Towards Autonomous Reinforcement Learning: Automatic Setting of Hyper-parameters using Bayesian Optimization

Juan C. Barsce [1] Jorge A. Palombarini [1 2]
Ernesto C. Martínez [3]

[1]UTN - Fac. Reg. Villa María

[2]CIT Villa María CONICET-UNVM

[3]INGAR - CONICET UTN Fac. Reg. Santa Fe

XLIII CLEI - 46 JAIIO - Septiembre 2017

Agenda

1. Introducción
2. Aprendizaje por Refuerzos
3. Enfoque Propuesto
4. Caso de Estudio
5. Conclusiones y Trabajos Futuros

Introducción

Resumen del estado del arte de Machine Learning (ML)

- Incremento del poder computacional — Vasta cantidad de aplicaciones y sistemas de ML.
- Auge del Deep Learning.
- Idea detrás de modelos ML — **abstraer al usuario** de la necesidad de programar explícitamente la solución.

Introducción

En la práctica, el usuario no se abstrae totalmente del aprendizaje, pues tiene que:

- Configurar el agente que va a resolver su problema.
- Asegurar que aprende correctamente bajo esa configuración.
- Cambiarla por una mejor si la misma no es satisfactoria.

Introducción

Desafío fundamental en ML:

- Encontrar una configuración que haga que el agente aprenda de la forma más efectiva posible.
- La configuración se compone principalmente por los **híper-parámetros** del modelo.

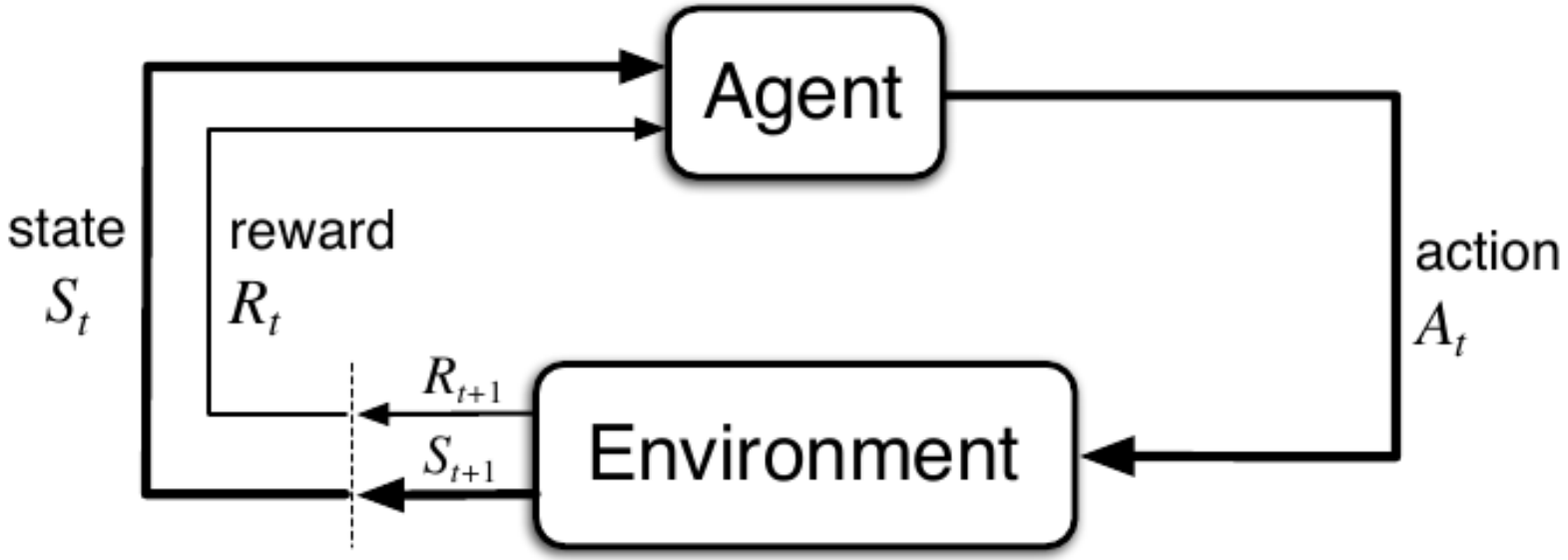
Introducción

Híper-parámetros comúnmente configurados a mano. Problemas:

- El ajuste manual lleva tiempo.
- Dificultad para comparar rendimiento de distintos modelos.
- Desconocimiento del impacto de los híper-parámetros sobre el modelo — uso de configuración por defecto.

Aprendizaje por Refuerzos

Aprendizaje por Refuerzos (Sutton y Barto, 1998)



Aprendizaje por Refuerzos

- Formalizado como Proceso de Decisión de Markov

$$MDP\_Finito = (S, A, P(.,.,.), R(.), \gamma)$$

- Resolver un *MDP* para maximizar la recompensa es el objetivo detrás de los distintos algoritmos de RL.
- Un algoritmo de RL busca, $\forall s$, política[1]

$$\underbrace{\pi^*(s) \rightarrow a}_{\text{Política óptima}} \mid v_{\pi^*} = \max_{\pi} v_\pi(s) \text{ donde } \underbrace{v_\pi(s) = \mathbb{E}_\pi(R_t \mid S_t = s)}_{\text{Recompensa esperada}}$$

[1] Política — función del agente con la que decide qué acción tomar en cada estado

Aprendizaje por Refuerzos

$$MDP\_Finito = (S, A, P(.,.,.), R(.), \gamma)$$

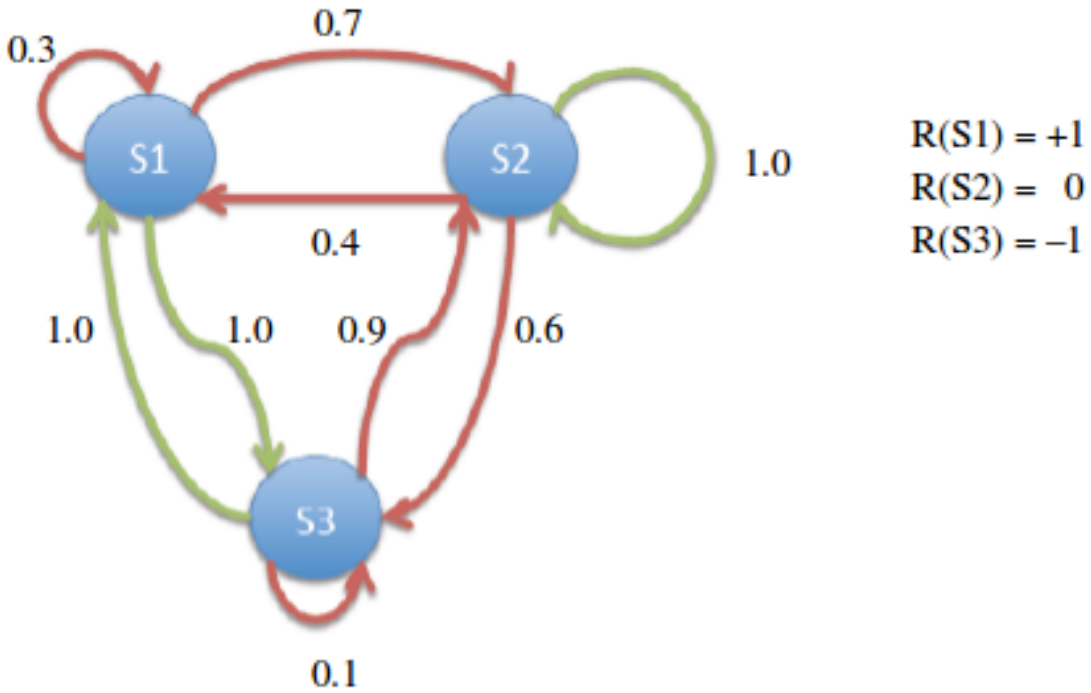
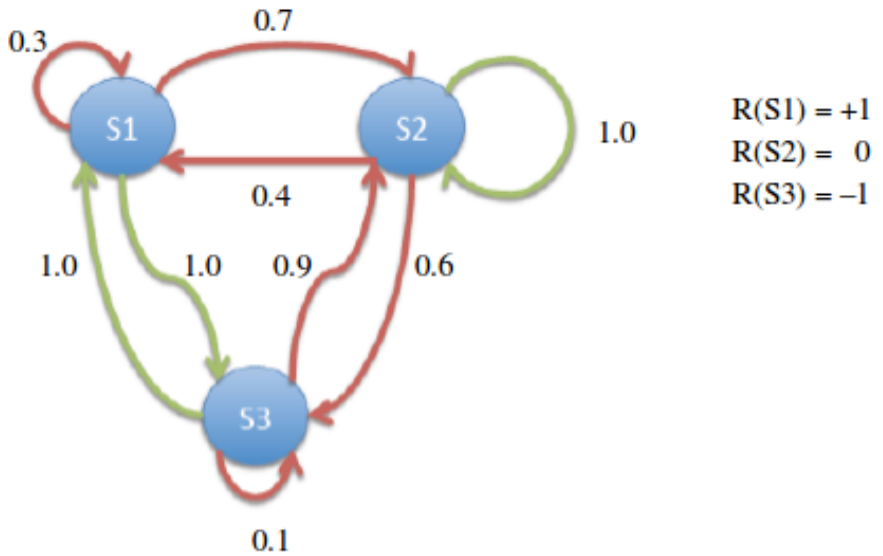


Figura 1: Ejemplo de proceso de Decisión de Markov

Aprendizaje por Refuerzos



Ejemplo de episodio de MDP (desde tiempo t):

$$\underbrace{S2}_{s_t}, \underbrace{R}_{a_t}, \underbrace{S1}_{s_{t+1}}, \underbrace{+1}_{r_{t+1}}, \underbrace{V}_{a_{t+1}}, \underbrace{S3}_{s_{t+2}}, \ldots, \underbrace{S1}_{s_{t+k}}, \underbrace{+1}_{r_{t+k}}$$

Recompensa total: $R_t = +1 - 1 + 1 + \cdots + 1$

Aprendizaje por Refuerzos

- Recompensa total: $R_t = +1 - 1 + 1 + \cdots + 1$:
- Descontando R_t por γ para estados futuros a partir de s_t teniendo en cuenta la estocacidad, quedaría

$$\begin{aligned} R_t &= +1 + \gamma(-1) + \gamma^2(+1) + \cdots + \gamma^k(+1) \\ &= +1 + \gamma((-1) + \gamma(+1) + \cdots + \gamma^{k-1}(+1)) \\ &= +1 + \gamma R_{t+1} \end{aligned}$$

Aprendizaje por Refuerzos

- γ es uno de los híper-parámetros de θ que en conjunto regulan la forma en la que aprende el agente.
- **Objetivo de este trabajo**: generar automáticamente un vector θ^* que maximice la R_t esperada.
- Expresado formalmente:
  $\mathbb{E}_\pi(R_t \mid S_t = s, \theta^*) \geq \mathbb{E}_\pi(R_t \mid S_t = s, \theta), \forall s, \theta.$

Enfoque Propuesto

- Planteado como problema de optimización:
  $\max_{\theta} \mathbb{E}_{\pi}(R_t \mid S_t = s, \theta), \forall s$ para un agente de RL, A.

Surgen dos problemas:

1. $\mathbb{E}_{\pi}(R_t \mid S_t = s, \theta_1)$ incomparable con $\mathbb{E}_{\pi}(R_t \mid S_t = s, \theta_2)$ si $\theta_1 \neq \theta_2$ (por la amplitud de la recompensa).
2. Ineficiente tratarlo como problema de optimización de híper-parámetros tradicional.
   - Recompensas demoradas
   - Alto costo computacional en cada corrida del agente

Enfoque Propuesto

- Solución propuesta: establecer una función objetivo $f_A(\theta)$ que represente el rendimiento de A en episodios bajo θ.
- Dos $f_A(\theta)$ objetivo planteadas:

$$f_A(\theta) = \sum_{i=1}^{n_{episodios}} t_i / n_{episodios} \quad (1)$$

$$f_A(\theta) = \sum_{i=1}^{n_{episodios}} s_i / n_{episodios} \quad (2)$$

donde $s_i = 1$ si el episodio fue exitoso; $s_i = 0$ en caso contrario

Enfoque Propuesto

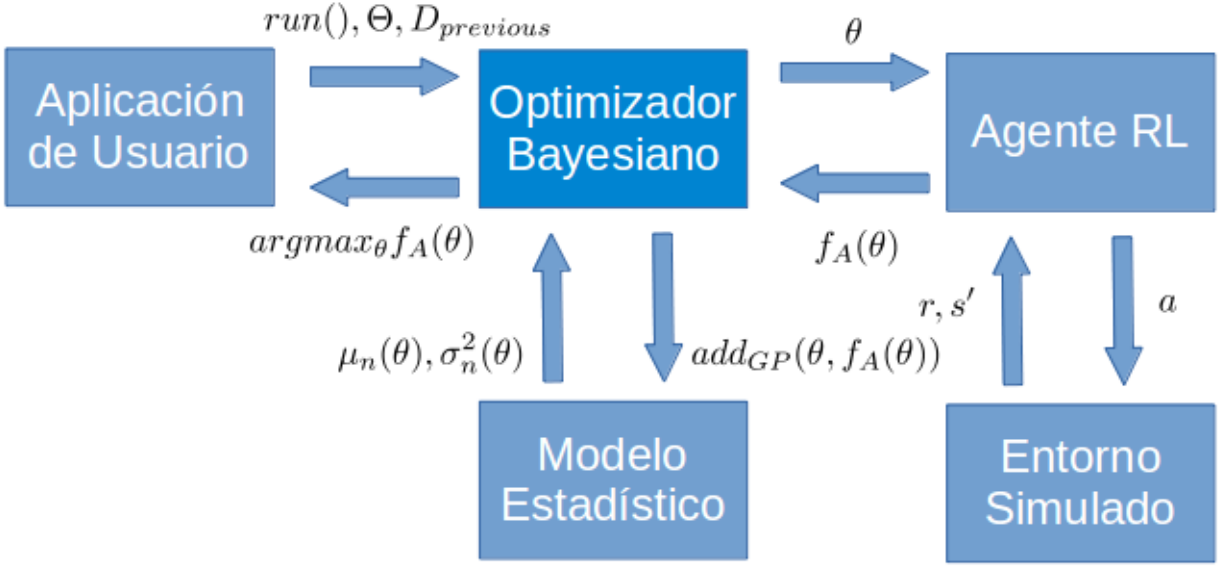


Figura 2: Marco de referencia RLOpt

Algoritmo 1: RLOpt framework.

Entrada: Θ

para $n = 1$ *hasta* $episodios_{OB}$ **hacer**

  $\theta \leftarrow argmax_{\theta}(\alpha, \alpha_{opt})$

  para $corrida = 1$ *hasta* $corrida_{\theta}$ **hacer**

    $iniciar(A, episodios_A)$

    $f_{A-promedio}(\theta) \leftarrow 0$

    para $ep = 1$ *hasta* $episodes_A$ **hacer**

      $reiniciar(A)$

      $correr(A \mid \theta)$

      $guardarEjecucion(A)$

      $f_{A-promedio}(\theta) \leftarrow f_{A-promedio}(\theta) + (f_{A-episodio}(\theta) + \sigma^2_{n-episodio})$

    fin

    $f_{A-promedio}(\theta) \leftarrow f_{A-promedio}(\theta)/episodios_A$

    $agregar_{GP}(\theta, f_{A-promedio}(\theta))$

  fin

fin

Salida : $\arg\max_{\theta} f_{A-prom}(\theta)$

Procesos Gaussianos

Regresión por Procesos Gaussianos (Rasmussen y Williams, 2006):

- Modelo estadístico no paramétrico caracterizado por $\mu_0(\theta)$ y $k(\theta_i, \theta_j)$
- Solución de forma cerrada con la cual se pueden hacer regresiones:

$$\mu_{n+1}(\theta) = \mu_0(\theta) + k(\theta)^T K^{-1}(Y - \mu_0)$$

$$\sigma^2_{n+1}(\theta) = k(\theta, \theta) - k(\theta)^T K^{-1} k(\theta)$$

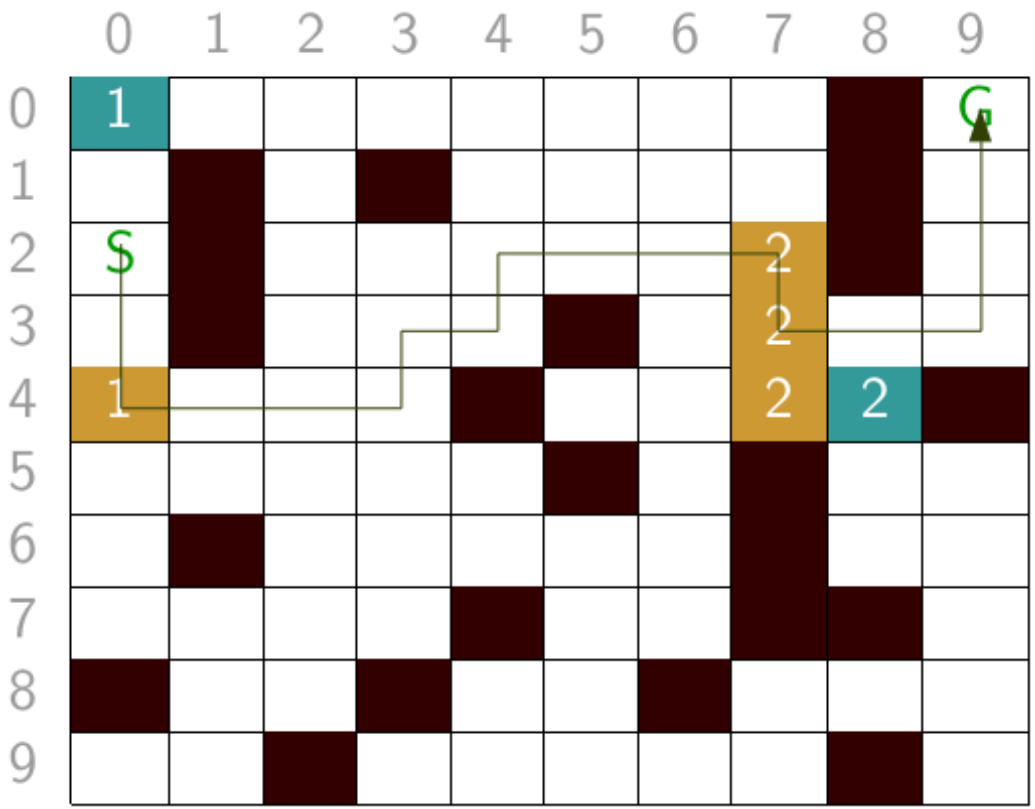
donde $Y = (f_A(\theta_1), f_A(\theta_2), ..., f_A(\theta_n))$

Caso de Estudio

Caso de estudio: "Grilla con doble bloqueo"

- Objetivo: demostrar que pueden existir configuraciones de θ que mejoren al estado del arte para dos optimizadores.
- Dos cambios en el entorno para probar cómo el agente se adapta a los mismos bajo una configuración θ.
- Configuración del estado del arte: hyperparámetros RL para arquitectura Soar (Laird, 2012) en este caso.

Caso de Estudio



1 Celda cambia en el primer bloqueo (en episodio 15)

2 Celda cambia en el segundo bloqueo (en episodio 30)

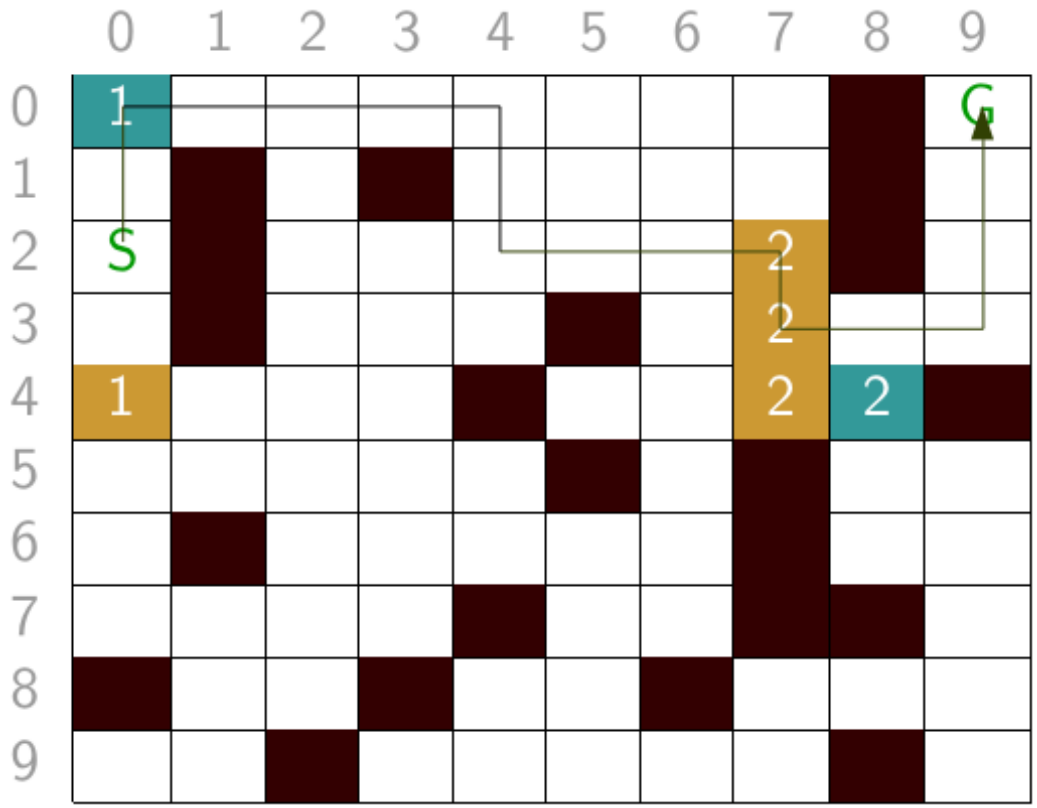
Obstáculo permanente

Obstáculo temporal

Obstáculo por surgir

Camino óptimo (17 pasos)

Caso de Estudio



1 Celda cambió en el primer bloqueo (en episodio 15)

2 Celda cambia en el segundo bloqueo (en episodio 30)

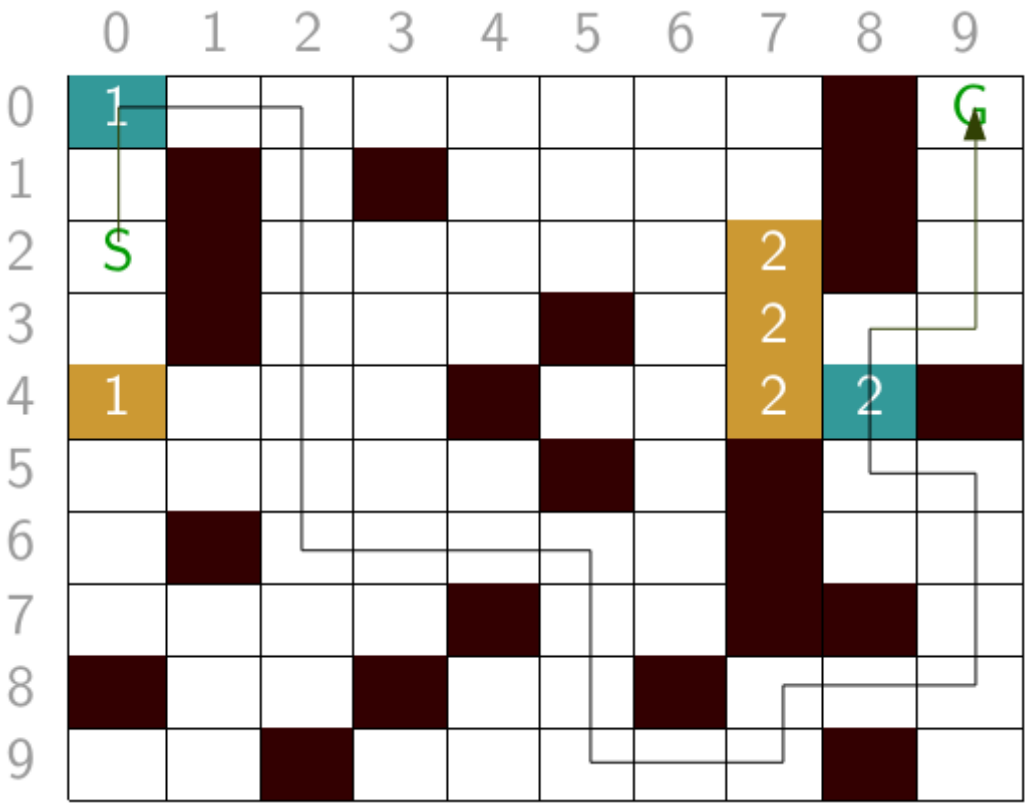
Obstáculo permanente

Obstáculo temporal

Obstáculo por surgir

Camino óptimo (17 pasos)

Caso de Estudio



1 Celda cambió en el primer bloqueo (en episodio 15)

2 Celda cambió en el segundo bloqueo (en episodio 30)

Obstáculo permanente

Obstáculo temporal

Obstáculo por surgir

Camino óptimo (31 pasos)

Caso de Estudio

- Cada optimizador se ejecutó 10 veces, cada una consistió en 30 consultas a $f_A(\theta)$ (meta-episodios).
- Medida de éxito utilizada:

$$s_i = \begin{cases} 1 & \text{si pasos } ep_i < \text{tiempo de corte} \\ 0 & \text{en caso contrario} \end{cases}$$

- Por cada meta-episodio i, un agente con una configuración θ_i corre 50 episodios. En cada i, el entorno vuelve a su estado original.
- Tiempo de corte de cada episodio: 400 pasos de tiempo.

Caso de Estudio

Detalles:

- Algoritmo RL: $\overbrace{SARSA(\lambda)}^{\theta=(\alpha,\epsilon,\gamma,\lambda)}$, política ϵ-greedy.
- $k(\theta_i, \theta_j)$ — Squared Exponential
- Función de adquisición — Expected Improvement
- Entorno: PC con 8Gb RAM y 4x3.20 Ghz.
- Duración de cada instancia de optimizador: promedio 135 minutos

Resultados

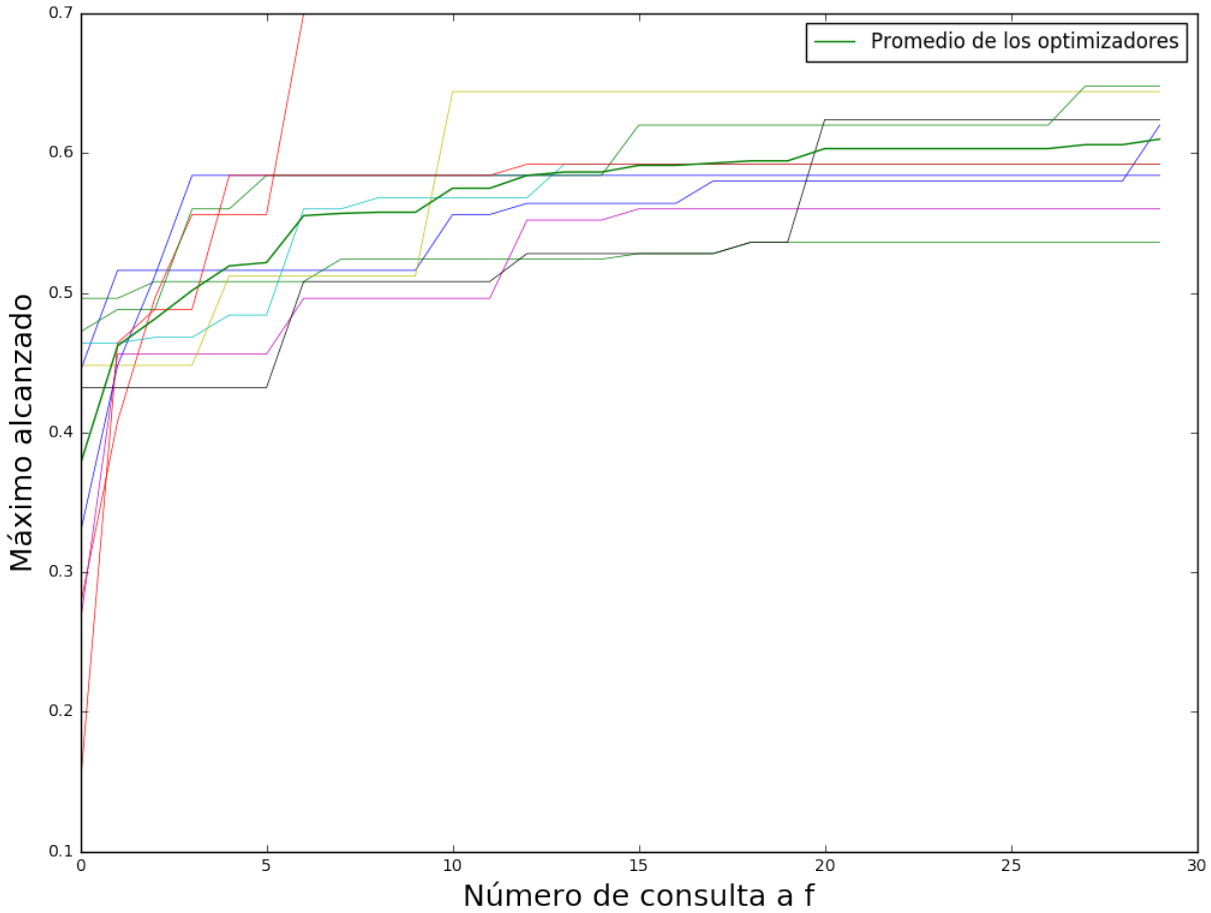
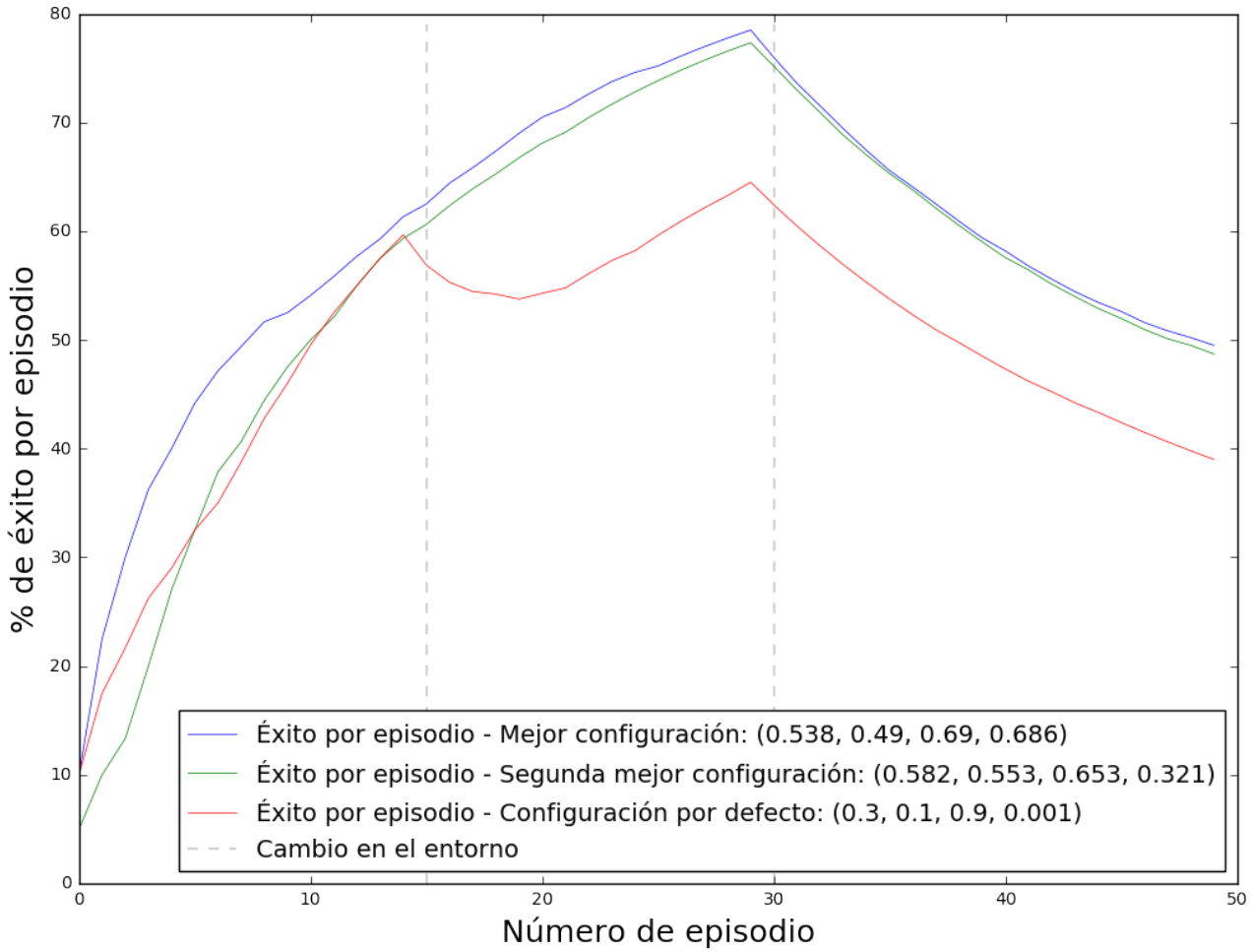


Figura 3: Convergencia hacia el máximo del optimizador con medida de

Resultados


% de éxito por episodio
Número de episodio
0
10
20
30
40
50
60
70
80
Éxito por episodio - Mejor configuración: (0.538, 0.49, 0.69, 0.686)
Éxito por episodio - Segunda mejor configuración: (0.582, 0.553, 0.653, 0.321)
Éxito por episodio - Configuración por defecto: (0.3, 0.1, 0.9, 0.001)
Cambio en el entorno

Observaciones finales

- Se presentó un marco de referencia para aproximar una política semi-óptima para un agente de RL.
- El mismo le permite al agente aprender automáticamente una buena política.
- Se mostró que, para un entorno determinado, existen híper-parámetros superiores al estado del arte.

Trabajos Futuros

- Extensión de las $f_A(\theta)$, incluyendo medidas que pertenezcan a una categoría $C_i \in \{C_1, C_2, ..., C_n\}$.
- Extensión del actual caso de estudio a un caso de estudio industrial para una tarea de replanificación de calendarios.
- Extensión de RLOpt a un nivel de abstracción mayor en cuanto a los híper-parámetros.
- Incorporación de otros modelos como random forests para comparar su desempeño al de los GP.

Muchas gracias!

Preguntas?