A Machine Learning Assisted Cell Selection Method for Drones in Cellular Networks

Sai Qian Zhang[1], Feng Xue[2], Nageen Himayat[2], Shilpa Talwar[2], and H.T. Kung[1]

[1]John A. Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, USA
[2]Intel Labs, Santa Clara, USA
zhangs@g.harvard.edu, {feng.a.xue, nageen.himayat, shilpa.talwar}@intel.com, kung@harvard.edu

***Abstract*—We apply machine learning techniques to predict the cell quality for the aerial drones connecting with a standard cellular network on the ground. Stationary and strong spatial correlation of the aerial channels allow for exploiting predictive techniques for optimal cell selection based on few available neighboring observations. Yet, drastic cell quality changes due to the side lobes of base-station antenna patterns require advanced solutions for accurate prediction. In this paper, we propose a conditional random field based framework to predict a drone's best (or top few) candidates for the serving cell. Our results, assuming realistic antenna patterns as well as errors in the location estimates, show a high prediction accuracy, thereby illustrating the feasibility of exploiting learning approaches to predict the aerial channel environment.**

I. Introduction

Drones, a.k.a. unmanned aerial vehicles (UAVs), are under dramatic growth in the last few years. Many new usages require enhanced support for robust and high throughput wireless communication, as well as relaxation of the restrictive regulation as exemplified by FAA's 2016 rule [1] mainly due to safety concern. Towards future large-scale, high density drone deployments, coexisting with manned airplanes (e.g., [2]), multiple-layer redundancy with good infrastructure support is needed. This is in addition to autonomous solutions such as auto-piloting consisting of only on-board technologies.

To address these needs, renovation based on today's cellular networks is being considered as a candidate solution. Cellular networks allow wide coverage, provide fast air and backhaul links, and have dedicated spectrum and infrastructure. To leverage these advantages, recently 3GPP initiated a study item on "Enhanced support for aerial vehicles" [3].

However, there are several challenges towards enabling and optimizing drone support in cellular networks, mainly due to the nature of their operation. These networks are primarily optimized for serving ground user equipments (UEs), with the base-station (BS) antennas tilted towards the ground direction. Therefore, with high probability the drone in the air are served by the side lobes of the BS's antennas. Even though, due to less blockage and diffraction in the air, the drone links enjoy good signal-to-noise ratio (SNR), they see many more BS's than the ground UEs and observe rapidly fluctuating gain pattern with nulls (see Figure 1). Recent study based on the tests and simulations has shown that the drones experience drastic interference and signal fluctuation in the air [12], [13], [10]. At the same time, drones cause significant interference to ground UEs in the uplink (UE to BS). These factors make interference mitigation and handover robustness two of the key issues in the 3GPP study (e.g., [4], [8], [9]).

Although complicated, the signal situations in the air are more predictable in nature due to the dominant Line-of-sight (LoS) channel, as well as less blockage and diffraction. Aerial channel properties are expected to be more correlated to spatial locations of observation points and their position relationship to the base station (BS) antennas, etc. This predictability can potentially be exploited in powerful ways to aid in system optimization and planning of drone deployments.

In this paper, we focus on a simple application of exploiting the predictability of the aerial environment. In particular we use machine learning methods to predict a drone's serving cell based on the known signal observations in the nearby region. Specifically, we adapt the conditional random field position (CRF) for predicting a drone's best serving cells at a given location. CRF [11] is a classical inference model widely used in computer vision [5] and natural language processing [6], and has been considered for the wireless system study [7]. For the cell selection problem considered in this paper, we construct a localized correlation region, consisting of a finite grid of locations, and build a random field to describe the best serving cells for each location in the grid. A parameterized joint distribution is trained by the 'ground truth' - the sampled locations. The performance of our approach is verified with the simulations under practical BS antenna patterns with typical 3GPP deployment setup. With a modest number of sampled locations, our CRF-based method can achieve 90% accuracy in selecting the best cells. Compared with the other two benchmark algorithms, KNN-CSA and CM, our CRF-based method shows a 4.7% and 4.5% improvements in absolute percentages, respectively (see Section IV). Moreover, we show that our CRF-based method is robust against localization errors resulting from limited accuracy in the Global Positioning System (GPS) positioning.

The results of this paper are illustrative of the promise of using the CRF model in predicting the aerial signal environment. That is, by processing and utilizing the known observations (e.g. from data gathered on best cells by past drone runs), a full picture of the aerial environment can be built for system optimization such as planning, traffic control

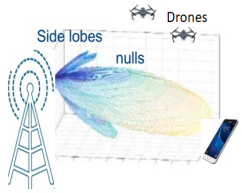

Fig. 1. Drones on a cellular network

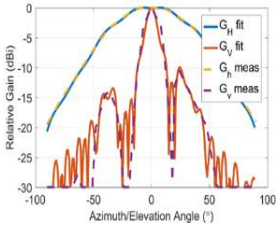

Fig. 2. BS antenna pattern

and so on. Compared with the constant scanning based on reference signals, prediction may also speed up initial access and lower power consumption by starting from top candidate cells. Moreover, such design may provide more freedom in introducing new method to address the excessive interference problem for drones. For example, methods based on directional antennas or beams are currently being considered [8], [9]. By pointing the beam towards a wanted cell, the desired signal is enhanced while those unwanted are suppressed. Without careful design, a traditional method would select the best cell based on signal measurement. This is impacted by the directional antenna beam in place, thus creating a chicken-egg problem. In contrast, our method would select the wanted (best) cell with accuracy without being impacted by the directional antenna pattern.

II. Problem Statement

In this section we formulate the problem of CRF to predict best cell for association or handover based on the drone's location. As noted, compared to the channels on the ground, aerial channels are more stationary due to less obstacles and scattering. However, there are a few factors complicating the predictive task: 1) Each BS's antenna height, tilting and antenna pattern may be different; 2) the signal propagation is subject to the properties of nearby terrestrial objects such as trees, hills, ground reflection, leading to shadowing effects. These properties introduce complex spatial correlations between nearby positions, supporting a proposition of describing their relationship with an appropriate joint distribution between random variables associated with these positions. Since in practice part of the air space can be sampled by specialized drone runs or by past observations, we formulate a cell quality prediction problem by exploiting such measurements as follows: *Given that a fraction of the air space has been measured such that their best serving cells are known, we infer the best serving cell for the rest of the space.*

Since the space is continuous, we represent it as a 3D grid, where the physical distance between the neighboring nodes is d meters. Each grid node v is associated with a random variable x_v, denoting the optimal cell for that position. We assume some nodes' best serving cells and signal-to-interference-plus-noise ratio (SINR) are known (see Figure 3), the problem then is to determine the true values of x_v for all the nodes. Towards this goal, we will introduce a joint distribution trained to describe the correlations between the random variables. The distribution needs to be adaptive and efficient for various wireless environments, then cell prediction is done by determining the marginal distribution. We will elaborate this more in the next section.

III. A CRF-based framework and algorithm for predicting serving cells

A. Introduction of CRF

We use a graphical model $G(V, E)$, where each node $v \in V$ is associated with a random variable (the *label*) $x_v \in X$, X being the possible labels. Each node is also associated with a *node potential* and each edge an *edge potential*. The node potential reflects the prior knowledge on the label assignment of each node, while the edge potential captures the relationship between nearby nodes, typically emphasizing the closer nodes. One can use $\phi(x_v)$ and $\psi(x_v, x_{v'})$ to denote the node potential for $v \in V$ and edge potential for $(v, v') \in E$, respectively. Let $x_V = \{x_{v \in V}\}$ represent the label selections for all the nodes in G. Table I gives a quick glossary of definitions. The joint probability $P_V(x_V)$ is defined as the product of all the node potentials and edge potentials in the following fashion:

$$P_V(x_V) = \frac{1}{Z} \prod_{v \in V} \phi(x_v) \prod_{(v,v') \in E} \psi(x_v, x_{v'}) \quad (1)$$

where Z is the normalization constant. Now we construct a

TABLE I
Definitions of parameters

Name	Description	Name	Description
G_0	grid that covers the whole area of interest	G_i	grid that centers at i with span of $a \times b \times c$ meters
E_0	set of edges in G_0	V_0	set of nodes in G_0
V_i	set of nodes in G_i	E_i	set of edges in G_i
x_v	cell selection of node v	x_V	cell selections of all $v \in V$
S_0	set of sample nodes in G_0	S_i	set of sample nodes in G_i
x_s^*	optimal cell at sample s	x_v^*	optimal cell at v
w_{kh}	parameter of node potential	$m_{vv'}$	parameter of edge potential
K	range where two nodes are correlated	H	number of levels of discretized SINR
$P_V(.)$	joint probability of cell selections of V	$P_v(.)$	marginal probability of cell selection of v
T	threshold on cell inference	D	set of training data
R	number of training datasets	μ_s^*	SINR from optimal cell at s
$N(v)$	set of neighbors of v	X	set of all possible cells

specialized CRF framework for our problem of predicting cells. As described in Section II, a grid, G_0, is set up to cover the whole area of interest in the 3D space, with physical distance between two neighboring nodes equal to d meters in all the 3D space. For an arbitrary location in the space, its cell prediction is that of its closest grid node's, allowing our design and analysis to be based only on the discrete grid. To apply CRF as in (1), we consider a graph $G_i(V_i, E_i)$ formed by nodes and edges within a rectangular cuboid of dimension $a \times b \times c$ meters that centers at node i. Let X denote the set of possible cell IDs, and x_v, $v \in V_i$ denote v's serving cell. There is a subset of nodes S_i in V_i such that each node

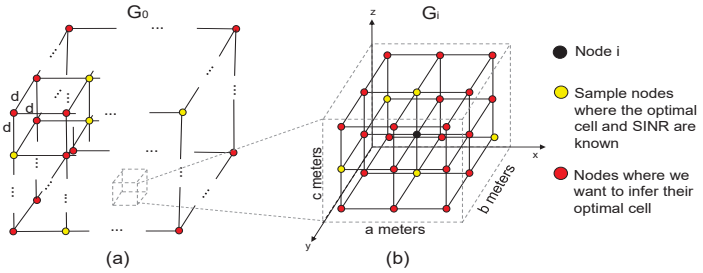

Fig. 3. (a) grid G_0, (b) sub-grid G_i centered at node i.

$s \in S_i$'s best serving cell, x_s^*, and the corresponding SINR, $SINR_s^*$, are known. Figure 3 shows G_0 and G_i whose sample nodes are shown in yellow and rest of nodes are shown in red. Assume further that the measured SINR's in S_i are within range $[SINR_{min}, SINR_{max}]$. Divide this range into H equal intervals and index each SINR with the corresponding interval, call this SINR index and denote the SINR index at s as u_s^*. We measure the distance between the nodes in terms of physical hops (p-hops), and it is defined as follows: for the node v in the grid, rank the rest of the nodes based on their line-of-sight distance to v. The rank of that node is defined as the its distance to v in p-hops. Now define the node and edge potential functions as follows:

$$\phi_v(x_v) = \exp(\sum_{k=0}^{K}\sum_{h=1}^{H} w_{kh} \sum_{1\{s\in S|d(s,v)=k\}} \mathbb{1}(u_s^* = h, x_s^* = x_v)) \quad (2)$$

$$\psi_{vv'}(x_v, x_{v'}) = \exp(m_{vv'}\mathbb{1}(x_v \neq x_{v'})) \quad (3)$$

where $\mathbb{1}(.)$ is the indicator function, and $d(s,v)$ is the number of p-hops between s and v on the graph. w_{kh} is a measure of correlation on the cell selection between a node and the nodes k p-hops away with SINR index h, allowing emphases based on both distance and SINR. K is the largest range where two nodes are correlated. $m_{vv'}$ measures the degree of penalty if we allocate different cells to the neighboring nodes v and v'. The smaller $m_{vv'}$ is, the higher consistency one requires on cell selection between neighboring nodes. Substituting (2) and (3) into (1), the joint probability equals:

$$P_{V_i}(x_{V_i}) = \frac{1}{Z}\prod_{v\in V_i}\exp(\sum_{k=0}^{K}\sum_{h=1}^{H} w_{kh}\mathbb{1}_{kh}) \prod_{(v,v')\in E_i}\exp(m_{vv'}\mathbb{1}_{vv'}) \quad (4)$$

where $\mathbb{1}_{kh} = \sum_{\{s\in S_i|d(s,v)=k\}} \mathbb{1}(x_s^* = x_v, u_s^* = h)$ and $\mathbb{1}_{vv'} = \mathbb{1}(x_v \neq x_{v'})$.

B. CRF Inference

When the parameters $\theta = \{w_{kh}, m_{vv'}\}$ are determined, node v's serving cell, x_v^*, is then predicted as the cell with the largest marginal probability $P_v(.)$. This could be done by directly calculating the marginal distribution. However this incurs high computational complexity as there are many terms in the joint distribution $P_{V_i}(x_{V_i})$. We now propose to approximate $P_{V_i}(x_{V_i})$ with another probability distribution $Q_{V_i}(x_{V_i})$ such that $Q_{V_i}(x_{V_i}) = \prod_{v\in V_i} Q_v(x_v)$. Because of Q's special format, x_v's marginal distribution is approximated 'independently' from other nodes. Formally, the approximation is based on the following optimization:

$$\min_{Q_{V_i}(x_v)} \sum_{x_{V_i}} Q_{V_i}(x_{V_i}) \log(\frac{Q_{V_i}(x_{V_i})}{P_{V_i}(x_{V_i})}) \quad (5)$$

$$s.t.\ Q_{V_i}(x_{V_i}) = \prod_{v\in V_i} Q_v(x_v) \quad (6)$$

$$\sum_{x_v\in X} Q_v(x_v) = 1 \quad \forall v \in V_i \quad (7)$$

Note (5) is the KL-divergence between $Q_{V_i}(x_{V_i})$ and $P_{V_i}(x_{V_i})$.

We propose the following iterative update to determine Q:

$$Q_v^{t+1}(x_v) = \frac{1}{Z}\exp(\sum_{k=0}^{K}\sum_{h=1}^{H} w_{kh}\mathbb{1}_{kh} + \sum_{v'\in N(v)}\sum_{x_{v'}\in X} Q_{v'}^t(x_{v'})m_{vv'}\mathbb{1}_{vv'}) \quad (8)$$

$$Z = \sum_{x_v\in X} exp(\sum_{k=0}^{K}\sum_{h=1}^{H} w_{kh}\mathbb{1}_{kh} + \sum_{v'\in N(v)}\sum_{x_{v'}\in X} Q_{v'}^t(x_{v'})m_{vv'}\mathbb{1}_{vv'})$$

is the normalization constant to ensure Q^{t+1} is a valid probability distribution, and $N(v)$ is v's neighbors in G_i. The algorithm is shown in Algorithm 1. *Remark:* Iteration

Algorithm 1: Iterative Algorithm for Q

```
Input: x_s^*, μ_s^*, s ∈ S_i
Output: Q_v(x_v), v ∈ V_i
Initialize Q_v^{t=0}(x_v), v ∈ V_i
while converge criteria not satisfied do
    for v ∈ V_i do
        for x_v ∈ X do
            Calculate Q_v^{t+1}(x_v) based on equation (8)
        end
    end
    t = t + 1
end
Return Q_v^t(x_v) for x_v ∈ X, v ∈ V_i
```

Algorithm 2: Cell Selection Protocol (CRF-CSP)

```
Input: G_0, S_0, x_s^*, u_s^*, s ∈ S_0
Offline:
for each node i ∈ V_0\S_0 do
    Calculate Q_i(x_i), the marginal probability of cell selection at i by
      running Algorithm 1 on G_i.
end
Online:
while SINR received from the current cell is lower than SINR_TH do
    Given the drones location, find the closest grid point i'
    Predict the drone's top T cells based on Q_{i'}(x_{i'})
    Try to connect in order.
    If one is found successful, then stop trying.
    Else, fall back to scanning method as in traditional design.
end
```

(8) is determined by solving the optimization based on (5)'s Lagrangian and letting the partial derivative equal zero. The details are in the appendix. Note that (5) is optimized relative to a single variable $Q_v(x_v)$ in each iteration. Hence (5) is monotonically nonincreasing. We observe from our simulation experiments that Q converges with good accuracy within 7 iterations and approximates P very well in practice.

C. A protocol for cell selection based on CRF

Given the CRF framework, we propose a cell selection protocol (CRF-CSP in Algorithm 2) for a drone to determine its serving cell. Note that the offline portion (Lines 3-5) can be computed in a data center and updated dynamically as the new information comes from additional drone runs. The cell prediction result for each grid node in the air can be stored on board of a drone's memory, or transmitted from a BS when needed.

D. CRF Learning

Now we discuss the training procedure to determine the parameters for the CRF. Suppose a training set is given as $D = \{D_{i_1}, D_{i_2}, ..., D_{i_R}\}$, $1 \le r \le R$ where the r-th training data $D_{i_r} = \{x^*_{v \in V_{i_r}}, u^*_{v \in V_{i_r}}\}$ contains the optimal cell ids and SINRs for each node in G_{i_r}. Our objective is to find the parameters $\theta = \{w_{kh}, m_{vv'}\}$, $1 \le k \le K, 1 \le h \le H$ and $(v, v') \in E_{i_r}$ that maximize the posterior distribution $P(\theta|D)$. By Bayes' rule, we have:

$$\prod_{r=1}^{R} P(D_{i_r}|\theta)P(\theta) = P(D|\theta)P(\theta) \propto P(\theta|D) \tag{9}$$

where $P(\theta)$ is the prior distribution. We take the prior distributions of w_{kh} and $m_{vv'}$ as Gaussian with means $\mu_{w_{kh}}$, μ_m and standard deviations $\sigma_{w_{kh}}$, σ_m, respectively:

$$P(\theta) = \prod_{k=1}^{K}\prod_{h=1}^{H} \mathcal{N}(w_{kh}; \mu_{w_{kh}}, \sigma_{w_{kh}}) \prod_{(v,v')\in E} \mathcal{N}(m_{vv'}; \mu_m, \sigma_m)$$

From (9), we have $\frac{1}{R} logP(\theta|D) = \frac{1}{R}\sum_{r=1}^{R} \log P(D_{i_r}|\theta) + \frac{1}{R}\log P(\theta) + C_1$, where C_1 is a constant not related to θ. After substituting the expressions for $P(D|\theta)$ and $P(\theta)$, we have:

$$\frac{logP(\theta|D)}{R} = -logZ(\theta) + \frac{1}{R}\sum_{r=1}^{R}\Bigg(\sum_{v\in V_{i_r}}\sum_{k=0}^{K}\sum_{h=1}^{H} w_{kh} \sum_{\{s\in S_{i_r}|d(s,v)=k\}} \mathbb{1}(x^*_s = x^*_v, u^*_s = h) + \sum_{(v,v')\in E_{i_r}} m_{vv'}\mathbb{1}(x^*_v \neq x^*_{v'})\Bigg) - \sum_{k=1}^{K}\sum_{h=1}^{H}\frac{(w_{kh}-\mu_{w_{kh}})^2}{2R\sigma^2_{w_{kh}}} - \sum_{(v,v')\in E}\frac{(m_{vv'}-\mu_m)^2}{2R\sigma^2_m} + C_2 \tag{10}$$

where $Z(\theta)$ is the normalization constant for $P(D_{i_r}|\theta)$ and C_2 represents the terms not related to θ. To find $w_{kh}, m_{vv'}$ that maximize (10), the gradients can be derived as:

$$\frac{\partial \frac{1}{R} logP(\theta|D)}{\partial w_{kh}} = E_D\Big(\sum_{v\in V}\sum_{\{s\in S|d(s,v)=k\}} \mathbb{1}_{kh}\Big) - E_P\Big(\sum_{v\in V}\sum_{\{s\in S|d(s,v)=k\}} \mathbb{1}_{kh}\Big) + \frac{\mu_{w_{kh}} - w_{kh}}{R\sigma^2_{w_{kh}}} \tag{11}$$

$$\frac{\partial \frac{1}{R} logP(\theta|D)}{\partial m_{vv'}} = E_D(\mathbb{1}_{vv'}) - E_P(\mathbb{1}_{vv'}) + \frac{\mu_m - m_{vv'}}{R\sigma^2_m} \tag{12}$$

Here $E_D(.)$ is the empirical expectation over training set D and $E_P(.)$ is the expectation over $P(.|\theta)$. The details are omitted due to page limit; see [7] for a similar derivation.

We then apply gradient ascent to find a set of optimal parameters.

TABLE II
SIMULATION SETTINGS

Cell layout	57-cell, UMa, ISD=500m
Frequency	2.1 GHz carrier
Drone height	[50, 250] meter, uniform
BS	TX Power = 46dBm, tilting=10 deg
Shadowing	5dB with correlation distance 50m
path loss	free space

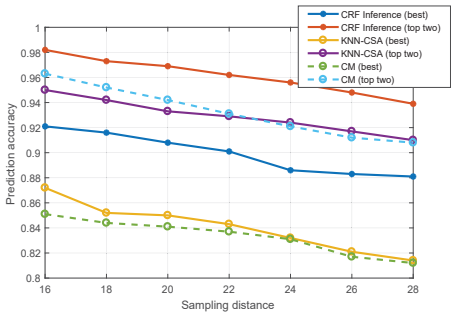

Fig. 4. Prediction accuracy of the algorithms

IV. SIMULATION-BASED ANALYSIS

We use a 57-cell system model based on 3GPP LTE cellular modeling assumptions to evaluate the design. The basic parameters are summarized in Table II. To capture the drone channels in the air, each BS is modeled with a realistic antenna pattern shown in Figure 2. We investigate the situation between 50-250 meters in the air, and the path-loss model is assumed to be free-space. A lognormal shadowing of 5dB with spatial correlation distance 50m is modeled. The physical distance d between the neighboring grid nodes is 2 meters in our graphical model, covering from 50m to 250m in the air.

As expected, one can show that the correlation between neighboring nodes' best cells gets higher as the distance gets closer. Since the cell selection for drone is a new research topic, it is hard to find the benchmark algorithm from the literature, therefore we devise two benchmark algorithms to compare with our design:

KNN-CSA: *Given a node v, consider all the sample nodes within K p-hops whose best serving cell is known. The majority vote among the sample nodes is the prediction.*

Cone Method (CM): *Build a cone and with apex v and a small apex angle α. Use the cone to scan the space in all possible directions. For each sample point s inside the cone, if its best serving cell m also falls inside the cone, with received power level (RSRP) at s, P^m_s, we then can approximate node v's RSRP from cell m by considering the distance difference between v and s to cell m, assuming the path loss formula is known (e.g. free-space). Then v will choose the serving cell with the highest RSRP. The intuition here is that the node s between v and a 'good' cell can serve as a reference point.*

To train the CRF model, 100 random subgrids were selected, assuming the best cell of each node within the subgrid is known. Each subgrid has a span of $a = b = c = 40m$. For prior distribution, we set $\mu_{w_{kh}} = \mu_m = 0.1$, and $\sigma_{w_{kh}} = \sigma_m = 1$ for all k, h. As the size of training data

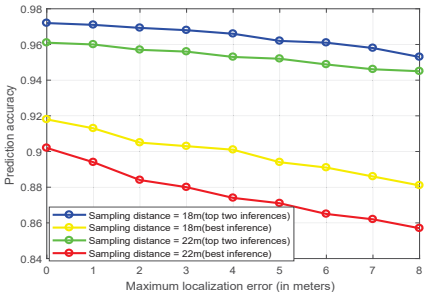

Fig. 5. Prediction accuracies with different localization errors

gets larger, the prior distribution will have little effect on the posterior distribution. We also fix $K = 10$ and $H = 5$ for 3. For inference, we randomly select 1000 locations and run CRF-CSP to infer their best cells.

Figure 4 shows the prediction accuracies of CRF-CSP, KNN-CSA and CM with different distances (in meters) between the neighboring sampling points. Here the parameters for CM and KNN-CSA are selected for each sampling density so that their prediction accuracies are highest. In addition to just predicting one single best cell, we also consider the probability of finding the best cell within the top two predictions. We notice that the prediction accuracy decreases for all the three algorithms as the sampling distance increases. The prediction accuracy of CRF is 4.7% and 4.9% higher on average than KNN-CSA and CM for the best inference result and 2.7% and 2.6% higher on average than KNN-CSA and CM for the top two best inference results.

Since CRF-CSP relies on the localization information to search the closest grid point, next we investigate the effect of GPS positioning error. In practice, the GPS localization error is usually smaller than 7.8m [14]. Figure 5 shows the changes on prediction accuracy with difference localization errors when the sampling distances are 18m and 22m. In this figure, the localization accuracy is interpreted in terms of the maximum localization error. Given the ground truth position of drone to be (x, y, z) and the maximum localization error equals δ, the position reported by GPS is a uniform random variable distributed in a ball centered at (x, y, z) with radius δ. We can see that prediction accuracy decreases very slowly with localization error. When the maximum localization error is 8m, the CRF prediction accuracies are still above 85%.

V. Concluding Remarks

In this paper we introduce a novel CRF based method for predicting a node's best serving cell, allowing the CRF framework to be used for wireless communications related applications. Our initial results demonstrate high accuracy and better performance compared to two simple heuristic methods. This suggests that the correlation inherent in the stationary aerial wireless environment can be exploited, but points to the need to use carefully designed tools. Our analysis is based on SINR without explicit modeling the interference; further analyses are needed for more complex deployment scenarios, and on integration with existing methods.

VI. Appendix

A. Derivation for the equation (8)

Proof. We solve this optimization problem by calculating its Lagrangian multiplier. Denote $\widetilde{\phi}_v(x_v) = \sum_{k=0}^{K} \sum_{h=1}^{H} w_{kh} \mathbb{1}_{kh}$ and $\widetilde{\psi}_{vv'}(x_v, x_{v'}) = m_{vv'} \mathbb{1}_{vv'}$. We have:

$$\log P_{V_i}(x_{V_i}) = -logZ + \sum_{v \in V_i} \widetilde{\phi}_v(x_v) + \sum_{(v,v') \in E_i} \widetilde{\psi}_{vv'}(x_v, x_{v'}) \tag{13}$$

Substitute the above expression into the objective function, we have:

$$\sum_{x_{V_i}} Q_{V_i}(x_{V_i}) \log(\frac{Q_{V_i}(x_{V_i})}{P_{V_i}(x_{V_i})}) = \sum_{x_{V_i}} Q_{V_i}(x_{V_i}) log(Q_{V_i}(x_{V_i})) + \sum_{x_{V_i}} Q_{V_i}(x_{V_i})(logZ - \sum_{v \in V_i} \widetilde{\phi}_v(x_v) - \sum_{(v,v') \in E_i} \widetilde{\psi}_{vv'}(x_v, x_{v'})) \tag{14}$$

Since $Q_{V_i}(x_{V_i}) = \prod_{v \in V_i} Q_v(x_v)$, the Lagrangian $L(x_{V_i}, \lambda_v)$, equals:

$$\sum_{v \in V_i} \sum_{x_v} Q_v(x_v) log(Q_v(x_v)) - \sum_{v \in V_i} \sum_{x_v} Q_v(x_v) \widetilde{\phi}_v(x_v) - \sum_{(v,v') \in E_i} \sum_{x_v} \sum_{x_{v'}} Q_v(x_v) Q_{v'}(x_{v'}) \widetilde{\psi}_{vv'}(x_v, x_{v'}) + logZ + \lambda_v (1 - \sum_{x_v} Q_v(x_v))$$

Take the derivative of $L(x_{V_i}, \lambda_v)$ w.r.t $Q_v(x_v)$, we get:

$$\frac{\partial L(x_{V_i}, \lambda_v)}{\partial Q_v(x_v)} = -\widetilde{\phi}_v(x_v) - \sum_{v' \in N(v)} \sum_{x_{v'}} Q_{v'}(x_{v'}) \widetilde{\psi}_{vv'}(x_v, x_{v'}) + logQ_v(x_v) + 1 - \lambda_v \tag{15}$$

In (15), we have $\frac{\partial \log Z}{\partial Q_v(x_v)} = 0$ since $\log Z$ is not related to $Q_v(x_v)$. By making $\frac{\partial L(x_{V_i}, \lambda_v)}{\partial Q_v(x_v)} = 0$ and $\frac{\partial L(x_{V_i}, \lambda_v)}{\partial \lambda_v} = 0$, we have:

$$Q_v(x_v) \propto exp(\widetilde{\phi}_v(x_v) + \sum_{v' \in N(v)} \sum_{x_{v'}} Q_{v'}(x_{v'}) \widetilde{\psi}_{vv'}(x_v, x_{v'})) \tag{16}$$

for all $x_v \in X$ and $\sum_{x_v \in X} Q_v(x_v) = 1$. Substitute the expression of node potential and edge potential to (17) and normalize the result, we get (8). □

References

[1] FAA, "Summary of small unmanned aircraft rule (Part 107)," https://www.faa.gov/uas/media/Part_107_Summary.pdf.

[2] https://utm.arc.nasa.gov/index.shtml

[3] RP-170779, "Study on enhanced support for aerial vehicles," 3GPP RSG RAN1 Meeting #75, Dubrovnik, Croatia, March 6-9, 2017.

[4] 3GPP, "Study on enhanced LTE support for aerial vehicles," 3GPP TR 36.777 (Release 15), 2018.

[5] J. Shi, et al. "Normalized Cuts and Image Segmentation", IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 22, no. 8, Aug. 2000

[6] M. Zissman. "Automatic language identification using Gaussian mixture and hidden Markov models." in Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 1993.

[7] Sai Qian Zhang, H.T. Kung, Youngjune Gwon. "InferBeam: A Fast Beam Alignment Protocol for Millimeter-wave Networking." https://arxiv.org/pdf/1802.03373.pdf

[8] 3GPP, R1-1718296, "Downlink interference mitigation for aerial vehicules," 3GPP TSG RAN WG1 #90B, Prague, 10/2017.

[9] 3GPP R1-1720053, "On Interference Mitigation schemes for DL, Study Item on Enhanced Support for Aerial Vehicles," 3GPP TSG RAN WG1 #91, Reno, 11/2017.

[10] Qualcomm Inc., "LTE unmanned aircraft systems trial report," May 2017.

[11] C.Bishop. Pattern Recognition and Machine Learning, Springer, 2006.

[12] B. Van der Bergh, A. Chiumento and S. Pollin, "LTE in the sky: trading off propagation benefits with interference costs for aerial nodes," IEEE Communications Magazine, vol. 54, no. 5, pp. 44-50, 2016.

[13] E. Teng, et al., "Aerial sensing and characterization of three-dimensional RF fields," in Proceedings of Second International Workshop on Robotic Sensor Networks, Seattle, 2015.

[14] GPS standard positioning service performance standard: http://www.gps.gov/technical/ps/2008-SPS-performance-standard.pdf