Distributed Computing Models for Distributed Sensor Networks

Carlos Soto (csoto@bnl.gov), Allison McComiskey, and Kerstin Kleese van Dam, *Brookhaven National Lab*

In the coming decades, communications networks are certain to bring faster data rates, greater reliability, and higher device densities than ever, enhancing a variety of compute tasks and enabling wholly new scientific applications. Beyond direct increases to capacity and throughput, future networks may enable new computational capabilities that boost scientific workloads. Climate modeling, for example, may leverage network-streamed data from geographically distributed sensors to feed arbitrarily large computational models that themselves run on distributed computing architectures. Such an approach may yield substantial scalability advantages, but it requires addressing numerous challenges, from networking techniques to sensor coordination to computing paradigms.

High-density, distributed measurement systems for climate applications are becoming a reality with the recent development of high-resolution, miniaturized, and low-cost sensors. These high-density observations enable data-driven approaches that can provide more accurate short-term and hyper-local forecasts and reduce uncertainties in long-term climate projections. For example, data assimilation (periodic nudging of the model to observed conditions) improves forecasts over purely physical modeling approaches, and machine learning algorithms using real-time observations are proving that accurate, minutes-ahead energy forecasts—critical for efficient solar renewable energy operation—are within reach. These approaches depend on robust operation of the underlying measurement systems, as well as computational resources that can handle the growing data volumes. At present, communication remains one of the primary challenges in continuous, reliable data delivery from new measurement systems. As these sensor networks expand, modeling capacity may emerge as an additional challenge.

Distributed models already are being explored in Artificial Intelligence[1], and their relevance will only amplify as network technologies improve to support them. Today's 5G and Wi-Fi 6 networks are already promising data throughputs (10 gbps) comparable to first-generation DDR RAM from 20 years ago (albeit with latencies ~1 million times slower). Future networks have the potential to reach and exceed current RAM and even inter-processor bus data rates. One consequence is that distributed computing architectures may be possible with effectively unlimited "RAM in the cloud," supporting arbitrarily large data and models in "memory" for AI, physics, and other compute classes. The trade-off is these architectures may be unavoidably limited by network latency. This is in contrast to today's sensor network deployments, which are largely limited by connection availability, quality, and throughput.

Emerging technological advances that may help drive distributed models include edge computing and compute-everywhere paradigms, such as Analysis on the Wire (AoW)[2]. By distributing compute resources to be physically closer to data sources and sinks and beginning processing as data are still in transit, these paradigms may alleviate network utilization, balance compute loads, and potentially reduce latencies. Approaches like AoW, in particular, may lead to fine-grained parallelism akin to existing supercomputing programming models, but they must contend with geographic distribution and heterogeneous compute and network resources. Additional challenges and research questions include: how to efficiently allocate compute resources with respect to sensor network topologies; determining the functional bounds for network capabilities; how to coordinate data movements between sensing and compute nodes (principally for real-time applications); and, perhaps most importantly, how to effectively program and perform experiments in such an environment, e.g., how to develop performant software models that not only scale to distributed sensors and compute nodes, but also are robust to the platform heterogeneities and latencies.

[1] Mayer, Ruben, and Hans-Arno Jacobsen. "Scalable Deep Learning on Distributed Infrastructures: Challenges, Techniques, and Tools." ACM Computing Surveys (CSUR), 2020.

[2] Bhattacharyya, Shilpi, Dimitrios Katramatos, and Shinjae Yoo. "Why wait? Let us start computing while the data is still on the wire." Future Generation Computer Systems 89 (2018):563-574.