Logical locality entails frugal distributed computation over graphs (extended abstract)

Stéphane Grumbach and Zhilin Wu

INRIA-LIAMA*

Abstract. First-order logic is known to have limited expressive power over finite structures. It enjoys in particular the locality property, which states that first-order formulae cannot have a global view of a structure. This limitation ensures their low sequential computational complexity. We show that the locality impacts as well on their distributed computational complexity. We use first-order formulae to describe the properties of finite connected graphs, which are the topology of communication networks, on which the first-order formulae are also evaluated. We show that over bounded degree networks and planar networks, first-order properties can be frugally evaluated, that is, with only a bounded number of messages, of size logarithmic in the number of nodes, sent over each link. Moreover, we show that the result carries over for the extension of first-order logic with unary counting.

1 Introduction

Logical formalisms have been widely used in many areas of computer science to provide high levels of abstraction, thus offering user-friendliness while increasing the ability to verify properties. In the field of databases, first-order logic constitutes the basis of relational query languages, which allow to write queries in a declarative manner, independently of the physical implementation. In this paper, we propose to use logical formalisms to express properties of the topology of communication networks, that can be verified in a distributed fashion over the networks themselves.

We focus on first-order logic over graphs. First-order logic has been shown to have limited expressive power over finite structures. In particular, it enjoys the locality property, which states that first-order formulae are local [Gai82], in the sense that local areas of the graphs are sufficient to evaluate them.

First-order properties have been shown to be computable with very low complexity in both sequential and parallel models of computation. It was shown that first-order properties can be evaluated in linear time over classes of bounded degree graphs [See95] and over classes of locally tree-decomposable graphs[1] [FG01].

* INRIA – LIAMA – CASIA – Chinese Academy of Sciences – PO Box 2728 – Beijing 100080 – PR China – Stephane.Grumbach@inria.fr, zlwu@liama.ia.ac.cn

[1] Locally tree-decomposable graphs generalize bounded degree graphs, planar graphs, and graphs of bounded genus.

These results follow from the locality of the logic. It was also shown that they can be evaluated in constant time over Boolean circuits with unbounded fan-in (AC^0) [Imm89]. These bounds lead us to be optimistic on the complexity of the distributed evaluation of first-order properties.

We consider communication networks based on the message passing model [AW04], where nodes exchange messages with their neighbors. The properties to be evaluated concern the graph which forms the topology of the network, and whose knowledge is distributed over the nodes, which are only aware of their 1-hop neighbors. We thus focus on connected graphs.

In distributed computing, the ability to solve problems locally has attracted a strong interest since the seminal paper of Linial [Lin92]. The ability to solve global problems in distributed systems, while performing as much as possible local computations, is of great interest to ensure scalability. Moreover relying as much as possible on local information improves fault-tolerance. Finally, restricting the computation to local areas allows to optimize time and communication complexity.

Naor and Stockmeyer [NS95] showed that there were non-trivial locally checkable labelings that are locally computable, while on the other hand some lower-bounds have been exhibited, thus resulting in non-local computability results [KMW04,KMW06].

Different notions of local computation have been considered. The most widely accepted restricts the time of the computation to be constant, that is independent of the size of the network [NS95], while allowing messages of size $O(\log n)$, where n is the size of the network. This condition is rather stringent. Naor and Stockmeyer [NS95] show their result for a restricted class of graphs (eg bounded odd degree). Godard et al. used graph relabeling systems as the distributed computational model, defined local computations as graph relabeling systems with locally-generated local relabeling rules, and characterized the classes of graphs that are locally computable [GMM04].

Our initial motivation is to understand the impact of the logical locality on the distributed computation, and its relationship with local distributed computation. It is easy to verify though that there are simple properties (expressible in first-order logic) that cannot be computed locally. Consider for instance the property "There exist at least two distinct triangles", which requires non-local communication to check the distinctness of the two triangles which may be far away from each other. Nevertheless, first-order properties do admit simple distributed computations.

We thus introduce frugal distributed computations. A distributed algorithm is *frugal* if during its computation only a bounded number of messages of size $O(\log n)$ are sent over each link. If we restrict our attention to bounded degree networks, this implies that each node is only receiving a bounded number of messages. Frugal computations resemble local computations over bounded degree networks, since the nodes are receiving only a bounded number of messages, although these messages can come from remote nodes through multi-hop paths.

We prove that first-order properties can be frugally evaluated over bounded degree networks and planar networks (Theorem 2 and Theorem 4). The proofs are obtained by transforming the centralized linear time evaluation algorithms [See95,FG01] into distributed ones satisfying the restriction that only a bounded number of messages are sent over each link. Moreover, we show that the results carry over to the extension of first-order logic with unary counting. While the transformation of the centralized linear time algorithm is simple for first-order properties over bounded degree networks, it is quite intricate for first-order properties over planar networks. The most intricate part is the distributed construction of an ordered tree decomposition for some subgraphs of the planar network, inspired by the distributed algorithm to construct an ordered tree decomposition for planar networks with bounded diameter in [GW09].

Intuitively, since in the centralized linear time computation each object is involved only a bounded number of times, in the distributed computation, a bounded number of messages sent over each link could be sufficient to evaluate first-order properties. So it might seem trivial to design frugal distributed algorithms for first-order properties over bounded degree networks and planar networks. Nevertheless, this is not the case, because in the centralized computation, after visiting one object, any other object can be visited, but in the distributed computation, only the *adjacent* objects (nodes, links) can be visited.

The paper is organized as follows. In the next section, we recall classical graph theory concepts, as well as Gaifman's locality theorem. In Section 3, we consider the distributed evaluation of first-order properties over respectively bounded degree and planar networks. Finally, in Section 4, we consider the distributed evaluation of first-order logic with unary counting.

2 Graphs, first-order logic and locality

In this paper, our interest is focused to a restricted class of structures, namely finite graphs. Let $G = (V, E)$, be a finite graph. We use the following notations. If $v \in V$, then $deg(v)$ denotes the *degree* of v. For two nodes $u, v \in V$, the *distance* between u and v, denoted $dist_G(u, v)$, is the length of the shortest path between u and v. For $k \in \mathbb{N}$, the *k-neighborhood* of a node v, denoted $N_k(v)$, is defined as $\{w \in V | dist_G(v, w) \leq k\}$. If $\bar{v} = v_1 ... v_p$ is a collection of nodes in V, then the k-neighborhood of $\bar{v}$, denoted $N_k(\bar{v})$, is defined by $\bigcup_{1 \leq i \leq p} N_k(v_i)$. For $X \subseteq V$, let $\langle X \rangle^G$ denote the subgraph induced by X.

Let $G = (V, E)$ be a connected graph, a *tree decomposition* of G is a rooted labeled tree $\mathcal{T} = (T, F, r, B)$, where T is the set of vertices of the tree, $F \subseteq T \times T$ is the child-parent relation of the tree, $r \in T$ is the root of the tree, and B is a labeling function from T to 2^V, mapping vertices t of T to sets $B(t) \subseteq V$, called *bags*, such that

1. For each edge $(v, w) \in E$, there is a $t \in T$, such that $\{v, w\} \subseteq B(t)$.
2. For each $v \in V$, $B^{-1}(v) = \{t \in T | v \in B(t)\}$ is connected in T.

The *width* of $\mathcal{T}$, $width(\mathcal{T})$, is defined as $\max\{|B(t)|-1 | t \in T\}$. The tree-width of G, denoted $tw(G)$, is the minimum width over all tree decompositions of G. An *ordered tree decomposition* of width k of a graph G is a rooted labeled tree $\mathcal{T}=(T,F,r,L)$ such that:

- (T,F,r) is defined as above,
- L assigns each vertex $t \in T$ to a $(k+1)$-tuple $\overline{b^t}=(b_1^t,\cdots,b_{k+1}^t)$ of vertices of G (note that in the tuple $\overline{b^t}$, vertices of G may occur repeatedly),
- If $L'(t) := \{b_j^t | L(t)=(b_1^t,\cdots,b_{k+1}^t), 1 \le j \le k+1\}$, then (T,F,r,L') is a tree decomposition.

The *rank* of an (ordered) tree decomposition is the rank of the rooted tree, i.e. the maximal number of children of its vertices.

We consider first-order logic (FO) over the signature E, where E is a binary relation symbol. The syntax and semantics of first-order formulae are defined as usual [EF99]. The *quantifier rank* of a formula φ is the maximal number of nestings of existential and universal quantifiers in φ.

A *graph property* is a class of graphs closed under isomorphisms. Let φ be a first-order sentence, the graph property defined by φ, denoted $\mathcal{P}_\varphi$, is the class of graphs satisfying φ.

The distance between nodes can be defined by first-order formulae $dist(x,y) \le k$ stating that the distance between x and y is no larger than k, and $dist(x,y) > k$ is an abbreviation of $\neg dist(x,y) \le k$. In addition, let $\bar{x}=x_1...x_p$ be a list of variables, then $dist(\bar{x},y) \le k$ is used to denote $\bigvee\limits_{1 \le i \le p} dist(x_i,y) \le k$.

Let $\varphi(\bar{y})$ be a first-order formula with free variables $\bar{y}$, $k \in \mathbb{N}$, and $\bar{x}$ be a list of variables not occurring in $\varphi(\bar{y})$, then the formula bounding the quantifiers of $\varphi(\bar{y})$ to the k-neighborhood of $\bar{x}$, denoted $(\varphi(\bar{y}))^{(k)}(\bar{x})$, can be defined easily in first-order logic by using formulae $dist(\bar{x},y) \le k$. For instance, if $\varphi(\bar{y}) := \exists z \psi(\bar{y},z)$, then

$$(\varphi(\bar{y}))^{(k)}(\bar{x}) := \exists z \left(dist(\bar{x},z) \le k \wedge (\psi(\bar{y},z))^{(k)}(\bar{x}) \right).$$

We can now recall the notion of logical locality introduced by Gaifman [Gai82,EF99].

Theorem 1. *[Gai82] Let φ be a first-order formula with free variables $u_1,...,u_p$, then φ can be written in Gaifman Normal Form, that is into a Boolean combination of (i) sentences of the form:*

$$\exists x_1 ... \exists x_s \left(\bigwedge_{1 \le i < j \le s} d(x_i,x_j) > 2r \wedge \bigwedge_i \psi^{(r)}(x_i) \right) \tag{1}$$

and (ii) formulae of the form $\psi^{(t)}(\overline{y})$, where $\overline{y}=y_1...y_q$ such that $y_i \in \{u_1,...,u_p\}$ for all $1 \le i \le q$, $r \le 7^{k-1}$, $s \le p+k$, $t \le \left(7^k-1\right)/2$ (k is the quantifier rank of φ)[2]*.*

[2] The bound on r has been improved to $4^k - 1$ in [KL04]

Moreover, if φ is a sentence, then the Boolean combination contains only sentences of the form (1).

The locality of first-order logic is a powerful tool to demonstrate non-definability results [Lib97]. It can be used in particular to prove that counting properties, such as the parity of the number of vertices, or recursive properties, such as the connectivity of a graph, are not first-order.

3 Distributed evaluation of FO

We consider a message passing model of distributed computation [AW04], based on a communication network whose topology is given by a graph $G = (V, E)$ of diameter Δ, where E denotes the set of bidirectional *communication links* between nodes. From now on, we restrict our attention to *finite connected graphs*.

We assume that the distributed system is asynchronous and has no failure. The nodes have a unique *identifier* taken from $1, 2, \cdots, n$, where n is the number of nodes. Each node has distinct local ports for distinct links incident to it. The nodes have *states*, including final accepting or rejecting states.

For simplicity, we assume that there is only one query fired in the network by a *requesting node*. We also assume that a *breadth-first-search (BFS) tree* rooted on the requesting node has been pre-computed in the network[3], such that each node stores locally the identifier of its parent in the BFS-tree, and the states of the ports with respect to the BFS-tree, which are either "parent" or "child", denoting the ports corresponding to the tree edges, or "horizon", "upward", "downward", denoting the ports corresponding to the non-tree edges to some node with the same, smaller, or larger depth in the BFS-tree. The computation terminates, when the requesting node reaches a final state.

Let $\mathcal{C}$ be a class of graphs. A distributed algorithm is said to be *frugal* over $\mathcal{C}$ if there is a $k \in \mathbb{N}$ such that for any network $G \in \mathcal{C}$ of n nodes and any requesting node in G, the distributed computation terminates, with only at most k messages of size $O(\log n)$ sent over each link. If we restrict our attention to bounded degree networks, frugal distributed algorithms imply that each node only receives a bounded number of messages. Frugal computations resemble local computations over bounded degree networks, since the nodes receive only a bounded number of messages, although these messages can come from remote nodes through multi-hop paths.

Let $\mathcal{C}$ be a class of graphs, and φ an FO sentence, we say that φ can be distributively evaluated over $\mathcal{C}$ if there exists a distributed algorithm such that for any network $G \in \mathcal{C}$ and any requesting node in G, the computation of the distributed algorithm on G terminates, with the requesting node in the accepting state if and only if $G \models \varphi$. Moreover, if there is a frugal distributed algorithm to do this, then we say that φ can be frugally evaluated over $\mathcal{C}$.

[3] The pre-computation of the BFS tree can be done in $O(\Delta)$ distributed time and with $O(\Delta)$ messages sent over each link [BDLP08]

For centralized computations, it has been shown that Gaifman's locality of FO entails linear time evaluation of FO properties over classes of bounded degree graphs and classes of locally tree-decomposable graphs [See95,FG01]. In the following, we show that it is possible to design frugal distributed evaluation algorithms for FO properties over bounded degree and planar networks, by carefully transforming the centralized linear time evaluation algorithms into distributed ones with computations on each node well balanced.

3.1 Bounded degree networks

We first consider the evaluation of FO properties over bounded degree networks. We assume that each node stores the degree bound k locally.

Theorem 2. *FO properties can be frugally evaluated over bounded degree networks.*

Theorem 2 can be shown by using Hanf's technique [FSV95], in a way similar to the proof of Seese's seminal result [See95].

Let $r \in \mathbb{N}$, $G = (V, E)$, and $v \in V$, then the r-*type* of v in G is the isomorphism type of $\left(\langle N_r(v)\rangle^G, v\right)$. Let $r, m \in \mathbb{N}$, G_1 and G_2 be two graphs, then G_1 and G_2 are said to be (r, m)-equivalent if and only if for every r-type τ, either G_1 and G_2 have the same number of vertices with r-type τ or else both have at least m vertices with r-type τ. G_1 and G_2 are said to be k-equivalent, denoted $G_1 \equiv_k G_2$, if G_1 and G_2 satisfy the same FO sentences of quantifier rank at most k. It has been shown that:

Theorem 3. *[FSV95] Let $k, d \in \mathbb{N}$. There exist $r, m \in \mathbb{N}$ such that r depends only on k, m depends on both k and d, and for any graphs G_1 and G_2 with maximal degree no more than d, if G_1 and G_2 are (r, m)-equivalent, then $G_1 \equiv_k G_2$.*

Let us now sketch the proof of Theorem 2, which relies on a distributed algorithm consisting of three phases. Suppose the requesting node requests the evaluation of some FO sentence with quantifier rank k. Let r, m be the natural numbers depending on k, d specified in Theorem 3.

Phase I The requesting node broadcasts messages along the BFS-tree to ask each node to collect the topology information in its r-neighborhood;
Phase II Each node collects the topology information in its r-neighborhood;
Phase III The r-types of the nodes in the network are aggregated through the BFS-tree to the requesting node up to the threshold m for each r-type. Finally the requesting node decides whether the network satisfies the FO sentence or not by using the information about the r-types.

It is easy to see that only a bounded number of messages are sent over each link in Phase I and II. Since the total number of distinct r-types with degree bound d depends only upon r and d and each r-type is only counted up to a threshold m, it turns out that over each link, only a bounded number of messages are sent in Phase III as well. So the above distributed evaluation algorithm is frugal over bounded degree networks.

3.2 Planar networks

We now consider the distributed evaluation of FO properties over planar networks.

A *combinatorial embedding* of a planar graph $G = (V, E)$ is an assignment of a cyclic ordering of the set of incident edges to each vertex v such that two edges (u, v) and (v, w) are in the same face iff (v, w) is immediately before (v, u) in the cyclic ordering of v. Combinatorial embeddings, which encode the information about boundaries of the faces in usual embeddings of planar graphs into the planes, are useful for computing on planar graphs. Given a combinatorial embedding, the boundaries of all the faces can be discovered by traversing the edges according to the above condition.

We assume in this subsection that a combinatorial embedding of the planar network is distributively stored in the network, i.e. a cyclic ordering of the set of the incident links is stored in each node of the network.

Theorem 4. *FO properties can be frugally evaluated over planar networks.*

For the proof of Theorem 4, we first recall the centralized linear time algorithm to evaluate FO properties over planar graphs in [FG01][4].

Let $G = (V, E)$ be a planar graph and φ be an FO sentence. From Theorem 1, we know that φ can be written into Boolean combinations of sentences of the form (1),

$$\exists x_1 ... \exists x_s \left(\bigwedge_{1 \leq i < j \leq s} d(x_i, x_j) > 2r \wedge \bigwedge_i \psi^{(r)}(x_i) \right).$$

It is sufficient to show that sentences of the form (1) are linear-time computable over G. The centralized algorithm to evaluate FO sentences of the form (1) over planar graphs consists of the following four phases:

1. Select some $v_0 \in V$, let $\mathcal{H} = \{G[i, i + 2r] | i \geq 0\}$, where $G[i, j] = \{v \in V | i \leq dist_G(v_0, v) \leq j\}$;
2. For each $H \in \mathcal{H}$, compute $K_r(H)$, where $K_r(H) := \{v \in H | N_r(v) \subseteq H\}$;
3. For each $H \in \mathcal{H}$, compute $P_H := \{v \in K_r(H) | \langle H \rangle^G \models \psi^{(r)}(v)\}$;
4. Let $P := \cup_H P_H$, determine whether there are s distinct nodes in P such that their pairwise distance is greater than $2r$.

In the computation of the 3rd and 4th phase above, an automata-theoretical technique to evaluate Monadic-Second-Order (MSO) formulae in linear time over classes of graphs with bounded tree-width [Cou90,FG06,FFG02] is used. In the following, we recall this centralized evaluation algorithm.

MSO is obtained by adding set variables and set quantifiers into FO, such as $\exists X \varphi(X)$ (where X is a set variable). MSO has been widely studied in the

[4] In fact, in [FG01], it was shown that FO is linear-time computable over classes of locally tree-decomposable graphs.

context of graphs for its expressive power. For instance, 3-colorability, transitive closure or connectivity can be defined in MSO [Cou08].

The centralized linear time evaluation of MSO formulae over classes of bounded tree-width graphs goes as follows. First an ordered tree decomposition $\mathcal{T}$ of the given graph is constructed. Then from the given MSO formula, a tree automaton $\mathcal{A}$ is obtained. Afterwards, $\mathcal{T}$ is transformed into a labeled tree $\mathcal{T}'$, finally $\mathcal{A}$ is run over $\mathcal{T}'$ (maybe several times for formulae containing free variables) to get the evaluation result.

In the rest of this section, we design a frugal distributed algorithm to evaluate FO sentences over planar networks by adapting the above centralized algorithm. The main difficulty is to distribute the computation among the nodes such that only a bounded number of messages are sent over each link during the computation.

Phase I The requesting node broadcasts the FO sentence of the form (1) to all the nodes in the network through the BFS tree;
Phase II For each $v \in V$, compute $C(v) := \{i \geq 0 | v \in G[i, i+2r]\}$;
Phase III For each $v \in V$, compute $D(v) := \{i \geq 0 | N_r(v) \subseteq G[i, i+2r]\}$;
Phase IV For each $i \geq 0$, compute $P_i := \{v \in V | i \in D(v), \langle G[i, i+2r]\rangle^G \models \psi^{(r)}(v)\}$;
Phase V Let $P := \bigcup_i P_i$, determine whether there are s distinct nodes labeled by P such that their pairwise distance is greater than $2r$.

Phase I is trivial. Phase II is easy. In the following, we illustrate the computation of Phase III, IV, and V one by one.

We first introduce a lemma for the computation of Phase III.

For $W \subseteq V$, let $K_i(W) := \{v \in W | N_i(v) \subseteq W\}$. Let $D_i(v) := \{j \geq 0 | v \in K_i(G[j, j+2r])\}$.

Lemma 1. *For each $v \in V$ and $i > 0$, $D_i(v) = C(v) \cap \bigcap_{w:(v,w)\in E} D_{i-1}(w)$.*

With Lemma 1, $D(v) = D_r(v)$ can be computed in an inductive way to finish Phase III: Each node v obtains the information $D_{i-1}(w)$ from all its neighbors w, and performs the in-node computation to compute $D_i(v)$.

Now we consider Phase IV.

Because $\psi^{(r)}(x)$ is a local formula, $\psi^{(r)}(x)$ can be evaluated separately over each connected component of $G[i, i+2r]$ and the results are stored distributively.

Let C_i be a connected component of $G[i, i+2r]$, and $w_1^i, \cdots, w_l^i$ be all the nodes contained in C_i with distance i from the requesting node. Now we consider the evaluation of $\psi^{(r)}(x)$ over C_i.

Let C_i' be the graph obtained from C_i by including all ancestors of $w_1^i, \cdots, w_l^i$ in the BFS-tree, and C_i^* be the graph obtained from C_i' by contracting all the ancestors of $w_1^i, \cdots, w_l^i$ into one vertex, i.e. C_i^* has one more vertex, called the virtual vertex, than C_i, and this vertex is connected to $w_1^i, \cdots, w_l^i$. It is easy to see that C_i^* is a planar graph with a BFS-tree rooted on v^* and of depth at most $2r+1$. So C_i^* is a planar graph with bounded diameter.

An ordered tree decomposition for planar networks with bounded diameter can be distributively constructed with only a bounded number of messages sent over each link as follows [GW09]:

- Do a depth-first-search to decompose the network into blocks, i.e. biconnected components;
- Construct an ordered tree decomposition for each nontrivial block: Traverse every face of the block according to the cyclic ordering at each node, triangulate all those faces, and connect the triangles into a tree decomposition by utilizing the pre-computed BFS tree;
- Finally the tree decompositions for the blocks are connected together into a complete tree decomposition for the whole network.

By using the distributed algorithm for the tree decomposition of planar networks with bounded diameter, we can construct distributively an ordered tree decomposition for C_i^*, while having the virtual vertex in our mind, and get an ordered tree decomposition for C_i.

With the ordered tree decomposition for C_i, we can evaluate $\psi^{(r)}(x)$ over C_i by using the automata-theoretical technique, and store the result distributively in the network (each node stores a Boolean value indicating whether it belongs to the result or not).

A distributed post-order traversal over the BFS tree can be done to find out the connected components of all $G[i, i+2r]$'s and construct the tree decompositions for these connected components one by one.

Finally we consider Phase V.

Label nodes in $\bigcup_i P_i$ with P.

Then consider the evaluation of FO sentence φ' over the vocabulary $\{E, P\}$,

$$\exists x_1 ... \exists x_s \left(\bigwedge_{1 \leq i < j \leq s} d(x_i, x_j) > 2r \wedge \bigwedge_i P(x_i) \right).$$

Starting from some node w_1 with label P, mark the vertices in $N_{2r}(w_1)$ as Q, then select some node w_2 outside Q, and mark those nodes in $N_{2r}(w_2)$ by Q again, continue like this, until w_l such that either $l = s$ or all the nodes with label P have already been labeled by Q.

If $l < s$, then label the nodes in $\bigcup\limits_{1 \leq i \leq l} N_{4r}(v_i)$ as I. Each connected component of $\langle I \rangle^G$ has diameter no more than $4lr < 4sr$. We can construct distributively a tree decomposition for each connected component of $\langle I \rangle^G$, and connect these tree decompositions together to get a complete tree-decomposition of $\langle I \rangle^G$, then evaluate the sentence φ' by using this complete tree decomposition.

4 Beyond FO properties

We have shown that FO properties can be frugally evaluated over respectively bounded degree and planar networks. In this section, we extend these results to FO unary queries and some counting extension of FO.

From Theorem 1, an FO formula $\varphi(x)$ containing exactly one free variable x can be written into a Boolean combinations of sentences of the form (1) and local formulae $\psi^{(t)}(x)$. Then it is not hard to prove the following result.

Theorem 5. *FO formulae $\varphi(x)$ with exactly one free variable x can be frugally evaluated over respectively bounded degree and planar networks, with the results distributively stored on the nodes of the network.*

Counting is one of the ability that is lacking to first-order logic, and has been added in commercial relational query languages (e.g. SQL). Its expressive power has been widely studied [GO92,GT95,Ott96] in the literature. Libkin [Lib97] proved that first-order logic with counting still enjoys Gaifman locality property. We prove that Theorem 2 and Theorem 4 carry over as well for first-order logic with unary counting.

Let FO(#) be the extension of first-order logic with unary counting. FO(#) is a two-sorted logic, the first sort ranges over the set of nodes V, while the second sort ranges over the natural numbers $\mathbb{N}$. The terms of the second sort are defined by: $t := \#x.\varphi(x) \mid t_1 + t_2 \mid t_1 \times t_2$, where φ is a formula over the first sort with one free variable x. Second sort terms of the form $\#x.\varphi(x)$ are called *basic* second sort terms.

The atoms of FO(#) extend standard FO atoms with the following two unary counting atoms: $t_1 = t_2 \mid t_1 < t_2$, where t_1, t_2 are second sort terms. Let t be a second sort term of FO(#), $G = (V, E)$ be a graph, then the interpretation of t in G, denoted t^G, is defined as follows:

- $(\#x.\varphi(x))^G$ is the cardinality of $\{v \in V | G \models \varphi(v)\}$;
- $(t_1 + t_2)^G$ is the sum of t_1^G and t_2^G;
- $(t_1 \times t_2)^G$ is the product of t_1^G and t_2^G.

The interpretation of FO(#) formulae is defined in a standard way.

Theorem 6. *FO(#) properties can be frugally evaluated over respectively bounded degree and planar networks.*

The proof of the theorem relies on a normal form of FO(#) formulae.

5 Conclusion

We have shown that logical formulae used to express properties of graphs, which constitute the topology of communication networks, can be evaluated very efficiently over these networks. Their distributed computation, although not local, can be done *frugally*, that is with a bounded number of messages of logarithmic size exchanged over each link, over respectively bounded degree and planar networks. The frugal computation, introduced in this paper, generalizes local computation and offers a large spectrum of applications. Moreover the results carry over to the extension of first-order logic with unary counting. The distributed time used in the frugal evaluation of FO properties over bounded degree networks is $O(\Delta)$, while that over planar networks is $O(n)$.

We assumed that a BFS tree is pre-computed and stored distributively in the network. Evidently the BFS-tree varies when the requesting node is chosen differently. Since a BFS-tree is a tree 2-spanner [CC95] of the network, we can actually assume that a tree 2-spanner, independent of the choice of the requesting node, is distributively pre-computed and stored in the network, and we still guarantee the frugality of the computation by adapting slightly the distributed evaluation algorithms in Section 3.

Beyond its interest for logical properties, the frugality of distributed algorithms, which ensures an extremely good scalability of their computation, raises fundamental questions, such as deciding what can be frugally computed. We leave as an open problem the question of deciding whether for instance a Hamiltonian path can be computed frugally.

References

[AW04] Hagit Attiya and Jennifer Welch. *Distributed Computing: Fundamentals, Simulations and Advanced Topics*. Wiley-Interscience, 2004.

[BDLP08] Christian Boulinier, Ajoy K. Datta, Lawrence L. Larmore, and Franck Petit. Space efficient and time optimal distributed BFS tree construction. *Inf. Process. Lett.*, 108(5):273–278, 2008.

[Bod93] Hans L. Bodlaender. A linear time algorithm for finding tree-decompositions of small treewidth. In *ACM STOC*, 1993.

[CC95] Leizhen Cai and Derek G. Corneil, Tree Spanners, SIAM J. Discret. Math., 8(3), 359-387, 1995.

[Cou90] Bruno Courcelle. Graph rewriting: An algebraic and logic approach. In *Handbook of Theoretical Computer Science, Volume B: Formal Models and Sematics (B)*, pages 193–242. Elsevier and MIT Press, 1990.

[Cou08] Bruno Courcelle. *Graph algebras and monadic second-order logic*. In preparation, to be published by Cambridge University Press, 2008.

[EF99] H.D. Ebbinghaus and J. Flum. *Finite model theory*. Springer, 1999.

[FFG02] Jörg Flum, Markus Frick, and Martin Grohe. Query evaluation via tree-decompositions. *J. ACM*, 49(6):716–752, 2002.

[FG01] Markus Frick and Martin Grohe. Deciding first-order properties of locally tree-decomposable structures. *J. ACM*, 48(6):1184–1206, 2001.

[FG06] J. Flum and M. Grohe. *Parameterized Complexity Theory*. Springer, 2006.

[FSV95] Ronald Fagin, Larry J. Stockmeyer, and Moshe Y. Vardi. On a monadic NP vs monadic co-NP. *Inf. Comput.*, 120(1):78–92, 1995.

[Gai82] H. Gaifman. On local and non-local properties. In *Proceedings of the Herbrand Symposium, Logic Colloquium'81, North Holland*, 1982.

[GMM04] Emmanuel Godard, Yves Métivier, and Anca Muscholl. Characterizations of classes of graphs recognizable by local computations. *Theory Comput. Syst.*, 37(2):249–293, 2004.

[GO92] Erich Grädel and Martin Otto. Inductive definability with counting on finite structures. In *Computer Science Logic, CSL*, pages 231–247, 1992.

[GT95] Stéphane Grumbach and Christophe Tollu. On the expressive power of counting. *Theor. Comput. Sci.*, 149(1):67–99, 1995.

[GW09] Stéphane Grumbach and Zhilin Wu. On the distributed evaluation of MSO on graphs. manuscript, 2009.

[Imm89] Neil Immerman. Expressibility and parallel complexity. *SIAM J. Comput.*, 18(3):625–638, 1989.

[KL04] H. Jerome Keisler and Wafik Boulos Lotfallah. Shrinking games and local formulas. *Ann. Pure Appl. Logic*, 128(1-3):215–225, 2004.

[KMW04] Fabian Kuhn, Thomas Moscibroda, and Roger Wattenhofer. What cannot be computed locally! In *ACM PODC*, 2004.

[KMW06] Fabian Kuhn, Thomas Moscibroda, and Roger Wattenhofer. The price of being near-sighted. In *Seventeenth ACM-SIAM SODA*, 2006.

[Lib97] Leonid Libkin. On the forms of locality over finite models. In *LICS*, pages 204–215, 1997.

[Lin92] Nathan Linial. Locality in distributed graph algorithms. *SIAM J. Comput.*, 21(1):193–201, 1992.

[NS95] Moni Naor and Larry J. Stockmeyer. What can be computed locally? *SIAM J. Comput.*, 24(6):1259–1277, 1995.

[Ott96] Martin Otto. The expressive power of fixed-point logic with counting. *J. Symb. Log.*, 61(1):147–176, 1996.

[Pel00] David Peleg. *Distributed computing: a locality-sensitive approach*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2000.

[See95] Detlef Seese. Linear time computable problems and logical descriptions. *Electr. Notes Theor. Comput. Sci.*, 2, 1995.

[TW68] J.W. Thatcher and J.B. Wright. Generalized finite automata theory with an application to a decision problem of second-order logic. *Math. Systems Theory*, 2(1):57–81, 1968.