

# Progetto Sistemi Distribuiti e Cloud Computing

## Applicazione distribuita di stima della distanza e rilevamento guasti

Alessandro Finocchi  
alessandro.22082001@gmail.com

Università degli studi di Roma Tor vergata  
Ingegneria informatica

Settembre 2024



**TOR VERGATA**  
UNIVERSITY OF ROME

School of Engineering

1 Introduzione

2 Soluzione

3 Risultati

4 Conclusione

# Roadmap

1 Introduzione

2 Soluzione

3 Risultati

4 Conclusione

# Introduzione



# Introduzione

## Il problema

- In reti di grandi dimensioni alcune scelte possono essere facilitate se si conosce la propria posizione e quella degli altri partecipanti della rete
- Es: load-balancing, replicazione dei dati, allocazione di risorse...
- Problema: il concetto di posizione nelle reti differisce da quello cui siamo abituati



# Introduzione

## L'obiettivo

- Creare un sistema di gossiping i cui partecipanti hanno assegnate delle coordinate sintetiche
- La distanza tra le coordinate di due host deve predire con precisione l'rtt tra di essi
- Tutti i peer della rete hanno a disposizione la lista di coordinate aggiornata di tutti gli altri
- In caso di crash di alcuni nodi, tramite un meccanismo di tolleranza ai guasti la rete è in grado di riassestarsi



# Roadmap

1 Introduzione

2 Soluzione

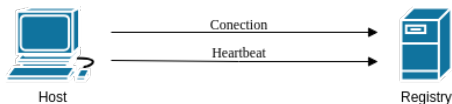
3 Risultati

4 Conclusione

# Soluzione

## Il registry

- Un nuovo partecipante può richiedere di entrare nella rete contattando il registry ad un indirizzo noto
- Il registry gli fornirà un insieme di nodi ancora up-and-running tramite i quali iniziare a scambiare informazioni sulle coordinate
- Periodicamente l'host invierà un messaggio di heartbeat per comunicare al registry che è ancora parte della rete

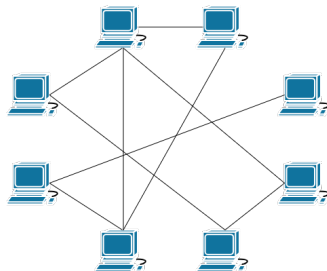




# Soluzione

## La rete

- A ciascun peer basta conoscere solo un sottoinsieme degli altri peer
- Una volta dentro, il nuovo host comincerà a partecipare a 3 diversi protocolli



# Soluzione

## Protocollo di Membership

- Permette di mantenere una vista parziale degli altri host
- Ogni nodo ha bisogno solo di pochi campioni della lista completa, anche presi a random
- Grazie a questo protocollo si riesce a mantenere sempre una vista di peer che sono ancora connessi



# Soluzione

## Protocollo di Vivaldi: l'idea

- Vivaldi si ispira ad un sistema reale di masse e molle
- Le masse sono l'analogia degli host
- Le molle che collegano due masse sono l'analogia dell'rtt tra i due rispettivi host
- La cui lunghezza naturale di una molla è la distanza (rtt) tra le masse (host)



# Soluzione

## Protocollo di Vivaldi: visualizzazione

- Si cerca di risolvere un problema di rilassamento delle molle
- All'inizio tutte le masse si trovano nell'origine e poi si rilassa il sistema



# Soluzione

## Protocollo di Vivaldi: visualizzazione



# Soluzione

## Protocollo di Vivaldi: miglioramenti

- Tramite l'utilizzo di un filtro a percentile mobile si filtrano le stime degli rtt calcolati per aumentare la stabilità
- Differenziando tra coordinate d'applicazione e di sistema, si minimizza la necessità di aggiornamenti, quindi il traffico diminuisce
- Il problema delle violazioni alla disuguaglianza triangolare è risolto adottando uno spazio non euclideo dotato di altezza



# Soluzione

## Protocollo di gossiping delle coordinate

- Le coordinate di sistema vengono costantemente aggiornate
- Quando si rileva un cambiamento significativo in quelle di sistema, vengono aggiornate le coordinate d'applicazione
- Le coordinate di applicazione sono diffuse quando cambiano



# Roadmap

1 Introduzione

2 Soluzione

**3 Risultati**

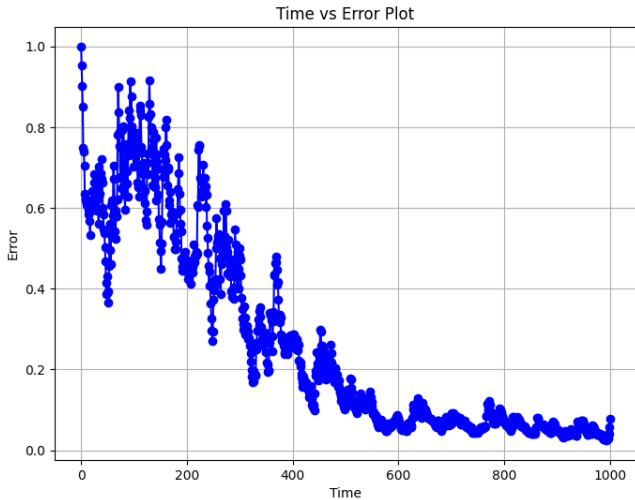
4 Conclusione



- Il protocollo di membership adotta una selezione dei peer di tipo "swapper" per interagire con più host del sistema possibile
- Si è utilizzato come spazio quello non euclideo dotato di altezza
- Le coordinate hanno 3 dimensioni più l'altezza
- Per velocizzare la convergenza come iperparametri sono stati scelti:

$$cc = 0.25 \quad ce = 0.25$$

- Il filtro è l'mp, con una finestra lunga 16 ed il percentile 25-esimo
- Si sono usate le euristiche "Energy" e "Relative" come test per verificare un cambiamento significativo nelle coordinate di sistema



# Roadmap

1 Introduzione

2 Soluzione

3 Risultati

4 Conclusione

- Evitare partizioni di rete nel protocollo di membership
- Aggiungere all'algoritmo di vivaldi la gravità per ridurre il drift
- Implementare un protocollo di snapshotting delle coordinate



# Conclusione

## Referenze

- Frank Dabek, Russ Cox, Frans Kaashoek, and Robert Morris.  
*Vivaldi: a decentralized network coordinate system*
- J. Ledlie, P. Pietzuch, and M. Seltzer. *Stable and accurate network coordinates*
- Alberto Montresor. *Gossip and Epidemic Protocols*
- Jelasity, Voulgaris, Guerraoui, Kermarrec, and van Steen.  
*Gossip-based peer sampling*
- Sanghwan Lee, Zhi-Li Zhang, Sambit Sahu, Debanjan Saha.  
*On Suitability of Euclidean Embedding for Host-based Network Coordinate Systems*
- Benedikt Elser, Andreas Forschler, and Thomas Fuhrmann  
*Tuning Vivaldi: Achieving Increased Accuracy and Stability*
- Jonathan Ledlie, Paul Gardner, and Margo Seltzer. *Network Coordinates in the Wild*

*GRAZIE PER  
L'ATTENZIONE!*

