

Optimal single-path information propagation in gradient-based algorithms propone una mejora en los algoritmos de gradiente utilizando en sistema distribuidos como redes inalámbricas e internet de las cosas (IoT). Los gradientes se emplean para calcular distancia entre dispositivos y propagar información dentro de una red mediante comunicación local entre nodos. Son esenciales para tareas como la difusión de toda la redacción de información distribuida la formación de canales de comunicación y la coordinación espacial.

Sin embargo, los métodos clásicos presentan limitaciones al trabajar en entornos dinámicos, como lentitud en la adaptación a cambios, errores de estimación y poca estabilidad ante el ruido. Para resolver estos problemas los autores proponen el BIS (Bounded Information Speed) Gradient, un algoritmo que incorpora la información temporal para controlar la velocidad con la que se propaga la información. Este enfoque permite una reacción más rápida y estable frente a los cambios en la red, manteniendo al mismo tiempo la precisión en las estimaciones.

Las pruebas realizadas muestran que el BIS Gradient supera a los algoritmos tradicionales (Classic CEF y FLEX) en precisión, velocidad de recuperación y estabilidad. Por ello se presenta como un bloque fundamental para la programación agregada y la computación espacial aplicable a sistema distribuidos que requieren resiliencia, eficiencia y adaptación continua a entornos cambiantes.

SISTEMA DISTRIBUIDOS (IoT, Redes, CPS)  
Dispositivos  $\leftrightarrow$  comunicación local  $\leftrightarrow$  Coordinación autónoma.

GRADIENTES  
calculan distancias y guían  
la programación de información.

Difusión (Broadcast)  
Propaga mensajes

Recolección (Data col.)  
Agrega datos al origen.

BIS GRADIENT (Bounded info speed)  
\* usa velocidad  $v$  para controlar difusión  
\* Aumenta precisión y reactividad.  
\* FLEX damping mejora estabilidad.

\* Comunicación por canales.  
\* Recolección de datos distribuidos.  
\* Coordinación y resiliencia en IoT