Duet: Estimating User Position and Identity in Smart Homes Using Intermittent and Incomplete RF-Data

DEEPAK VASISHT, ANUBHAV JAIN, CHEN-YU HSU, ZACHARY KABELAC, DINA KATABI,
Massachusetts Institute of Technology

Although past work on RF-based indoor localization has delivered important advances, it typically makes assumptions that hinder its adoption in smart home applications. Most localization systems assume that users carry their phones on them at home, an assumption that has been proven highly inaccurate in past measurements. The few localization systems that do not require the user to carry a device on her, cannot tell the identity of the person; yet identification is essential to most smart home applications. This paper focuses on addressing these issues so that smart homes can benefit from recent advances in indoor localization.

We introduce Duet, a multi-modal system that takes as input measurements from both device-based and device-free localization. Duet introduces a new framework that combines probabilistic inference with first order logic to reason about the users' most likely locations and identities in light of the measurements. We implement Duet and compare it with a baseline that uses state-of-art WiFi-based localization. The results of two weeks of monitoring in two smart environments show that Duet accurately localizes and identifies the users for 94% and 96% of the time in the two places. In contrast, the baseline is accurate 17% and 42% respectively.

CCS Concepts: • **Human-centered computing** → *Ubiquitous and mobile computing systems and tools*;

Additional Key Words and Phrases: RF-based Indoor Positioning, Multi-modal Sensor System

ACM Reference Format:
Deepak Vasisht, Anubhav Jain, Chen-yu Hsu, Zachary Kabelac, Dina Katabi. 2018. Duet: Estimating User Position and Identity in Smart Homes Using Intermittent and Incomplete RF-Data. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 2, Article 84 (June 2018), 21 pages. https://doi.org/10.1145/3214287

1 INTRODUCTION

The vision of the smart home has captivated academia and industry since the 90's, when Bill Gates revealed the design of his new home where the lighting, music, and art collection change according to the person's taste and liking. Over the years, we have made advances towards that vision. Today, users can control heating, lighting, and appliances from their cell phones. However, current smart homes rely on explicit input from the user. We have not yet realized the complete vision of a smart home that tracks its occupants, monitors their habits, and adapts to their liking organically.

RF-based indoor localization can play a central role in realizing this vision of the smart home, by enabling the home to be aware of its occupants' locations and react to their presence. It can allow a smart home to track the user as she sits on the TV couch and tune to her favorite channel, alert the parents if, during their absence, the

Correspondence to: Deepak Vasisht <deepakv@mit.edu>.
Author's address: Deepak Vasisht, Anubhav Jain, Chen-yu Hsu, Zachary Kabelac, Dina Katabi,
Massachusetts Institute of Technology.

2474-9567/2018/6-ART84 $15.00
https://doi.org/10.1145/3214287

babysitter enters their bedroom instead of the baby's room, and turn the alarm clock off when the user actually leaves the bed. Further, there is much past work on RF-based localization [17, 18, 34, 40] and many of these systems achieve sub-meter accuracy.

The problem, however, is that RF-based localization systems typically rely on two assumptions neither of which is valid in the home scenario. First, a majority of RF-based localization systems assume that the user can be localized using signals emitted by their personal device, e.g., a cellphone. Yet, at home, it is typical for people to leave their phones behind, either on a counter or in their bags, or connected to a charger, as they go about their activities. In particular, a recent study [10, 26] has shown that people spend 54% of the time away from their cellphones at home.

Second, RF-based localization systems assume that the RF signal along the direct line-of-sight to the user is not completely blocked. Although recent systems ([17, 34, 40]) can successfully localize in the presence of multipath, they fundamentally require the presence of the signal along the direct path. Yet, typical home structures may completely block the signal along the direct line of sight from the user to the receiver. TV screens, mirrors, HVAC are all metallic bodies that block RF signals. Thick structures such as pillars and in-wall plumbing can have the same effect. The blocking effect is further exacerbated by that, homes tend to have a single access point, unlike the enterprise where each location is covered by several access points. Once that single line-of-sight is blocked, there is no way to localize the user.

Some past work [3, 4, 35] has demonstrated device-free-localization, which can track a person using the RF signal reflected off her body. Such systems eliminate the need for having the user carry her phone, but they introduce their own challenges. First, they exacerbate the blocking problem. Since they rely purely on weak, low-power RF reflections off people's bodies, they are more likely to be blocked by home structures like TV, mirrors, HVAC, etc. Further, even when not blocked, they have a limited reach due to their lower power (100x lower power than Wi-Fi [3]). Second, device-free localization lacks the notion of identity. These systems are unable to identify who is sitting on the TV couch and who is cooking in the kitchen; they may end up tuning the TV to the wrong channel or making the temperature too high for the person in the kitchen.

This paper introduces Duet, a location tracking system that is customized for the smart home. In contrast to past work, which has focused on new signal processing algorithms to infer physical location from radio signals, Duet focuses on addressing the above practical problems so that smart home applications can benefit from existing indoor localization algorithms.

Duet is a multi-modal system –i.e., a Duet AP has both a Wi-Fi radio for device-based localization and FMCW radio for device-free localization. Duet leverages device-free localization to track people's movements in the home even when they carry no personal devices on them. It then uses device-based localization to tag users' trajectories with their identities based on the users' intermittent interactions with their phones.

Duet's design has the following components which together deliver perpetual localization and identification:

- **Identification:** At first blush, it might seem that one can simply localize people using device-free localization, localize devices using Wi-Fi, and then identify each person by matching him with the nearest personal device. This however does not work. While the average accuracy of RF-localization is sub-meter, the 90th percentile error can be several meters. Further, localization error are likely higher in the home environment where, unlike the enterprise, one cannot cover each location by 3 to 5 APs. Such errors can easily position a device closer to the wrong person. Furthermore, being close to a cellphone does not mean that the person is using the device, and hence provides no information about the identity of the person. For identification, Duet considers only scenarios where the phone and user are moving and their movements are correlated. In this case, Duet uses the distance between the device and the person to infer a probability over them being matched. Whether this probability translates into the user being matched with the device is subject to other dependencies as explained below.

- **Blockage/Coverage:** Duet aims to locate a user even when she is outside the radio coverage area. To do so, Duet leverages that smart home applications care about the user's symbolic location rather than her absolute location. For example, they want to learn whether the user in the bedroom or kitchen, whether she sat on the TV couch or at her desk, etc. Duet extracts the entry and exit boundaries to such symbolic spaces –e.g., the doors to each room, the bedside, the side of the couch for sitting. It then leverages this information to reason about entry and exit events to individual spaces. Even if parts of these spaces may be occluded or out of coverage areas, Duet can still identify the symbolic location of the user based on the entry and exit events. To encode this intuition, we leverage a Hidden Markov Model (HMM) for each symbolic space to reason about entry and exit events in a probabilistic manner. We note that entering and leaving at the specified boundaries is not a hard constraint –e.g. if a user enters the bed area at the bed foot, Duet corrects itself once it observes signals from the user in the bed.
- **Dependencies:** We recognize that there are dependencies between identity and location because the same person cannot be in two places at the same time. There are also dependencies between different symbolic locations. Say Duet sees two people as they enter their empty living room but later localizes two people on the couch and one person on the armchair, which are both in the living room. This would be a contradiction. To deal with such dependencies, Duet introduces a framework which takes the probabilities of matching identities with trajectories as well as the HMMs for all symbolic locations, and reasons holistically about the various dependencies. Duet's reasoning framework combines probabilistic inference with first order logic. Specifically, at any point in time, Duet has an estimate of the most likely state of the home, where "state" refers to the identities and locations of people. Duet models each state as a set of variables, where each variable corresponds to a person in the environment. Each variable has three kinds of constraints: (a) possibility: a set of people that this variable can be assigned to, (b) impossibility: a set of people this variable cannot be assigned to and (c) area: the symbolic area of interest occupied by the person corresponding to this variable. This design allows the state to be very expressive. For example, it can accommodate constraints like: either Alice or Bob is in the kitchen, or, one person who is not Charlie is on the TV couch. Duet can then reason about user location and identity by solving a satisfiability problem. While a general satisfiability problem in first-order logic is not decidable (i.e., it is impossible to determine if there exists a valid assignment to variables), Duet's formulation of the satisfiability problem in this context is decidable.

While we have discussed Duet in the context of smart homes, its benefits extend to other smart environments. Even at work, users carry their phones only 56% of the time [10, 26]. Users often leave their phones on their desks, connected to the charger, and step out to chat with other employees or fetch some documents from the printer.

We have implemented Duet and evaluated it empirically in both, a home setting and an office environment. We ran our evaluation over an extended period (e.g., two weeks) with the actual occupants of the space. Further, we localize the occupants using their own devices (iPhones and Android phones) as they normally use them. To our knowledge, this is the first study that reports localization performance for real users using their own devices in their normal environment.

We compare Duet with a baseline that uses the-state-of-the-art WiFi-based localization [40]. We also evaluate the ability of Duet to deal with intermittent coverage and sparse interaction with devices and the resulting impact on accurate localization and identification. Our results reveal:

- Duet can accurately identify the person corresponding to a trajectory 95% of the times at home, and 93% of the times in the office. 36% and 56% of the correct identities (in home and office respectively) correspond to matching a device with the nearest user, while the rest were inferred using Duet's first order logic model.
- In our experiments, over 45% of the time, the users were either blocked or outside the coverage area. Duet was able to correctly identify the symbolic space occupied by a user 96% of the time in the home and 94% of the time in the office environment.

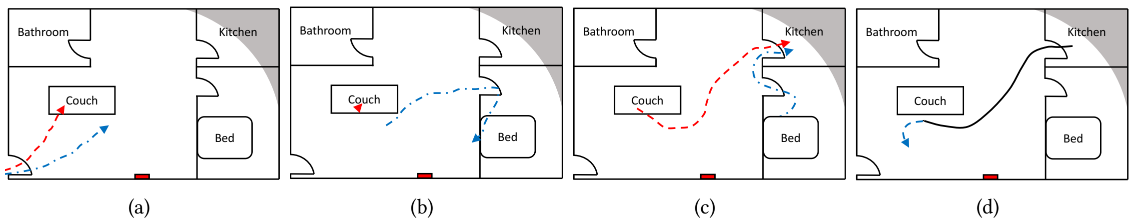


Fig. 1. **Example:** Alice (red) and Bob (blue) are tracked using Duet. The corresponding events and states are described in table 1.

Table 1. **Example:** This table lists a sequence of home events (left column) and the corresponding Duet states (right column).

Event	Duet State
Alice (red) and Bob (blue) come in with their phones (Fig. 1(a)).	$v_1 = (\{Alice\}, LivingRoom)$, $v_2 = (\{Bob\}, LivingRoom)$
Alice and Bob leave their phones behind; Alice goes to the couch to watch TV and Bob goes to the bed.	$v_1 = (\{Alice\}, Couch)$, $v_2 = (\{Bob\}, Bed)$
Alice goes to the kitchen followed by Bob	$v_1 = (\{Alice\}, Kitchen)$, $v_2 = (\{Bob\}, Kitchen)$
Bob comes out of the kitchen	$v_1 = (\{Alice, Bob\}, Kitchen)$, $v_2 = (\{Alice, Bob\}, LivingRoom)$
Bob checks email on his phone	$v_1 = (\{Alice\}, Kitchen)$, $v_2 = (\{Bob\}, LivingRoom)$

- Overall, for the home and office environment, Duet's accuracy in identifying the symbolic location of a person is 96% and 94% respectively. In contrast, the baseline's accuracy is 17% and 42% respectively. The low accuracy of the WiFi-based localization baseline is mainly due to its assumption that users and their phones are co-located.

Limitations and Scope: As is common in pervasive computing [10, 11, 26], Duet assumes that the identity of the person is associated with her cell phone. In some cases, this may not be true. For example, a parent may give her kid her cellphone to play games. It may be easy to infer this usage through side channel information like the apps being used or the network traffic being generated [12, 21, 32]. We leave such an extension to future work.

2 ILLUSTRATIVE EXAMPLE

Let us start with a simple example to show how Duet works. Consider the home in Fig. 1. Let the symbolic areas of interest be: kitchen, living room, bedroom, couch, and bed. For simplicity, assume that the only area out of radio coverage is the shaded area in the kitchen[1]. We will go through a sequence of events and the corresponding sequence of states that Duet outputs. Recall, a state is a set of user variables, where each variable is defined by three constraints: a set of possible identities (P), a set of impossible identities (I), and a symbolic location (R). In this simple example, there are no impossible identities. So, we simplify the notation for each variable to (P, R).

Duet has four components: an identity-matching subsystem, one HMM that tracks access to each symbolic area, a probabilistic first order logic framework to leverage dependencies, and the device-free and device-based localization radios. Here, we illustrate how these components work together to contribute to the resulting state (the details of each component are left to later sections).

The home in our example has two occupants Alice and Bob. Table 1 shows a sequence of events in the first column and the corresponding Duet states in the second column. First, Alice and Bob come home from work and

[1]The radio coverage of Wi-Fi based systems is typically larger than the radio coverage of device-free localization systems. Thus, for the rest of the paper, the term 'radio coverage' has been used to refer to the range of the device-free localization system.

enter the living room carrying their cell phones. The corresponding Duet state is $v_1 = (\{Alice\}, LivingRoom)$, $v_2 = (\{Bob\}, LivingRoom)$ – i.e., Duet detects two people, identifies them as Alice and Bob, and locates them in the living room.

To produce this state, Duet has relied on its device-free localization which measured two user trajectories (the red and blue trajectories in Fig. 1(a)). These trajectories were fed to the HMM for the living room as observations. As a result, the HMM state changed and indicated two people in the living room. Duet's identity matching component used WiFi-based localization to track the location of Alice's and Bob's phones and correlated it with the user trajectories. Recognizing that the two trajectories are highly correlated with the two phones, the identity matching component translated the high correlation into a high probability that the red trajectory is Alice and the blue one is Bob. The logic took the outputs of the identity-matching component and the HMM, and decided that the best explanation of the evidence is that there are two people in the living room who are Alice and Bob.

Next, both Alice and Bob leave their phones on the coffee table to charge them. Alice (marked in red in Fig. 1(b)) goes to the couch to watch TV, while Bob goes to the bedroom to take a nap. As the bed is close to the living room, with just a wall between them, the errors in device-free localization lead Bob's final location to be inside the living room. However, since the blue trajectory in Fig. 1(b) shows an entry through the bedroom door but no corresponding exit through the door, Duet's HMM for the bedroom concludes that Bob is in the bedroom with high probability. Similarly, since the blue trajectory crossed the bedside but did not cross back, the HMM for the bed identifies Bob in the bed. Based on the evidence from both HMMs, the logic concludes that Bob is on his bed and could not have crossed to the living room through the wall. Thus, these events result in an update of Duet's state as follows: $v_1 = (\{Alice\}, Couch)$, $v_2 = (\{Bob\}, Bed)$.

After some time, Alice goes to the Kitchen; later Bob also follows her to the Kitchen (Fig. 1(c)). Since only the kitchen's door is within radio coverage, the device loses track of the exact location of Alice and Bob inside the Kitchen. However, the device-free trajectories when fed as observations to the kitchen HMM, will cause the state to indicate that Alice and Bob are in the kitchen. In the absence of contradictory evidence, Duet's logic agrees and outputs the state: $v_1 = (\{Alice\}, Kitchen)$, $v_2 = (\{Bob\}, Kitchen)$.

Next, Bob leaves the Kitchen and comes back to the living room. Device-free localization outputs a trajectory leaving the kitchen to the living room (Fig. 1(d)). The kitchen HMM and the living room HMM will show one person in each room. However since both Alice and Bob were outside the coverage area in the kitchen, Duet lost all the contextual information about them. Thus, the logic outputs: $v_1 = (\{Alice, Bob\}, \{Kitchen\})$, $v_2 = (\{Alice, Bob\}, LivingRoom)$. This state implies that one of Alice and Bob is in the kitchen and the other is in the living room. At this moment, Duet presents a fuzzy state about the home, where two different outcomes are possible, unless further evidence is presented.

Finally, Bob picks up his phone to check his emails (Fig. 1(d)). This allows the identity matching component to tag the black trajectory in the living room as Bob with high probability. The respective HMMs still have the same states. Based on the interaction, the logic component can resolve the state and outputs: $v_1 = (\{Alice\}, Kitchen)$, $v_2 = (\{Bob\}, LivingRoom)$. Duet would also go back to resolve the prior state recognizing that earlier Bob left the Kitchen and Alice stayed behind.

This example illustrates the following:

- Duet can correct errors made by the underlying localization systems. This happened when the device-free localization system showed Bob in the living room, while he actually was on his bed.
- Duet can localize users even when they are outside the radio range – e.g., positioning Alice and Bob in the kitchen, even though the radio cannot sense them.
- Duet can successfully handle incomplete data and support late binding. This happened when Bob stepped out of the kitchen but Duet could not immediately tell the identity of the person leaving the kitchen. Duet

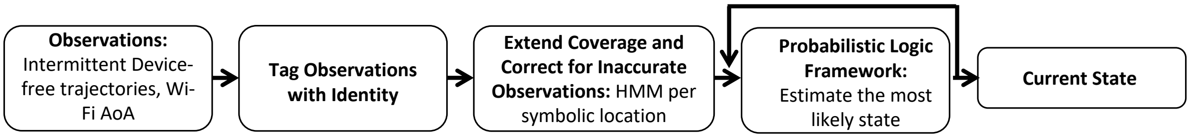


Fig. 2. The different components of Duet.

then resolved this uncertainty once Bob interacted with his phone. This allowed Duet to go back to the earlier state and decide that Alice stayed in the Kitchen and Bob stepped out of it.

3 DUET OVERVIEW

Duet is a wireless sensor that hangs on the wall like a picture frame (Fig. 3). It takes as input the floor-map annotated with Duet's own location, and symbolic areas of interest such as the bedroom, the bed, the TV couch, etc. It also takes the MAC address of the occupants' phones, which it uses as their IDs. Duet tracks the location of the people in the home with respect to the symbolic areas.

Fig. 2 shows Duet's four components and its workflow:

1. *Collecting Location Observations:* Duet tracks people using device-free localization and tracks their phones using Wi-Fi localization.
2. *Tagging Observations with Identity:* Duet has a subsystem that tags users' trajectories with identity based on the correlation between user movements and phone movements. The tagging is probabilistic and includes the option that the trajectory may be for a visitor, unknown a priori to Duet.
3. *Extending Coverage and Correcting for Inaccurate Observation:* Duet assigns an HMM to each symbolic area. The HMM takes the location measurements as observations and reasons about them to identify the number of people in its symbolic area.
4. *Reasoning with Probabilistic Logic:* Finally, the logic collects all evidence from the other components and estimates the most likely state of the home, where *state* refers to the identities and locations of people with respect to the symbolic areas.

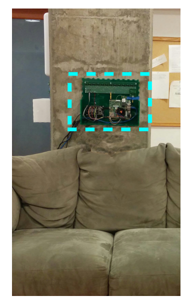


Fig. 3. The Duet sensor (in blue rectangle) in a deployment.

The next few sections describe each component in detail. We note that while Duet can deliver its service with one sensor, Duet can also combine information across sensors to cover a large home.

4 COLLECTING LOCATION OBSERVATIONS

Duet takes as input two types of location measurements:

4.1 Device-Free Observations

Duet is equipped with a multi-antenna FMCW sensor similar to that in [3], which uses RF-reflections to return the location of nearby individuals. The sensor returns a measurement every 20ms. Each measurement contains the location of the detected people. We assume that a person does not move much within 20ms, and hence nearby locations in consecutive measurements belong to the same person and identify her trajectory. We connect such consecutive nearby location estimates to create continuous stretches which we call tracklets. A tracklet looks similar to the blue or red lines in Fig. 1 A tracklet is discontinued whenever the signal from the moving user is too weak. This can happen when the user exits the coverage region or becomes blocked by a TV screen, a mirror, a thick closet, etc. It can also occur when the user stops moving.[2] Tracklets have no identity. This is because the wireless signal reflects off all nearby humans. Thus, in contrast to a Wi-Fi transmission which is always tagged by its sender's MAC address, in the absence of additional information, we cannot tell whose trajectory is captured by a particular tracklet.

4.2 Wi-Fi Based Observations

To add identity to tracklets, Duet uses the intuition that if a user is carrying her device, the motion of the device in the environment will match the motion of the user. Thus, Duet takes as input the MAC address of the devices associated with the home occupants. Duet can track multiple devices for the same user, e.g., her cellphone and laptop. However, for simplicity, we will assume that each occupant is recognized by one Wi-Fi device −her phone. Duet then measures how each device moves and try to correlate each tracklet with the motion of a phone.

One way to track the motion of phones is to continuously localize each of them using a WiFi-based positioning technique [17, 40]. Such systems typically require combining measurements across multiple Wi-Fi access points. However, we want Duet to be effective even if the user deploys only one Duet sensor in her home, i.e., one Wi-Fi access point. Thus, instead of measuring the location of the phone, we measure the angle of arrival (AoA) of its signal. Specifically, we use the Wi-Fi radio on our Duet sensor to compute the spatial direction along which the phone's signal is received at our sensor. Due to multipath effects, the signal arrives along multiple directions. However, one of those directions should be the direct path from the phone to sensor.

There are many ways for measuring the power received along a spatial direction. In particular, for a multiple antenna radio, the power P received along a direction θ at time t is:

$$P(\theta, t) = |\sum_i h_i(t) e^{\frac{j2\pi i l cos(\theta)}{\lambda}}| \tag{1}$$

where $h_i(t)$ is the channel measured at time t on the i^{th} antenna and l is the distance between two antennas. These measurements can be further refined by leveraging that Wi-Fi uses multiple frequencies called OFDM subcarriers ([17]). [3]

To summarize, Duet uses two types of location observations: user tracklets, and the power received from each phone along each spatial direction (output of Eqn. 1).

5 TAGGING OBSERVATIONS WITH IDENTITY

Next, for each tracklet, and for each phone, we want to estimate the likelihood that the person in the tracklet carried the phone. This provides us information about the identity of the person whose motion is captured by the tracklet. But how can we match a phone's movement with a tracklet? As explained in the previous section, the phone's movement is captured by the power received from it along each spatial direction −i.e., Eqn. 1. Thus, we need to correlate the phone's AoA power profile in Eqn. 1 with how the AoA power profile would look like if the phone was located with the person in the tracklet.

Say the tracklet shows that the person is in location $(x(t), y(t))$ at time t. Let us define $d_i(t)$ as the distance between antenna i and the person's location at time t. If the person has the phone on her at time t, then the

[2]When a user stops walking, his breathing signal can still be registered if he is close to the sensor. Yet since the inhale-exhale motion is very small, the ability to track a person based on those reflections becomes weaker −i.e., the coverage area of a static person is smaller than that of a moving person.

[3]We note that continuously pinging the phone to measure its AoA can significantly drain its battery. Duet pings phones only when it observes a person moving − i.e., observes a tracklet.

amount of power that Duet would receive from the phone as a function of direction can be estimated by:

$$P_{est}(\theta, t) = |\sum_i \tilde{h}_i(t) e^{\frac{j2\pi i l cos(\theta)}{\lambda}}| \quad (2)$$

$$\text{where,} \quad \tilde{h}_i(t) = \frac{1}{d_i(t)} e^{\frac{-j2\pi d_i(t)}{\lambda}} \quad (3)$$

In essence, Eqn. 3 estimates the wireless channel information as if the phone was at the same position at the users and then, Eqn. 2 uses these channel estimates to construct the power of the signal coming along each direction.

We can measure the similarity between the phone's measured AoA profile, and the estimated one assuming the phone is carried by the moving person. In particular, let $\vec{P}(t)$ be the vector of received power along all spatial directions and $\vec{P}_{est}(t)$ the vector of estimated power along all spatial directions, had the phone been carried by the person. The similarity between them is captured by the normalized dot product of the two vectors: $s(t) = \frac{\vec{P}(t) \circ \vec{P}_{est}(t)}{|\vec{P}(t)| * |\vec{P}est(t)|}$, where $\circ$ represents the vector dot product and |.| represents the L-2 norm. We can, then, estimate the similarity for the duration of the tracklet as $\sum_t s(t)$, where the sum is taken over all of the phone's measurements that happen to occur during the tracklet. The larger the similarity, the more likely that the person in the tracklet was carrying that phone, and hence the more likely that the person in the tracklet is the one who owns the phone.

We end with two notes:

- *Estimating Probabilities:* We would like to use the similarity function to compute a probability distribution over which phone may be carried by the person in the tracklet. We can do this by normalizing the similarities so that they sum up to 1. However, as we do that we have to account for the possibility of the user in the tracket not carrying any of the phones known to Duet. To account for this case, we imagine a virtual phone whose $\vec{P}(t)$ at any time is a random sample from past measurements of $\vec{P}(t)$ values across all phones.
- *The Effect of Multipath:* In computing P_{est}, we have ignored the effect of multipath. While the multipath effect is easy to measure using an antenna array, it is very hard to predict multipath because such a prediction depends on the material and location of the walls and the furniture. However, two factors alleviate this problem. First, even in the case of multipath, $P(\theta, t)$, is high for θ corresponding to the direct path. Second, as prior work ([17, 40]) has observed, multipath changes over time and only the direct path is consistently observed. Given that the multipath changes over the duration of the tracklet, it is unlikely to cause consistent bias.

6 EXTENDING COVERAGE & CORRECTING OBSERVATIONS

Location observations can be erroneous or missing, as when the person leaves the coverage area. To deal with such issues, Duet extracts the entry and exit boundaries to the monitored symbolic areas −e.g., the doors to each room, the bedside, the side of the couch for setting. It then leverages this information to reason about entry and exit events to individual symbolic spaces. Even if parts of these spaces may be occluded or out of coverage, Duet can still identify the symbolic location of the user based on the entry and exit events. This approach works even when multiple people enter the out-of-coverage area.

So, how does one keep track of people entering and exiting a symbolic space? A naive solution is to mark the possible entrances and exits and increment or decrement a counter whenever a tracklet enters or leaves the space. However, this approach is not robust, primarily because location errors in device-free localization can lead to erroneous entrances and exits, leading to an incorrect estimate of the people count. To add robustness to the people count estimation, Duet uses a Hidden Markov Model (HMM). For a specific area, the HMM takes as input a sequence of observations extracted out of the device-free system and outputs the number of people in the area. Below we give a short primer on HMM followed by our particular HMM design.

6.1 HMM Background

A hidden Markov model is a statistical model that tries to explain a sequence of observations with a sequence of hidden states. The key assumption underlying an HMM is that the probability of an observation given the current state is independent of any other observation or state. Mathematically, let us model the observation at time, t, as o_t and the hidden state as s_t. If $P(X)$ denotes the probability of event X, then the assumption can be represented as:

$$P(o_t|s_0, s_1, s_2, ..., s_{t-1}, s_t) = P(o_t|s_t) \tag{4}$$

Thus, the joint probability of having a sequence of states, $\{s_t\}_{t=1}^{N_t}$ and observations, $\{o_t\}_{t=1}^{N_t}$ is given by:

$$P(\{o_t, s_t\}_{t=1}^{N_t}) = P(s_0)P(o_0|s_0)\prod_{t=1}^{N_t} P(o_t|s_t)P(s_t|s_{t-1}) \tag{5}$$

To model a problem as an HMM one has to define a set of possible states $\mathcal{S} = \{\mathcal{S}_1, ..., \mathcal{S}_M\}$, a set of observations $O = \{O_1, ..., O_N\}$, a $M \times M$ transition probability matrix $\mathbf{T}$, and a $M \times N$ emission probability matrix $\mathbf{E}$. Transition probability is the probability of transitioning from state, $\mathcal{S}_i$ to $\mathcal{S}_j$ in consecutive time steps, i.e., $\mathbf{T}_{ij} = P(s_t = \mathcal{S}_i|s_{t-1} = \mathcal{S}_j)$. The emission probability is the probability of an observation given a state, i.e, $\mathbf{E}_{ij} = P(o_t = O_j|s_t = \mathcal{S}_i)$. The set of observations and states is typically picked by the designer, and the transmission probabilities and emission probabilities are learnt from the data.

6.2 Design of the HMM

How should we pick the HMM states and observations for our application? One option is to directly use the location measurements as observations, and let the HMM determine when a tracklet enters an area, exits an area or stays within the area. This design, however, requires a lot of data to learn the HMM model. Further, the learned HMM will be dependent on the particular environment where the HMM was trained and cannot be transferred to a new home. Instead, we use a more compact model that can be trained in one place and used in new homes.

HMM Observations: For each time instant, the observation can be: (a) an entry event (O_{entry}), (b) an exit event (O_{exit}), (c) number of tracklets (O_k) observed inside the area in this time step. This choice of observations does not require any information specific to the location and thus, can be trained only once. The entry and exit event signals are soft and can be overturned by the HMM based on the complete sequence of observations. In fact, this is exactly why we model the number of tracklets observed in an area as an observation in itself. For instance, if the system sees someone inside, even though it did not see an entry event due to a positioning error, the HMM will correct it to say that one person entered this area. Finally, we reiterate that the number of people inside an area is not necessarily equal to the number of tracklets observed in the area because the area is not completely covered by the device-free localization system.

HMM States: We use the HMM states to represent the number of people in the area at time t. Let us denote the state corresponding to k people as $\mathcal{S}_k$. Further, to allow the HMM to be expressive enough to detect exits and entrances, we introduce two new states for each value of k, $\mathcal{S}_{k\to k+1}$ and $\mathcal{S}_{k\to k-1}$. As the name implies, these states correspond to entry and exit events respectively.

Learning and Inference: Once the HMM model is defined, learning and inference can follow the standard algorithms in literature. The goal of the learning algorithm is to learn the transition probabilities matrix, T, and the emission probabilities matrix, E. In the absence of priors, the typical way to learn T and E is to compute their statistics using labeled data. So, we manually label a sequence of data with the correct states and use it to learn T and E. After the learning phase, the model can be used independent of the area, as described before. In the inference phase, the HMM is given a sequence of observations and is required to find a sequence of states that explain the observations. This is done using the Viterbi algorithm [13].

Algorithm 1 Hypothesis Update Algorithm

```
▷ Given: Hypothesis 𝓗_in = (C_in, b_in); Event as 4-tuple of person, area, type and score, e = (p, r, type, score)
▷ Output: A set of hypothesis, 𝓗_out
▷ Initialize H_out to an empty set.
if type == entry then
    v_new.P = p, v_new.I = [], v_new.R = r
    for v ∈ C_in do
        if p ≠ 𝓤 then v.I = v.I ∪ p end if
    end for
    𝓗_out = {(C_in ∪ v_new, b_in × score)}
else if type == exit then
    for v ∈ C_in do
        if |(v.P − v.I) ∩ {p}| > 0 and v.R == r then
            C_new = C_in\v
            for v' ∈ C_new do
                if p ≠ 𝓤 then v'.I = v'.I ∪ p end if
            end for
            𝓗_out = 𝓗_out ∪ {(C_new, b_in × score)}
        end if
    end for
else
    for v ∈ C_in do
        if |(v.P − v.I) ∩ {p}| > 0 and v.R == r then
            C_new = C_in\v
            for v' ∈ C_new do v'.I = v'.I ∪ p end for
            v.P = {p}, v.I = {}
            𝓗_out = 𝓗_out ∪ {(C_new ∪ v, b_in × score)}
        end if
    end for
end if
```

7 REASONING WITH PROBABILISTIC LOGIC

Duet uses the information it collects to infer the most likely state of the home. Note that the state of the home is different from the HMM state; the state of the home refers to the identities and locations of people with respect to the symbolic areas of interest.

Duet has to reason about the information in a manner similar to how a human does it. For example, if Alice and Bob enter the bedroom and someone walks out, then Duet should be able to tell that the person walking out is either Alice or Bob. In addition, if the Wi-Fi data shows that the person inside is carrying Alice's phone, then the system should be able to reason out that the person who had walked out was Bob.

To make such reasoning, we want to model three constraints. First, a human cannot be present in two different locations at the same time. Second, a human cannot walk into a location that he/she already occupies. Third, anyone walking out of a location has to be one of the people already present in the room.

These constraints are across different dimensions. The first constraint is spatial, the second one is temporal, and the third one is temporal and requires information about multiple people at the same time. This rich diversity in the constraints is great for aiding localization, but is not easy to model. For instance, it cannot be done via an HMM.

7.1 Context Model

To reason about people location and identity, Duet needs a model that captures the information it has and the constraints it needs to obey. We call this model "the context", and describe it below.

Hypothesis: We model the context, C as a set of N competing hypotheses, i.e. $C = \{\mathcal{H}_i | i = 1...N\}$, where $\mathcal{H}_i$ denotes the i^{th} hypothesis. Each hypothesis is defined by a pair: a state, C_i and a belief score b_i. A higher belief score indicates a high probability of the system being in the corresponding state. While we used a single state in our illustrative example described in section 2, we use multiple hypotheses to enable the system to recover from errors made by the underlying localization systems. At any time, the system uses the hypothesis with the highest score to make decisions about symbolic locations of the users.

State: As discussed in section 2, each state is a set of variables, i.e. $C_i = \{v_j | j = 1, ...K\}$, where v_j is the j^{th} variable and K is the number of variables. Here, each variable represents the constraints on the identity of a person. Each variable, v, is represented by three constraints: (a) a set of identities (say Alice, Bob, etc) that can be assigned to this variable, P; (b) a set of identities that *cannot* be assigned to it, I and (c) the symbolic location of this person, R (say bathroom, bedroom, bed, couch). Exactly one identity can be assigned to one variable in a valid state. For completeness, if P and I have overlapping elements, the impossibility takes precedence, i.e., any person assigned to variable v must belong to the set difference, $P - I$. Finally, we use $\mathcal{U}$ to represent the set of all identities in the universe, i.e., if $\mathcal{U} \subset P$, then anyone can be assigned to v, except people in I. Allowing the identity $\mathcal{U}$ enables Duet to handle cases when there is no identity information available (for example, because their phone is powered off). Similarly, it enables Duet to deal with users who do not own/carry a device on them or whose device is not registered with the system.

Events: Duet uses three types of events: exit, entry and interaction. Recall, Duet uses one HMM per symbolic location to track the number of people in that location. Every change in the number of people in a symbolic location corresponds to an event. Specifically, the HMM state $\mathcal{S}_{k \to k+1}$ corresponds to an entry event and the HMM state $\mathcal{S}_{k \to k-1}$ corresponds to an exit event. These events are caused by a tracklet that entered or exited the space. In addition, Duet observes tracklets that are contained within one symbolic location. Any interaction with a phone during these tracklets is considered as an event. Duet's identity matching module assigns a probability score to each of the possible identities corresponding to all tracklets. Thus, each event is defined by a *set* of 4-tuples: $\{(person, region, type, score)\}$, where the $person$ is the identity, $region$ is the symbolic location corresponding to this event, $type$ is entry or exit or interaction, and the $score$ is the score assigned to this tracklet-identity pair by Duet's identity matching module. Finally, when two regions are adjacent, exit from one region corresponds to an entry into another. In that case, the exit event is assumed to have occurred before the entry event.

7.2 Updating Context

Each hypothesis in a context gets updated after every event. Each hypothesis in the current context can fork into multiple hypotheses (since the events are probabilistic in nature) or collapse if the state corresponding to the hypothesis is invalid. We will define how we check for state validity later. Here, we discuss the algorithm for updating a hypothesis based on an event.

For simplicity, consider a single hypothesis ($\mathcal{H}_{in} = (C_{in}, b_{in})$) and an event with only one possible identity. The exact algorithm to use the event to update the hypothesis is given in Algorithm 1. At a high level, if the event is of the type entry, then, a new variable v_{new} is created for the new person who has just entered the symbolic location, r, specified by the event. The possible identity set for this variable is set to the identity for this event, p. This new variable is now added to the state corresponding to this hypothesis. Further, the identity, p, is added to the impossibility set of every other variable in the state (since this identity cannot be assigned to any other person). Analogously, for an event of type exit, the algorithm removes a variable from the state and for an interaction event, it applies additional constraints to the current state. For exit events, the identity of the exiting variable is retained for future events caused by the same tracklet. As in our example in section 2, when the identity matching component failed to identify who exited the kitchen (thus labelling the identity as $\mathcal{U}$), the system could leverage the fact that the kitchen has only two possible people, Alice and Bob. Thus, the exit tracklet (and subsequent entry to living room) could have matched to any one of them.

The above description is for the case when the identity matching module matches each event with just one identity. In the more general case, the event can be matched with multiple identities, i.e., either Bob entered the room for a score of s_1 or Alice entered the room for a score of s_2. In this case, the event set has multiple elements with different identities and scores. This causes the input hypothesis to fork into multiple hypotheses obtained by applying algorithm 1 on each element in the event set.

So far, we discussed the case when the context has a single hypothesis ($\mathcal{H}_{in} = (C_{in}, b_{in})$) . In general, Algorithm 1 is applied to each hypothesis in the current context. The updated context is the union of the hypotheses returned by Algorithm 1.

7.3 Checking State Validity

So far, we have considered how to model context as a set of hypotheses and how to update these hypotheses based on events. However, there is a very important aspect of this problem that remains to be discussed. How do you determine the validity of a state obtained after the update?

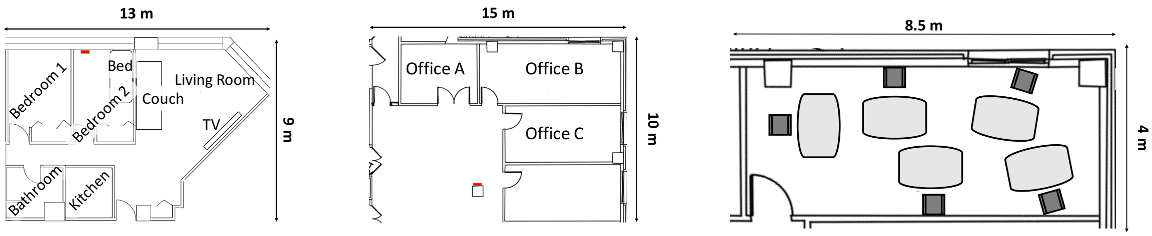
This notion of state validity is useful for two reasons. First, it prunes out invalid states and helps us identify the correct configuration out of the multiple possible states. Second, pruning out states with invalid configurations increases the constraints on the system that can be carried forward. To formalize this notion of states and validity checking, we leverage first-order logic theory. Similar logic tools have previously been used in the context of protocol verification, model-checking, etc, but Duet presents the first formulation of contextual states for estimating and tracking user location information.

Definition: A state is defined to be valid if there exists an assignment of identities to variables, wherein exactly one identity is assigned to each variable. To formalize this notion of validity, let us define the predicate $In(x, X)$ to mean that an element x is in the set X. Further, let us define the predicate $Ar(x, R)$ to denote that a person with identity x is physically present in area R. Recall, each variable v is defined by a triplet $\{P, I, R\}$, where P is the set of people that can be assigned to v, I is the set of people that cannot be assigned to v and R is the area that this person is in. Thus, we can denote the constraint on the assignment of an identity to v as $\exists x[In(x, v.P \backslash v.I) \wedge Ar(x, v.R)]$ ($v.R$ is the value of R corresponding to variable v and $v.P \backslash v.I$ represents the set difference $v.P - v.I$). Thus, for a state which is a set of K variables, we need to model these constraints for each variable. The constraint becomes: $\bigwedge_{i=1}^{K} \exists x_i [In(x_i, v_i.P \backslash v_i.I) \wedge Ar(x_i, v_i.R)]$. Here, $\bigwedge$ represents the logical *and* operator applied to multiple values.

We have modeled that every variable needs to be assigned an identity. Now, we add the constraint that no two variables can be assigned the same identity. So, we can define a state to be valid if and only if the following constraint (technically, a formula) is satisfiable:

$$\exists x_1, ..., x_K \bigwedge_{i=1}^{K} [In(x_i, v_i.P \backslash v_i.I) \wedge Ar(x_i, v_i.R)] \bigwedge_{i=1}^{K} \bigwedge_{j=1}^{i-1} [x_i \neq x_j] \tag{6}$$

A formula in first order logic is satisfiable if there exists a valuation (an assignment of variables to constants) that makes the formula true. In our formulation, the set of constants are the possible identities that can be assigned. By incorporating the exclusive assignment constraints in Eqn. 6, we have modeled the state validity problem as a satisfiability problem in first-order logic. However, a general first-order satisfiability problem is not decidable, i.e., no algorithm today can ascertain if an assignment, that can make the formula true, exists.



(a) Home (b) Office (c) Office B (Zoomed In)

Fig. 4. **Experimental Setup:** We deployed Duet in a 2 bedroom apartment as well as an office environment. The red box shows the location of our sensor. At home, we tracked access to the bed, two bedrooms, TV couch, kitchen and bathroom. In the office environment, we tracked access to the three offices. Within office B, we tracked access to each of the five desks.

Checking Validity: Duet leverages the insight that the formula in Eqn. 6 is not a generic first order formula, but has a specific structure that can be leveraged to decide satisfiability. Specifically, there exist two conditions that one can leverage. First, any variable v with $\mathcal{U} \subset v.P$ is always satisfiable. Second, any variable v with $\mathcal{U} \not\subset v.P$ and $|v.P \backslash v.I| = 1$, must be assigned the value in $v.P \backslash v.I$, as it has only one possibility. Duet builds on these insights to design the algorithm to check satisfiability. In appendix A, we present the detailed algorithm and a proof of correctness.

7.4 Discussion

To conclude, we discuss two design choices in our reasoning framework. First, the choice to model context as multiple hypotheses as opposed to one hypothesis allow us to correct for errors made by the reasoning framework itself. By tracking multiple hypotheses simultaneously, we allow future events to assign higher scores to currently less probable hypotheses and make them more likely in the future. For instance, in our example in section 2, when Bob exited the kitchen, imagine the system incorrectly estimated that the most likely outcome was that Alice exited the kitchen, and Bob was still inside. If we had just one deterministic hypothesis, even if the system later sees Bob entering the couch in the living room with very high probability, it would not be able to make the correct inference. It will simply reject the possibility of Bob being in the living room, since Bob is in the kitchen in the current state. But, in the current model, due to multiple hypotheses with different probabilities, Duet would continue to maintain both hypotheses, one where Bob exited the kitchen and the other where Alice exited the kitchen. When it makes a mistake, the incorrect hypothesis will win over temporarily but over time, as new events unfold, the correct hypothesis will become more likely.

Secondly, one potential problem with the approach above is that the number of hypotheses expands with each iteration. For N hypotheses in a context and T possible events, we get at least NT hypotheses in the next context. Duet controls the hypothesis space in two ways. First, hypotheses with invalid states get pruned out by using Algorithm 2. Second, Duet ranks all the hypotheses in the order of their belief scores. Then, it rejects all except the L most probable hypotheses[4].

8 IMPLEMENTATION

We implemented Duet using off-the-shelf radios augmented with a custom PCB design. To implement a device-free positioning system, we use a multi-antenna FMCW radio similar to that used in [3]. The radio provides continuous real-time location measurements of individuals in the environment sampled every 20 ms. The algorithms developed in [3] are used to tease apart the signal reflections from multiple individuals when multiple individuals occupy the same space.

[4]We set $L = 10$ for our experiments.

To get channel measurements for Wi-Fi packets transmitted by user devices, we use the two-antenna Intel 7265 Wi-Fi chipset, which provides channel state information (CSI) for each Wi-Fi packet. The 7265 chipset has just two antennas and hence cannot provide high-resolution angle of arrival (AoA). To improve the resolution, we built an 8-antenna switched array on a custom PCB. The array can be controlled by the Wi-Fi driver upon packet reception to switch the RF-chain from one antenna to the next. This allows Duet to create the equivalent of 8-antenna array using a 2-antenna off-the-shelf Wi-Fi chipset. As in past work, we correct for CFO by taking the ratio of the channels received on the two RF chains of the Wi-Fi card.

Duet's algorithms described in sections 4, 5, 6 and 7 are implemented in Python and MATLAB. The HMM is trained on 9 days of manually labeled data before the evaluation deployment. Because of its design, the HMM can generalize and does not need to be trained separately for each area. The same model is used for all of our experiments, even though they are conducted in different spaces.

8.1 Evaluation Setup

We evaluate Duet in two different settings: a two-bedroom apartment measuring 9 m by 13 m (Fig. 4(a)), an office area measuring 10 m by 15 m (Fig. 4(b)). In the apartment, the symbolic locations are the bedrooms, the bed, the kitchen, the bathroom, and the TV couch, as shown in Fig. 4(a). In the office space, the symbolic locations are the offices marked as A, B and C in the figure. Further, within office B, we track access to 5 individual desks shown in Fig. 4(c), i.e., each desk-chair pair is marked as a symbolic area.

These environments vary in terms of size, occupancy, and symbolic locations. The home is inhabited by 2 people, with two other frequent visitors; the office space has 9 occupants. Specifically, Office A has one occupant, Office B has five and Office C has three occupants. In addition, this space has occasional visitors. The smallest area tracked in the home is the couch which is about 1.3 m^2. In the open area, each desk (and chair) occupy about 1.5 m^2.

Our experiments were conducted over two weeks. The location of our device is shown as a red rectangle in Fig. 4. The system operates in real-time. We ask users to register their devices in our system and consent to be continuously monitored. The users do not need to install any apps to be tracked by the system. As users enter the monitored environment, Duet automatically updates the current list of devices based on their Wi-Fi transmissions and starts tracking them.

All of our experiments are conducted in natural, dynamic environments with *no change to user devices or behavior*. To the best of our knowledge, our system is the first to present real-time continuous localization with users' own devices.

Ground Truth: To obtain the ground truth, we placed cameras in the common areas and hand-labeled all data for ground truth by watching the camera videos. For areas of interest inside a bedroom (i.e., the bed), the occupant of that bedroom used a camera that he controls and a diary to keep track of bed use. All occupants consented for the camera monitoring.

Baseline: For baseline, we deploy an angle-of-arrival based Wi-Fi localization system in each of these spaces. Specifically, we deploy multiple access points and compute angle-of-arrival of the cellphones of all users using the MUSIC algorithm, which is the state-of-the-art algorithm for Wi-Fi localization. Then, we triangulate these measurements across the access points to get the actual location of the user devices. We re-iterate that these multiple Wi-Fi access points *do not* feed into Duet and are deployed just for a baseline comparison. The baseline uses two 8-antenna Wi-Fi radios positioned along the two axes in each of the spaces. The use of 8 antennas on the access points delivers accurate Angle-of-arrival based localization. To ensure that the baseline is accurate, we measure the localization accuracy of the baseline. The baseline achieves a median localization accuracy of 97 cm, which is comparable to the state-of-the-art work in [17].

9 RESULTS

We present our evaluation of Duet below.

9.1 Overall Accuracy

We start by reporting the overall accuracy of Duet and comparing it with the Wi-Fi-based localization baseline. To calculate accuracy, we measure the percentage of time for which a user's symbolic location is correctly estimated. For example, if a user is in their bed from 5 PM to 6 PM, and the system estimates that the person is in the bed from 5:30 PM to 6:00 PM and not on the bed otherwise, the system would be 50% accurate.

We measure the accuracy for each day of deployment and plot the mean and standard deviation in Fig. 5. The wide bars in the figure show the average over days, whereas the small bars show the standard deviation. As seen in the figure, Duet can achieve a high accuracy of over 96% in home and 94% for the office area. In comparison, the Wi-Fi localization baseline does much worse and is able to correctly identify the symbolic location only 17% of the times at home, 42% of the time in the office.

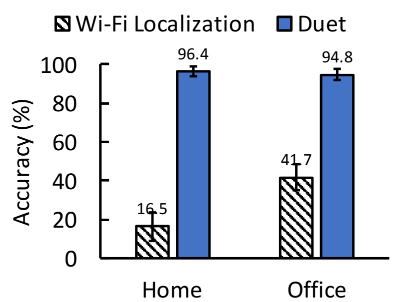


Fig. 5. **Accuracy:** The figure compares localization and identification accuracy of Duet with Wi-Fi localization baseline.

The primary reason for the large errors in the baseline is that the users do not carry their cellphone with them all the time. Since the baseline system just monitors the cellphones of the users, it fails to notice when a user enters or exits a symbolic location without their cellphone. Also note that while past work shows that users tend to be within arm's reach of their phones 46% percent of the time in the home and 56% percent of the time otherwise, these numbers put an upper bound on the accuracy of the baseline. The actual accuracy of the baseline is significantly lower because being close does not mean being in the same symbolic location. In particular, a significant percentage of the time that the users spend at home is in their beds. Their phones are typically close but not in the bed. In contrast, Duet addresses these issues and dramatically improves the location accuracy.

9.2 Error Breakdown in Duet

The above section shows that Duet's errors are limited to a few percent. But where do those errors come from? We classify the errors into three types: (a) Insufficient Context: The user is not carrying her device and there is no prior contextual information about the user, (b) Incorrect Wi-Fi tagging: the user is carrying her device but her tracklet is incorrectly tagged with the id of another user. This could result from localization errors or a user interacting with another person's device. (c) Error Propagation: the event is incorrectly marked because of an incorrect notion of the current state due to incorrect Wi-Fi tags from the past. As can be seen in Fig. 6, a majority of the errors are because of the lack of context when a person enters the environment (e.g., a user entering the coverage area and leaving it without interacting with a phone and hence providing no context to infer her/his identity.) The overall low error rate and the fact that most errors are due to lack of contextual information imply that Duet's probabilistic logic is reasoning accurately about various constraints and dependencies.

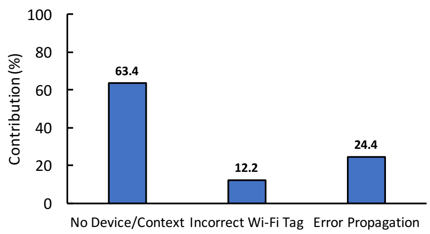


Fig. 6. Breakdown of the errors experienced in Duet's deployments

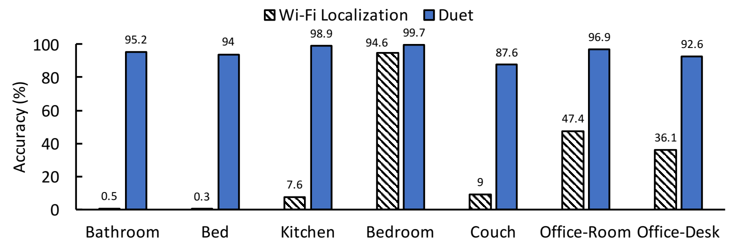


Fig. 7. Accuracy results for different symbolic locations

9.3 Accuracy in Sub-spaces

To further understand the working of Duet and user behavior, we measure the accuracy for different areas in our deployments. We plot the variation of accuracy over the different types of physical spaces in Fig. 7. The results indicate an interesting trend. For the Wi-Fi baseline, the accuracy goes up to 95% in the bedroom, but falls down to almost zero in the bathroom and the kitchen. This is explained by user behavior. When users are at home, they tend to leave their devices in one spot, such as the bedroom, but typically do not carry it to the kitchen or the bathroom. Thus, the Wi-Fi localization baseline correctly estimates user's current area, only if she is in the space where she left her device (in this case, the bedroom).

In contrast, Duet can easily reason about these spaces by keeping track of contextual information. For instance, if a user comes into the bedroom, sets her alarm on the phone and sleeps on the bed, we can use the brief interaction with the phone to identify the user and continue to track her to her bed. As a result, for Duet, the accuracy is consistently over 90% for all the spaces.

Finally, observe that the performance of both Duet and the Wi-Fi baseline degrades when monitoring the desks in Office B as compared to monitoring the rooms themselves. This stems from two reasons. First, Office B represents a very cluttered environment. Each of the desks in this space has two large monitor screens and a couple of PC's. Consequently, the presence of a large amount of metal in this area leads to the direct path of the signal being blocked more often, leading to more errors.Secondly, each of the desks is a smaller symbolic location and naturally, localization errors have larger impact on accuracy.

9.4 Identity Matching

The previous results show the overall accuracy for localization and identification. Here, we zoom in on the identification problem. We plot the percentage of trajectories that could be correctly tagged with the user identities if we just relied on the matching algorithm. Specifically, we pick the highest probability identity for each tracklet and compare the identity with the ground truth identity for the same tracklet. The percentage of tracklets correctly labeled by the matching algorithm is plotted in Fig. 8. For reference, we also plot the percentage of correctly labeled tracklets if we use the end-to-end Duet pipeline.

As seen in the plot, the percentage of tracklets correctly labeled by the matching algorithm is 36% in home, 56% in office area. These numbers go up to 95% and 93% once Duet applies its HMMs and probabilistic logic. This result shows that one cannot simply tag a tracklet with the identity of the nearest phone. This is because of two reasons: people do not carry their phones on them all the time, and both device-based and device-free localization make errors. Duet can correct these errors and fill in missing information based on its other components (the HMM's and the first order logic) and correctly identify the people in over 93% of the tracklets.

9.5 Expanding Coverage

An objective of Duet is to expand the coverage of the underlying localization systems in space and time. To assess if Duet could successfully achieve this in our deployment, we plot the percentage of time a person is located by the device-free localization system in Fig. 9.

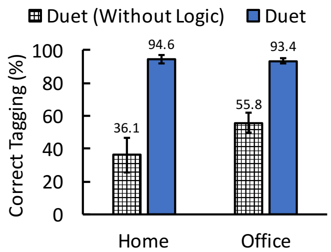


Fig. 8. **Tagging Accuracy:** Duet's reasoning framework improves the accuracy of identifying people corresponding to tracklets.

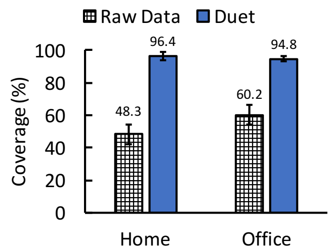


Fig. 9. **Coverage:** Duet expands the coverage of the underlying device-free system in both home and office deployments.

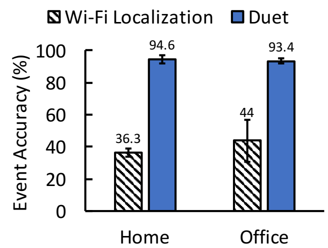


Fig. 10. **Event Accuracy:** The figure plots the accuracy of detecting entry and exit events.

As seen in the figure, the coverage provided by the device-free localization system is about 48% of the time in the home deployment and about 60% of the time in the office deployment. Note that there are two reasons for a person to be out of coverage. First, they are out of range of the device or they are blocked completely by metallic bodies like TV, HVAC, etc. Second, since device-free systems try to isolate humans from surrounding reflectors based on motion. Thus, if a human is still, they are unlikely to be tracked by the device. Duet can overcome this lack of coverage by leveraging the HMMs to track access to symbolic spaces. Thus, even if the user sleeps on the bed and disappears due to lack of motion, Duet uses the information that the user entered the bed and never exited it to continue to identify her symbolic location.

9.6 Event Detection Accuracy

We note that some smart home applications are reactive, such as changing the TV channel, adapting the lights to the user's liking, or turning the alarm off once the user steps out of her bed. These applications care about when the user steps in or out of the symbolic area. Thus instead of the percentage of time a user is localized accurately, which is the metric in section 9.1, their measure of accuracy focuses on detecting entrance and exit events and the identity of the user performing the event. To capture the interest of this class of applications, we also measure the event detection accuracy of Duet. Specifically, we consider each entry or exit from a symbolic area as an event. The accuracy is measured as the percent of such events for which the entry/exit is correctly detected *and* the identity of the user performing the action is correctly labeled. The event has to be detected and tagged with the user identity in less than 5 seconds to be considered correct.

We plot the event detection accuracy of Duet as well as the Wi-Fi localization baseline in Fig. 10. Duet achieves a high accuracy of around 94% in both the environments. On the other hand, the accuracy achieved by Wi-Fi localization is much lower due to the reason described before, namely, people are not carrying their phones on them as well as the localization errors. Duet can use past information to weed out incorrect choices and hence, more accurately identify the people performing the actions.

10 RELATED WORK

RF-based Localization: Duet is inspired by past work on RF-based localization. Over the last two decades, researchers have used variety of localization techniques that exploit RSSI [7, 9, 28, 44], angle of arrival [14, 15, 40], time of flight [34, 41], or a combination of these concepts [17–19, 31, 36] to achieve sub-meter median location accuracy for RF devices. Some papers have even demonstrated methods that can localize a device using one access point [34, 38]. However, past literature focuses on improving localization accuracy assuming the user carries a wireless device on her. In contrast, Duet focuses on scenarios where users often leave their devices behind, at their desks, in their bags, or connected to a charger. Duet introduces a new design that can reason about intermittent and partial location data to deliver perpetual localization.

Duet also builds on past work on device-free localization [3, 4, 35, 42, 45]. These systems typically do not have a notion of identity. The few that identify different users do it only when the user walks on a specific predefined path, for which the system is calibrated [2, 37, 39]. In contrast, Duet focuses on identifying users independent of the path they walk. Further, it presents new techniques that allow localization systems to reason about intervals during which the user is out of coverage.

Use of Contextual Information to Augment Localization: Our work is also related to past papers that leverage contextual information to improve the performance of localization systems [1, 5, 29, 33]. These solutions, however, assume that location measurements are always available, and focus on improving the accuracy by leveraging the floor map or the fact that people do not walk across walls. In contrast, Duet focuses on scenarios where location fixes are intermittent and hence there is a need to reason both across space and time to understand how people move at home.

Some past work also focuses on leveraging sensors deployed in the environment to understand the interactions between people and surrounding objects [8, 16, 27, 30]. For example, they may tag all windows and doors to detect the opening/closing of a window/door and use this information to infer the location of a user [8]. When deployed, such sensors can play an important role in augmenting Duet to capture interactions with specific objects. However, a system that localizes people purely based on their interaction with a large number of tagged objects is hard to maintain over time, and furthermore, lacks identity information.

Other Sensing Modalities: Beyond RF-based localization systems, other sensing modalities like visible light [20, 22, 23, 43, 46, 47] or acoustic signals [6, 24, 25] have been used for indoor positioning. Much like RF positioning systems, they either require the user to carry a device to be localized [6, 20, 24, 25, 43, 46, 47] or lack identity information [22, 23]. We believe these systems are orthogonal to Duet and can complement the framework proposed in this paper by presenting additional streams of information or replacing one of the two streams of data used in Duet.

11 CONCLUSION

We present Duet, a system for continuous localization and identification of users in a smart home. We base our design on a logical reasoning formulation that can enable a smart home to reason about the location and identity of different users in the environment, even when partial, intermittent location data is available. Our experience from two weeks of continuous deployment in users' natural environments suggests that Duet can act as a substrate for different kinds of smart home applications, ranging from playing music to long term analytics of user behavior.

ACKNOWLEDGMENTS

The authors thank the members of the NETMIT group for their insightful discussions, anonymous reviewers for their valuable comments and all the volunteers for participating in our study.

APPENDIX A ALGORITHM TO CHECK STATE VALIDITY

The algorithm to check the validity of states in Duet's probabilistic reasoning model is described in algorithm 2. We prove the correctness of Algorithm 2 below. Specifically, we prove that the algorithm returns "true" if and only if the formula in Eqn. 6 is satisfiable.

We will prove this by the principle of mathematical induction. For the base case, consider the case of a state that has a single variable, v. In this case, if the algorithm returns true, there exist at least one value in $v.P \backslash v.I$ that can be assigned to v. This implies that the state is valid or the corresponding first-order formula is satisfiable. Similarly, if the state is valid, there exists at least one value that can be assigned to the variable, v. This implies $v.P \backslash v.I$ is non-empty. It is trivial to see that the algorithm will return true in this case.

Algorithm 2 Algorithm to Check State Validity

▹ Given: State, $C = \{v_i\}_{i=1}^{K}$
▹ Preprocessing Step: Remove variables (and corresponding constraints) that have $\mathcal{U} \subset v.P$ as they are always satisfiable
▹ Output: True if the state is valid, false otherwise
function CHECK_VALIDITY(C)
    for $i = 1, ..., K$ **do**
        if $|v_i.P \backslash v_i.I| == 0$ **then**
            return false
        else
            for $j = 1, ..., |v_i.P \backslash v_i.I|$ **do**
                $v'.P = (v_i.P \backslash v_i.I)_j, v'.I = [], v'.R = v_i.R$, where $(X)_j$ is the j^{th} element in set X
                $x_i = (v_i.P \backslash v_i.I)_j$
                for $k = i + 1, ...K$ **do**
                    $v_k.I = v_k.I \cup \{(v_i.P \backslash v_i.I)_j\}$
                end for
                $C_1 = \bigcup_{l=i+1}^{K} v_l$
                if check_validity(C_1) **then** return true **end if**
            end for
            return false
        end if
    end for
end function

Now, let us assume for all $k < K$ variables in a state, the algorithm returns true if and only if the corresponding first-order formula is satisfiable. If we prove that this assumption implies that the statement holds for $k = K$, then by the principle of mathematical induction, the statement must be true for all integer values of k, which is what we aim to prove. First, let us assume that the formula is satisfiable, i.e. there exist constants $p_1, p_2, ..p_K$, such that setting $x_i = p_i$ makes the formula true. Thus, for all values of i, $p_i \in v_i.P \backslash v_i.I$. Since, $p_1 \in v_1.P \backslash v_1.I$, the algorithm must try $x_1 = p_1$. When the algorithm tries $x_1 = p_1$, it makes a recursive call to the algorithm with an updated state, C_1, where each variable has ruled out p_1. Clearly, the assignment $x_i = p_i$ satisfies the updated formula for C_1, since each p_i is unique. Since the formula for C_1 is satisfiable and has $k - 1$ variables, the algorithm must return true on this recursive call (using our assumption). Thus, if the formula is satisfiable, the algorithm returns true.

To prove the converse, let us assume that the algorithm returned true for a state, C, with K variables. We need to prove that the formula corresponding to state C is satisfiable. To return true for a state of size $K > 1$, for at least one value of $x_1 \in v_1.P \backslash v_1.I$ (say p_1), the recursive call to the algorithm returned true for the corresponding updated state C_1. Since, C_1 has less than K variables, this implies that there exists an assignment of variables which satisfies the formula corresponding to C_1. Finally, none of those variables can be assigned p_1 because it was removed as a possibility by the algorithm before the recursive call was performed. Thus, there exist a valid assignment to variables for the formula corresponding to C. Using principles of mathematical induction, this proves that the algorithm returns true if and only if the state is valid.

REFERENCES

[1] Heba Abdelnasser, Reham Mohamed, Ahmed Elgohary, Moustafa Farid Alzantot, He Wang, Souvik Sen, Romit Roy Choudhury, and Moustafa Youssef. 2016. SemanticSLAM: Using Environment Landmarks for Unsupervised Indoor Localization *(Transactions on Mobile Computing)*.

[2] Fadel Adib, Chen-Yu Hsu, Hongzi Mao, Dina Katabi, and Fredo Durand. 2015. RF-Capture: Capturing the Human Figure Through a Wall *(SIGGRAPH Asia)*.
[3] Fadel Adib, Zachary Kabelac, and Dina Katabi. 2015. Multi-person Localization via RF Body Reflections *(NSDI)*.
[4] Fadel Adib, Zach Kabelac, Dina Katabi, and Robert C. Miller. 2014. 3D Tracking via Body Radio Reflections *(NSDI)*.
[5] Imad Afyouni, Cyril Ray, and Christophe Claramunt. 2012. Spatial Models for Context-Aware Indoor Navigation Systems: A Survey *(JOSIS)*.
[6] Martin Azizyan, Ionut Constandache, and Romit Roy Choudhury. 2009. SurroundSense: Mobile Phone Localization via Ambience Fingerprinting *(ACM MobiCom)*.
[7] Victor Bahl and Venkat Padmanabhan. 2000. RADAR: An In-Building RF-based User Location and Tracking System *(INFOCOM)*.
[8] Paolo Barsocchi, Stefano Chessa, Erina Ferro, Francesco Furfari, and Francesco Potorti. 2011. Context Driven Enhancement of RSS-based Localization Systems *(ISCC)*.
[9] Krishna Chintalapudi, Anand Padmanabha Iyer, and Venkata N. Padmanabhan. 2010. Indoor Localization Without the Pain *(MobiCom)*.
[10] Anind K. Dey, Katarzyna Wac, Denzil Ferreira, Kevin Tassini, Jin-Hyuk Hong, and Julian Ramos. 2011. Getting Closer: An Empirical Investigation of the Proximity of User to Their Smart Phones *(UbiComp)*.
[11] Denzil Ferreira, Jorge Goncalves, Vassilis Kostakos, Louise Barkhuus, and Anind K. Dey. 2014. Contextual Experience Sampling of Mobile Application Micro-usage *(MobileHCI)*.
[12] Andrey Finkelstein, Ron Biton, Rami Puzis, and Asaf Shabtai. 2017. Classification of Smartphone Users Using Internet Traffic. *CoRR Arxiv* (2017).
[13] G. D. Forney. 1973. The Viterbi Algorithm. *Proc. IEEE* (1973).
[14] Jon Gjengset, Jie Xiong, Graeme McPhillips, and Kyle Jamieson. 2014. Phaser: Enabling Phased Array Signal Processing on Commodity Wi-Fi Access Points. *MobiCom* (2014).
[15] Kiran Joshi, Steven Hong, and Sachin Katti. 2013. PinPoint: Localizing Interfering Radios *(NSDI)*.
[16] Avinash Kalyanaraman, Dezhi Hong, Elahe Soltanaghaei, and Kamin Whitehouse. 2017. Forma Track: Tracking People Based on Body Shape. *ACM IMWUT* (2017).
[17] Manikanta Kotaru, Kiran Joshi, Dinesh Bharadia, and Sachin Katti. 2015. SpotFi: Decimeter Level Localization Using Wi-Fi *(SIGCOMM)*.
[18] Swarun Kumar, Stephanie Gil, Dina Katabi, and Daniela Rus. 2014. Accurate Indoor Localization with Zero Start-up Cost *(MobiCom)*.
[19] Swarun Kumar, Ezzeldin Hamed, Dina Katabi, and Li Erran Li. 2014. LTE Radio Analytics Made Easy and Accessible *(SIGCOMM)*.
[20] Ye-Sheng Kuo, Pat Pannuto, Ko-Jen Hsiao, and Prabal Dutta. 2014. Luxapose: Indoor Positioning with Mobile Phones and Visible Light *(ACM MobiCom)*.
[21] Huaxin Li, Zheyu Xu, Haojin Zhu, Di Ma, Shuai Li, and Kai Xing. 2016. Demographics inference through Wi-Fi network traffic analysis *(INFOCOM)*.
[22] Tianxing Li, Chuankai An, Zhao Tian, Andrew T. Campbell, and Xia Zhou. 2015. Human Sensing Using Visible Light Communication *(ACM MobiCom)*.
[23] Tianxing Li, Qiang Liu, and Xia Zhou. 2016. Practical Human Sensing in the Light *(ACM MobiSys)*.
[24] Hongbo Liu, Yu Gan, Jie Yang, Simon Sidhom, Yan Wang, Yingying Chen, and Fan Ye. 2012. Push the Limit of WiFi Based Localization for Smartphones *(ACM Mobicom)*.
[25] J. Oh and J. Um. 2018. Acoustic Signal-Based Indoor Global Coordinates System for Smartphones. *IEEE Sensors Journal* (2018).
[26] Shwetak N. Patel, Julie A. Kientz, Gillian R. Hayes, Sooraj Bhat, and Gregory D. Abowd. 2006. Farther Than You May Think: An Empirical Investigation of the Proximity of Users to Their Mobile Phones *(UbiComp)*.
[27] D. J. Patterson, D. Fox, H. Kautz, and M. Philipose. 2005. Fine-grained Activity Recognition by Aggregating Abstract Object Usage *(ISWC)*.
[28] Anshul Rai, Krishna Kant Chintalapudi, Venkata N. Padmanabhan, and Rijurekha Sen. 2012. Zee: Zero-effort Crowdsourcing for Indoor Localization *(MobiCom)*.
[29] Montserrat Ros, Joshua Boom, Gavin de Hosson, and Matthew D'Souza. 2012. Indoor Localisation Using a Context-Aware Dynamic Position Tracking Model *(International Journal of Navigation and Observation)*.
[30] Christoph Scholz, Martin Atzmueller, and Gerd Stumme. 2014. *Unsupervised and Hybrid Approaches for On-line RFID Localization with Mixed Context Knowledge.*
[31] Souvik Sen, Jeongkeun Lee, Kyu-Han Kim, and Paul Congdon. 2013. Avoiding Multipath to Revive Inbuilding Wi-Fi Localization *(MobiSys)*.
[32] Suranga Seneviratne, Aruna Seneviratne, Prasant Mohapatra, and Anirban Mahanti. July 2014. Your Installed Apps Reveal Your Gender and More! *SIGMOBILE Mob. CCR* (July 2014).
[33] Zheng Sun, Rick Farley, Telis Kaleas, Judy Ellis, and Kiran Chikkappa. 2011. Cortina: Collaborative COntext-aware Indoor Positioning Employing RSS and RToF Techniques *(PERCOM)*.
[34] Deepak Vasisht, Swarun Kumar, and Dina Katabi. 2016. Decimeter-Level Localization with a Single Wi-Fi Access Point *(NSDI)*.

[35] Ju Wang, Hongbo Jiang, Jie Xiong, Kyle Jamieson, Xiaojiang Chen, Dingyi Fang, and Binbin Xie. 2016. LiFS: Low Human-effort, Device-free Localization with Fine-grained Subcarrier Information *(MobiCom)*.
[36] Jue Wang and Dina Katabi. 2013. Dude, Where's My Card?: RFID Positioning That Works with Multipath and Non-line of Sight *(SIGCOMM)*.
[37] Wei Wang, Alex X. Liu, and Muhammad Shahzad. 2016. Gait Recognition Using Wi-Fi Signals *(UbiComp)*.
[38] F. Wen and C. Liang. 2015. Fine-Grained Indoor Localization Using Single Access Point With Multiple Antennas. *IEEE Sensors Journal* (2015).
[39] Tong Xin, Bin Guo, Zhu Wang, Mingyang Li, and Zhiwen Yu. 2016. FreeSense: Indoor Human Identification with Wi-Fi Signals. *CoRR Arxiv* (2016).
[40] Jie Xiong and Kyle Jamieson. 2013. ArrayTrack: A Fine-grained Indoor Location System *(NSDI)*.
[41] Jie Xiong, Karthikeyan Sundaresan, and Kyle Jamieson. 2015. ToneTrack: Leveraging Frequency-Agile Radios for Time-Based Indoor Wireless Localization *(MobiCom)*.
[42] Chenren Xu, Bernhard Firner, Yanyong Zhang, Richard Howard, Jun Li, and Xiaodong Lin. 2012. Improving RF-based Device-free Passive Localization in Cluttered Indoor Environments Through Probabilistic Classification Methods *(IPSN)*.
[43] Zhice Yang, Zeyu Wang, Jiansong Zhang, Chenyu Huang, and Qian Zhang. 2015. Wearables Can Afford: Light-weight Indoor Positioning with Visible Light *(ACM MobiSys)*.
[44] Moustafa Youssef and Ashok Agrawala. 2005. The Horus WLAN Location Determination System *(MobiSys)*.
[45] Moustafa Youssef, Matthew Mah, and Ashok Agrawala. 2007. Challenges: Device-free Passive Localization for Wireless Environments *(MobiCom)*.
[46] Chi Zhang and Xinyu Zhang. 2016. LiTell: Robust Indoor Localization Using Unmodified Light Fixtures. In *Proceedings of the 22Nd Annual International Conference on Mobile Computing and Networking (ACM MobiCom)*.
[47] Shilin Zhu and Xinyu Zhang. 2017. Enabling High-Precision Visible Light Localization in TodayâĂŹs Buildings *(ACM MobiSys)*.

Received February 2018; revised April 2018; accepted April 2018