q_{ij}(\mathbf{s}_{ij}, a_i, a_j)$$

- Regla de actualización con Q-Learning¹:

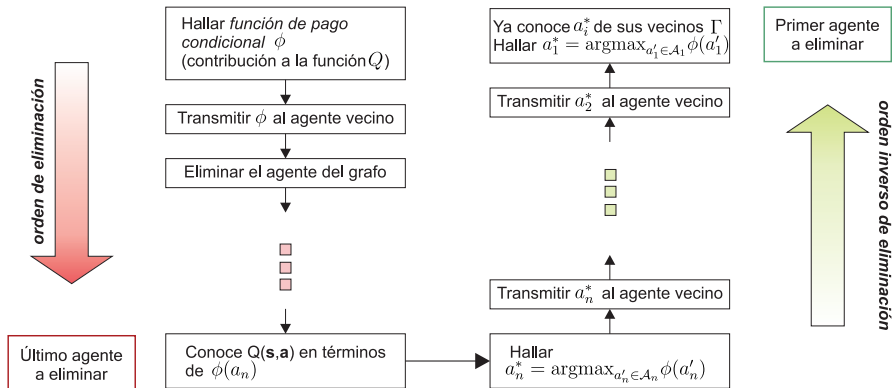
$$Q_{ij}^{k+1}(\mathbf{s}_{ij}^k, a_i^k, a_j^k) = (1 - \alpha) Q_{ij}^k(\mathbf{s}_{ij}^k, a_i^k, a_j^k) + \alpha \left[\frac{r_i^{k+1}}{|\Gamma(i)|} + \frac{r_j^{k+1}}{|\Gamma(j)|} + \gamma Q_{ij}^k(\mathbf{s}_{ij}^{k+1}, a_i^*, a_j^*) \right]$$

$$a_i^*, a_j^* \in \operatorname{argmax}_{\mathbf{a}' \in \mathcal{A}} Q(\mathbf{s}, \mathbf{a}')$$

¹Propuesto por: J. Kok en *Cooperation and Learning in Cooperative Multiagent Systems*. Ph.D thesis, University of Amsterdam, 2006.

Método 1: Q-Learning y grafos de coordinación

Algoritmo de eliminación de variable (VE): resuelve el problema de coordinación, encontrando $\mathbf{a}^* = \operatorname{argmax}_{\mathbf{a}} Q(\mathbf{s}, \mathbf{a})$



Método 1: Q-Learning y grafos de coordinación

Algoritmo de aprendizaje

Algoritmo 1 Q-Learning con eliminación de variable (Q-VE)

Inicialice $Q_{ij}(s_{ij}, a_i, a_j)$ de manera optimista

Para cada episodio **haga**

 Inicialice el estado observado s_i para todos los agentes i

Para cada periodo de decisión k **haga**

 Obtenga la acción conjunta \mathbf{a} con ϵ -greedy como método de selección

 Para cada agente aplique $a_i \in \mathbf{a}$, observe r_i^k y s_i^k

 Para todos los arcos (i, j) forme $s_{ij}^k = s_i^k \cup s_j^k$

 Obtenga $\mathbf{a}^* = \max_{\mathbf{a}} Q(\mathbf{s}^k, \mathbf{a}) \rightarrow \text{ELIMINACIÓN VARIABLE}$

Para cada arco (i, j) **haga**

$$Q_{ij}(s_{ij}^{k-1}, a_i^{k-1}, a_j^{k-1}) := Q_{ij}(s_{ij}^{k-1}, a_i^{k-1}, a_j^{k-1}) + \alpha \left[\frac{r_i^k}{|\Gamma(i)|} + \frac{r_j^k}{|\Gamma(j)|} + \gamma Q_{ij}(s_{ij}^k, a_i^*, a_j^*) - Q_{ij}(s_{ij}^{k-1}, a_i^{k-1}, a_j^{k-1}) \right]$$

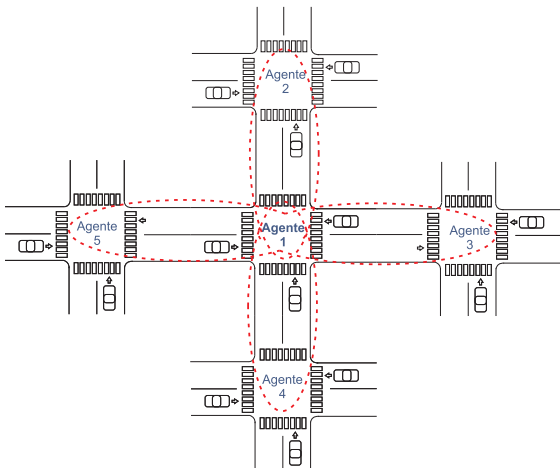
Fin Para

 Para cada agente i actualice $s_i^{k-1} \leftarrow s_i^k$

Fin Para

Fin Para

Método 2: Q-Learning y *best response*



El agente i :

- Exhibe necesidad de coordinación con una vecindad NB_i
- Participa en un juego de dos jugadores con cada vecino $NB_i[j]$

Método 2: Q-Learning y *best response*

El agente i en cada periodo k :

- 1 Estima la política de sus vecinos:

$$\theta_{i, \text{NB}_i[j]}(s_{i, \text{NB}_i[j]}^{k-1}, a_{\text{NB}_i[j]}^{k-1}) = \frac{v(s_{i, \text{NB}_i[j]}^{k-1}, a_{\text{NB}_i[j]}^{k-1})}{\sum_{a_{\text{NB}_i[j]} \in \mathcal{A}_{\text{NB}_i[j]}} v(s_{i, \text{NB}_i[j]}^{k-1}, a_{\text{NB}_i[j]})}$$

Método 2: Q-Learning y *best response*

② Actualiza los factores Q con cada vecino:

$$Q_{i, \text{NB}_i[j]}^k(s_{i, \text{NB}_i[j]}^{k-1}, a_{i, \text{NB}_i[j]}^{k-1}) = (1 - \alpha) Q_{i, \text{NB}_i[j]}^{k-1}(s_{i, \text{NB}_i[j]}^{k-1}, a_{i, \text{NB}_i[j]}^{k-1}) + \alpha \left[r_i^k + \gamma \max_{a' \in \mathcal{A}} Q(s^k, a') \right]$$

Método 2: Q-Learning y *best response*

2 Actualiza los factores Q con cada vecino:

$$Q_{i, \text{NB}_i[j]}^k(s_{i, \text{NB}_i[j]}^{k-1}, a_{i, \text{NB}_i[j]}^{k-1}) = (1 - \alpha) Q_{i, \text{NB}_i[j]}^{k-1}(s_{i, \text{NB}_i[j]}^{k-1}, a_{i, \text{NB}_i[j]}^{k-1}) + \alpha \left[r_i^k + \gamma \text{br}_i^k \right]$$

$$\text{br}_i^k = \max_{a_i \in \mathcal{A}_i} \left[\sum_{a_{\text{NB}_i[j]} \in \mathcal{A}_{\text{NB}_i[j]}} Q_{i, \text{NB}_i[j]}(s_{i, \text{NB}_i[j]}^k, a_{i, \text{NB}_i[j]}) \times \theta_{i, \text{NB}_i[j]}(s_{i, \text{NB}_i[j]}^k, a_{\text{NB}_i[j]}) \right]$$

Best response

- Función de pago $Q_i()$
- Estimativo θ_{-i} sobre las estrategias de los agentes vecinos

La estrategia $a_i \in \mathcal{A}_i$ para el jugador i es una *best response* si para todo a_i' se cumple:

$$Q_i(a_i, \theta_{-i}) \geq Q_i(a_i', \theta_{-i})$$

Método 2: Q-Learning y *best response*

- 3 Selecciona acción que corresponde a *best response* respecto a su vecindad:

$$a_i^k = \operatorname{argmax}_{a_i \in \mathcal{A}_i} \left[\sum_{j \in \{1, 2, \dots, |\text{NB}_i|\}} \sum_{a_{\text{NB}_i[j]} \in \mathcal{A}_{\text{NB}_i[j]}} Q_{i, \text{NB}_i[j]}(s_{i, \text{NB}_i[j]}^k, [a_i \cup a_{\text{NB}_i[j]}]) \times \theta_{i, \text{NB}_i[j]}(s_{i, \text{NB}_i[j]}^k, a_{\text{NB}_i[j]}) \right]$$

²Basado en: El-Tantawy *et al.* en *Multiagent Reinforcement Learning for MARLIN-ATSC*. IEEE Transactions on Intelligent Transportation Systems, 2013.

Método 2: Q-Learning y *best response*

Algoritmo 2 Q-Learning con best response (Q-BR)

Inicialice los factores $Q_{i,NB_i[j]}$ y modelo de estimación de política $\theta_{i,NB_i[j]}$

Para cada episodio **haga**

 Inicialice el estado observado s_i para todos los agentes i

Para cada periodo de decisión k **haga**

 Obtenga y aplique la acción conjunta \mathbf{a} , con ϵ -greedy como método de selección

Para cada agente $i \in \{1, 2, \dots, |N|\}$ **haga**

Para cada agente vecino $j \in \{1, 2, \dots, |NB_i|\}$ **haga**

 Observe $s_i^k, r_i^k, s_{NB_i[j]}^k$ y $a_{NB_i[j]}^{k-1}$

 Forme el estado conjunto $s_{i,NB_i[j]}^k$ y la acción conjunta $a_{i,NB_i[j]}^{k-1}$

 Actualice el modelo $\theta_{i,NB_i[j]}(s_{i,NB_i[j]}^{k-1}, a_{NB_i[j]}^{k-1})$ de la política

 Encuentre la *best response* br_i^k

 Actualice $Q_{i,NB_i[j]}^k(s_{i,NB_i[j]}^{k-1}, a_{i,NB_i[j]}^{k-1})$

 Actualice $s_{i,NB_i[j]}^{k-1} \leftarrow s_{i,NB_i[j]}^k$

Fin Para

 Encuentre al acción a_i^* que corresponde a *best response* con todos los vecinos

Fin Para

Fin Para

Fin Para

Espacio de estados y acciones

Vector de estado

Estado de un agente de i accesos ($i \in \{\text{norte, este, sur, oeste}\}$), conformado por:

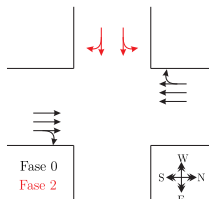
- Hora de día (h)
- Fase actual (k)
- Máxima longitud de cola (veh) por acceso (q_i)
- Tiempo de espera (min) de los vehículos por acceso (w_i)

Discretizado por *Vector Quantization*

Acciones

Fase a aplicar con duración mínima:

- Q-VE: 20 segundos
- Q-BR: 14 segundos



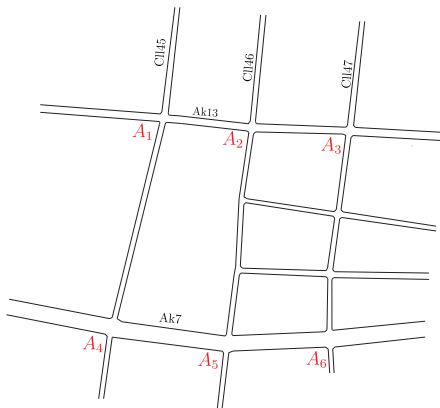
Función de recompensa para cada agente

$$r_i = - \sum_{k=1}^{numAcc_i} \beta_q(q_k)^{\theta_q} + \beta_w(w_k)^{\theta_w}$$

Donde:

- $numAcc_i$: número de accesos que tiene el agente i
- q_k : máxima longitud de cola del acceso k
- w_k : tiempo de espera de los vehículos en el acceso k
- β_q y β_w : coeficientes para el equilibrio de magnitudes de las variables q y w .
- θ_q y θ_w : términos potencia para equilibrar las longitudes de cola y tiempos de espera en los accesos.

Marco de prueba



Datos de flujos vehiculares y programas de fases de la Secretaría de Movilidad de Bogotá

Parámetros de simulación:

- Interacción con las agentes a través del simulador SUMO
- *Observación del sistema*: 5 episodios
- *Aprendizaje de política*: 150 episodios
- *Prueba de política*: 5 episodios

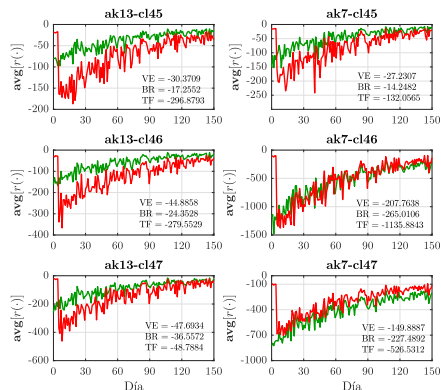
Detalles de ejecución:

- Entrenamiento en AWS
- Duración: 36 horas aprox.

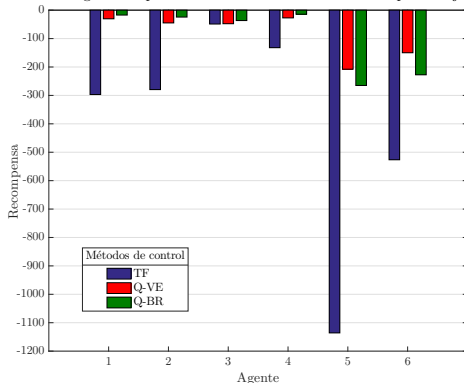
Resultados

Curva de aprendizaje

Curva de aprendizaje



Avg. Recompensa durante los últimos 10 días de aprendizaje



Resultados

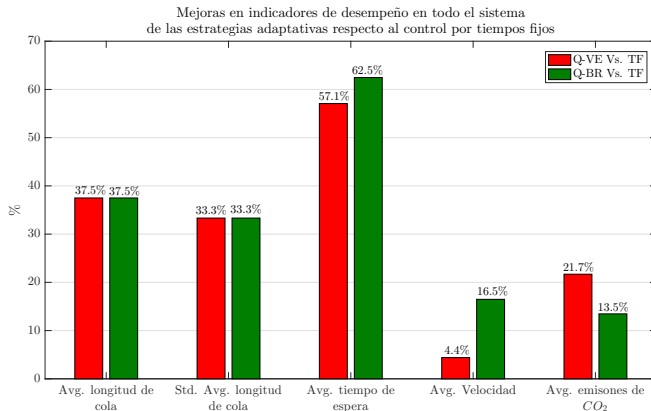
Indicadores de desempeño

- Máxima longitud de cola promedio por intersección (veh)
- Desviación estándar del promedio de las longitudes de cola en los accesos de las intersecciones (veh)
- Tiempo de espera promedio por vehículo (s/veh)
- Velocidad promedio (m/s)
- Emisiones promedio de CO_2 por intersección (mg)

Resultados

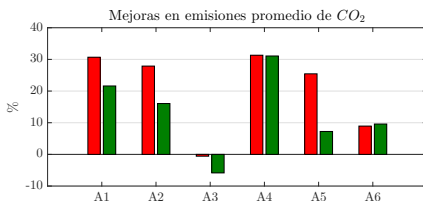
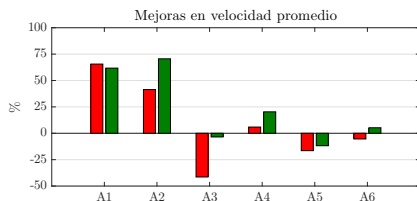
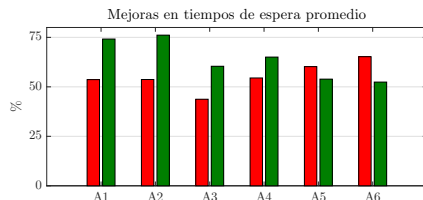
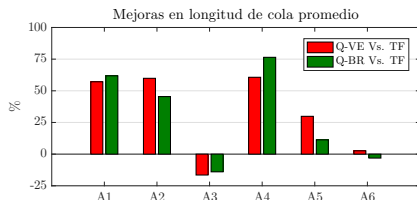
Indicadores de desempeño del sistema

	<i>Avg. longitud de cola (veh)</i>	<i>Std. Avg. longitud de cola (veh)</i>	<i>Avg. tiempo de espera (s/veh)</i>	<i>Avg. velocidad promedio (Km/h)</i>	<i>Avg. emisiones de CO₂ (mg)</i>
TF	8	3	38,9	15,2	5140,9
Q-VE	5	2	16,7	15,9	4025,8
Q-BR	5	2	14,6	18,2	4448,3



Resultados

Indicadores de desempeño por agente



Resultados

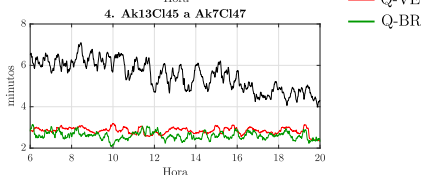
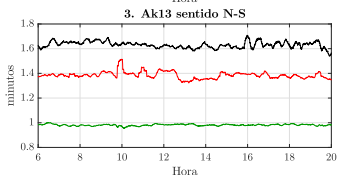
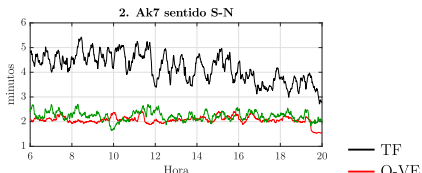
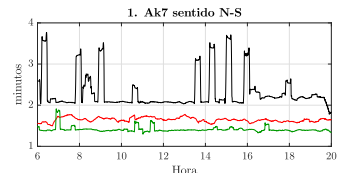
Tiempo de viaje para rutas principales

Tiempo de viaje promedio (min)

	Ruta 1	Ruta 2	Ruta 3	Ruta 4
TF	2.21	3.98	1.62	5.37
Q-VE	1.67	2.09	1.39	2.85
Q-BR	1.40	2.22	0.98	2.61

Mejoras en el tiempo de viaje promedio

	Ruta 1	Ruta 2	Ruta 3	Ruta 4
Q-VE Vs. TF	24,43 %	47,49 %	14,20 %	46,93 %
Q-BR Vs. TF	36,65 %	44,22 %	39,51 %	51,40 %



— TF
— Q-VE
— Q-BR

Conclusiones

- Las políticas aprendidas buscan priorizar el paso continuo de grupos de movimientos, en lugar del paso de flujos paralelos.
- Para el sistema de tránsito, la aplicación del principio de localidad en la selección de las acciones de los agentes, conlleva a una política con mejor desempeño en los objetivos de minimización.
- El control adaptativo por medio de Q-BR presenta menor tiempo de viaje promedio y menores variaciones.

Conclusiones

- Las políticas aprendidas buscan priorizar el paso continuo de grupos de movimientos, en lugar del paso de flujos paralelos.
- Para el sistema de tránsito, la aplicación del principio de localidad en la selección de las acciones de los agentes, conlleva a una política con mejor desempeño en los objetivos de minimización.
- El control adaptativo por medio de Q-BR presenta menor tiempo de viaje promedio y menores variaciones.

	<i>Grafos de coordinación</i>	<i>Best response</i>
Obtención de a^*	Exacta por medio de VE	A nivel de vecindades
Escalabilidad	No facilmente	Completamente
Comunicaciones entre agentes	Sujetas a cambios	Definidas <i>a priori</i>
Observabilidad del sistema	Menor	Mayor

- 1 Tener en cuenta otros objetivos en la función de recompensa (ej. velocidad promedio)
- 2 Tratar el problema con un espacio de estados continuo
- 3 Si se afronta el problema con un espacio de acciones continuo (ej. duraciones de las fases en una secuencia de fases fija), se podría incorporar el algoritmo de Consensus para la maximización de la función de recompensa

Videos del control adaptativo por Q-BR

Simulaciones

Hora pico 6am-7am

Hora valle 9am-10am

Hora pico 12m-1pm

¿Preguntas?