# Random walks in a dynamic small-world space: robust routing in large-scale sensor networks

Behnam A. Rezaei
Electrical Engineering Department
University of California, Los Angeles
,California 90095
Email: behnam@ee.ucla.edu

Nima Sarshar
Electrical and Computer
Engineering Department
McMaster University
,Hamilton, Ontario L8S 4K1
nima@grads.ece.mcmaster.ca

Vwani P. Roychowdhury
Electrical Engineering Department
University of California, Los Angeles
,California 90095
Email: vwani@ee.ucla.edu

***Abstract*— The task of moving data (i.e., the *routing* problem) in large-scale sensor networks has to contend with several obstacles, including severe power constraints at each node and temporary but random failures of nodes, which render well-known routing schemes designed for traditional communication networks (e.g., IP, telephone, mobile ad-hoc, and cellular) ineffective. In this paper, we consider the open problem of finding optimum routes between any fixed source-destination pair in a large-scale network, such that the communication load (i.e., the required power) is distributed among all the nodes, the overall latency is minimized, and the algorithm is decentralized and robust. While a recent work has addressed this problem in the context of a grid topology and shown how to obtain a load-balanced routing, the transmissions are restricted to be among near-neighbors and the overall latency grows linearly with the number of nodes.**

**In this paper, we show how one can route messages between source and destination nodes along random small-world topologies using a decentralized algorithm. Specifically, nodes make connections independently (based only on the source and destination information in the packets), according to a distribution that guarantees an average latency of $O(log^2(N))$, while preventing hotspot regions by providing almost uniform distribution of traffic load over all nodes. Surprisingly, the randomized nature of the network structure keeps the average per-node power consumption almost the same as in the case of a grid topology (i.e., local transmissions), while providing an exponential reduction in latency, resulting in a highly fault-tolerant and stable design capable of working in very dynamic environments.**

## I. Introduction

The production of small, low-power, low-cost and multi-functional sensors has recently become possible in light of advances in digital wireless communication, MEMS technology and digital electronics [1], [2]. Large-scale networks of these new sensors are being conceived for a wide range of industrial applications.

In many applications, such as image recognition or pattern recognition, there is often high demand on a few particular sensors in the network relative to other sensors, creating *hot-spots* in the network. In such cases, network performance will be limited by the power available to the few sensors carrying the high traffic load. The delay time it takes for a signal to travel across the network, or the latency, is another important issue for many applications in large-scale sensor networks. Since the number of nodes in the type of network considered here can be orders of magnitude larger than the number of nodes in a traditional network, we expect that the latencies associated with these networks could become prohibitive. It is imperative, then, that we design our routing algorithm to both minimize latency and efficiently balance the load on the network. In this paper, we propose an algorithm for finding optimum routes between any fixed source and destination points in a large-scale network, where optimality is defined in terms of certain desirable macroscopic properties, particularly load balance, latency and fault tolerance.

Recently, there have been several attempts to adapt algorithms from other large-scale systems, such as biological, statistical mechanical, and social networks to address the routing problem in sensor networks. All such attempts must optimize macroscopic properties like latency, scalability, etc., while retaining a simple set of microscopic rules, in keeping with the low computational power of the individual nodes. In the context of the problem we are addressing, Servetto *et al* [3] presented the first such results, which show how a proper random walk on a grid can provide load balanced, multi-path routing between a source-destination pair. This class of protocols considers multiple path routing between a fixed source and destination. Servetto *et al* proposed an algorithm that distributes the load evenly over all the nodes at the same grid distance to source and destination, i.e. over all nodes lying along a diagonal perpendicular to the line between source and destination. Their algorithm achieves load balance over the nodes; however, it suffers from a latency that scales linearly with the size of the network.

In this paper, continuing the line of work of Servetto *et al* [3], we propose a new local algorithm (i.e., a decentralized algorithm where the intermediate nodes make a decision about where to transmit the received package based solely on the information about the source and the destination and its own position in the network) for passing data based on a random walk on a dynamic small world model [7], [6]. The proposed algorithm finds multiple paths between a fixed source and destination node pair. We show how this abstract model can be utilized in sensor networks and how it proves to be robust to path loss and limited power requirement of sensors. The proposed scheme achieves a trade-off between latency and power consumption. In particular, it achieves poly-logarithmic latency with size of the network and the power requirement is distributed over the network. Thus, the average power requirement *per node* remains scalable.

Multi-path routing is associated with an extensive and rich literature. In the context of sensor networks, several different multi-path algorithms have been studied. These have typically fallen into one of three broad classes: flooding techniques, cluster methods, and and gradient methods. These traditional multi-path routing algorithms are essentially different from our algorithm and the problem that we are addressing. These algorithms[8], [9], [10], [4] consider the problem of finding the desired content in the network, then finding multiple paths between the destination and source. It is assumed that the network links are not configurable, and the subsequent routing has to be performed using the links that are available. We, however, address another aspect of the multi-path routing, that is finding multiple paths between a source and destination while the position of the destination is known. Moreover, our problem is different in nature since we design how the network should be formed such that it has right properties, i.e. low latency, load balanced and fault tolerance; each routing configures its own path and links.

The rest of the paper is organized as follows. In Section II we describe in detail our decentralized random-walk algorithm for a grid network topology. We then define and calculate figures of merit for this algorithm in Section III. In Section IV we characterize effect of initial conditions using Monte Carlo simulations. Finally, the performance of the proposed method and quantitative comparison to existing methods are summarized in Section V.

## II. Proposed algorithm

We consider an $N+1$ by $N+1$ grid of nodes $\{(i,j)| -N/2 < i < N/2, -N/2 < j < N/2\}$, each with total energy $E_t$. We assume a path loss model of $r^{-\alpha}$, where $\alpha \geq 2$ depends on the physical model. The routing problem is defined as follows: a sink node, $S$, randomly located in the network, makes inquiries for data at some position $D$.[1] This inquiry process sets up a path between $S$ and $D$, and a stream of data is sent back to the sink node using the same path. To analyze a fixed transmission pair, we assume without loss of generality that the destination node is at the origin. We propose a random, decentralized routing algorithm which uses collaboration of all the sensors in the network, instead of relying on a finite subset of the nodes. A new routing path is generated from source to destination for each transmission. Local, independent decisions at the node level are used to construct a new routing path as an instance of a small world network.

Before the algorithm is executed, we assume that the source node connects to a randomly chosen node within a circle of radius $N/4$ of the destination. We will discuss this assumption further later in this section. The proposed algorithm proceeds as follows:

**Contraction Random Walk Algorithm**

1) Each packet is assumed to have only source and destination node information.
2) Upon receiving a packet, the receiving node $v$ selects a long range contact node $u$ from the set of all nodes. The probability of a node $u$ being selected is $\frac{1/d^2}{\Sigma_{\forall u} 1/d^2}$, where $d$ is defined as the Euclidean distance between $u$ and $v$, $d((i,j),(k,l)) = \sqrt{(k-i)^2 + (l-j)^2}$. It is easily shown that $\Sigma_{\forall u} 1/d^2 < 4\ln(6N)$ [7].
3) If $d(v,t) > d(u,t)$, then $v$ passes the inquiry to $u$. If $d(v,t) \leq d(u,t)$, then $v$ passes the inquiry to its nearest neighbor closer to $t$.
4) Steps 2 and 3 are repeated until inquiry reaches the destination node $t$.

The algorithm essentially follows a shortest- path route on an instance of a small world network created for each routing.

In the algorithm description above, we assumed that the source node connects to a randomly chosen node inside or on the perimeter a circle of radius $N/4$ around the destination; this requires the source node to always use long-range connections for its communications. There is a cost associated with long range connections (e.g. power requirement is proportional to distance); thus, in this model, the sink node requires much higher resources than regular nodes. Often a time this is the common scenario in practical applications since a sink node with unlimited resources makes queries from different parts of the network. Whereas in some application all the nodes are homogenous and the source sensor has same power limitations as other sensors in the network. We may relax this assumption by performing a random walk similar to contraction walk introduced above but away from the source node in an expansion phase followed by the contraction phase described above to achieve uniform distribution over the network, a solution very similar to Servetto *et. al.* model.

**Expansion Random Walk Algorithm**

1) Each packet is assumed to have only source and destination node information.
2) Upon receiving a packet, the receiving node, $v$, selects a long range contact node, $u$ randomly from the set of all nodes. The probability that node $u$ is selected is $\frac{1/d^2}{\Sigma_{\forall u} 1/d^2}$, where $d$ is defined as the Euclidean distance between $u$ and $v$, $d((i,j),(k,l)) = \sqrt{(k-i)^2 + (l-j)^2}$.
3) If $d(v,s) > d(u,s)$, and $d(u,v) < d(v,s)$, $v$ passes the inquiry to $u$. Otherwise, $v$ passes the inquiry to its nearest neighbor further away from $s$.
4) Steps 2 and 3 are repeated until the inquiry arrives at a node at distance greater than $N/4$ from the source.

[1]Note that throughout this paper, we assume that sensors have access to positioning information. There are several ways this can be achieved, including (i) in many applications, the destination is usually a spatial location and not a particular sensor node, in which case the routing algorithm will find the destination sensor, $T$, that is the closest sensor to the location $D$. (ii) One could first perform a local search to identify the location of the node with a desired content, and thereby fix the destination location, (iii) one could first execute a distributed position determination algorithm to locate the sensor nodes and their relative distances.

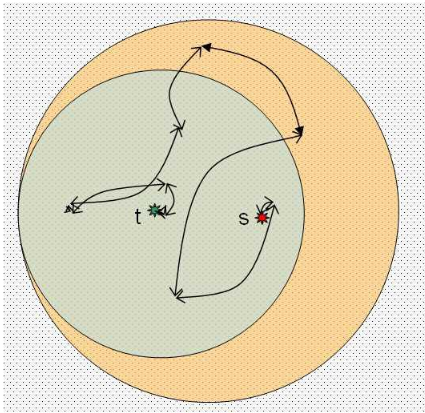



Fig. 1. A schematic of the routing algorithm. An expansion phase (orange) originates at the source node and expands outward until it reaches a node within at least $N/4$ of destination node. It is then followed by a contraction phase to get to the destination node. The routing path is formed through a random walk on the small world formed as a combination of local and long-range connections.

Figure 1 is a simple schematic of how an expansion phase is followed by a contraction phase to get from a fixed source to a fixed destination through multiple paths.

## III. Network performance

In this section, we calculate the approximate distributions of different performance metrics in our model network. Specifically, we are interested in calculating the latency and average power consumption of transmissions routed across a large-scale network using the proposed algorithm. We define the *phase* of the routing algorithm as follows: if the lattice distance from the current node to the destination is between $2^j$ and $2^{j+1}$, we say that the algorithm is in phase $j$.

### A. Latency

In this section, we show that the average latency of our algorithm is considerably less than O(N) hops and, in fact, increases only polylogarithmically with network size, thus achieving scalable latency.

*Proposition 3.1:* For the defined network structure and routing scheme, the average latency is $O(log(N)^2)$.

*Proof:* The polylogarithmic dependence of the latency on $N$ is the direct result of the general theorem in [7], since an instance of the small world structure is created for each routing. A full proof of this proposition for general small world models is given in [7]; here, we provide a simple intuitive proof.

The long-range probability distribution ($P \propto 1/d^2$) assures that, at all resolutions, the probability of going to next resolution is bounded by $\Omega(1/\log(N))$. Thus, in expected $\log(N)$ steps, the algorithm will move to the next phase. The algorithm passes through at most $O(\log(N))$ phases; therefore, the algorithm will reach the last phase (i.e., within $O(\log(N))$ steps of the destination node) in expected steps of $O(\log(N)^2)$. ■

### B. Per-Node Range Distribution and Load

In this section, we derive the stationary connection distribution for a node at distance $r$ from the destination node. We also calculate the probability that a node at distance $r$ participates in a given route. These results provide an insight on how the algorithm uses collaboration of all the nodes in the network.

*Proposition 3.2:* The probability that a node at distance $r$ from the destination participates in a given routing path is proportional to $\frac{1}{r^2}$.

*Proof:* Here we just give intuition behind the proof; Approximate solution using a continuous approximation approach for the stationary distribution of the underlying 2D Markov chain is given in [11].

The algorithm spends on average $\log(N)$ (and almost surely less than $\log(N)^2$) steps in each phase. Phase $j$ with $r_j = 2^j$ has $O(r_j^2)$ nodes, so the likelihood of any node in phase $j$ participating in a given route is $\log(N)/r_j^2$ on average, or $\log(N)^2/r_j^2$ at most. ■

This shows that the distribution is close to $O(\frac{1}{r^2})$ within a polylog factor of $r$.

*Proposition 3.3:* The probability that a node at distance $r$ from the destination makes a connection to a node at distance $x$ is proportional to $1/x$ for $x \leq r$.

*Proof:* The probability of making a long range connection to a node at distance $x$ is $1/x^2$. There are $O(x)$ nodes at distance $x$, which results in a distribution of $O(1/x)$. ■

### C. Power Distribution and Load Balance

In Section II, we introduced a routing algorithm for an idealized network model that takes advantage of collaboration of the nodes in the network to achieve low latency. In this section, we define the power requirement for communication between sensors as a function of their distance. i.e. $P \propto d^{\alpha}$. Then we compute the total power required for a single transmission between source and destination and also average per node power requirement.

*Lemma 3.4:* For the network model described above, with a source and destination pair at distance O(N) times nearest-neighbor distance, the total power requirement for a routing with latency $\Delta = \delta$ is at least $(\frac{N}{\delta})^{\alpha} P_0$.

*Proof:* The lemma follows from the fact that, for a routing with latency $\delta$, there is at least one communication with range $1/\delta$, which is $N/\delta$ times the range of nearest-neighbor communication. ■

Lemma 3.4 proves that scalability in both the total power and the latency can not be achieved simultaneously. While total power requirement can not scale with low latency routing but per node power requirement can still scale by distributing the load over the network.

*Theorem 1:* The total power requirement, $P_t$, grows as $O((\frac{N}{\log^2(N)})^{\alpha})P_0$, but the *per-node* power requirement for a node at distance $r$ grows as $O(\frac{1}{log(r)}(r^{\alpha-2}))P_0$.

*Proof:* This is the direct result of the proposition 3.1, since this algorithm enjoys the same efficiency as the abstract model. The detail of the proof is given in [11]

| | |
|---|---|
| Latency | $O(\log(N)^2)$ |
| Node Activity | $O(1/r^2)$ |
| Per Node Average Energy | $O(\frac{1}{log(r)}(r^{\alpha-2}))P_0$ |

TABLE I

THE SCALING PROPERTIES OF THE PROPOSED ABSTRACT MODEL WITH $N$(SIZE OF THE GRID). $r$ IS THE DISTANCE TO THE DESTINATION NODE.

The average power requirement over time for a node at distance $r$ from source , $P(r)$, is given as

$$\begin{aligned} P(r) &= \Pi_r \overline{P}_r \\ &\approx \frac{1}{log(r)}(r^{\alpha-2})P_0 \end{aligned} \tag{1}$$

Thus, for the limiting case of $\alpha = 2$, the average per-node power requirement is *independent of the network size* $N$, and depends on the distance to destination, $r$ approximately as $(r^{\alpha-2})$. ∎

### D. Fault Tolerance and Security

The ramifications of our algorithm for network security and fault tolerance are summarized as follows:

- An attack in which nodes are disabled at random must create at least $O(N^2)$ faulty nodes to be effective. Thus, data can be retrieved from the network unless a constant fraction of the nodes in the network are disabled.
- If an attacker disables $O(1)$ nodes, at least one faulty node must be within a distance $r < polylog(N)$ of the destination in order for the attack to be effective. Thus, an attacker must target the destination point very accurately in order to effectively block transmissions to or from this point.

Here, we have designated an attack or a failure *effective* if the average number of routes broken is $\Omega(1/polylog(N))$. A path is *broken* if one node in the path is not active. For a node failure at distance $r$ from the destination node, the probability that this node is on the current route , i.e., the probability that the route is broken, is $O(\frac{1}{r^2 log(N)^2})$; thus, for a finite size attack in which the number of faulty nodes is O(1), $O(\frac{1}{r^2 log(N)^2})$ of the packets will not be received. In order for a single node failure to block a constant fraction of transmissions, then, the faulty node must be within a distance $r < log(N)$ of the destination. If nodes are chosen randomly, then $r$ is O(N) on average, so at least $O(N^2)$ node failures, a constant fraction of all the nodes, are required to disable a fraction of communications.

Table I summarizes figures of merit for different solutions.

The results we have derived so far do not take into account initial condition.

## IV. NUMERICAL SIMULATIONS

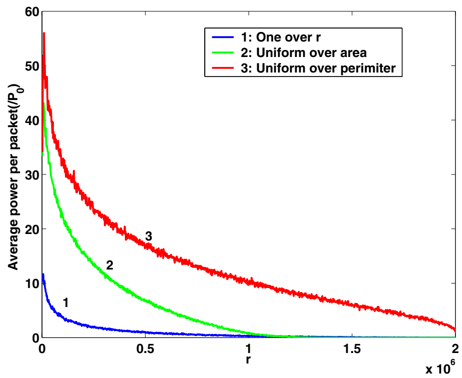
In section II we calculated different properties of the proposed algorithm. However we did not consider the effect of initial conditions. The transient and mixing times for the expansion and contraction phases of the proposed algorithm can not be exactly calculated analytically, so we rely on numerical simulations to verify that the results hold for various initial conditions. Contraction phase simulations were performed for three different initial scenarios.

1) A sink node with unlimited resources makes a connection to a node within circle of radius $N/4$ around the destination with uniform probability distribution.
2) A sink node with unlimited resources makes a connection to a node within the circle of radius $N/4$ around the destination with probability proportional to $1/r^2$, where $r$ is the distance $r$ to the destination.
3) A sink node with limited resources undergoes an expansion phase followed by a contraction phase.

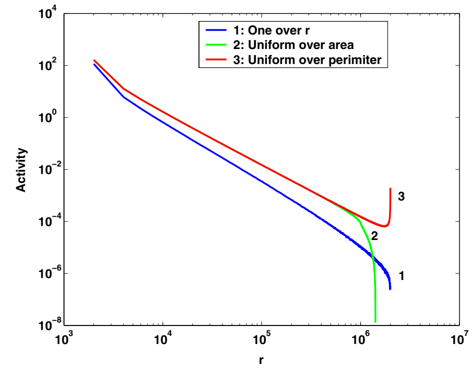
For each of these scenarios we verify scalability of latency and distribution of the load and show that initial condition does not change the results. We have performed Monte Carlo simulations for a grid of $N$ by $N$ sensors for $N = 4 \times 10^6$, using a path loss exponent of $\alpha = 2$. The average power and participation activity are calculated for a total of $M = 10^6$ packet transmissions, for a fixed randomly chosen source and destination for different initial conditions. Figures 2(a) and 2(b) show the power and activity distributions as a function of distance to destination for different initial conditions. Note that for uniform perimeter initial condition the activity is slightly higher near perimeter. The simulation results for $1/r^2$ initial condition follow exact distribution as calculated in previous section. Figures 2 shows the total energy consumption distribution for a 2-D grid network using the proposed algorithm for different initial conditions. The total energy needed to transmit packets with scalable latency is distributed over the network so that each node needs the same total energy as it would need to communicate with its nearest neighbor. The simulation results show that the contraction algorithm is extremely robust to initial conditions, and the mixing time for achieving a stationary distribution is very small. The average latency for different scenarios were 391 for uniform area initial condition, 430 for uniform perimeter initial condition, and 182 for $1/r^2$ initial condition compared to predicted latency between $log(N/4)^2 = 172$ and $log(N/4)^3 = 2259$ from analysis. Similar simulations have been performed for expansion followed by a contraction scenario showing scalability of power and latency[11].

## V. CONCLUSION

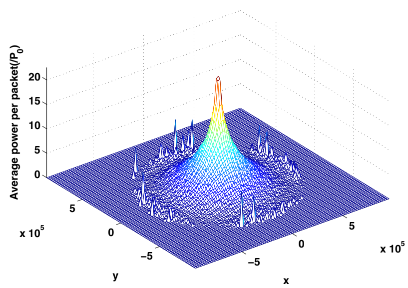
We have proposed a random walk routing scheme on dynamic small worlds for load balanced routing in large-scale sensor networks. Often in sensor network applications, a stream of data is required between two fixed sensors, e.g. an inquiry from a sensor closest to a particular position. Traditional algorithms do not provide sufficient load balancing for these applications, creating hotspots in the network. We address this problem by finding multiple random paths between the source and destination points. All the traditional solutions minimize latency or power requirement separately, but a trade



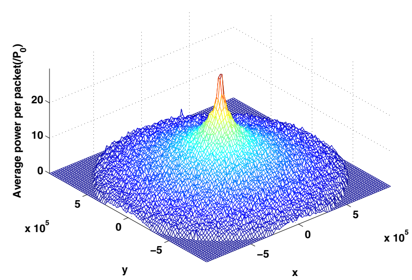

(a) Average power requirement for entire network per packet, as a function of distance to destination. The y-axis is scaled by $P_0$, the Power required for a local transmission of a single packet. The simulation is performed with three different initial conditions.



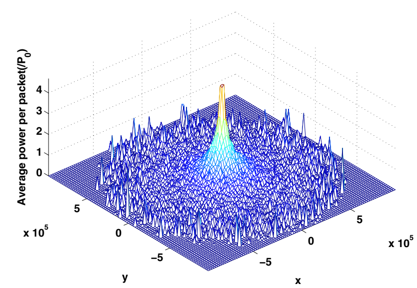

(b) Average activity for entire network per packet, as a function of distance to destination. The simulation is performed with three different initial conditions.




(c) With uniform initial condition over circle radius $N/4$.




(d) With uniform initial condition over perimeter of a circle of radius $N/2$.




(e) With $1/r^2$ distribution over a circle of radius $N/2$.

Fig. 2. Average power requirement for a 2-D grid network using the proposed algorithm. The total energy needed to transmit packets with scalable latency is distributed over the network so that each node needs same total energy as for nearest-neighbor routing. The simulation includes $m = 10^6$ packet transmissions on an $N$ by $N$ grid, where $N = 10^6$.

off has remained as an open problem. While any routing algorithm with scalable latency will not have a scalable total power, proper distribution of the load can lead to a scalable *per node* power requirement. The decentralized algorithm distributes the total power requirement over the entire sensor network, thus minimizing hot spots. For each routing instance, a simple decentralized algorithm will construct a short path of logarithmic number of hops or links. The imposed local construction rules are designed so that nodes farther from the source have less chance of participating in the routing path, while initiating longer connections with higher probabilities. The randomized nature of the routing provides a high degree of fault tolerance and any adversary requires a precise knowledge of the source and the destination sensors to effectively interrupt communication between the pair.

## References

[1] D. Estrin, L. Girod, G. Pottie, and M. Srivastava. Instrumenting the world with wireless sensor networks. In *International Conference on Acoustics, Speech, and Signal Processing*, volume 4, page 2033, 2001.

[2] G. J. Pottie and W. J. Kaiser. Wireless integrated network sensors. *Communications of the ACM*, 43(5):51–58, 2000.

[3] S. D. Servetto, G. Barrenechea. "Constrained Random Walks on Random Graphs: Routing Algorithms for Large Scale Wireless Sensor Networks", Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications, 2002.

[4] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks", In MobiCOM, Boston, MA, August 2000.

[5] D. Ganesan, R. Govindan, S. Shenker and D. Estrin, "Highly Resilient, Energy Efficient Multipath Routing in Wireless Sensor Networks," In Mobile Computing and Communications Review (MC2R '02), 2002

[6] D. Watts and S. Strogatz, "Collective dynamics of small-world networks", Nature, 363:202–204, 1998

[7] J. Kleinberg, "The small world phenomenon: An algorithmic perspective". In Proceedings of the 32nd ACM Symposium on Theory of Computing, May 2000.

[8] W. R. Heinzelman, J. Kulik, and H. Balakrishnan "Adaptive Protocols for Information Dissemination in Wireless Sensor Networks," MobiCom 1999.

[9] M. Lin, K. Marzullo, S. Masini, "Gossip versus deterministic flooding: Low message overhead and high reliability for broadcasting on small networks," UCSD Technical Report TR CS99- 0637. http://citeseer.nj.nec.com/278404.html

[10] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in 33rd Annual Hawaii Int. Conf. on System Sciences, 2000, pp. 30053014.

[11] B A Rezaei, N Sarshar, and V P Roychowdhury,"Load Balanced, Low Latency Routing in Large Scale Sensor Networks,"Preprint.