

# Localization and Tracking of Mobile Attackers Using Hyperbolic Position Bounding in Vehicular Networks

Christine Laurendeau and Michel Barbeau  
School of Computer Science, Carleton University  
1125 Colonel By Drive, Ottawa, ON Canada K1S 5B6  
Tel: 613-520-2600; Fax: 613-520-4334  
E-mail: {claurend,barbeau}@scs.carleton.ca

**Abstract**—A malicious insider in a wireless network may carry out a number of devastating attacks without fear of retribution by authenticating its messages with untraceable credentials. Hyperbolic position bounding (HPB) provides a mechanism to probabilistically estimate the candidate location of an attack message’s originator using received signal strength (RSS) reports, without knowledge of the transmitting power. We specialize the applicability of HPB into the realm of vehicular networks and provide alternate HPB algorithms to improve localization precision and computational efficiency. We extend HPB for tracking the consecutive locations of a mobile attacker. We evaluate the localization and tracking performance of HPB in a vehicular scenario featuring a variable number of receivers and a known navigational layout. We find that HPB can position a transmitting device within stipulated guidelines for emergency services localization accuracy.

**Index Terms**—Insider Attacks, Location Estimation, Vehicular Networks, Mobile Networks, Wireless Security, Wireless Networks

## I. INTRODUCTION

Insider attacks pose an often neglected threat scenario when devising security mechanisms for emerging wireless technologies. For example, traffic safety applications in vehicular networks aim to prevent fatal collisions and preemptively warn drivers of hazards along their path, thus preserving numerous lives. Unmitigated attacks upon these networks stand to severely jeopardize their adoption and limit the scope of their deployment.

The advent of public key cryptography, where a node is authenticated through the possession of a public/private key pair certified by a trust anchor, has addressed the primary threat posed by an outsider without valid credentials. But a vehicular network safeguarded through a Public Key Infrastructure (PKI) is only as secure as the means implemented to protect its member nodes’ private keys. An IEEE standard has been proposed for securing vehicular communications in the Dedicated Short Range Communications Wireless Access in Vehicular Environments (DSRC/WAVE) [12]. This standard advocates the use of digital signatures to secure vehicle safety broadcast messages, with tamper proof devices storing secret keys and cryptographic algorithms in each vehicle. Yet a convincing body of existing literature questions the resistance of such devices to a motivated attacker, especially in technologies

that are relatively inexpensive and readily available [2], [3]. In the absence of strict distribution regulations, for example if tamper proof devices for vehicular nodes are available off the shelf from a neighborhood mechanic, a supply chain exists for experimentation with these devices for the express purpose of extracting private keys. The National Institute of Standards and Technology (NIST) has established a certification process to evaluate the physical resistance of cryptographic processors to tampering, according to four security levels [20]. However, tamper resistance comes at a price. High end cryptographic processors certified at the highest level of tamper resistance are very expensive, for example an IBM 4764 coprocessor costs in excess of 8000 USD [11]. Conversely, lower end tamper evident cryptographic modules, such as smartcards, feature limited mechanisms to prevent cryptographic material disclosure or modification and only provide evidence of tampering after the fact [25]. The European consortium researching solutions in vehicular communications security, SeVeCom, has highlighted the existence of a gap in tamper resistant technology for use in vehicular networks [23]. While low end devices lack physical security measures and suffer from computational performance issues, the cost of high end modules is prohibitive. The gap between the two extremes implies that a custom hardware and software solution is required, otherwise low end devices may be adopted and prove to be a boon for malicious insiders.

Vehicle safety applications necessitate that each network device periodically broadcast position reports, or beacons. A malicious insider generating false beacons whose digital signature is verifiable can cause serious accidents and possibly loss of life. Given the need to locate the transmitter of false beacons, we have put forth a mechanism for attributing a wireless network insider attack to its perpetrator, assuming that a malicious insider is unlikely to use a digital certificate linked to its true identity. Our hyperbolic position bounding (HPB) algorithm probabilistically delimits the candidate location of an attack message’s originating device, assuming neither the cooperation of the attacker nor any knowledge of the effective isotropic radiated power (EIRP) [14]. The received signal strength (RSS) of an attack message at a number of trusted receivers is employed to compute multiple hyperbolic areas whose intersection contains the source of the signal, with a

degree of confidence.

We demonstrate herein that the HPB mechanism is resistant to varying power attacks, which are a known pitfall of RSS-based location estimation schemes. We present three variations of HPB, each with a different algorithm for computing hyperbolic areas. We extend HPB to include a mobile attacker tracking capability. We simulate a vehicular scenario with a variable number of receiving devices, and we evaluate the performance of HPB in both localizing and tracking a transmitting attacker, as a function of the number of receivers. We compare the HPB performance against existing location accuracy standards in related technologies, including the Federal Communications Commission (FCC) guidelines for localizing a wireless handset in an emergency situation.

Section II reviews existing work in vehicular node location determination and tracking. Section III outlines the HPB mechanism in its generic incarnation. Section IV presents three flavours of the HPB algorithm for localizing and tracking a mobile attacker. Section V evaluates the performance of the extended HPB algorithms. Section VI discusses the simulation results obtained. Section VII concludes the paper.

## II. RELATED WORK

A majority of wireless device location estimation schemes presume a number of constraints that are not suitable for security scenarios. We outline these assumptions and compare them against those inherent in our HPB threat model in [13]. For example, a number of publications related to the location determination of vehicular devices focus on self-localization, where a node seeks to learn its own position [4], [21]. Although the measurements and information provided to these schemes are presumed to be trustworthy, this assumption does not hold for finding an attacker invested in avoiding detection and eviction from the network.

Some mechanisms for the localization of a vehicular device by other nodes are based on the principle of location verification, where a candidate position is proposed, and some measured radio signal characteristic, such as time of flight or RSS, is used to confirm the vehicle's location. For example in [10] and [6], Hubaux, Capkun and Luo adapt Brands and Chaum's distance bounding scheme [5] for this purpose. Yet a degree of cooperation is expected on the part of an attacker for supplying a position. Additionally, specialized hardware is necessary to measure time of flight, including nanosecond-precision synchronized clocks and accelerated processors to factor out relatively significant processing delays at the sender and receiver. Xiao *et al.* [26] employ RSS values for location verification, but they assume that all devices, including malicious ones, use the same EIRP. An attacker with access to a variety of radio equipment is unlikely to be constrained in such a manner.

Location verification schemes for detecting false position reports may be beacon-based or sensor-based. Leinmüller *et al.* [16] filter beacon information through a number of plausibility rules. Because each beacon's claimed position is corroborated by multiple nodes, consistent information is assumed to be correct, based on the assumption of an honest

majority of network devices. This presumption leaves the scheme vulnerable to Sybil attacks [8]. If a rogue insider can generate a number of Sybil identities greater than the honest majority, then the attacker can dictate the information corroborated by a *dishonest majority* of virtual nodes. In ensuring a unique geographical location for a signal source, our HPB-based algorithms can detect a disproportionate number of co-located nodes.

Tang *et al.* [24] put forth a sensor-based location verification mechanism where video sensors, such as cameras and RFID readers, can identify license plates. However, cameras perform sub-optimally when visibility is reduced, for example at night or in poor weather conditions. This scheme is supported by PKI-based beacon verification and correlation by an honest majority, which is also vulnerable to insider and Sybil attacks. Another sensor-based mechanism is suggested by Yan *et al.* [27], using radar technology for local security and the propagation of radar readings through beacons on a global scale. Again, an honest majority is assumed to be trustworthy for corroborating the beacons, both locally and globally.

Some existing literature deals explicitly with mobile device tracking, including the RSS-based mechanisms put forth by Mirmotahhary *et al.* [19] and by Zaidi and Mark [28]. These presume a known EIRP and require a large number of transmitted messages so that the signal strength variations can be filtered out.

## III. HYPERBOLIC POSITION BOUNDING

The log-normal shadowing model predicts a radio signal's large scale propagation attenuation, or *path loss*, as it travels over a known transmitter-receiver (T-R) distance [22]. The variations in signal strength experienced in a particular propagation environment, also known as the *signal shadowing*, behave as a log-normal random variable with mean zero and a standard deviation obtained from experimental measurements. In this model, the path loss over T-R distance  $d$  is computed as:

$$L(d) = \bar{L}(d_0) + 10\eta \log\left(\frac{d}{d_0}\right) + X_\sigma \quad (1)$$

where  $d_0$  is a pre-defined reference distance close to the transmitter,  $\bar{L}(d_0)$  is the average path loss at the reference distance, and  $\eta$  is a path loss exponent dependent upon the propagation environment. The signal shadowing is represented by a random variable  $X_\sigma$  with zero mean and standard deviation  $\sigma$ .

In [14], we adapt the log-normal shadowing model to estimate a range of T-R *distance differences*, assuming that the EIRP is unknown. The minimum and maximum bounds of the distance difference range between a transmitter and a receiver pair  $R_i$  and  $R_j$ , with confidence level  $\mathcal{C}$ , are computed as:

$$\Delta d_{ij}^- = \begin{aligned} & (d_0 \times 10^{(\mathcal{P}^- - RSS_i - \bar{L}(d_0) - z\sigma)/10\eta}) \\ & - (d_0 \times 10^{(\mathcal{P}^- - RSS_j - \bar{L}(d_0) + z\sigma)/10\eta}) \end{aligned} \quad (2)$$

$$\Delta d_{ij}^+ = \begin{aligned} & (d_0 \times 10^{(\mathcal{P}^+ - RSS_i - \bar{L}(d_0) + z\sigma)/10\eta}) \\ & - (d_0 \times 10^{(\mathcal{P}^+ - RSS_j - \bar{L}(d_0) - z\sigma)/10\eta}) \end{aligned} \quad (3)$$

where  $RSS_k$  is the RSS measured at receiver  $R_k$ ,  $[\mathcal{P}^-, \mathcal{P}^+]$  represents a dynamically estimated EIRP interval,  $z = \Phi^{-1}(\frac{1+\mathcal{C}}{2})$  represents the normal distribution constant associated with a selected confidence level  $\mathcal{C}$ , and  $[-z\sigma, +z\sigma]$  is the signal shadowing interval associated with this confidence level. The amount of signal shadowing taken into account in the T-R distance difference range is commensurate with the degree of confidence  $\mathcal{C}$ . For example, a confidence level of  $\mathcal{C} = 0.95$ , where  $z = 1.96$ , encompasses a larger proportion of signal shadowing around the mean path loss than  $\mathcal{C} = 0.90$ , where  $z = 1.65$ . A higher confidence level, and thus a larger signal shadowing interval, translates into a wider range of T-R distance differences.

Hyperbolas are computed at the minimum and maximum bounds,  $\Delta d_{ij}^-$  and  $\Delta d_{ij}^+$  respectively, of the distance difference range. The resulting candidate hyperbolic area for the location of a transmitter is situated between the minimum and maximum hyperbolas and contains the transmitter with probability  $\mathcal{C}$ . The intersection of hyperbolic areas computed for multiple receiver pairs bounds the position of a transmitting attacker with an aggregated degree of confidence, as demonstrated in [15].

#### IV. LOCALIZATION AND TRACKING OF MOBILE ATTACKERS

We demonstrate that by dynamically computing an EIRP range, we render the HPB mechanism impervious to varying power attacks. We propose three variations of HPB for computing sets of hyperbolic areas and the resulting candidate areas for the location of a transmitting attacker. We also describe our HPB-based approach for estimating the mobility path of a transmitter in terms of location and direction of travel.

##### A. Mitigating Varying Power Attacks

The use of RSS reports has been criticized as a sub-optimal tool for estimating T-R distances due to their vulnerability to varying power attacks [7]. An attacker that transmits at an EIRP other than the one expected by a receiver can appear to be closer or farther simply by transmitting a stronger or weaker signal. Our HPB-based algorithms are immune to such an exploit, since no fixed EIRP value is expected. Instead, measured RSS values are leveraged to compute a likely EIRP range, as demonstrated in Heuristic 1.

In order for HPB to compute a set of hyperbolic areas between pairs of receivers, a possible EIRP range  $[\mathcal{P}^-, \mathcal{P}^+]$  is dynamically estimated for a given attack message. We use the log-normal shadowing model captured in Equation (1) for this purpose. The path loss  $L(d)$  is replaced with its equivalent, the difference between the EIRP and the  $RSS_k$  measured at a given receiver  $R_k$ .

**Heuristic 1. EIRP Range Computation.** Let  $\mathbb{R}$  be the set of all receivers within range of an attack message. Let  $\tilde{R}_m$  be the maximal RSS receiver and thus be estimated as the closest receiver to the message transmitter, such that  $\tilde{R}_m \in \mathbb{R}$  and  $RSS_m \geq RSS_j$  for all  $R_j \in \mathbb{R}$ . Given that  $EIRP = \bar{L}(d_0) +$

$10\eta \log(d/d_0) + RSS + X_\sigma$  from the log-normal shadowing model, let the EIRP range  $[\mathcal{P}_k^-, \mathcal{P}_k^+]$  at any receiver  $R_k$  be determined, with confidence  $\mathcal{C}$ , as:

$$\mathcal{P}_k^- = \bar{L}(d_0) + 10\eta \log(d_{mk}/d_0) + RSS_k - z\sigma \quad (4)$$

$$\mathcal{P}_k^+ = \bar{L}(d_0) + 10\eta \log(d_{mk}/d_0) + RSS_k + z\sigma \quad (5)$$

where  $d_{mk}$  is the Euclidian distance between  $R_k$  and  $\tilde{R}_m$ , for any  $R_k \in \mathbb{R} \setminus \{\tilde{R}_m\}$ . The estimated EIRP range  $[\mathcal{P}^-, \mathcal{P}^+]$  employed by a transmitter is the smallest probable interval with which every receiver  $R_k \in \mathbb{R} \setminus \{\tilde{R}_m\}$  can reach  $\tilde{R}_m$ . Since  $\mathcal{P}^-$  must be smaller than  $\mathcal{P}^+$ , we iterate through the ordered sets  $\{\mathcal{P}_k^-\}$  and  $\{\mathcal{P}_k^+\}$ , for all  $R_k \in \mathbb{R} \setminus \{\tilde{R}_m\}$ , to find a supremum of EIRP values with minimal shadowing that is lower than an infimum of maximal shadowing EIRP values. Assuming the size of  $\mathbb{R}$  is  $n$ , and thus the size of  $\mathbb{R} \setminus \{\tilde{R}_m\}$  is  $n - 1$ , we compute the estimated EIRP range  $[\mathcal{P}^-, \mathcal{P}^+]$  as follows:

```

i ← n - 1
j ← 1
while i > 0 and j < n
  if P_i^- < P_j^+
    then { P^- ← P_i^-
          P^+ ← P_j^+
          exit
    }
  if i > 1
    then { if P_{i-1}^- < P_j^+
           then { P^- ← P_{i-1}^-
                 P^+ ← P_j^+
                 exit
           }
    }
  i ← i - 1
  j ← j + 1

```

The only case where the pseudo-code above can fail is if every  $\mathcal{P}_i^-$  is greater than every  $\mathcal{P}_j^+$  for all  $1 \leq i, j \leq n - 1$ . This is impossible, since Equations (4) and (5) together indicate that for any  $k$ ,  $\mathcal{P}_k^-$  must be smaller than  $\mathcal{P}_k^+$ .

The log-normal shadowing model indicates that, for a fixed T-R distance, the expected path loss is constant, albeit subject to signal shadowing, regardless of the EIRP used by a transmitter. Any EIRP variation induced by an attacker translates into a corresponding change in the RSS values measured by all receivers within radio range. As a result, an EIRP range computed with Heuristic 1 incorporates an attacker's power variation and is commensurate with the actual EIRP used, as are the measured RSS reports. The values cancel each other out when computing an HPB distance difference range, yielding constant values for the minimum and maximum bounds of this range, independently of EIRP variations.

**Lemma 1. Varying Power Effect.** Let  $\mathbb{R}$  be the set of all receivers within range of an attack message. Let a probable EIRP range  $[\mathcal{P}^-, \mathcal{P}^+]$  for this message be computed as set forth in Heuristic 1. Let the distance difference range  $[\Delta d_{ij}^-, \Delta d_{ij}^+]$  between a transmitter and receiver pair  $R_i, R_j$  be calculated according to Equations (2) and (3). Then any increase (or decrease) in the EIRP of a subsequent message influences a corresponding proportional increase (or decrease)

in RSS reports, effecting no measurable change in the range of distance differences  $[\Delta d_{ij}^-, \Delta d_{ij}^+]$  estimated with a dynamically computed EIRP range.

*Proof:* Let an original EIRP range  $[\mathcal{P}_k^-, \mathcal{P}_k^+]$  computed for all receivers  $R_k \in \mathbb{R}$  yield an estimated global EIRP range  $[\mathcal{P}^-, \mathcal{P}^+]$ . Let a new varying power attack message be transmitted such that the EIRP includes a power increase (or decrease) of  $\Delta\mathcal{P}$ . Then for every  $R_k \in \mathbb{R}$ , the corresponding  $\widehat{RSS}_k$  for the new attack message reflects the same change in value from the original  $RSS_k$ , for  $\widehat{RSS}_k = RSS_k + \Delta\mathcal{P}$ . Given new  $\widehat{RSS}_k$  values for all  $R_k \in \mathbb{R}$ , the resulting EIRP range  $[\widehat{\mathcal{P}}^-, \widehat{\mathcal{P}}^+]$  computed with Heuristic 1 include the same change  $\Delta\mathcal{P}$  over the original range of values  $[\mathcal{P}^-, \mathcal{P}^+]$ :

$$\begin{aligned}\widehat{\mathcal{P}}^- &= \sup\{\widehat{\mathcal{P}}_k^-\} \\ &= \sup\{\widehat{L}(d_0) + 10\eta \log(d_{mk}/d_0) + \widehat{RSS}_k - z\sigma\} \\ &= \sup\{\widehat{L}(d_0) + 10\eta \log(d_{mk}/d_0) + RSS_k + \Delta\mathcal{P} - z\sigma\} \\ &= \sup\{\mathcal{P}_k^- + \Delta\mathcal{P}\} \\ &= \mathcal{P}^- + \Delta\mathcal{P}\end{aligned}$$

Conversely, we see that  $\widehat{\mathcal{P}}^+ = \mathcal{P}^+ + \Delta\mathcal{P}$ .

As a result, the distance difference range  $[\Delta \widehat{d}_{ij}^-, \Delta \widehat{d}_{ij}^+]$  for the new message is equal to the original range  $[\Delta d_{ij}^-, \Delta d_{ij}^+]$ :

$$\begin{aligned}\Delta \widehat{d}_{ij}^- &= (d_0 \times 10^{(\widehat{\mathcal{P}}^- - \widehat{RSS}_i - \widehat{L}(d_0) - z\sigma)/10\eta}) \\ &\quad - (d_0 \times 10^{(\widehat{\mathcal{P}}^- - \widehat{RSS}_j - \widehat{L}(d_0) + z\sigma)/10\eta}) \\ &= (d_0 \times 10^{(\mathcal{P}^- + \Delta\mathcal{P} - RSS_i - \Delta\mathcal{P} - \widehat{L}(d_0) - z\sigma)/10\eta}) \\ &\quad - (d_0 \times 10^{(\mathcal{P}^- + \Delta\mathcal{P} - RSS_j - \Delta\mathcal{P} - \widehat{L}(d_0) + z\sigma)/10\eta}) \\ &= (d_0 \times 10^{(\mathcal{P}^- - RSS_i - \widehat{L}(d_0) - z\sigma)/10\eta}) \\ &\quad - (d_0 \times 10^{(\mathcal{P}^- - RSS_j - \widehat{L}(d_0) + z\sigma)/10\eta}) \\ &= \Delta d_{ij}^-\end{aligned}$$

The same logic can be used to demonstrate that  $\Delta \widehat{d}_{ij}^+ = \Delta d_{ij}^+$ . ■

A varying power attack is thus ineffective against HPB, as the placement of hyperbolic areas remains unchanged.

## B. HPB Algorithm Variations

The HPB mechanism estimates the originating location of a single attack message from a static snapshot of a wireless network topology. Given sufficient computational efficiency, the algorithm executes in near real-time to bound a malicious insider's position at the time of its transmission.

Hyperbolic areas constructed from Equations (2) and (3) are used by HPB to compute a candidate area for the location of a malicious transmitter.

**Definition 1. Hyperbolic Area.** Let  $\mathbb{G}$  be the set of all  $(x, y)$  coordinates in the Euclidian space within radio range of a malicious transmitter. Let  $\mathcal{H}_{ij}^-$  be the hyperbola computed from the minimum bound of the distance difference range between receivers  $R_i$  and  $R_j$  with confidence level  $\mathcal{C}$ , as defined by Equation (2). Let  $\mathcal{H}_{ij}^+$  be the hyperbola computed from the maximum bound of the distance difference range between  $R_i$

and  $R_j$  with the same confidence, as defined by Equation (3). Then we define the hyperbolic area  $\mathcal{A}_{ij}$  as situated between the hyperbolas  $\mathcal{H}_{ij}^-$  and  $\mathcal{H}_{ij}^+$  with confidence level  $\mathcal{C}$ . More formally, if  $\delta(a, b)$  represents the Euclidian distance between any two points  $a$  and  $b$ , then:

$$\mathcal{A}_{ij} = \{p_k : \Delta d_{ij}^- \leq \delta(p_k, R_i) - \delta(p_k, R_j) \leq \Delta d_{ij}^+ \text{ for all } p_k \in \mathbb{G}\}$$

where  $\Delta d_{ij}^-$  and  $\Delta d_{ij}^+$  are defined in Equations (2) and (3).

A set of hyperbolic areas may be computed according to three different algorithms, depending on the set of receiver pairs considered.

**Definition 2. Receiver Pair Set.** Let  $\Omega$  be any set of unique receivers  $R_k$ . Then  $\mathcal{S}^\Omega$  is defined as the exhaustive set of unique, ordered receiver pairs in  $\Omega$ :

$$\mathcal{S}^\Omega = \{\{R_i, R_j\} : R_i, R_j \in \Omega \text{ and } i < j\}$$

where  $s_h \neq s_k$  for all  $s_h, s_k \in \mathcal{S}^\Omega$  where  $h \neq k$ , and  $|\mathcal{S}^\Omega| = \binom{n}{2}$  where  $n = |\Omega|$ .

Our original HPB algorithm employs all possible receiver pairs to compute a candidate area.

**Algorithm 1.  $\mathbf{A}^\alpha$ : All-pairs Algorithm.** The all-pairs algorithm  $\mathbf{A}^\alpha$  computes hyperbolic areas between every possible pair of receivers. Let  $\mathbb{R}$  be the set of all receivers within range of an attack message. Let  $\mathcal{S}^\mathbb{R}$  represent the set of all unique, ordered receiver pairs in  $\mathbb{R}$ , as put forth in Definition 2. Then the set of hyperbolic areas  $\mathbb{H}^\alpha$  between all receiver pairs is stated as follows:

$$\mathbb{H}^\alpha = \{\mathcal{A}_{ij}, \mathcal{A}_{ji} : \mathcal{A}_{ij}, \mathcal{A}_{ji} \text{ are computed as in Def. 1 for every } \{R_i, R_j\} \in \mathcal{S}^\mathbb{R}\}$$

The  $\mathbf{A}^\alpha$  algorithm generates hyperbolic areas for every possible receiver pair, for a total of  $\binom{n}{2}$  pairs given  $n$  receivers, as put forth in Algorithm 1. While this approach works adequately for four receivers, additional receiving devices have the effect of dramatically increasing computation time, as well as reducing the success rate due to the accumulated amount of signal shadowing excluded. The HPB execution time is based on the number of hyperbolic areas computed, which in turn is contingent upon the number of receivers. For  $\mathbf{A}^\alpha$ ,  $n$  receivers locate a transmitter with a complexity of  $\binom{n}{2} = \frac{n \times (n-1)}{2} \approx O(n^2)$ .

An alternate algorithm  $\mathbf{A}^\beta$  groups receivers in sets of size  $r$ , with one intermediate candidate area computed for each set, given all possible receiver pairs in that set. The final candidate area for a transmitter consists of the intersection of all set candidate areas.

**Algorithm 2.  $\mathbf{A}^\beta$ : r-pair Set Algorithm.** The r-pair set algorithm  $\mathbf{A}^\beta$  groups receivers in sets of size  $r$  and computes the intersection of all receiver pairs candidate areas for each set. Let  $\mathbb{R}$  be the set of all receivers within range of an attack message. Let  $\Psi$  represent the disjoint partition of  $(m-1)$  sets of  $r$  receivers, with the  $m^{\text{th}}$  element of  $\Psi$  containing the

remaining receivers:

$$\Psi = \left\{ \begin{array}{l} \psi_k : \psi_k \subseteq \mathbb{R} \text{ for } 1 \leq k \leq m \\ \text{and } |\psi_k| = r \text{ if } k < m \\ \text{and } 2 \leq |\psi_k| \leq r \text{ if } k = m \end{array} \right\}$$

where  $\psi_h \cap \psi_k = \emptyset$  for all  $\psi_h, \psi_k \in \Psi$  with  $h \neq k$ . Let  $\mathcal{S}^{\psi_k}$  represent the set of all unique, ordered receiver pairs in a given set of receivers  $\psi_k \in \Psi$ , as put forth in Definition 2. Then the set of hyperbolic areas  $\mathbb{H}^\beta$  computed for sets of  $r$  receivers is stated as follows:

$$\mathbb{H}^\beta = \left\{ \mathcal{A}_{ij}, \mathcal{A}_{ji} : \mathcal{A}_{ij}, \mathcal{A}_{ji} \text{ are computed as in Def. 1} \right. \\ \left. \text{for every } \{R_i, R_j\} \in \mathcal{S}^{\psi_k} \text{ for all } \psi_k \in \Psi \right\}$$

For the  $\mathbf{A}^\beta$  algorithm, the number of hyperbolic areas depends on the set size  $r$ , as well as the number of receivers  $n$ . Thus  $\mathbf{A}^\beta$  locates a transmitter with a complexity of  $\left(\frac{n}{r} + 1\right) \times \binom{n}{2} \approx O(n)$ . For a small value of  $r$ , for example  $r = 4$ , the execution time is proportional to at most  $\left(\frac{3n}{2} + 6\right)$ .

A third HPB algorithm, the perimeter-pairs variation  $\mathbf{A}^\gamma$ , establishes a rudimentary perimeter around a transmitter's radio range. The logical center of the transmission range is computed as the centroid of all receiver coordinates. This range is partitioned into four quadrants from the center, along two perpendicular axes. Four perimeter receivers are identified as the farthest in each quadrant from the center. Hyperbolic areas are computed between the perimeter receiver pairs and between every remaining non-perimeter receiver at the perimeter receivers in the other three quadrants.

**Algorithm 3.  $\mathbf{A}^\gamma$ : Perimeter-pairs Algorithm.** The perimeter pairs algorithm  $\mathbf{A}^\gamma$  partitions a transmitter's radio range into four quadrants. Four perimeter receivers are determined. Hyperbolic areas are computed between all pairs of perimeter receivers, as well as between every perimeter receiver at the non-perimeter receivers of other quadrants. Let  $\mathbb{R}$  be the set of all receivers within range of an attack message. Let  $R_\chi = (x_c, y_c)$  be the centroid of all  $R_i \in \mathbb{R}$ . Let  $\mathbb{Q}$  be the disjoint set of all receivers  $R_i \in \mathbb{R}$  partitioned into four quadrants from the centroid  $R_\chi$ :

$$\mathbb{Q} = \left\{ \begin{array}{l} Q_k : Q_k = \{R_i : R_i \in \mathbb{R} \text{ and } R_i = (x_i, y_i) \text{ and} \\ x_i \geq x_c \text{ and } y_i \geq y_c \text{ for } k = 1 \\ x_i < x_c \text{ and } y_i \geq y_c \text{ for } k = 2 \\ x_i < x_c \text{ and } y_i < y_c \text{ for } k = 3 \\ x_i \geq x_c \text{ and } y_i < y_c \text{ for } k = 4 \} \end{array} \right\}$$

Let the set  $\mathcal{N}$  of perimeter receivers contain one receiver  $\rho_k$  for each of the four quadrants, such that  $\rho_k$  is the farthest receiver from the centroid  $R_\chi$  in quadrant  $k$ :

$$\mathcal{N} = \left\{ \begin{array}{l} \rho_k : \rho_k = q_i \text{ such that } q_i \in Q_k \text{ and} \\ \delta(q_i, R_\chi) \geq \delta(q_j, R_\chi) \text{ for all } q_j \in Q_k \\ \text{for all } Q_k \in \mathbb{Q} \end{array} \right\}$$

where  $\delta(a, b)$  represents the Euclidian distance between any two points  $a$  and  $b$ . Also let the set of non-perimeter receivers

in a given quadrant be determined as all receivers in that quadrant other than the perimeter receiver:

$$\bar{\mathcal{N}} = \left\{ \bar{\rho}_k : \bar{\rho}_k = \{Q_k \setminus \{\rho_k\}\} \text{ for every } Q_k \in \mathbb{Q} \right\}$$

Let  $\mathcal{S}^\mathcal{N}$  represent the set of all unique, ordered perimeter receiver pairs, as put forth in Definition 2. Then the set of hyperbolic areas  $\mathbb{H}^\gamma$  is stated as follows:

$$\mathbb{H}^\gamma = \left\{ \mathcal{A}_{ij}, \mathcal{A}_{ji} : \mathcal{A}_{ij}, \mathcal{A}_{ji} \text{ are computed as in Def. 1} \right. \\ \left. \text{for every } \{R_i, R_j\} \in \{\mathcal{S}^\mathcal{N} \cup \{\{R_i, R_j\} : \right. \\ R_i = \rho_k \text{ for every } \rho_k \in \mathcal{N} \text{ and } R_j \in \bar{\rho}_m \\ \left. \text{for every } \bar{\rho}_m \in \bar{\mathcal{N}} \text{ where } m \neq k\}\} \right\}$$

For example, Figure 1 illustrates a transmitter  $T$  and a set of receivers. The grid is partitioned into four quadrants from the computed receiver centroid. The set of perimeter receivers, as the farthest receivers from the centroid in each quadrant (I to IV), form a rudimentary bounding area for the location of the transmitter. The  $\mathbf{A}^\gamma$  algorithm computes hyperbolic areas between all pairs of perimeter receivers, in this case between all possible pairs in  $\mathcal{N} = \{R_3, R_4, R_7, R_5\}$ . Additional receiver pairs are formed between the remaining non-perimeter receivers  $\{R_1, R_2, R_6, R_8\}$  and the perimeter receivers of other quadrants. Receiver  $R_6$ , for instance, is situated in quadrant II, so it is included in a receiver pair with each perimeter receiver in  $\{R_3, R_7, R_5\}$ .

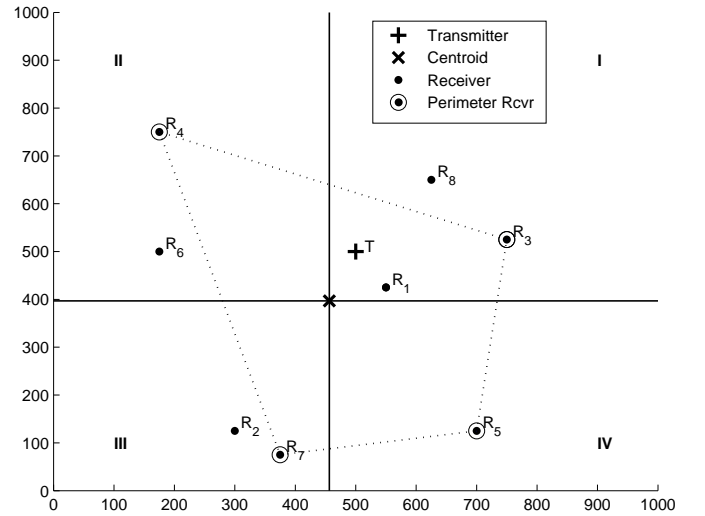


Fig. 1. Example of Perimeter Receivers

In terms of complexity, the  $\mathbf{A}^\gamma$  algorithm is equivalent to  $\mathbf{A}^\beta$ . Given  $n$  receivers and four perimeter receivers such that  $|\mathcal{N}| = 4$ ,  $\mathbf{A}^\gamma$  executes in time  $\binom{4}{2} + 3(n-4) = 3n-6 \approx O(n)$ .

The candidate area for the location of a malicious transmitter is computed as the intersection of a set of hyperbolic areas,  $\mathbb{H}^\alpha$ ,  $\mathbb{H}^\beta$ , or  $\mathbb{H}^\gamma$ , determined according to Algorithm 1, 2 or 3.

**Definition 3. Candidate Area.** Let  $\mathbb{G}$  be the set of all  $(x, y)$  coordinates in our sample Euclidian space. Let  $\mathbb{V} \subseteq \mathbb{G}$  be the subset of all coordinates situated on the road layout of a vehicular scenario. Then the grid candidate area  $GA^\ell$ , where  $\ell \in \{\alpha, \beta, \gamma\}$ , is defined as the subset of grid points in  $\mathbb{G}$

situated in the intersection of every hyperbolic area computed according to Algorithm  $\mathbf{A}^\alpha$ ,  $\mathbf{A}^\beta$  or  $\mathbf{A}^\gamma$ .

$$GA^\ell = \left\{ p_k : p_k \in \mathbb{G} \text{ and } p_k \in \bigcap_{h=1}^{h \leq m} \mathcal{A}_h \in \mathbb{H}^\ell \right\}$$

where  $\ell \in \{\alpha, \beta, \gamma\}$  and  $m = |\mathbb{H}^\ell|$  }

Similarly, the vehicular candidate area  $VA^\ell$ , where  $\ell \in \{\alpha, \beta, \gamma\}$ , is defined as the subset of vehicular layout points in  $\mathbb{V}$  situated in the intersection of every hyperbolic area computed according to Algorithm  $\mathbf{A}^\alpha$ ,  $\mathbf{A}^\beta$  or  $\mathbf{A}^\gamma$ .

$$VA^\ell = \left\{ p_k : p_k \in \mathbb{V} \text{ and } p_k \in \bigcap_{h=1}^{h \leq m} \mathcal{A}_h \in \mathbb{H}^\ell \right\}$$

where  $\ell \in \{\alpha, \beta, \gamma\}$  and  $m = |\mathbb{H}^\ell|$  }

While a candidate area contains a malicious transmitter with probability  $\mathcal{C}$ , the tracking of a mobile device requires that a unique point in Euclidian space be deemed the likeliest position for the attacker. In free space, we can use the centroid of a candidate area, which is calculated as the average of all the  $(x, y)$  coordinates in this area. In a vehicular scenario, we use the road location closest to the candidate area centroid.

**Definition 4. Centroids.** The grid centroid of a given  $GA$ , denoted as  $G_\chi$ , consists of the average  $(x, y)$  coordinates of all points within the  $GA$ .

$$G_\chi = (x_G, y_G), \text{ such that } x_G = \frac{\sum_{i=1}^{|GA|} x_i}{|GA|} \text{ and } y_G = \frac{\sum_{i=1}^{|GA|} y_i}{|GA|},$$

for all  $p_i = (x_i, y_i) \in GA$

The vehicular centroid of a given  $VA$ , represented as  $V_\chi$ , is the closest vehicular point to the average coordinates of all points within the  $VA$ .

$V_\chi = v_k$ , such that  $v_k \in \mathbb{V}$  and  $p_h = (x_V, y_V)$ , where

$$x_V = \frac{\sum_{i=1}^{|VA|} x_i}{|VA|} \text{ and } y_V = \frac{\sum_{i=1}^{|VA|} y_i}{|VA|},$$

for all  $p_i = (x_i, y_i) \in VA$ ,

and  $\delta(p_h, v_k) \leq \delta(p_h, v_j)$ , for all  $v_j \in \mathbb{V}$

### C. Tracking a Mobile Attacker

We extend HPB to approximate the path followed by a mobile attacker, as it continues transmitting. By computing a new candidate area for each attack message received, a malicious node can be tracked using a set of consecutive candidate positions and the direction of travel inferred between these points. We establish a mobility path in our vehicular scenario as a sequence of vehicular layout  $(x, y)$  coordinates over time, along with a mobile transmitter's direction of travel at every point.

**Definition 5.** A mobility path  $\mathbb{P}$  is defined as a set of consecutive coordinates  $p_i = (x_i, y_i)$  and angles of travel  $\theta_i$  over a time interval  $T$ .

$$\mathbb{P} = \left\{ \{p_i, \theta_i\} : p_i = (x_i, y_i) \text{ is the transmitter location at } t_i \in T \text{ and } \theta_i = \text{atan2}(y_i - y_{i-1}, x_i - x_{i-1}) \right\}$$

where  $\text{atan2}$  is an inverse tangent function returning values over the range  $[-\pi, +\pi]$  to take direction into account.<sup>1</sup>

We estimate the mobility path  $\mathbb{P}$  taken by an attacker by executing an HPB algorithm for every attack message received over a time period  $T$ . The vehicular centroids of the resulting candidate areas constitute the estimated attacker positions, and the angle from one estimated point to the next determines the approximated direction of travel.

**Algorithm 4. Mobile Attacker Tracking.** Let  $\mathcal{M}$  be the set of consecutive attack messages received over a time interval. Then the estimated mobility path  $\hat{\mathbb{P}}$  of a transmitter over the message base  $\mathcal{M}$  is computed as follows:

$$\hat{\mathbb{P}} = \left\{ (\hat{p}_i, \hat{\theta}_i) : \hat{p}_i = (\hat{x}_i, \hat{y}_i) = V_{\chi_i} \text{ for } m_i \in \mathcal{M} \text{ and } \hat{\theta}_i = \text{atan2}(\hat{y}_i - \hat{y}_{i-1}, \hat{x}_i - \hat{x}_{i-1}) \right\}$$

For every attack message  $m_i \in \mathcal{M}$ , an estimated transmitter location  $\hat{p}_i$  must be determined. An execution of HPB using the RSS values corresponding to  $m_i$  yields a vehicular candidate area  $VA_i$ , as put forth in Definition 3. The road centroid of  $VA_i$  is computed as  $V_{\chi_i}$ , according to Definition 4. It is by definition the closest point in the vehicular layout to the averaged center of the  $VA_i$ , and thus the natural choice for an estimated value  $\hat{p}_i$  of the true transmitter location  $p_i$ . The direction of travel of a transmitter is stated in Definition 5 as the angle between consecutive positions in Euclidian space. We follow the same logic to compute the estimated direction of travel  $\hat{\theta}_i$  between transmitted messages  $m_{i-1}$  and  $m_i$  as the angle between the corresponding estimated positions  $\hat{p}_{i-1}$  and  $\hat{p}_i$ .

**Example.** Figure 2 depicts an example mobility path of a malicious insider, with consecutive traveled points labeled from 1 to 20. The transmitter broadcasts an attack message at every fourth location, labeled as points 4, 8, 12, 16 and 20.

For each attack message, we execute the  $\mathbf{A}^\gamma$  HPB variation, for confidence level  $\mathcal{C} = 0.95$ , using eight randomly positioned receivers, and a vehicular candidate area  $VA^\gamma$  is computed. The estimated locations and directions of travel are depicted in Figure 3. The initial point's direction of travel cannot be estimated, as there is no previous point from which to ascertain a traveled path. In this example, point 4 is localized at 100 meters from its true position, points 8, 16 and 20 at 25 meters, while point 12 is found in its exact location.

## V. PERFORMANCE EVALUATION

We describe a simulated vehicular scenario to evaluate the localization and tracking performance of the extended HPB mechanisms described in Section IV-B. We compare the success rates of the  $\mathbf{A}^\alpha$ ,  $\mathbf{A}^\beta$  and  $\mathbf{A}^\gamma$  algorithms at estimating

<sup>1</sup>As first defined for the Fortran 77 programming language [1]

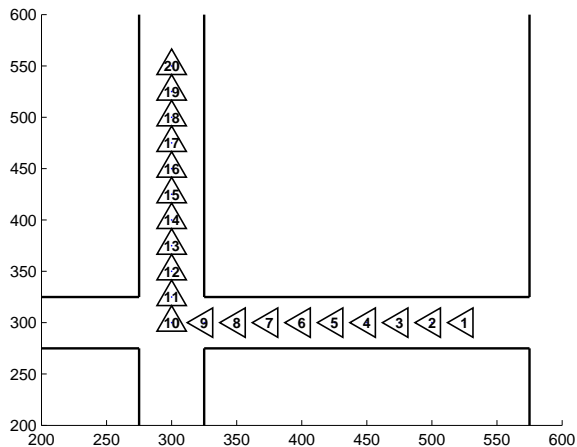


Fig. 2. Example of Attacker Mobility Path

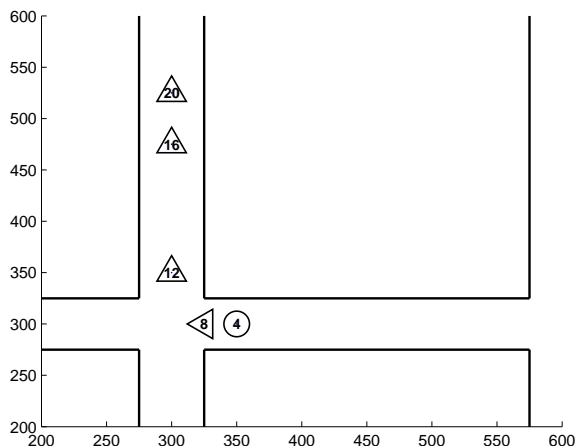


Fig. 3. Example of Mobile Attacker Localization

a malicious transmitter’s location within a candidate area, as well as the relative sizes of the grid and vehicular candidate areas. We model a mobile transmitter’s path through a vehicular scenario and assess the success in tracking it by measuring the distance between the actual and estimated positions, in addition to the difference between the approximated direction of travel and the real one.

#### A. Hyperbolic Position Bounding of Vehicular Devices

Our simulation uses a one square kilometer urban grid, as depicted in Figure 4. We evaluate the all-pairs  $A^\alpha$ , 4-pair set  $A^\beta$  and perimeter-pairs  $A^\gamma$  HPB algorithms with four, eight, 16 and 32 receivers. In each HPB execution, four of the receivers are fixed road-side units (RSUs) stationed at intersections. The remaining receivers are randomly positioned on-board units (OBUs), distributed uniformly on the grid streets. Every HPB execution also sees a transmitter placed at a random road position within the inner square of the simulation grid. We assume that in a sufficiently dense urban

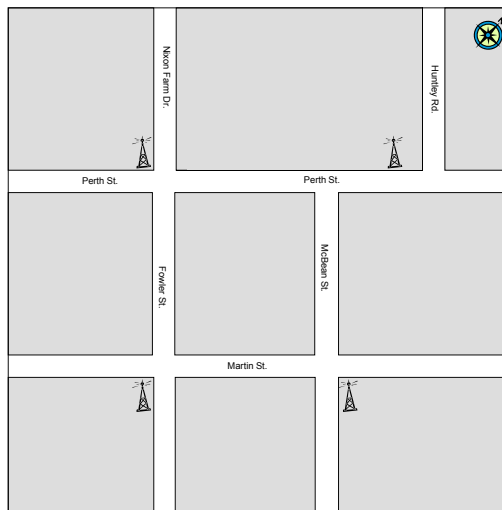


Fig. 4. Urban Scenario – Richmond, Ontario

setting, RSUs are positioned at most intersections. As a result, any transmitter location is geographically surrounded by four RSUs within radio range. For each defined number of receivers and two separate confidence levels  $C \in \{0.95, 0.90\}$ , the HPB algorithms,  $A^\alpha$ ,  $A^\beta$  and  $A^\gamma$ , are executed 1000 times. For every execution, RSS values are generated for each receiver from the log-normal shadowing model. We adopt existing experimental path loss parameter values from large scale measurements gathered at 2.4 GHz by Liechty *et al.* [17], [18]. From  $\eta = 2.76$  and a signal shadowing standard deviation  $\sigma = 5.62$ , we augment the simulated RSS values with an independently generated amount of random shadowing to every receiver in a given HPB execution. Since the EIRP used by a malicious transmitter is unknown, a probable range is computed according to Heuristic 1.

For every HPB execution, whether the  $A^\alpha$ ,  $A^\beta$  or  $A^\gamma$  algorithm is used, we gather three metrics: the success rate in localizing the transmitter within a computed candidate area  $GA$ ; the size of the unconstrained candidate area  $GA$  as a percentage of the one square kilometer grid; and the size of the candidate area restricted to the vehicular layout  $VA$  as a percentage of the grid. The success rate and candidate area size results we obtain are deemed 90% accurate within a 2% and 0.8% confidence interval respectively.

The comparative success rates of the  $A^\alpha$ ,  $A^\beta$  and  $A^\gamma$  approaches are illustrated in Figure 5, for confidence level  $C = 0.95$ . While  $A^\gamma$  exhibits the best localization success rate, every algorithm sees its performance degrade as more receivers are included. With four receivers for example, all three variations successfully localize a transmitter 94-95% of the time. However with 32 receivers,  $A^\gamma$  succeeds in 79% of the cases, while  $A^\beta$  and  $A^\alpha$  do so in 71% and 50% of executions. Given that each receiver pair takes into account an amount of signal shadowing based on the confidence level  $C$ , it also probabilistically ignores a portion  $(1 - C)$  of the shadowing. As more receivers and thus more receiver pairs are added, the error due to excluded shadowing accumulates. The results obtained for confidence level  $C = 0.90$  follow the

same trend, although the success rates are slightly lower.

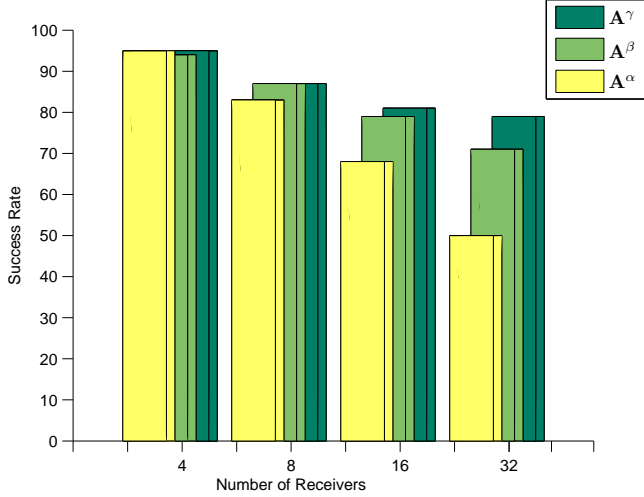


Fig. 5. Success Rate for  $C = 0.95$

Figures 6 and 7 show the grid and vehicular candidate area sizes associated with our simulation scenario, as computed with algorithms  $A^\alpha$ ,  $A^\beta$  and  $A^\gamma$ , for confidence level  $C = 0.95$ . The size of the grid candidate area  $GA$  corresponds to 21% of the simulation grid, with four receivers, for both  $A^\beta$  and  $A^\alpha$ , while  $A^\gamma$  narrows the area to only 7%. In fact, the  $A^\gamma$  approach exhibits a  $GA$  size that is independent of the number of receivers. Yet for  $A^\beta$  and  $A^\alpha$ , the  $GA$  size is noticeably lower with additional receivers. This finding reflects the use of perimeter receivers with  $A^\gamma$ . These specialized receivers serve to restrict the  $GA$  to a particular portion of the simulation grid, even with few receivers. However, this variation does not fully exploit the presence of additional receiving devices, as these only support the  $GA$  determined by the perimeter receivers. The size of the vehicular candidate area  $VA$  follows the same trend, with a near constant size of 0.64% to 1% of the grid for  $A^\gamma$ , corresponding to a localization granularity within an area less than  $100\text{ m} \times 100\text{ m}$ , assuming the transmitter is aboard a vehicle traveling on a road. The  $A^\beta$  and  $A^\alpha$  algorithms compute vehicular candidate area sizes that decrease as more receivers are taken into account, with  $A^\alpha$  yielding the best localization granularity. But even with four receivers,  $A^\beta$  and  $A^\alpha$  localize a transmitter within a vehicular layout area of 1.6% of the grid, or  $125\text{ m} \times 125\text{ m}$ .

Generally, both the  $GA$  and  $VA$  sizes decrease as the number of receivers increase, since additional hyperbolic areas pose a higher number of constraints on a candidate area, thus decreasing its extent. We see in Figures 6 and 7 that  $A^\beta$  consistently yields larger candidate areas than  $A^\alpha$  for the same reason, as  $A^\alpha$  generates a significantly greater number of hyperbolic areas. For example, while  $A^\alpha$  computes an average  $GA^\alpha$  of 10% and 3% of the simulation grid with eight and 16 receivers,  $A^\beta$  yields areas of 15% and 9% respectively. By contrast,  $A^\gamma$  yields a  $GA$  size of 5-6%, but its reliability is greater, as demonstrated by the higher success rates achieved. The nearly constant 5%  $GA$  size computed with  $A^\gamma$  has an average success rate of 81% for 16 receivers, while the 9%  $GA$  generated by  $A^\beta$  is 79% reliable and the 3%  $GA$  obtained

with  $A^\alpha$  features a dismal 68% success rate. Indeed, Figures 5 and 6 taken together indicate that smaller candidate areas provide increased granularity at the cost of lower success rates, and thus decreased reliability. This phenomenon is consistent with the intuitive expectation that a smaller area is less likely to contain the transmitter.

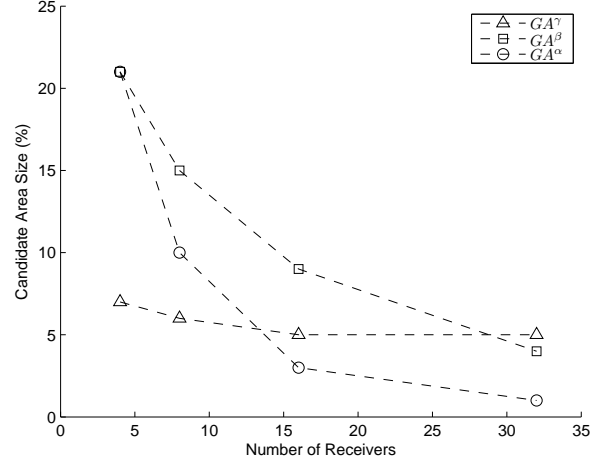


Fig. 6. Grid Candidate Area Size for  $C = 0.95$

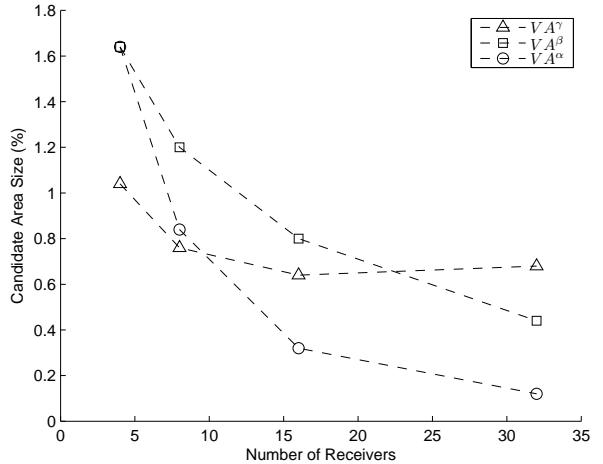


Fig. 7. Vehicular Candidate Area Size for  $C = 0.95$

## B. Tracking a Vehicular Device

We generate 1000 attacker mobility paths  $\mathbb{P}$ , as stipulated in Definition 5, of 20 consecutive points evenly spaced at every 25 meters. Each path begins at a random start location along the central square of the simulation grid depicted in Figure 4. We keep the simulated transmitter location within the area covered by four fixed RSUs, presuming that an infinite grid features at least four RSUs within radio range of a transmitter. The direction of travel for the start location is determined randomly. Each subsequent point in the mobile path is contiguous to the previous point, along the direction of travel. Upon reaching an intersection in the simulation



grid, a direction of travel is chosen randomly among the ones available from the current position, excluding the reverse direction.

The  $\mathbf{A}^\alpha$ ,  $\mathbf{A}^\beta$  and  $\mathbf{A}^\gamma$  algorithms are executed at every fourth point  $p_i$  of each mobility path  $\mathbb{P}$ , corresponding to a transmitted attack signal at every 100 meters. The algorithms are executed for confidence levels  $\mathcal{C} \in \{0.95, 0.90\}$ , with each of four, eight, 16 and 32 receivers. In every case, the receivers consist of four static RSUs, and the remainder are OBUs randomly placed at any point on the simulated roads.

For each execution of  $\mathbf{A}^\alpha$ ,  $\mathbf{A}^\beta$  and  $\mathbf{A}^\gamma$ , a vehicular candidate area  $VA$  is computed, and its centroid  $V\chi$  is taken as the probable location of the transmitter, as described in Algorithm 4. Two metrics are aggregated over the executions: the root mean square *location error*, as the distance in meters between the actual transmitter location  $p_i$  and its estimated position  $\hat{p}_i = V\chi_i$ ; and the root mean square *angle error* between the angle of travel  $\theta_i$  for each consecutive actual transmitter location and the angle  $\hat{\theta}_i$  computed for the approximated locations.

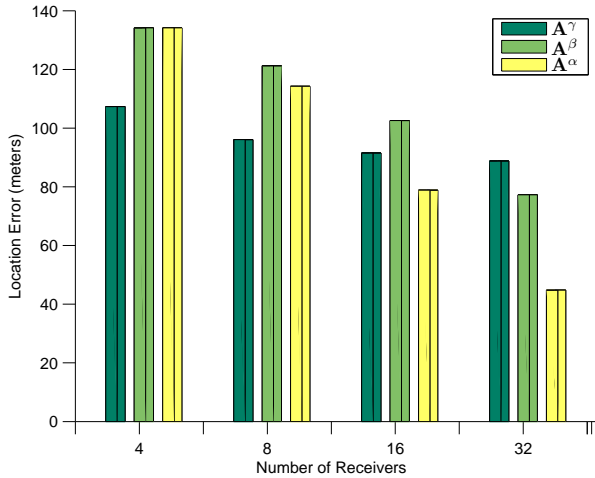


Fig. 8. Location Error for  $\mathcal{C} = 0.95$

The location error for the  $\mathbf{A}^\alpha$ ,  $\mathbf{A}^\beta$  and  $\mathbf{A}^\gamma$  algorithms, given confidence level  $\mathcal{C} = 0.95$ , is illustrated in Figure 8. As expected, the smaller  $VA$  sizes achieved with a greater number of receivers for  $\mathbf{A}^\alpha$  and  $\mathbf{A}^\beta$  correspond to a more precise transmitter localization. The location error associated with the  $\mathbf{A}^\alpha$  algorithm is smaller, compared to  $\mathbf{A}^\beta$ , for the same reason. Correspondingly, the nearly constant  $VA$  size obtained with  $\mathbf{A}^\gamma$  yields a similar result for the location error. For instance with confidence level  $\mathcal{C} = 0.95$ , eight and 16 receivers produce a location error of 114 and 79 meters respectively with  $\mathbf{A}^\alpha$ , but of 121 and 102 meters with  $\mathbf{A}^\beta$ . The location error with  $\mathbf{A}^\gamma$  is once more nearly constant, at 96 and 91 meters. The use of all receiver pairs to compute a  $VA$  with  $\mathbf{A}^\alpha$  allows for localization that is up to 40-50% more precise than grouping the receivers in sets of four or relying on perimeter receivers when 16 or 32 receiving devices are present. Despite its granular localization performance, the  $\mathbf{A}^\alpha$  approach works best with large numbers of receivers,

which may not consistently be realistic in a practical setting. Another important disadvantage of the  $\mathbf{A}^\alpha$  approach lies in its large complexity of  $O(n^2)$  for  $n$  receivers, when compared to  $\mathbf{A}^\beta$  and  $\mathbf{A}^\gamma$  with a complexity of  $O(n)$ , as discussed in Section IV-B.

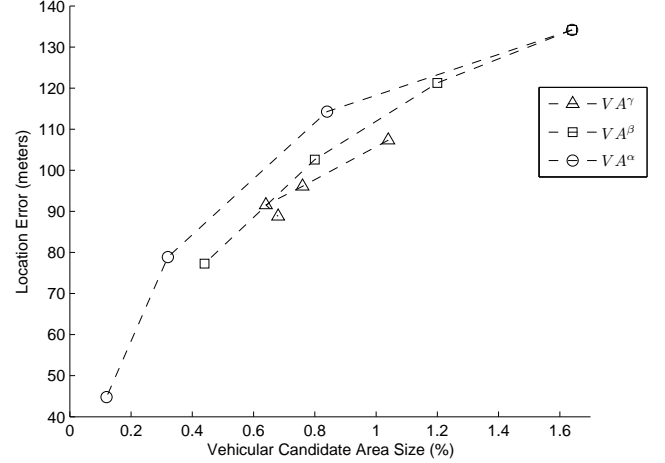


Fig. 9. Location Error for Vehicular Candidate Area Size

Figure 9 plots the root mean square location error in terms of  $VA$  size for the three algorithms. While  $\mathbf{A}^\alpha$  and  $\mathbf{A}^\beta$  yield smaller  $VAs$  for a large number of receivers, the  $VAs$  computed with  $\mathbf{A}^\gamma$  offer more precise localization with respect to their size. For example, a 0.7%  $VA$  size obtained with  $\mathbf{A}^\gamma$  features a 96 meter location error, while a similar size  $VA$  computed with  $\mathbf{A}^\beta$  and  $\mathbf{A}^\alpha$  generate a 102 and 114 meter location error, respectively.

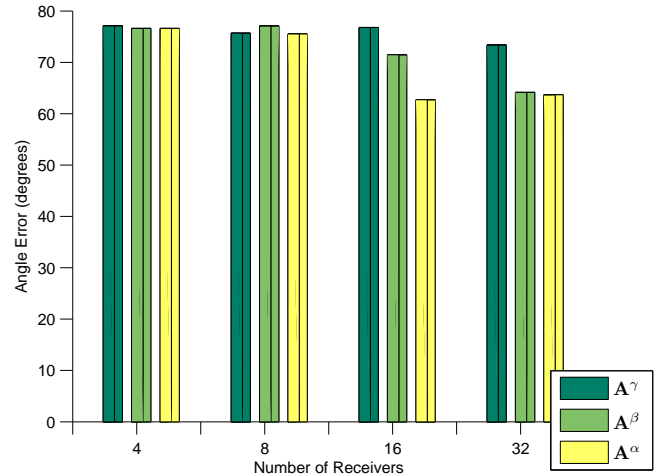


Fig. 10. Direction of Travel Angle Error for  $\mathcal{C} = 0.95$

The error in estimating the direction of travel exhibits little variation in terms of number of receivers and choice of HPB algorithm, as shown in Figure 10. With eight and 16 receivers, for confidence level  $\mathcal{C} = 0.95$ ,  $\mathbf{A}^\beta$  approximates the angle of travel between two consecutive points within  $77^\circ$  and  $71^\circ$

respectively, whereas  $\mathbf{A}^\alpha$  estimates it within  $76^\circ$  and  $63^\circ$ .  $\mathbf{A}^\gamma$  exhibits a slightly higher direction error at  $76^\circ$  and  $77^\circ$ . It should be noted that for all three algorithms, for all numbers of receivers, the range of angle errors only spans  $14^\circ$ . So while the granularity of localization is contingent upon the HPB methodology used and the number of receivers, the three variations perform similarly in estimating the general direction of travel.

## VI. DISCUSSION

The location error results of Figure 8 shed an interesting light on the HPB success rates discussed in Section V-A. For example in the presence of 32 receivers, for confidence level  $\mathcal{C} = 0.95$ , only 50% of  $\mathbf{A}^\alpha$  executions yield a candidate area containing a malicious transmitter, as shown in Figure 5. Yet the same scenario localizes a transmitter with a root mean square location error of 45 meters of its true location, whether it lies within the corresponding candidate area or not. This indicates that while a candidate area may be computed in the wrong position, it is in fact rarely far from the correct transmitter location. This may be a result of our strict definition of a successful execution, where only a candidate area in the intersection of all hyperbolic areas is considered. We have observed in our simulations that a candidate area may be erroneous solely because of a single misplaced hyperbolic area, which results in either a wrong location or an empty candidate area. In our simulations tracking a mobile attacker, we notice that while  $\mathbf{A}^\gamma$  and  $\mathbf{A}^\beta$  generate an empty  $VA$  for 10% and 14% of executions,  $\mathbf{A}^\alpha$  does so in 31% of the cases. This phenomenon is likely due to the greater number of hyperbolic areas generated with the  $\mathbf{A}^\alpha$  approach and the subsequent greater likelihood of erroneously situated hyperbolic areas. While the success rates depicted in Figure 5 omit the executions yielding empty candidate areas as inconclusive, future work includes devising a heuristic to re-compute a set of hyperbolic areas in the case where their common intersection is empty.

In comparing the location accuracy of HPB with related technologies, we find that, for example, differential GPS devices can achieve less than 10 meter accuracy. However, this technology is better suited to self-localization efforts relying on a device's assistance, and cannot be depended upon for the position estimation of a non-cooperative adversary. The FCC has set forth regulations for the network-based localization of wireless handsets in emergency 911 call situations. Service providers are expected to locate a calling device within 100 meters 67% of the time and within 300 meters in 95% of cases [9]. In the minimalist case involving four receivers, the HPB perimeter-pairs variation  $\mathbf{A}^\gamma$  localizes a transmitting device with a root mean square location error of 107 meters. This translates into a location accuracy of 210 meters in 95% of cases and of 104 meters in 67% of executions. While the former case is fully within FCC guidelines, the latter is very close. With a larger number of receivers, for example eight receiving devices,  $\mathbf{A}^\gamma$  yields an accuracy of 188 meters 95% of the time and of 93 meters in 67% of cases. Although HPB is designed for the location estimation of a malicious insider,

its use may be extended to additional applications such as 911 call origin localization, given that its performance closely matches the FCC requirements for emergency services.

## VII. CONCLUSION

We extend a hyperbolic position bounding (HPB) mechanism to localize the originator of an attack signal within a vehicular network. We devise two additional approaches to compute hyperbolic areas between pairs of trusted receivers by grouping them in sets and establishing perimeter receivers. We demonstrate that due to the dynamic computation of a probable EIRP range utilized by an attacker, our HPB algorithms are impervious to varying power attacks. We extend the HPB algorithms to track the location of a mobile attacker transmitting along a traveled path.

The performance of all three HPB variations is evaluated in a vehicular scenario. We find that the grouped receivers method yields a localization success rate up to 11% higher for a 6% increase in candidate area size over the all-pairs approach. We also observe that the perimeter-pairs algorithm provides a more constant candidate area size, independently of the number of receivers, for a success rate up to 13% higher for a 2% increase in candidate area size over the all-pairs variation. We conclude that the original HPB mechanism using all pairs of receivers produces a smaller localization error than the other two approaches, when a large number of receiving devices are available. We observe that for a confidence level of 95%, the former approach localizes a mobile transmitter with a granularity as low as 45 meters, up to 40-50% more precisely than the grouped receivers and perimeter-pairs methods. However, the computational complexity of the all-pairs variation is significantly greater, and its performance with fewer receivers is less granular than the perimeter-pairs method. Of the two approaches with complexity  $O(n)$ , the perimeter-pairs method yields a success rate up to 8% higher for consistently smaller candidate area sizes, location and direction errors.

In a vehicular scenario, we achieve a root mean square location error of 107 meters with four receivers and of 96 meters with eight receiving devices. This granularity is sufficient to satisfy the FCC-mandated location accuracy regulations for emergency 911 services. Our HPB mechanism may therefore be adaptable to a wide range of applications involving network-based device localization assuming neither target node cooperation nor knowledge of the EIRP.

We have demonstrated the suitability of the hyperbolic position bounding mechanism for estimating the candidate location of a vehicular network malicious insider and for tracking such a device as it moves throughout the network. Future research is required to assess the applicability of the HPB localization and tracking mechanisms in additional types of wireless and mobile technologies, including wireless access networks such as WiMAX/802.16.

## ACKNOWLEDGMENT

The authors gratefully acknowledge the financial support received for this research from the Natural Sciences and Engineering Research Council of Canada (NSERC) and the

## REFERENCES

- [1] American National Standards Institute. Programming Language FORTRAN. ANSI Standard X3.9-1978, 1978.
- [2] R. Anderson, M. Bond, J. Clulow, and S. Skorobogatov. Cryptographic Processors – A Survey. *Proceedings of the IEEE*, 94(2):357–369, February 2006.
- [3] R. Anderson and M. Kuhn. Tamper Resistance – A Cautionary Note. In *Proceedings of the 2nd USENIX Workshop on Electronic Commerce*, pages 1–11, November 1996.
- [4] A. Boukerche, H. A. Oliveira, E. F. Nakamura, and A. A. Loureiro. Vehicular Ad Hoc Networks: A New Challenge for Localization-Based Systems. *Computer Communications*, 31(12):2838–2849, July 2008.
- [5] S. Brands and D. Chaum. Distance-Bounding Protocols. In *Advances in Cryptology: Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques*, volume 765 of *Lecture Notes in Computer Science*, pages 344–359. Springer Berlin / Heidelberg, 1994.
- [6] S. Capkun and J.-P. Hubaux. Secure Positioning in Wireless Networks. *IEEE Journal on Selected Areas in Communications*, 24(2):221–232, February 2006.
- [7] Y. Chen, K. Kleisouris, X. Li, W. Trappe, and R. P. Martin. The Robustness of Localization Algorithms to Signal Strength Attacks: A Comparative Study. In *Distributed Computing in Sensor Systems: Proceedings of the 2nd IEEE International Conference (DCOSS)*, volume 4026 of *Lecture Notes in Computer Science*, pages 546–563. Springer Berlin / Heidelberg, 2006.
- [8] J. R. Douceur. The Sybil Attack. In *Peer-to-Peer Systems: Proceedings of the First International Workshop (IPTPS)*, volume 2429 of *Lecture Notes in Computer Science*, pages 251–260. Springer Berlin / Heidelberg, 2002.
- [9] Federal Communications Commission. 911 Service. FCC Code of Federal Regulations, Title 47, Part 20, Section 20.18, October 2007.
- [10] J.-P. Hubaux, S. Capkun, and J. Luo. The Security and Privacy of Smart Vehicles. *IEEE Security and Privacy*, 2(3):49–55, May–June 2004.
- [11] IBM. IBM 4764 PCI-X Cryptographic Coprocessor. [Online] <http://www.ibm.com/>.
- [12] IEEE Intelligent Transportation Systems Committee. IEEE Trial-Use Standard for Wireless Access in Vehicular Environments - Security Services for Applications and Management Messages. IEEE Std 1609.2-2006, July 2006.
- [13] C. Laurendeau and M. Barbeau. Hyperbolic Location Estimation of Malicious Nodes in Mobile WiFi/802.11 Networks. In *Proceedings of the 2nd IEEE LCN Workshop On User Mobility and Vehicular Networks (ON-MOVE)*, October 2008.
- [14] C. Laurendeau and M. Barbeau. Insider Attack Attribution Using Signal Strength Based Hyperbolic Location Estimation. *Security and Communication Networks*, 1(4):337–349, July–August 2008.
- [15] C. Laurendeau and M. Barbeau. Probabilistic Evidence Aggregation for Malicious Node Position Bounding in Wireless Networks. To appear in *Journal of Networks*, accepted October 2008.
- [16] T. Leinmüller, E. Schoch, and F. Kargl. Position Verification Approaches for Vehicular Ad Hoc Networks. *IEEE Wireless Communications*, 13(5):16–21, October 2006.
- [17] L. C. Liechty. Path Loss Measurements and Model Analysis of a 2.4 GHz Wireless Network in an Outdoor Environment. Master's thesis, Georgia Institute of Technology, August 2007.
- [18] L. C. Liechty, E. Reifsnider, and G. Durgin. Developing the Best 2.4 GHz Propagation Model from Active Network Measurements. In *Proceedings of the 66th IEEE Vehicular Technology Conference*, pages 894–896, September 2007.
- [19] N. Mirmotahhary, A. Kohansal, H. Zamiri-Jafarian, and M. Mirsalehi. Discrete Mobile User Tracking Algorithm via Velocity Estimation for Microcellular Urban Environment. In *Proceedings of the IEEE Vehicular Technology Conference (VTC Spring)*, pages 2631–2635, May 2008.
- [20] National Institute of Standards and Technology. Security Requirements for Cryptographic Modules. Federal Information Processing Standards Publication 140-2, NIST, May 2001.
- [21] R. Parker and S. Valaee. Vehicular Node Localization Using Received-Signal-Strength Indicator. *IEEE Transactions on Vehicular Technology*, 56(6):3371–3380, November 2007.
- [22] T. S. Rappaport. *Wireless Communications: Principles and Practice*. Prentice-Hall, New Jersey, second edition, 2002.
- [23] SeVeCom. Security Architecture and Mechanisms for V2V/V2I. Deliverable Report D2.1, Secure Vehicle Communication, August 2007. Antonio Kung (editor).
- [24] L. Tang, X. Hong, and P. G. Bradford. Privacy-preserving Secure Relative Localization in Vehicular Networks. *Security and Communication Networks*, 1(3):195–204, May–June 2008.
- [25] D. E. Williams. A Concept for Universal Identification. White paper, SANS Institute, December 2001.
- [26] B. Xiao, B. Yu, and C. Gao. Detection and Localization of Sybil Nodes in VANETs. In *Proceedings of the 2006 Workshop on Dependability Issues in Wireless Ad Hoc Networks and Sensor Networks (DIWANS)*, pages 1–8, September 2006.
- [27] G. Yan, S. Olariu, and M. C. Weigle. Providing VANET Security Through Active Position Detection. *Computer Communications*, 31(12):2883–2897, July 2008.
- [28] Z. R. Zaidi and B. L. Mark. Real-Time Mobility Tracking Algorithms for Cellular Networks Based on Kalman Filtering. *IEEE Transactions on Mobile Computing*, 4(2):195–208, March–April 2005.

**Christine Laurendeau** received her B.Sc. and M.Sc. in Computer Science from the University of Ottawa in 1989 and 1992. She is currently a Ph.D. candidate at Carleton University's School of Computer Science in Ottawa, Canada. She focuses her research efforts on wireless communications security, specifically the security of wireless access networks and vehicular communications.

**Michel Barbeau** is a professor of Computer Science. He got a Bachelor, a Master's and a Ph.D., in Computer Science, from Université de Sherbrooke, Canada ('85), for undergraduate studies, and Université de Montréal, Canada ('87 & '91), for graduate studies. From '91 to '99, he was a professor at Université de Sherbrooke. During the '98-'99 academic year, he was a visiting researcher at the University of Aizu, Japan. Since 2000, he works at Carleton University, Canada. The topic of wireless communications has been his main research interest. He puts his efforts more particularly on the topics of wireless security, vehicular communications and wireless access network management. He also conducts work on small satellite software and AI for computer games.