Optimal Path Construction with Decode and Forward Relays in mmWave Backhaul Networks

Yan Yan, *Qiang Hu* and *Douglas M. Blough*
School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, 30332

***Abstract*—In this paper, we consider the problem of constructing paths using decode and forward (DF) relays for millimeter wave (mmWave) backhaul communications in urban environments. Due to the large number of obstacles in urban environments, line-of-sight (LoS) wireless links, which are necessary for backhaul communication, often do not exist between small-cell base stations. To address this, some earlier works proposed creating multi-hop paths that use mmWave relay nodes with LoS communication between every pair of consecutive nodes to form logical links between base stations. We present algorithms, based on a novel widest-path formulation of the problem, for selecting decode and forward relay node locations in such paths. Our main algorithm is the first polynomial-time algorithm that constructs a relay path with a throughput that is proven to be the maximum possible. We also present variations of this algorithm for constrained problems in which: 1) each possible relay location can host only one relay node, and 2) minimizing the number of hops in the relay path is also an objective. For all of the proposed algorithms, the achievable throughput and numbers of relays are evaluated through simulation based on a 3-D model of a section of downtown Atlanta. The results show that, over a large number of random cases, our algorithm can always find paths with very high throughput using a small number of relays. We also compare and contrast the results with our earlier work that studied the use of amplify-and-forward (AF) relays for the same scenario.**

I. Introduction

The current cellular spectrum is experiencing difficulty in keeping up with the explosive growth of mobile data demand. MmWave communication, with its enormous amount of spectrum and multi-Gigabit-per-second (Gbps) data rates, is considered a strong candidate for broadband radio access and backhaul in 5G cellular networks. However, there are a number of differences between mmWave communications and lower-frequency communications, and so there are several challenges that need to be addressed before mmWave communications for 5G become a reality. Among these challenges are a high propagation loss and sensitivity to blockage of mmWave signals. Due to these issues, for mmWave in very high data rate use cases such as backhaul, the communication range could be limited to a few hundred meters or less. To both extend the range of communications and to deal with obstacles, the use of relays for mmWave communications has been proposed [1] [2] [3] [4].

The primary relaying strategies are amplify-and-forward (AF) and decode-and-forward (DF). In AF relaying, each relay node simply amplifies the received signal and forwards it to the next node without decoding and re-encoding it. In DF relaying, each relay node decodes the received signal, re-encodes it, and forwards it to the next node. Because AF relaying amplifies noise along with intended signal at each hop, while DF relaying eliminates the noise at each hop, the end-to-end performance of DF should be superior to that of AF [5]. Hence, in this paper, we focus primarily on the DF relay case. However, at the end of the paper, we do compare the DF relay results with existing work that studied AF relays.

The average error performance analysis of the multihop-based DF protocol is analyzed in [6]–[10]; however, these works only considered the performance of the protocol but didn't consider the path construction problem. A few works have considered the relay path construction problem with DF relays [3] [11] [12] previously. However, in [11] [12], the possible paths are given and the methods simply choose the best one, whereas in our work, possible relay locations are given and the goal is to *construct* an optimal path that goes through a subset of the given locations. The closest prior work to the work in this paper is our previous work in [3]. However, the algorithm of [3] focuses on very short relay paths with minimum or near-minimum number of relays. While the algorithm of [3] could be extended to find the overall maximum-throughput path, its time complexity grows exponentially with the length of the path and so it is impractical as a general solution to the optimal-throughput path construction problem. Thus, our algorithm presented herein is the first polynomial-time algorithm with provably optimal throughput performance for the DF relay selection problem in mmWave backhaul.

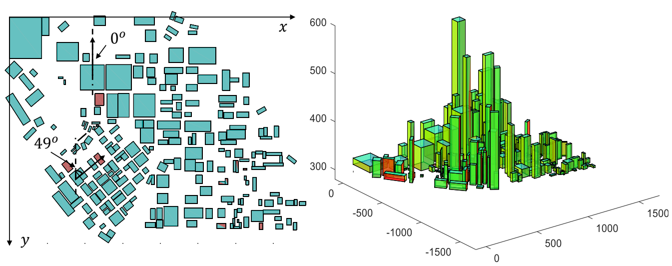
Our new path construction algorithms are based on a novel weighted directed graph model, where nodes in the graph represent possible pairs of consecutive links in a relay path. We transform the maximum-throughput relay path construction problem into a widest path problem in this graph. Our first algorithm constructs a provably optimal path, in terms of throughput, with no limitation on length and under the assumption that multiple relay nodes can be placed at the same location. We are also able to constrain the graph formulation to prevent multiple relays at the same location. While our solution to this constrained problem does not guarantee the optimal throughput under this constraint, we demonstrate through simulation that the throughput is extremely close to an upper bound, making the algorithm near-optimal. Finally, to account for latency considerations, we modify the algorithm to find high-throughput paths with a small number of relays. As validated by a large number of random cases in a realistic wireless network setting, this final algorithm finds paths with very high throughput and a small number of relays.

978-1-7281-4905-9/20/$31.00 ©2020 IEEE

II. Preliminaries

A. Network model

We consider mmWave wireless backhaul networks for dense small cell deployments in urban areas. Specifically, we focus on the case where a multi-Gbps backhaul logical link has to be constructed between a given pair of base stations (BSs), say s and d. If a relatively short line-of-sight (LoS) link between s and d exists, it usually can meet the demand. However, due to the well-known *blockage effect* on mmWave signals, LoS links will often be unavailable in obstacle-rich urban environments. To address this issue, we propose to deploy dedicated mmWave relays, which are used to form a multi-hop path between a BS pair with each hop being a LoS link. We assume a set of candidate locations for deploying relays is given. Since mmWave signals in 5G scenarios are highly directional and nodes could be placed at different heights in urban settings, we consider 3D effects



(a) (b)

Fig. 1. Building topology in downtown Atlanta. (a) Top view. (b) 3D view. [3]

Due to the short wavelengths, mmWave antenna arrays can have a large number of antenna elements, which allow very narrow beamwidth antennas to be deployed. The highly directional signals that result, together with the blockage effect in mmWave, the many obstacles in the urban environment, and the 3D nature of the network topologies, make interference a fairly rare occurrence in our mmWave backhaul setting. For these reasons, we ignore interference in our initial analyses and designs. However, as in our prior work on AF relaying [4], we can check the final constructed paths for interference and iterate the construction process, if necessary, to find interference-free paths in the small number of cases where the original path contains interference. The last set of results we present in Section VI demonstrate that this process finds interference-free paths with virtually no reduction in performance compared to the paths generated without considering interference.

B. Channel model and propagation assumptions

The channel estimation penalty is negligible when the signal to noise ratio (SNR) is high and the small scale fading is mild. In this situation, the maximum achievable rate is closely approximated by the capacity of a continuous time additive white Gaussian noise (AWGN) channel with the same SNR [13]. Our approach, presented in subsequent sections, produces very short LoS physical links, which will operate in the high SNR region and, therefore, the AWGN assumption is reasonable and link capacities can be estimated by Shannon's equation:

$$C = B\log_2(1 + \min\{\mathrm{SINR}, \mathrm{SINR}_{\max}\}) \ , \tag{1}$$

where B is the bandwidth of the channel and SINR is the signal to interference plus noise ratio at the receiver. In real networks, the data rate is determined by the coding and modulation schemes and has a maximum achievable value based on the technology deployed. The inclusion of $\mathrm{SINR}_{\max}$ reflects this reality (without it, the capacity can become infinitely high, which is clearly unrealistic). We also note that our approach ensures that there is no interference along the relay paths that are constructed. This, combined with the very short LoS links, will produce very stable SINR values (and therefore very stable data rates also), which can be used for path construction at network deployment time.[1]

SINR is defined by:

$$\mathrm{SINR} = \frac{P_r}{N_T + I} = \frac{P_r}{KTB + I} \ , \tag{2}$$

where P_r is the power of the transmitter's signal at the receiver side, N_T is the power of thermal noise, and I represents the interference power. For the thermal noise, K is Boltzmann's constant and T is the temperature.

The Friis transmission equation is used to calculate the receive power P_r:

$$P_r(d) = P_t \times G_t \times G_r \times (\frac{\lambda}{4\pi L})^{\eta} \times e^{-\alpha L} \ , \tag{3}$$

where P_t is the transmit power, G_t and G_r are antenna gains of the transmitting and receiving antenna, respectively, λ is the wavelength of the signal, L is the transmission distance, η is the path loss exponent, and α is the attenuation factor due to atmospheric absorption.

C. Throughput analysis for DF relaying path

Consider the simplest relay path consisting of a source station (s), a single relay station (r), and a destination station (d), as shown in Fig. 2. Let h_{sr} be the channel gain from s to r and h_{rd} be the channel gain from r to d.

For the data transmission along this two-hop path, r decodes the received signal y_{sr} to obtain a signal $\widehat{x}$, which it then forwards to d, LoS link from s to d is assumed to be unavailable in our obstacle-rich environments. The received signal y_{rd} at d is:

$$y_{rd} = h_{rd}\widehat{x} + n_t \tag{4}$$

[1] We do not consider temporary physical link blockages in this work, since we assume that base stations and relays are deployed on tops of buildings and temporary blockages will therefore be rare.

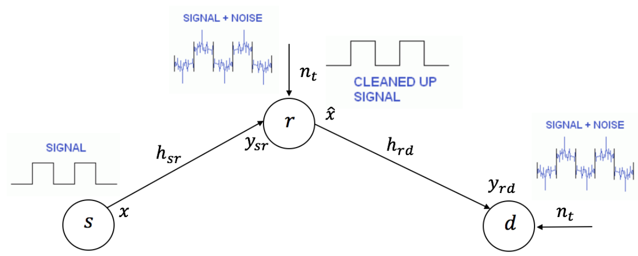


Fig. 2. Simplest DF relay path

It is observed that in case the interference is eliminated, the added noise in the received signal is removed by the decoding at a DF relay node, which then regenerates and re-encodes the signal to be forwarded to the next hop and eventually to the destination. Therefore, unlike the case in AF relay paths, where the maximum end-to-end throughput is determined by the end-to-end SNR [4], our previous work [3] shows that the maximum end-to-end throughput of a DF relay path is determined not only by the capacity of each individual link but also by the link schedule.

Assume the same traffic demand on each individual physical link along a path is D bits. The time demand f_i of link i can be obtained as,

$$f_i = D/C_i \tag{5}$$

From Theorem 1 in [3]: the minimum schedule length t in a multi-hop interference-free relay path is equal to the maximum demand sum of two consecutive links, i.e.,

$$t = \max_{1\leq i\leq |V_l|-1} f_i + f_{i+1} = \max_{1\leq i\leq |V_l|-1} D(1/C_i + 1/C_{i+1}), \tag{6}$$

where $|V_l|$ is the number of nodes in a relay path l. Equation (6) shows the minimum schedule length t corresponding to the maximum throughput. Thus, to obtain the maximum throughput T among all possible DF relay paths $\mathcal{L}$ between a pair of BSs, we have to find a path $l \in \mathcal{L}$ with the minimum schedule length $\min_{l\in\mathcal{L}}\{t_l\}$.

$$T = \max_{l\in\mathcal{L}}\{D/t_l\} = \max_{l\in\mathcal{L}}\{\min_{1\leq i\leq |V_l|-1}\{\frac{C_i C_{i+1}}{C_i + C_{i+1}}\}\} \tag{7}$$

The problem of finding a maximum-throughput DF relay path is mathematically equivalent to solving Eq. 7. In [3], the authors present a heuristic algorithm that finds a maximum throughput DF relay path with a very limited number of hops; however, due to its high computation complexity, it cannot be used to find a path with globally optimal throughput over paths with an arbitrary number of hops. In the next section, we propose a novel graph model that can be used to transform this problem into a well-known widest path problem in graph theory, such that a DF relay path with globally optimal throughput can be found in polynomial time.

III. Novel weighted directed graph model

In the widest path problem in graphs, the objective is to find a path between two designated vertices such that the weight of the minimum-weight edge in the path is maximum among all paths between the two vertices [15]. We note that this problem is similar in form to Eq. 7 that it maximizes some minimum quantity over the paths between two vertices. However, there is a clear gap between these two problems, because in the widest path problem, the weights are defined on the edges of the graph, while in Eq. 7, the weights are defined on pairs of links. To bridge this gap, we propose to transform the original graph, which consists of BSs, candidate relay locations, and possible physical links, into a novel weighted directed graph model where:

- the vertices represent 3-tuples of nodes that form a pair of links from the original graph,
- the edges connect each two vertices where the second link in the link pair of one vertex is the same as the first link in the link pair of the other vertex,
- the weight of each edge is defined as $\frac{C_i C_{i+1}}{C_i + C_{i+1}}$, where link i and link $i+1$ are the consecutive link pair specified by the edge's starting vertex.

An example of an original network graph is shown in Fig. 3 and its corresponding new weighted graph is shown in Fig. 4. In the original network graph, A represents the source BS, F represents the destination BS, B, C, D, and E represent possible relay locations, an edge between two nodes indicates that there is a LoS link between them, and the edge weights represent the capacities of the associated links. Next, we describe how the new weighted graph of Fig. 4 is constructed from the original graph of Fig. 3.

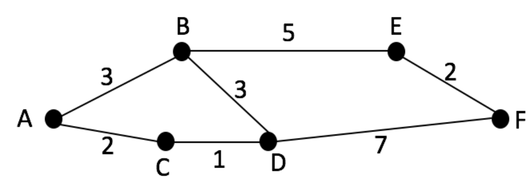


Fig. 3. Original Weighted Graph

As mentioned, the vertices of the new graph are 3-tuples representing a pair of consecutive links in the original graph. For example, the vertex ABE represents the link (A, B) followed by the link (B, E) in the original graph. The first two columns of Table I list the nodes and their neighbors from the original graph. From the nodes and their neighbors, we generate combinations of the form neighbor, node, neighbor to form the set of vertices of the new graph. The last column of Table I shows the vertices in the new graph generated in this way. Note that not all possible combinations become new vertices. This is because the paths that we consider start from the source BS (A) and end at the destination BS (F). Thus, no path links should end at the source or start from the destination. In the table, this means that A can only appear in combinations as the first element of the tuple and

F can only appear as the last element. Also, in the new graph, A is the source node and F is the sink node. For consistency in labeling, we denote those vertices by 00A and F00, respectively. Note that the generated vertices in the new graph are directional and different notations with the same 3 nodes of the original graph have different meaning, e.g., BDC and CDB both use edges (B, D) and (D, C) but in opposite order.

TABLE I
CREATE NEW VERTICES FOR WEIGHTED GRAPH

Nodes	Neighbors	New Vertices
B	A, D, E	ABD, ABE, DBE, EBD
C	A, D	ACD
D	B, C, F	BDC, BDF, CDB, CDF
E	B,F	BEF
A	B, C	00A
F	D,F	F00

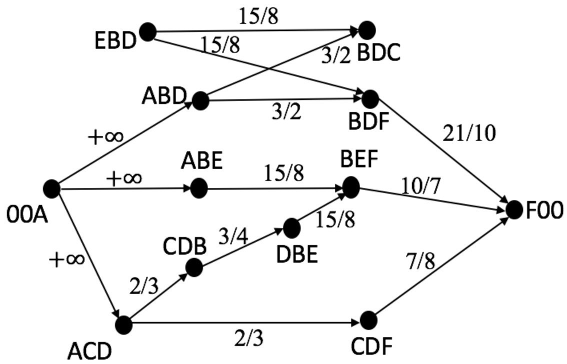


Fig. 4. New Weighted Directed Graph

As mentioned earlier, an edge exists between two vertices in the new graph if the second edge of the first vertex's 3-tuple is the same as the first edge of the second vertex's 3-tuple. As for the source and destination vertices, since they are in the special form of 00S and D00, we define that all vertices of the form SYZ are neighbors of 00S and, similarly, all vertices of the form WXD are neighbors of D00.

The last elements of the new weighted graph are its edge weights. Referring to Eq. 7, we set the weight of each edge in the new graph as the weight value of the edge's starting vertex, i.e. $\frac{C_i C_{i+1}}{C_i + C_{i+1}}$, where link i and link $i+1$ are the consecutive link pair specified by the edge's starting vertex. For example, the weight of the edge (EBD, BDC) is the weight of EBD, which (referring to the capacities in Fig. 3) is $\frac{5\times 3}{5+3} = 15/8$. We set the weight of any edge emanating from the source vertex to infinity, which allows those edges to be used in any path without constraining the path's throughput.

Once we have generated a new weighted directed graph from an original graph, we find the widest path in the new graph, which represents the maximum throughput path in the original graph, as shown by Eq. 7. The path in the new graph can be easily converted to the corresponding path in the original graph. For example, the widest path in the graph of Fig. 4 is 00A $\rightarrow$ ABD $\rightarrow$ BDF $\rightarrow$ F00 and this corresponds to the path A $\rightarrow$ B $\rightarrow$ D $\rightarrow$ F in the original graph.

IV. Basic DF Relay Path Construction Algorithms

A. Optimal-Throughput Path Construction Algorithm

Algorithm 1 Finding the DF relay path with maximum throughput using the method of widest path search

Input: $srcId$ (original source station), $dstId$ (original destination station), src (new source vertex), dst (new destination vertex), V_o (original vertices), V_n (new vertices), N_o (original neighbor map), N_n (new neighbor map)
Output: $OriginalPath$

Function NewGraphPath($Graph, Source, Destination$)
$width[src] = +Inf$
for each new vertex V in Graph **do**
  $width[V] = -Inf$;
  $prevIndex[V] = undefined$
Q = unvisited set in new Graph
while Q is not empty **do**
  u: a vertex in Q with largest $width[\,]$; remove from Q
  if $width[u] = -Inf$ **then**
    break;
  for each new neighbor n of u **do**
    $alt = max(width[n], min(width[u], width[n,u]))$
    if $alt > width[n]$ **then**
      $width[n] = alt$;
      $previous[n] = u$;
return $width; prevIndex$
Function newpathRecov($src, dst, prevIndex$)
$cur = dst$
while $cur \neq src$ **do**
  $newnodespath.add(cur)$;
  $cur = preIndex[cur]$;
$path.add(cur)$;
return $newpath$;
Function pathRecov($srcId, dstId, prevIndex, newpath$)
$path.add(dstId)$
for $0 < i < newnpath.size - 1$ **do**
  $path.add(newnpath[i][1])$;
$path.add(srcId)$;
return $path$;

Algorithm 1 shows the pseudo-code of our algorithm based on the widest path problem, which finds the maximum minimum weighted path in the new weighted graph, constructed as described in the previous section, and then converts the widest path in the new graph to an optimal-throughput path in the original graph. The following theorem states the optimality of Algorithm 1 and gives its time complexity:

Theorem 1. *For a given source, destination base station pair and a given set of possible relay locations, Algorithm 1 produces a decode-and-forward relay path with highest throughput in time $O(|E|^4)$, where $|E|$ is the number of line-of-sight links between different possible relay locations.*

Proof. The optimality of the constructed relay path follows directly from the graph construction of the previous section.

The time complexity of the widest-path algorithm is $O(|V|^2)$, where $|V|$ is the number of vertices of the modified graph. There is at most one vertex in the modified graph for each pair of edges in the original graph. Since $|E|$ is the number of edges in the original graph, $|V| \in O(|E|^2)$ and the overall complexity is then $O(|E|^4)$. □

In addition to producing a relay path with maximum throughput, Algorithm 1 also provides an upper bound throughput value against which the results of other non-optimal algorithms can be compared.

B. Path Construction Algorithm without Repeat Nodes

As we evaluated Algorithm 1 on different data sets, we discovered that it can produce relay paths in which the same relay location occurs multiple times. For example, if S is the source base station and D is the destination base station, then the path 00S → SIJ → IJK → JKL → KLI → LID → D00 in the modified graph is converted to path S → I → J → K → L → I → D in the original graph and this path passes through relay location I twice. This situation arises because the optimal algorithm constructs widest paths on *link pairs*, rather than simply on *links*. So, in the given example, link (S, I) pairs better with link (I, J) than it does with link (I, D), while link (L, I) pairs well with link (I, D). Because of these link pairings, relay location I appears twice in the path.

In some scenarios, it might be possible to deploy two relay nodes at the same location by physically separating their antennas and this situation would not cause any major problem. However, other scenarios might have physical constraints that prevent multiple relay nodes from being deployed at the same location. To handle scenarios in which each possible relay location can host at most one relay, we modify Algorithm 1 by not searching links in the modified graph that would produce a repeat node in the original graph. It turns out that this modification some times causes optimal paths to be missed in these new scenarios. However, in the next subsection, we demonstrate that this modified algorithm still produces paths that have very close to optimal throughput.

C. Simulation Results

In this part, we provide simulation results for the relay paths that are constructed by Algorithm 1 and the modified algorithm without repeat nodes. As mentioned earlier, we use an actual 3D topology of a section of downtown Atlanta to drive the simulations, as was done in [3] [4]. This topology contains 227 buildings higher than 5 meters, and for each building with a height between 20 and 200 meters, one of its rooftop corners is randomly picked as a candidate BS position and the diagonal corners of these rooftops are picked as possible relay locations (183 positions in total). We place one base station in each 200m×200m area and, therefore, we have 42 BS positions in total. We limit the maximum physical link distance to 300 meters, because longer LoS paths rarely exist in a dense urban environment due to signal attenuation and blockages.

100 BS pairs are randomly chosen for each separation in the meter range of $[20, 200)$, $[200, 400)$, $[400, 600)$, $[600, 800)$, and $[800, 1000)$. The fixed parameter values mentioned in previous equations are shown in Table II. Due to the short LoS links used in the backhaul and the high SINR at the receiver, we ignore the relatively small random attenuation due to shadowing effects in our analysis. However, implementation loss (5dB), noise figure (5dB), and heavy rain attenuation (10dB/km) are considered in the analysis by subtracting an additional link margin $L_m = 10\text{dB} + 10\text{dB/km} \times d$ when calculating the received power. As mentioned earlier, these values are the same as those used in our earlier works [3] [4], in order to facilitate comparison with these works.

TABLE II
PARAMETERS OF SIMULATION ENVIRONMENT

B	2.16 GHz	P_t	1 W	G_t, G_r	21.87 dBi
λ	5mm	η	2.0	SINR_{max}	50 dB
L_m	10 dB	α	16 dB/km		

We compare three different cases. First, we use Algorithm 1 to find an optimal-throughput relay path that might contain multiple occurrences of the same relay location. Second, we use Dijkstra's algorithm to find a path with the minimum *number* of hops, which is a simple best-path heuristic. Third, we use the modified algorithm to find a relay path where each possible relay location occurs at most once.

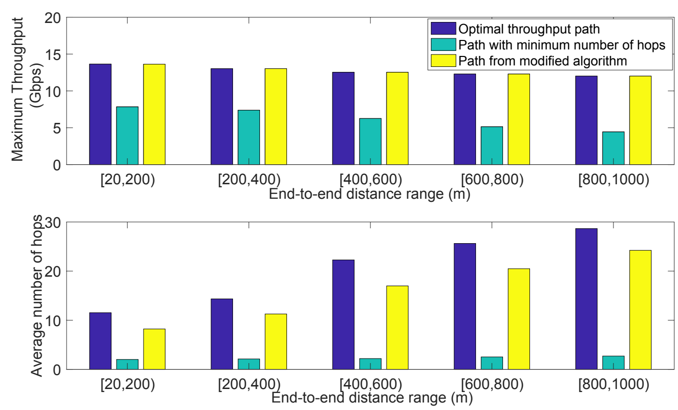


Fig. 5. Average throughput and number of hops among constructed paths with and without repeated nodes

Fig. 5 compares the average throughput and the average number of hops that are produced for these three cases. Note that the optimal throughput produced by Algorithm 1 is 2–3 times higher than that achieved by minimum-hop paths. The price that is paid for optimality, however, is a significant increase in the number of hops (and therefore the number of deployed relays) in the paths. For example, in the most extreme case (maximum BS separation), the average number of hops needed for the optimal-throughput path is about 29, while minimum-hop paths use an average of only around 4 hops.

It is interesting to note that the modified algorithm that does not allow repeat nodes produces an average throughput that is very close to that of the optimal algorithm. Thus, while the modified algorithm is sub-optimal in some cases, in practice it is very close to optimal. Eliminating repeat nodes reduces the lengths of the paths somewhat, but the number of hops is still quite high on average, e.g. around 24 for the maximum separation case.

The above results show that our algorithms are able to find paths with very high throughput. However, these paths can have a high cost in terms of the number of relays deployed. With a large number of relays, network stability, reliability, and delay will all be negatively affected. In many situations, achieving relatively high throughput while using a smaller number of relays might be preferable. In the next section, we consider the path construction problem where a throughput requirement is given and the goal is to minimize the number of relays while achieving the required throughput.

V. Minimum Hop Relay Path Construction With Throughput Constraint

Given a target percentage of maximum throughput, e.g. 90%, our throughput-constrained path construction algorithm consists of the following steps:

1) Use Algorithm 1 to find the maximum throughput value, and set the throughput threshold to be the specified percentage of the obtained value.
2) Prune all edges whose weights are smaller than the threshold value from the modified weighted graph.
3) Use Dijkstra's algorithm to find a minimum-hop path in the pruned graph.

Any path in the modified graph that contains an edge with weight below the target throughput threshold will not meet the target. Furthermore, once all such edges are pruned from the modified graph, *all* remaining paths meet the throughput target. Therefore, we construct a minimum-hop path from the pruned graph as the best path to satisfy the minimum throughput requirement.

As an example, we set 90% of the maximum throughput as the threshold value and compared the throughput-constrained algorithm to the algorithm from the previous section that maximizes throughput while constructing paths without repeat nodes. Fig. 6 shows that, for all base station separation ranges, the number of hops is reduced significantly with the throughput-constrained algorithm while only sacrificing 10% of the throughput. For example, in the 800m to 1000m BS separation case, the average number of hops is reduced from about 24 to about 9, which represents a substantial savings in the number of relays.

VI. Interference Considerations

As mentioned earlier, the results presented in the previous sections assume that interference does not occur due to the narrow beamwidths of mmWave antennas, blockages, and 3D effects. In the results presented in this section, we first ran the algorithm of Section IV-B with at most one node per relay location and we then checked the resulting paths for interference, assuming an antenna beamwidth of 11^o. If interference was found, we iterated the algorithm by removing

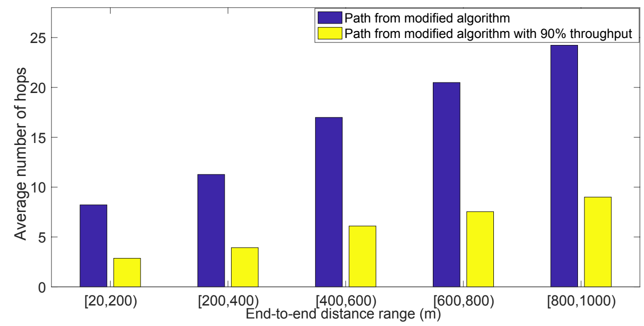


Fig. 6. Number of hops required for best paths without repeated nodes and 90% of maximum throughput paths

an interfering link and re-running it. In this way, we were able to find interference-free paths in all cases.

Figure 7 compares the interference-free paths with the paths generated under the assumption of no interference. The results show that interference-free paths can be found with virtually the same performance (throughput and number of hops) as the paths in which interference is ignored.

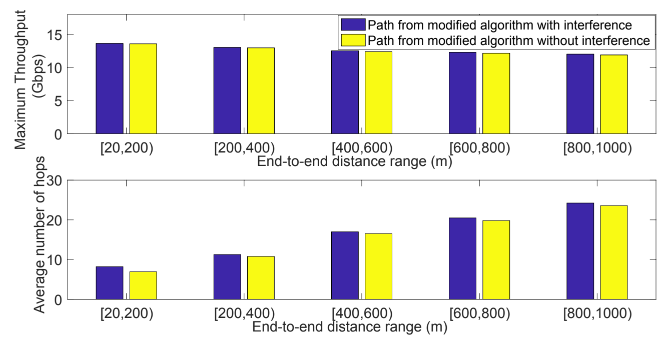


Fig. 7. Comparison between maximum throughput path (assuming no interference) and maximum throughput interference-free path

VII. Comparison between DF and AF

In this section, we compare two different kinds of fixed relaying protocols that are defined for wireless cooperative communications in mmWave backhaul situation. In fixed relaying protocols, we allow the relay either to amplify its received signal, maintaining a fixed average transmit power, or to decode and re-encode the received signal and then forward it to the destination. These two relaying strategies are known as amplify-and-forward and decode-and-forward, respectively. The main advantage of the DF strategy is that it eliminates the noise at relay nodes. However, the DF strategy has higher complexity and delay at each node due to modulation, demodulation, encoding, and decoding. The AF strategy is simpler and can, therefore, be implemented with lower cost. However, the AF strategy is prone to noise propagation effects because the relay node also amplifies the noise when the retransmitted signals are amplified. The noise amplification problem can degrade the signal quality.

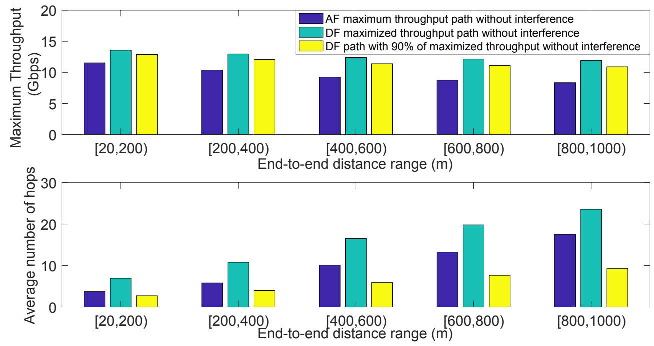


Fig. 8. Average maximum throughput and number of hops among all available paths for DF and AF protocol

Our prior work solved the problems considered herein for AF relay paths [4] using a simpler modification of Dijkstra's algorithm that worked for the AF setting. Fig. 8 compares the paths selected by that algorithm for the AF case with the DF algorithm for maximized throughput without repeat nodes.[2] In both cases, we added the interference checking step at the end and found new paths without interference for the small number of interference cases that occurred. The maximized throughput DF algorithm has 10–30% higher throughput than the AF algorithm but it also requires significantly more hops. However, if we compare the DF paths with 90% of maximized throughput to the AF paths, we find that the DF paths still have larger throughput than the AF paths and they are also shorter. This indicates that to find a relay path in the same environment, DF has better performance than AF both in terms of throughput and number of hops. Of course, as mentioned earlier, DF relays are more complex and, therefore, have higher cost than AF relays. Nevertheless, the performance gains of DF relays are quite significant.

VIII. Conclusion

In this paper, we have developed a novel widest-path formulation of the problem of finding high-throughput paths using decode-and-forward relays to support mmWave backhaul links. This novel formulation allowed us to develop a series of algorithms, the first of which has provably optimal throughput, that solve several variations of the problem. This represents the first polynomial-time algorithm for this problem with probably optimal throughput. Simulation results verified that all of our algorithms achieve very high throughputs and our final algorithm does so while using a small number of relays. We also demonstrated that relay paths using DF relays significantly outperform those with AF relays, both in terms of throughput and number of hops.

Acknowledgement

This research was supported in part by the National Science Foundation through Award CNS-1813242.

[2]For the AF algorithm, there is no benefit to having multiple relays in the same location, so we used the DF algorithm without repeat nodes for a fair comparison.

References

[1] Belbase, K., Zhang, Z., Jiang, H. and Tellambura, C., ”Coverage analysis of millimeter wave decode-and-forward networks With best relay selection”, *IEEE Access*, Vol. 6, pp. 22670–22683, 2018.

[2] Biswas, S., et al., ”On the performance of relay aided millimeter-wave networks”, *IEEE Journal of Selected Topics in Signal Processing*, 10.3, pp. 576–588, 2016.

[3] Hu, Q. and Blough, D.M., ”Relay selection and scheduling for millimeter wave backhaul in urban environments”, *Proc. IEEE Int'l Conf. on Mobile Ad Hoc and Sensor Systems*, pp. 206–214, 2017.

[4] Yan, Y., Hu, Q., and Blough, D.M., ”Path selection with amplify and forward Relays in mmWave backhaul networks”, *Proc. IEEE Int'l Symp. on Personal, Indoor, and Mobile Radio Communications*, 2018.

[5] Saleh, A.B., Redana, S., Raaf, B., Riihonen, T., Hamalainen, J. and Wichman, R., ”Performance of amplify-and-forward and decode-and-forward relays in LTE-advanced”, *IEEE Vehicular Technology Conference Fall*, 2009.

[6] Bao, V. N. Q. and Kong, H. Y., ”Error probability performance for multihop decode-and-forward relaying over Rayleigh fading channels”, *Proc. Int'l. Conf. Adv. Commun. Technol.*, Gangwon-Do, Korea, pp. 1512–1516, Feb. 2009.

[7] Morgado, E., et al., ”End-to-end average BER in multihop wireless networks over fading channels”, *IEEE Trans. Wireless Commun.*, Vol. 9, no. 8, pp. 2478–2487, Aug. 2010.

[8] Dhaka, K. , Mallik, R. K., and Schober, R., ”Performance analysis of decode-and-forward multi-hop communication: A difference equation approach”, *IEEE Trans. Commun.*, Vol. 60, no. 2, pp. 339–345, Feb. 2012.

[9] Huang, G., Wang, Y., and Coon, J., 2011, August. ”Performance of multihop decode-and-forward and amplify-and-forward relay networks with channel estimation”, *Proc. IEEE Pacific Rim Conference on Communications, Computers and Signal Processing,* pp. 352–357, 2011.

[10] Muenthetrakoon, W. and Khutwiang, K., ”SER of multi-hop decode and forward cooperative communications under Rayleigh fading channel”, *Proc. Int'l. Conf. Intell. Syst., Model., and Simul.*, pp. 318–323, 2011.

[11] Bhatnagar, M.R., ”Performance analysis of a path selection scheme in multi-hop decode-and-forward protocol”, *IEEE Communications Letters*, 16(12), pp. 1980–1983, 2012.

[12] Bhatnagar, M.R., Mallik, R.K., and Tirkkonen, O., ”Performance evaluation of best-path selection in a multihop decode-and-forward cooperative system”, *IEEE Transactions on Vehicular Technology*, 65(4), pp. 2722–2728, 2016.

[13] Du, J., Onaran, E., Chizhik, D., Venkatesan, S. and Valenzuela, R.A., ”Gbps user rates using mmWave relayed backhaul with high-gain antennas”, *IEEE Journal on Selected Areas in Communications*, 35(6), pp.1363-1372.

[14] Marzi, Z., Madhow, U. and Zheng, H., ”Interference analysis for mm-wave picocells”, *IEEE Global Communications Conference (GLOBECOM)*, pp. 1-6, 2015.

[15] Wang, Z. and Crowcroft, J., ”Bandwidth-delay based routing algorithms”, *Proceedings of GLOBECOM*, Vol. 3, 1995.