Vivaldi: A Decentralized Network Coordinate System

Xintong Wang, Han Zhang

EECS 589 Paper Review

1 Full Reference

Authors: Frank Dabek, Russ Cox, Frans Kaashoek and Robert Morris.

Title: Vivaldi: A Decentralized Network Coordinate System.

Venue: Proc. of ACM SIGCOMM '04, 34(4):15-26.

Date of publication: October 2004.

2 Paper Summary

In this paper, the authors propose a decentralized coordinate calculation and assignment algorithm, Vivaldi, along with a hybrid coordinate system combining Euclidean coordinates and directionless height element. The purpose of the algorithm is to accurately predict the round-trip times (RTT) between any two hosts directly from their coordinates without incurring too much overhead to contact each other first. Evaluation of the Vivaldi algorithm, conducted using a packet-level network simulation with both real world RTT data and synthetic data sets, shows that Vivaldi is accurate, robust against high error nodes, and rapidly responsive to network changes. By comparing the performances of Vivaldi under several possible coordinate spaces, the authors demonstrate that a two dimensional Euclidean model with height element can capture the Internet's underlying structure and predict the RTT very well.

3 Novelty and Contributions

3.1 Analogy to Mass-spring System in Physics

The first thing that enlightens us is the authors' analogy to mass-spring system when proposing the Vivaldi algorithm. They interpret the problem of minimizing prediction errors as a physics problem of minimizing the energy of a mass-spring system. Although previously in this course we have seen many modeling techniques, most of them are applied to simulate packet arrival and network traffic (for example, Poisson distribution and power law distribution in the Self-Similar paper [3] and Brownian motion in the Sprout paper [10]) but not the underlying structure of the network. We find this work's *analogy* between network coordinate spaces and mass-spring system and *equivalency* between squared-error function and spring energy especially fascinating.

This paper demonstrates the power of adopting ideas in physics to model the complex underlying structure of network. The analogy to mass-spring system makes the abstract and complicated graph of network hosts straightforward and clarifies the motion and trajectory of each node in such a complex network system: a node will move towards the direction decided by the sum of forces exerted by its neighbors.

It is noteworthy to mention that the authors are not the first one who proposed the spring relaxation model in the field of computer science. As the authors have explained in the related works section, a few previous works [2, 4, 7, 8, 9] have tried similar modeling in different areas. Though previous works successfully introduces this spring relaxation model, it is Vivaldi along with its applications that has helped the idea to reach broader audiences and influence future works.

3.2 The Proposed Height-vector Coordinates

The directionless height element introduced by the authors to the Euclidean coordinate system is another novel idea. This height element captures positive the access link latency and is different from adding a dimension to the two dimensional Euclidean space. As a result, the authors propose the height vector model and redefine vector operations.

We find this height element important because forces that cancel out each other in Euclidean plane may not model the non-negative height vector. Though previous study has proposed Euclidean coordinates, all models (including the hyperbolic model) fail to suggest that a directionless height element should be included to capture the latency from the node to the core.

3.3 The Adaptive Timestep

The intuition behind the adaptive timestep is similar to that of learning rate in gradient descent, which is commonly used to find a local minimum of a function. However, instead of using the *Bold Driver* technique [1] to tune the learning rate, the authors adapt the timestep in a more meaningful approach by defining the timestep as the ratio of local error to the sum of local error and remote error. This tuning approach reveals the intuition behind it: an inaccurate node sampling an accurate node should compromise to move, whereas an accurate node should not move too much when it samples an inaccurate one.

Though the Bold Driver trick should also work on this kind of fine-tuning, their adaptive approach is more than a trial-and-error tuning but has also shed light on the possible mechanism underneath.

3.4 Combining Principal Component Analysis (PCA) with Model Selection

In Section 5 of the paper, instead of simply increasing the number of dimensions included in the model to increase the prediction accuracy of the specific data sets, the authors use PCA to capture the dimensionality of Internet coordinates and decide the number of dimensions to include in a Euclidean space. This PCA analysis can not only help the coordinate system to generalize better to other data sets but also give insights on the complex structure of the internet (geographical distance and access-link delays are the two major components).

3.5 Sequential Order of Introducing Contributions

Another thing that we like about the paper is the sequential order of introducing two major contributions that are essential to the RTT coordinate system, namely the Vivaldi algorithm and the synthetic coordinates with height element. Since people are more familiar with the

[1] After each epoch, compare the network's loss to its previous value. If the error has decreased, increase the learning rate by a small proportion (typically 1% to 5%). If the error has increased by more than a tiny threshold (for example 10^{10}), undo the last weight change and decrease the learning rate sharply, typically by 50 %.

traditional Euclidean system due to its prevalence in real world applications, the authors first describe the Vivaldi algorithm and evaluate its performance under the Euclidean coordinates. Once readers understand Vivaldi, the paper introduces the hybrid coordinate system composed of Euclidean coordinates and directionless height element. They show the strengths of the hybrid coordinate system by comparing its error rate with that of Euclidean coordinates, both calculated with the Vivaldi algorithm.

This method gives us guidelines on how to write our own research paper and convey complex ideas. If the authors had described Vivaldi directly in the context of hybrid coordinate system, which should be the best combination to demonstrate the contributions, readers may have a hard time quickly understanding a complex algorithm under an unfamiliar coordinate system. When presenting complicated technical ideas in the future, it would be helpful to isolate the cryptic part while keeping the rest of the problem simple. This paper has done an excellent job in this narrative aspect.

4 Possible Improvements and Extensions

4.1 Comparison against Other Decentralized Coordinate Systems

This paper is neither the first of its kind in proposing a coordinate systems for RTT estimation nor the first in adopting decentralized versions. As the paper has pointed out, there are a few previous competing decentralized coordinate systems, such as PIC [1] and NPS [6], that operate in either different or similar manners. In particular, the biggest different between Vivaldi and NPS would be the underlying modeling performed for minimization: one uses the mass-spring model while the other uses Simplex algorithm, which is a popular algorithm adopted by many other systems.

However, this paper only directly compares the performances of Vivaldi with GNP [5], which also utilizes the Simplex algorithm. We believe as an extension of this paper, it would be interesting to compare Vivaldi against both other distributed algorithms and other existing coordinate systems on robustness of high-error nodes, responsiveness of network changes and accuracy respectively.

4.2 Limitations in a Packet-level Network Simulation

Evaluation of the algorithm is made based on experiments conducted using a packet-level network simulation with RTT data collected from the PlanetLab network test bed with 192 hosts and King data set 1740 DNS servers. As mentioned in Section 3.2, the collected RTTs do not change over time in the simulation. More specifically, queuing delay and changes in routing are ignored by this simulation model. However, queuing delay is a key component of network delay and should be taken into consideration.

Instead of modeling the underlying structure of the network, the authors could get real world data on packet queuing delay and discover the distribution of queuing delay of the two data sets. According to the same distribution, queuing delay could be assigned to each packet transmission. We believe by incorporating and simulating the queuing delay, the model could be more realistic and robust when applying to real world RRT predictions.

4.3 Merging Two Existing Vivaldi Coordinate Systems

As explicitly stated in the paper, the Vivaldi algorithm will experience significant start-up cost for an initial network, especially when many nodes are joining an empty coordinate system altogether. Once there are enough nodes in the network, adding more nodes to the system will incur smaller joining costs. In Section 4.4, the authors demonstrate the idea by running a simulation of network with 1000 nodes, later adding another 1000 nodes one by one into the network to calculate the joining cost.

This simulation approach raises our question on how to incorporate nodes at a faster pace. In order for Vivaldi system to scale beyond lab settings and eventually to a global scope, it would be helpful to investigate the possibility of merging multiple Vivaldi coordinate systems into an integrated one. More specifically, we hope separate clusters that form their Vivaldi coordinates independently could merge together more effectively. The paper does not discuss the feasibility of the above idea, in that it only focuses on either adding one node at a time or throwing multiple nodes all at once with random initial coordinates.

A promising future work is to figure out how to merge two existing Vivaldi coordinate systems together. We think the relative typologies of the two independent coordinate systems could be helpful to reduce the merging time if exploited properly.

Bibliography

[1] Costa, M., Castro, M., Rowstron, R., Key, P.: Pic: Practical internet coordinates for distance estimation. In: Distributed Computing Systems, 2004. Proceedings. 24th International Conference on. pp. 178–187. IEEE (2004)

[2] Hoppe, H.: Surface reconstruction from unorganized points. Ph.D. thesis, Department of Computer Science and Engineering, University of Washington (1994)

[3] Leland, W.E., Taqqu, M.S., Willinger, W., Wilson, D.V.: On the self-similar nature of ethernet traffic (extended version). In: ACM/IEEE Transactions on Networking. pp. 2(1):1–15 (1994)

[4] Mogul, J.C.: Efficient use of workstations for passive monitoring of local area networks. Tech. rep., Digital Western Research Laboratory (July 1990)

[5] Ng, E.: Gnp software (2003), https://www.cs.cmu.edu/ eugeneng/research/gnp/

[6] Ng, T.E., Zhang, H.: A network positioning system for the internet. In: Proc. USENIX Conference (2004)

[7] R.Rinaldi, M.Waldvoge: Routing and data location in overlay peer-to-peer networks. Tech. rep., IBM (July 2002)

[8] Shavitt, Y., Tankel, T.: Big-bang simulation for embedding network distances in euclidean space. In: IEEE Infocom (April 2003)

[9] Waldvogel, M., Rinaldi, R.: Efficient topology-aware overlay network. In: Hotnets-I (2002)

[10] Winstein, K., Sivaraman, A., Balakrishnan, H.: Stochastic forecasts achieve high throughput and low delay over cellular networks. In: Proc. of the 10th USENIX Conf. on NSDI. pp. 459–472 (2013)