Towards a Distributed Realization of Dynamic Virtual Locality in the NOW Setting

(EXTENDED ABSTRACT)

Pilar de la Torre *
University of New Hampshire

Matthew D. Plumlee †
University of New Hampshire

Contact Author: Pilar de la Torre; email: dltrr@cs.unh.edu; phone: (603) 862-2682.

1. Motivation, Goals and Challenges

Although the NOW architecture has succeeded in leveraging uniprocessor investment, the following question has remained virtually unexplored until now: Can similar levarging be made on the investment in networking infrastructure—LANs composed of switches, bridges, and management tools? A salient characteristic of this networking infrastructure is its own brand of locality, which we call *virtual-locality*.

1. *Virtual Locality:* The network can often viewed and operated as an aggregate of independent logical subcomponents. We call such an aggregate a *virtual locality configuration*, and each of its components a *virtually-localized subNOW*, or simply a *subNOW*. Moreover, the performance characteristics of certain fundamental operations within a virtual subNOW—such as routing, broadcast, and synchronization—can be significantly different from the performance characteristics of the other subNOWs, particularly from those of the network as a whole.

2. *Virtual Locality Shifting:* The virtual locality configurations available on a network can be changed centrally by a network manager, who manually (statically) operates management tools.

GOAL. *A central objective of this research is to systematically investigate the following question: To what extent can this virtual locality and shifting ability can be modeled theoretically and realized practically, so that it can be exploited across various NOWs by virtual locality-aware distributed and parallel algorithms?*

To achieve this goal, significant theoretical and practical hurdles remain to be systematically negotiated.

THEORETICAL MODELING. Two current groups of theoretical locality frameworks are of interest here. In the distributed setting, current theoretical frameworks for locality-sensitive computing are tightly coupled with the notion of 'proximity in network topology' [1]. In the distributed-memory parallel computation setting, issues such as congestion and portability are addressed by recent work which models a communication network by the notion of *abstract-locality* [2]. This notion abstracts a network into a collection of subnetworks as represented by their individual latency and bandwidth parameters.

- ▷ CHALLENGE. *Towards harnessing dynamic virtual locality, the target is a framework that extends and blends the above two frameworks in order to quantify the virtual-locality shifting capabilities of emerging technologies. The central strategy of our ongoing research is to develop an environment in which the theoretical and practical aspects of our progress can proceed hand in hand.*

MODEL REALIZATION. Existing implementations of general-purpose distributed-memory parallel programming environments for NOWs are typically built on top of point-to-point communication libraries that are unaware of the virtual-locality of the underlying LAN.

- ▷ CHALLENGE. *Towards exploiting virtual locality at the algorithmic level, the target is the development of communication primitives that are aware of virtual-locality, i.e., primitives that allow the dynamic specification of the virtual subNOW within which they operate.*

Current network management tools operate in essentially static and centralized manners. They are typically intended to be used by human managers who operate them manually, with system reaction time on the order of seconds.

- ▷ CHALLENGE. *Towards exploiting virtual locality shifting at the distributed-algorithm level, the target is the development of virtual locality-shifting primitives that can be invoked distributedly by the processors, and effected with reaction times on the order of milliseconds.*

2. Contributions and Significance

This abstract summarizes a theoretical framework and practical tools that we are constructing to address these goals and challenges.

ABSTRACT COST MODEL. We have further developed the theoretical framework of de la Torre and Kruskal [2] to characterize networks (including NOWs) in terms of a specific collection of virtual locality configurations, and the specific routing, synchronization, and locality-shifting cost parameters of each subnetwork in these configurations. Our framework supports the specification of various virtual locality configurations for a given network, giving the algorithm designer the ability to choose the configurations with as little or as much locality information as is required. This model allows the design of locality-aware distributed-parallel algorithms that have the potential to port their efficiency across networks with similar parameters, regardless of the underlying physical topology.

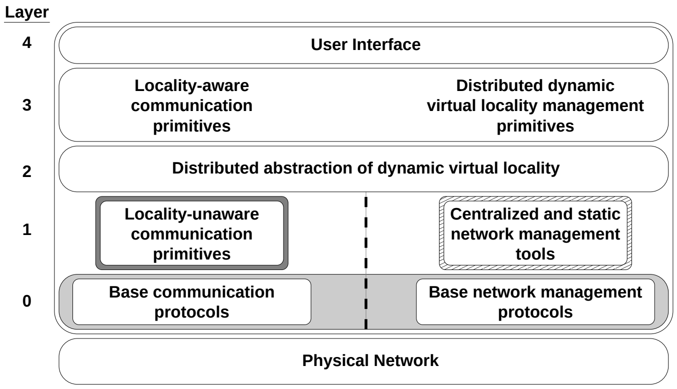
In order for these virtual locality configurations to be mapped

*The research of this author was supported in part under NSF Grant CCR-9410592.

†The research of this author was supported in part by Pilar de la Torre under NSF Grant CCR-9410592.

to physical NOWs in such a way that predicted costs map well to the measured costs, it is necessary to benchmark a NOW to determine its locality parameters. We have developed a systematic methodology for estimating the locality parameters of a NOW, which involves the benchmarking of certain "raw" characteristics, then performing linear regression and other analyses on these benchmarks.

DyViL: A DYNAMIC VIRTUAL LOCALITY SYSTEM ARCHITECTURE. We have designed a virtual locality system architecture, depicted in the figure below, that specifies a realization of our abstract model.



It provides a standard user interface (layer 4), giving the user locality-aware communication and synchronization facilities, as well as distributed, dynamic virtual locality shifting facilities. The system architecture specifies the structure of a system which transforms the basic protocols and/or libraries of layers 0 and 1 into the layer 4 dynamic virtual locality facilities. The key component in achieving this transformation is the distributed abstraction of dynamic virtual locality (layer 2)—a representation of the salient subNOW information for each processor in the form of a distributed data structure. Layer 3 uses layer 2 to integrate the layer 1 facilities, completing the transformation of these facilities into the user interface facilities of layer 4.

SYSTEM ARCHITECTURE REALIZATIONS. We have implemented prototype realizations of our system architecture for two TCP/IP-based platforms. Constant among all of the realizations is the implementation of the distributed abstraction of dynamic virtual locality (layer 2 of the system architecture). This implementation is independent of the layer 1 facilities "below" it. The realizations are based on one of two underlying communication components and on one of two network-management components. The selection of one component from layer 1 is independent from the selection of the other. The two locality-unaware communication components are the DRMA ("direct remote memory access") component, and the BSMP ("bulk-synchronous message-passing") component. The first allows asynchronous sending of communication, while the latter buffers communication until the end of a communication round. The two static network-management components are the XHUB component, and the SWITCH component, which perform management for the hub and switch platforms, respectively. The realizations based on these components from layer 1 make our empirical study of dynamic virtual locality possible in a way that is relevant to our abstract model.

PLATFORMS AND PRELIMINARY EMPIRICAL EVIDENCE. We have implemented and selected algorithms on different platforms to demonstrate and test our ideas and our model. The intent is to use the feedback from these tests to further develop the theoretical framework. We describe two platforms built using relatively inexpensive, off-the-shelf technology. We show how the locality of these platforms is mapped to the abstract model, and how the benchmarking is used to find the appropriate subNOWs and collect the specific values for the parameters in these subNOWs. We then contrast the performance of algorithms which exploit the measured locality with those that are locality-unaware. A brief summary of the empirical results follows:

Empirical Result 1 (Hub Platform):

PLATFORM DESCRIPTION: A Xyplex Series 9000 Hub, consisting of 72 Ethernet ports (10 Mbits), 18 repeaters, 3 backplanes, and a management processor to control the connections among them.

LOCALITY MAPPING: SubNOWs obtained by repeated halving, each with bandwidth and latency parameters directly proportional to the number of processors in a subNOW. The shifting parameters are all relatively constant, on the order of 10 seconds.

ALGORITHMS: Two different all-to-all broadcast algorithms: one that is locality-aware and uses the shifting primitives according to the repeated halving model, and one that is not locality-aware and uses only the network as a whole.

SIGNIFICANCE: Even with the slow reconfiguration times inherent in this platform, the locality-aware algorithm exhibits better performance for large, though reasonable input sizes. Our theoretical analysis suggests that if the shifting overhead were reduced to being on the order of milliseconds, the input sizes at which performance gains are seen could be reduced significantly. This points to improvements in network management hardware that could potentially be of greatest benefit to performance.

Empirical Result 2 (Switch Platform):

PLATFORM DESCRIPTION: A Bay Networks 28115 Adv switch, with 16 Fast-Ethernet ports (100 Mbits).

LOCALITY MAPPING: Any subset of two processors can constitute a subNOW, as well as the entire NOW. The entire NOW has bandwidth, latency, and shifting parameters which are significantly larger than the constant parameters for each subNOW of two processors: the NOW's parameters are proportional to the square root of the total number of processors it contains.

ALGORITHMS: Two different total exchange algorithms. The first total exchange algorithm exploits the agility of the platform (i.e. its low cost of shifting), and the significantly lower cost of communicating in disjoint pairs of processors, by using only disjoint pairs of processors. The other only uses the network as a whole.

SIGNIFICANCE: On this platform, the locality-aware algorithm outperforms the locality-unaware algorithm for all data sizes.

We are currently taking up the challenge to extend our ideas of dynamic, distributed network management to the emerging technology of virtual LANs.

References

[1] David Peleg. An Overview of Locality-Sensitive Distributed Computing (Preliminary Draft), 1996.

[2] P. de la Torre and C. P. Kruskal. Submachine Locality in the Bulk Synchronous Setting. In *Lecture Notes in Computer Science 1124, Euro-Par '96 Parallel Processing*, volume II, pages 352–358, Lyon, France, August 1996.