On Low-Power Multiple Unicast Network Coding Over a Wireless Triangular Grid

Sukwon Kim, Michelle Effros, and Tracey Ho
Department of Electrical Engineering
California Institute of Technology
Pasadena, CA, USA 91125
Email: {sukwon, effros, tho}@caltech.edu

***Abstract*— Power saving is an important issue in wireless networks. In this paper, we consider minimizing the power consumption in a network coded wireless network with multiple unicast sessions. We consider a simple network coding strategy termed "reverse carpooling", which uses only XOR operations at each node. This can reduce the number of transmissions and the corresponding power consumption. We investigate the use of this scheme on a wireless triangular grid network. First, we propose two polynomial time algorithms that approximately minimize the number of transmissions for unicasts with two and three source-destination pairs,respectively. We extend them to obtain a greedy algorithm for the general problem with an arbitrary number of source-destination pairs. We show by simulations that the algorithm reduces the power consumption significantly for multiple unicasts on a wireless triangular grid.**

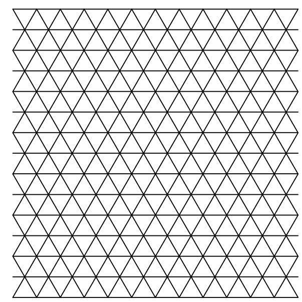

Fig. 1. The nodes of the network lie on the vertices of a triangular lattice.

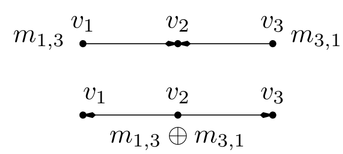

Fig. 2. Network coding provides an energy saving. For given two pairs unicasts, 4 transmissions are required without network coding. But,with network coding, 3 transmissions are required.

I. Introduction

Network coding is an interdisciplinary field of information theory and coding theory and is needed in general to attain the maximum information flow in a network. In network coding, routing is generalized by allowing each node to combine a number of the packets it has received to generate the output [1]. It is shown in [3,4] that network coding can be used for power saving in wireless networks, which is an important advantage over traditional routing. In [4], the minimum energy-per-bit for multicasting with network coding is obtained by solving a linear program. Network coding can also offer significant energy savings for the cases of some specific topologies [6]. Effros *et al.* [2] present simple XOR-based coding strategies called "reverse carpooling" and "star coding" in order to minimize the energy consumption by multiple unicasts on a shared wireless grid network. A recursive algorithm that employs a dynamic programming argument for optimizing the power consumption using above strategies is given in [2]. However, the complexity of this algorithm makes its solution impractical for large networks.

In this paper, we consider a minimization of the power consumption using reverse carpooling strategy for multiple unicasts problem on a wireless network and develop simple heuristic algorithms to obtain approximately optimal solutions in polynomial time. A wireless triangular grid is used as a simplified network model, as shown in Fig. 1. Each node corresponds to a vertex of a triangular grid, and it can broadcast information only to its six neighbor nodes. Each node directly receives all transmissions sent by its six neighbors. Thus there is direct communication only along edges connecting a node and its neighbor in graph. Fig. 2 illustrates the XOR based network coding operation we use. Node v_1 sends a packet m_{13} to node v_3 and node v_3 sends a packet m_{31} to node v_1. Without network coding, four transmissions are required. Each source transmits its packet to v_2, and v_2 transmits each packet in turn to its intended sink. With network coding, only three transmissions are required. Nodes v_1 and v_3 transmit m_{13} and m_{31} to v_2; node v_2 takes the bitwise binary sum of m_{13} and m_{31} and then broadcasts it. This strategy is generalized to information exchange between two nodes separated by arbitrarily many hops, in [5]. The term reverse carpooling is used in [2] to refer to the strategy of routing unicast flows to create path segments where such coding can occur. The upper bound of coding gain, which is the ratio between the number of transmissions required by non-coding approach to the minimum number of transmissions using reverse carpooling, is 2[1], and it is achievable by long paths and long sequence of packets. We propose two polynomial time heuristic algorithms which find approximate minimum energy solutions for two unicast sessions and three sessions, respectively. Moreover, based on the algorithm for two unicast sessions problem, we develop a greedy algorithm to solve the general problem with an arbitrary number of source-destination pairs. From simulations, it is shown that our greedy algorithm achieve significant coding gain on a wireless triangular grid.

Section II introduces the system model and the definitions. We present approximate optimal polynomial time algorithms for two unicast sessions problem in Section III and for three unicast sessions problem in Section IV. We propose a greedy algorithm for the general problem in Section V. Section VI gives our experimental results. Section VII concludes this paper.

Notation: Caligraphic letters are used for sets. O notation is employed to describe an asymptotically tight bound.

II. Preliminaries

A. System Model

1) Network: We define a *triangular grid* $\mathcal{G}$ = $(\mathcal{V},\mathcal{E})$ as the set of vertices $\mathcal{V}$ = $\{a(1{,}0)$ + $b(\frac{1}{2},\frac{\sqrt{3}}{2})$: $a,b \in \mathbb{Z}\}$ where $\mathbb{Z}$ denotes the set of integer and the set of directed edges $\mathcal{E}$= $\{(v,v') : \|v - v'\| = 1\}$ where for any $v,v' \in \mathcal{V}$, (v,v') denotes the arc connecting v and v'. Thus each node has six incident and six outgoing edges, each of length 1. The head and tail of edge $e = (v_i,v_j)$ are denoted by v_j = head(e) and v_i = tail(e), respectively. Together, the edges in $\mathcal{E}$ form lines at angles 0, 60°, and 120° from the horizontal, as shown in Fig. 1; we call these lines grid lines. A *path* is an ordered list of connected edges. Precisely, for any path $P = (e_1,e_2,..,e_k)$, we require $e_1,e_2,..,e_k \in\mathcal{E}$ and head(e_i) = tail(e_{i+1}) for $1 \leqslant i \leqslant k-1$. For any $1\leqslant i \leqslant j \leqslant k$, we call $P' = (e_i,e_{i+1},..,e_j)$ a sub-path of $P = (e_1,e_2,..,e_k)$ and write $P' \subseteq P$. When tail(e_i) = head(e_j) for some $i\leqslant j$, we call sub-path $P' = (e_i,e_{i+1},..,e_j)$ a self-loop. We restrict our attention to paths without self loops; Lemma 1 in Section II-B shows that for our purposes, there is no loss of generality in this restriction. We use $l(P)$ = $\sum_{e\in P}\|e\|$= $|P|$ to denote the length of path P. For any distinct vertices v,$v'\in\mathcal{V}$, we use $\mathcal{P}(v,v')$ to denote the set of all paths from v to v' in $\mathcal{G}$, $SP(v,v') = \arg\min_{P\in\mathcal{P}(v,v')}\{l(P)\}$ to denote the shortest path from v to v', and $d(v,v') = l(SP(v,v'))$ to denote the length of the shortest path.

2) Unicast: In a unicast session, a single source vertex $s \in \mathcal{V}$ transmits information to a single destination vertex $t \in \mathcal{V}$. In this paper, we consider multiple unicast sessions on a shared triangular grid. We specify a multiple unicast problem by describing the source and the destination for each unicast. For example, U= $\{(s_1,t_1)$, (s_2,t_2),..., $(s_n,t_n)\}$ is an n-unicast problem.

[1]We only investigate the effect of network coding on the networking layer. Unlike [5], the benefit of network coding on the multiple access layer is not considered.

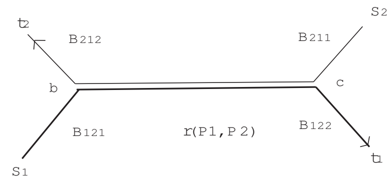

Fig. 3. Reverse carpooling solution of two unicast sessions with one reverse carpooling segment and four branches.

3) Reverse carpooling: Given a multiple unicast problem U= $\{(s_1,t_1)$, (s_2,t_2),...,$(s_n,t_n)\}$, a candidate solution S= $\{P_1,....,P_n\}$ is a list of paths such that $P_i \in P(s_i,t_i)$ for each i. For any edge $e = (v,v')\in\mathcal{E}$, we use e^R = (v',v) to denote the reversal of edge e. Likewise, for any path P = $(e_1,e_2,..e_k)$, we use P^R= $(e_k^R,e_{k-1}^R,...,e_1^R)$ to denote the reversal of path P. In candidate solution S, the opportunity to apply reverse carpooling arises when two paths, say P_i and P_j, contain sub-paths $P_i' \subseteq P_i$ and $P_j' \subseteq P_j$ satisfying $(P_i')^R$ = P_j'. Such a sub-path is called a reverse carpooling segment. Note that reverse carpooling may not actually occur between sub-paths P_i' and P_j' if one of them is involved in reverse carpooling with another sub-path. Since paths P_i and P_j may reverse carpool along multiple non-consecutive segments, we use $K(P_i,P_j)$ to denote the number of reverse carpooling segments shared by P_i and P_j and $r_k(P_i,P_j)$ to denote the kth sub-path shared by P_i and P_j. If $K(P_i,P_j) = 1$, we simplify our notation to $r(P_i,P_j)$ = $r_1(P_i,P_j)$. The sub-paths are numbered according to the order in which they appear in the first path (path P_i in $r_k(P_i,P_j)$). Each sub-path is as long as possible, and the sub-paths are disjoint.

To make these definitions precise, let $P_i = (e_1^{(i)},...,e_{l(P_i)}^{(i)})$ and $P_j = (e_1^{(j)},...,e_{l(P_j)}^{(j)})$. The following discussion defines t_k and $l_k = l(r_k(P_i,P_j))$ to be the start point (in P_i) and length, respectively, of $r_k(P_i,P_j)$ (provided that P_i and P_j have at least k reverse carpooling segments).

Initialize $t_0 = 0$, and $l_0 = 1$. Then for each subsequent $k\geqslant 1$ for which $t_{k-1}+l_{k-1}\leqslant l(P_i)$, let

$$\begin{aligned} t_k &= \min[\{n \in \{t_{k-1}+l_{k-1},...,l(P_i)\} : e_n^{(i)R} \in P_j\} \\ &\cup \{l(P_i)+1\}] \end{aligned}$$

If $t_k\leqslant l(P_i)$, let

$$l_k = \min\{n \in \{1,...,l(P_i)-t_k\} : e_{t_k+n}^{(i)R} \notin P_j\}.$$

Then $r_k(P_i,P_j)$ = $(e_{t_k}^{(i)},...,e_{t_k+l_k-1}^{(i)})$. We define branches B_{ijk} as B_{ijk} = $(e_{t_{k-1}+l_{k-1}}^{(i)},...,e_{t_k-1}^{(i)})$ if $t_k>$ $t_{k-1}+l_{k-1}$, and $B_{ijk} = \phi$ otherwise.

If $t_k>l(P_i)$, then P_i and P_j share fewer than k reverse carpooling segments, B_{ijk} = $(e_{t_{k-1}+l_{k-1}}^{(i)},...,e_{l(P_i)}^{(i)})$, and the process stops.

Fig. 3 shows an example with one reverse carpooling segment and four branches.

4) Network cost: The cost of a candidate solution is the energy consumed in a wireless network that transmits a single information stream along each path $P_i\in S$. When n = 1 (a

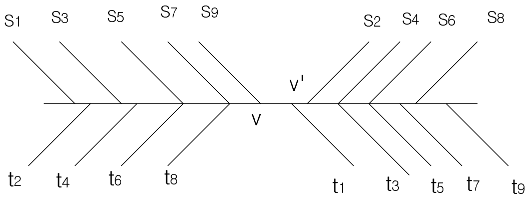

Fig. 4. Illustration of cost of edge (v, v'): 5 unicasts use edge (v, v') and 4 unicasts use edge (v', v).

single unicast session), we estimate the cost of candidate solution $S = \{P_1\}$ by the number of transmissions required to send a single packet from s_1 to t_1 along path P_1. Thus the cost of P_1 is the number of edges in path P_1, which equals $l(P_1)$. When $n>1$, the opportunity for reverse carpooling may arise. We approximate the cost saved using reverse carpooling by counting the link between nodes v and v' only once for each time we apply reverse carpooling along (v, v') and (v', v). For candidate solution S, we use $R(S, e)$ to denote the number of reverse carpooling opportunities along edge e using solution S. Thus

$$R(S, e) = \min \left\{ \sum_{P \in S} 1(e \in P), \sum_{P \in S} 1(e^R \in P) \right\}.$$

Similarly, we use $F(S, e)$ to denote the remainder of transmissions along edge e. Thus,

$$F(S, e) = \left(\sum_{P \in S} 1(e \in P) \right) - R(S, e).$$

We approximate the cost across each edge $e \in E$ using candidate solution S as

$$C(S, e) = \left(F(S, e) + \frac{1}{2} R(S, e) \right)$$

and the cost of S as

$$C(S) = \sum_{e \in E} C(S, e).$$

Fig. 4 gives an example. Edge(v, v') appears in five paths $(P_1, P_3, P_5, P_7, P_9)$. Edge (v', v) appears in four paths (P_2, P_4, P_6, P_8). Thus $R(S, (v, v'))$ = $R(S, (v', v))$ = 4, $F(S, (v, v'))$ = 1, $F(S, (v', v))$ = 0, and the combined contribution of edges (v, v') and (v', v) to our estimated cost $C(S)$ is $F(S, (v, v')) + \frac{1}{2} R(S, (v, v')) + F(S, (v', v)) + \frac{1}{2} R(S, (v', v))$ = 5.

The difference between the approximate cost $C(S)$ and the actual number of transmissions for a candidate solution S is at most the number of reverse carpooling segments. We show in Sections III and IV that in an n-unicast problem with $n \leqslant 3$, the number of reverse carpooling segments is at most $\binom{n}{2}$. Fig. 5 illustrates the difference between cost and number of transmissions. We approximate the number of transmissions by the cost $C(S)$ throughout the paper.

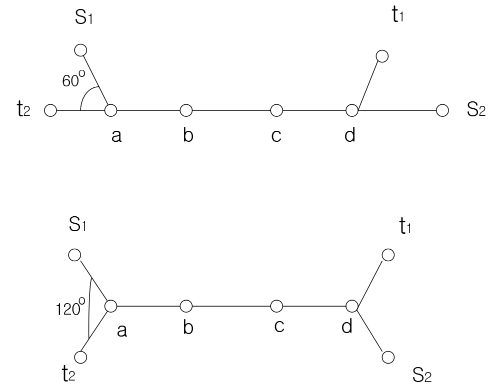

Fig. 5. Reverse carpooling is possible at nodes a,b,c,and d in first network. In the second network, reverse carpooling is not possible at nodes a, and d because t_2 cannot overhear the transmission from s_1 and t_1 cannot overhear the transmission from s_2. Thus the first network requires 6 transmissions, while the second network requires 8 transmissions. In both networks, we approximate the cost of S as $C(S)$ = 7. In general, for a reverse carpooling segment of n links shared by two unicast sessions, the actual number of transmissions is $n \pm 1$ while the approximate cost is n.

B. *Self-loops*

Lemma 1: Given a n-unicasts problem $U = \{(s_1, t_1), .., (s_n, t_n)\}$, there exists a minimal cost solution $S^* = (P_1, .., P_n)$ that contains no self-loops.

Proof: Suppose that P_1 has a self-loop $P_{11} = (e_i^{(1)}, .., e_j^{(1)}) \subseteq P_1$. We define an alternative solution $S' = (P_1', P_2, .., P_n)$ with $P_1' = P_1 - P_{11}$. Then,

$$C(S') \leqslant C(S' \cup P_{11}) = C(S^*) \leqslant C(S') + l(P_{11}).$$

By the optimality of S^*, $S' = S^*$ and we can remove P_{11} without increasing cost. By repeating this argument, we can remove all self-loops while maintaining the optimal cost $C(S^*)$. ■

III. TWO UNICAST SESSIONS PROBLEM

This section presents a polynomial-time algorithm that finds an optimal cost solution for two unicast sessions (s_1, t_1), (s_2, t_2) on a triangular grid.

Lemma 1 and Theorem 1 help to characterize the form of an optimal solution for two unicast sessions.

Theorem 1: Given a two-unicast problem $U = \{(s_1, t_1), (s_2, t_2)\}$, if $S^* = (P_1, P_2)$ is a minimal cost solution, then $K(P_1, P_2) \leqslant 1$, i.e., there is at most one reverse carpooling segment shared by P_1 and P_2.

Proof: Please see [7]. ■

For each $v \in \{a, b, c\}$ and $\theta \in \{0°, 60°, 120°\}$, let $g_{v,\theta}$ denote the grid lines at angle θ through vertex v. We write $g \in \triangle abc$ to specify that grid line g is contained in one or more of the angles in $\triangle abc$ and for each $\theta \in \{0°, 60°, 120°\}$ and $v \in \{a, b, c\}$, define

$$G_\theta = \{g_{v',\theta} : v' \in \{a, b, c\}, g_{v',\theta} \in \triangle abc\}.$$

$$G^v = \{g_{v,\theta'} : \theta' \in \{0°, 60°, 120°\}, g_{v,\theta'} \in \triangle abc\}.$$

Lemma 2: Given any $a, b, c \in \mathcal{V}$ that are not collinear, $|G_\theta| = 1$ for each $\theta \in \{0°, 60°, 120°\}$, and $|\cup_{v\in\{a,b,c\}} G^v| = |\cup_{\theta\in\{0°,60°,120°\}} G_\theta| = 3$.

Proof: Please see [7]. ■

Theorem 2: For any $a, b, c \in \mathcal{V}$ that are not collinear, we can find the largest set

$$\begin{aligned} P &= \{x \in \mathcal{V} : d(a,x) + d(b,x) + d(c,x) \\ &\leqslant d(a,y) + d(b,y) + d(c,y), \forall y \in \mathcal{V}\} \end{aligned}$$

in $O(1)$ time.

Outline of proof: By Lemma 2, a triangle $\triangle abc$ contains exactly three grid lines $\cup_{v\in\{a,b,c\}} G^v$. We show that the desired set contains all vertices in a triangle formed by $\cup_{v\in\{a,b,c\}} G^v$. The full proof is in [7]. ■

In theorem 3, we obtain the optimal cost reverse carpooling solution using Theorem 2 and compare its cost with the cost of the shortest paths solution, $l = d(s_1, t_1) + d(s_2, t_2)$.

Theorem 3: Given a two unicast problem $U = \{(s_1, t_1), (s_2, t_2)\}$, we can find a minimal cost solution $S^* = (P_1, P_2)$ in $O(1)$ time.

Outline of Proof: By Theorem 1, for two unicast sessions (s_1, t_1) and (s_2, t_2), there exists an minimal cost solution $S^* = (P_1, P_2)$ for which $K(P_1, P_2) \leqslant 1$. If $K(P_1, P_2) = 0$, using a shortest path for each unicast session gives the optimal solution. We use ($SP(s_1, t_1)$, $SP(s_2, t_2)$) to denote the shortest paths solution. Otherwise, S^* contains exactly one reverse carpooling segment $r(P_1, P_2)$. Let $r(P_1, P_2) = (e_k^{(1)}, e_{k+1}^{(1)}, ..., e_{k+m-1}^{(1)})$. We use $rc(s_1, t_2) = \text{tail}(e_k^{(1)})$ and $rc(t_1, s_2) = \text{head}(e_{k+m-1}^{(1)})$ to denote the two endpoints of $r(P_1, P_2)$ as shown in Fig. 3. For brevity, let b and c denote $rc(s_1, t_2)$ and $rc(t_1, s_2)$ respectively. Given locations of b and c, the reverse carpooling solution is $S(b, c) = (P_1, P_2)$ where

$$P_1 = SP(s_1, b) \cup r(P_1, P_2) \cup SP(c, t_1),$$

$$P_2 = SP(s_2, c) \cup (r(P_1, P_2))^R \cup SP(b, t_2).$$

We use $S' = (P'_1, P'_2)$ to denote the minimal cost reverse carpooling solution. We compare $C(S')$ with l. If $C(S') \leqslant l$, then $S^* = S'$. Otherwise, $S^* = (SP(s_1, t_1), SP(s_2, t_2))$. For a reverse carpooling solution with given b to be optimal, the location of c must satisfy

$$d(c, b) + d(c, t_1) + d(c, s_2) \leqslant d(p, b) + d(p, t_1) + d(p, s_2)$$

for all $p \in \mathcal{V}$.

By the construction in the proof of Theorem 2,

(i) If $\angle b$ in $\triangle bs_2t_1$ contains more than one grid line, then $c = b$. Since there is no cost reduction for reverse carpooling, the shortest paths solution is at least as good.

(ii) If $\angle s_2$ in $\triangle bs_2t_1$ contains more than one grid line, then $c = s_2$.

(iii) If $\angle t_1$ in $\triangle bs_2t_1$ contains more than one grid line, then $c = t_1$.

(iv) Otherwise, when each angle in $\triangle bs_2t_1$ contains one grid line as shown in Fig. 6, we assume that $\angle s_2$ in $\triangle bs_2t_1$ contains $g_{s_2,\alpha}$ and $\angle t_1$ in $\triangle bs_2t_1$ contains $g_{t_1,\beta}$ where $\alpha, \beta \in (0°, 60°, 120°)$, $\alpha \neq \beta$. We use $a_{\alpha,\beta}$ to denote the intersection between $g_{s_2,\alpha}$ and $g_{t_1,\beta}$. Then, by Theorem 2, we can get $c = a_{\alpha,\beta}$.

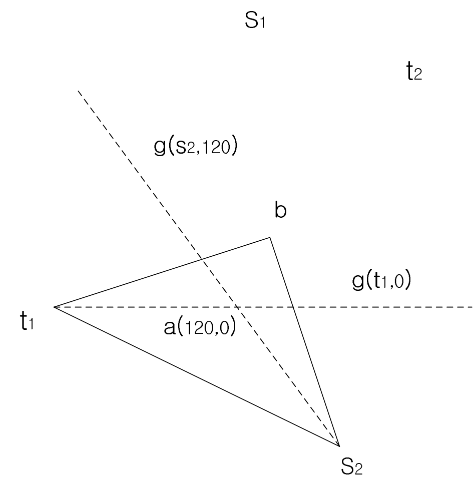

Fig. 6. Each angle in $\triangle bs_2t_1$ contains one grid line.

Thus, we only need to consider the following set of possible locations for c : $I = \{s_2, t_1, a_{0°,60°}, a_{0°,120°}, a_{60°,0°}, a_{60°,120°}, a_{120°,0°}, a_{120°,60°}\}$. For each $c \in I$, we use $b(c) \in \mathcal{V}$ to denote the corresponding location for the other end point b of the reverse carpooling segment. Then, we can find $b(c) \in \mathcal{V}$ such that

$$d(b(c), c) + d(b(c), s_1) + d(b(c), t_2) \leqslant d(p, c) + d(p, s_1) + d(p, t_2)$$

for all $p \in \mathcal{V}$ in $O(1)$ time, by Theorem 2. By comparing the costs of the reverse carpooling solutions for these 8 possibilities for c, we can obtain the minimum cost reverse carpooling solution, i.e. $C(S') = \min_{C\in I}\{C(S(b(c), c)\}$. Finally, we compare $C(S')$ with l and obtain an optimal cost solution. ■

IV. Three unicast sessions problem

In this section, we construct a polynomial-time algorithm which finds an optimal cost solution for three unicast sessions (s_1, t_1), (s_2, t_2), and (s_3, t_3). Theorem 4 helps to characterize the form of an optimal solution for three unicast sessions.

Theorem 4: Given a three-unicast problem $U = \{(s_1, t_1), (s_2, t_2), (s_3, t_3)\}$, there exists a minimal cost solution $S^* = (P_1, P_2, P_3)$ with $K(P_1, P_2) \leqslant 1$, $K(P_2, P_3) \leqslant 1$, and $K(P_2, P_3) \leqslant 1$.

Proof: Please see [7]. ■

Theorem 5: Given a three unicasts problem $U = \{(s_1, t_1), (s_2, t_2), (s_3, t_3)\}$, there exists a minimal cost solution S^* taking one of the six forms shown in Fig. 7, Fig. 8, Fig. 9, Fig. 10, Fig. 11, and Fig. 12 or those obtained by interchanging the different source and sink pairs. For the forms in in Fig. 9, Fig. 10, Fig. 11, and Fig. 12, we associate with each a set S of 2 or 4 points where $S \subseteq \{rc(s_1, t_2), rc(t_1, s_2), rc(s_1, t_3), rc(t_1, s_3), rc(s_2, t_3), rc(t_2, s_3)\}$ called fixed points. The minimal cost solution corresponding

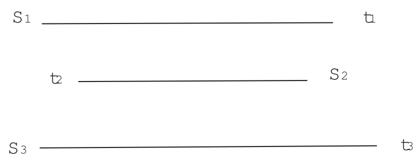

Fig. 7. Form 1 of Theorem 5 (case i) in the proof of Theorem 5) : In this case, $S^* = (SP(s_1,t_1), SP(s_2,t_2), SP(s_3,t_3))$.

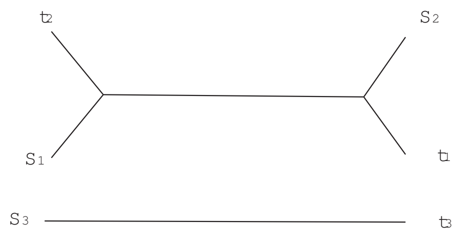

Fig. 8. Form 2 of Theorem 5 (case ii) - a) in the proof of Theorem 5) : S^* = $(S_{12}, SP(s_3,t_3))$.

to given locations of the fixed points can be found in $O(1)$ time.

Outline of Proof: We prove this theorem by characterizing all possible cases for a minimal cost solution $S^* = (P_1, P_2, P_3)$ satisfying $K(P_1,P_2) \leqslant 1$, $K(P_1,P_3) \leqslant 1$, $K(P_2,P_3) \leqslant 1$ and without any self-loops, which suffices by Lemma 1 and Theorem 4.

Case i) : $K(P_1,P_2)$ = 0, $K(P_2,P_3)$ = 0, and $K(P_1,P_3)$ = 0.

In this case, as shown in Fig. 7, the shortest paths for each unicast session give the minimal cost solution.

Case ii): 1$\leqslant$ $K(P_1,P_2)$ + $K(P_2,P_3)$ + $K(P_2,P_3)$

The path segments are labeled as shown in Fig. 3. We denote by p and q the points $rc(s_1,t_2)$ and $rc(t_1,s_2)$ respectively. Without loss of generality, we assume that $K(P_1,P_2)$ = 1. We use S_{12} = (P_1', P_2') to denote an optimal cost solution for (s_1,t_1) and (s_2,t_2) obtained by Theorem 3. We define S_{13} = (P_1', P_3') and S_{23} = (P_2', P_3') similarly.

Case ii) - a) $K(P_1,P_3)$ = 0, and $K(P_2,P_3)$ = 0.

Then, as shown in Fig. 8, S^* = $(S_{12}, SP(s_3,t_3))$.

Case ii) - b) $K(P_1,P_3)$ + $K(P_2,P_3)$ = 1.

Without loss of generality, we suppose that $K(P_1,P_3)$ = 1. Then, $K(P_2,P_3) = 0$, $K(B_{121},P_3)$$\leqslant$1, $K(r(P_1,P_2),P_3)$$\leqslant$1, and $K(B_{122},P_3)$$\leqslant$1.

Case ii) - b) - (1) $K(B_{121},P_3)$ = 1, and $K(B_{122},P_3)$ = 1.

In this case, $S^* = (S_{13}, SP(s_2,t_2))$.

Case ii) - b) - (2) $K(B_{121},P_3)$ + $K(B_{122},P_3)$ = 1.

Without loss of generality, we suppose that $K(B_{121},P_3)$ = 1. P_3 cannot reverse carpool along $r(P_1,P_2)$ since reverse carpooling occurs across pairs of unicast sessions. Hence P_3 reverse carpools only along B_{121}. As shown in Fig. 9, we use $S'_{121,3}$ = (B'_{121}, P''_3) to denote the optimal cost solution for

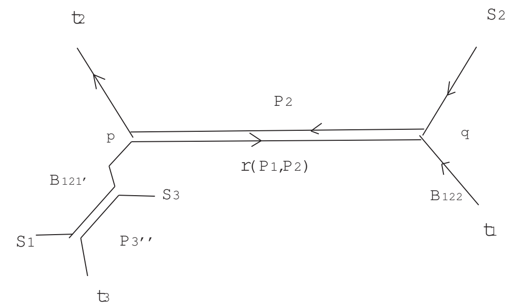

Fig. 9. Form 3 of Theorem 5 (case ii)- b)- 2) in the proof of Theorem 5).

(s_1,p) and (s_3,t_3) obtained by Theorem 3. Let

$$\begin{aligned} P_1'' &= (B'_{121} \cup SP(p,q) \cup SP(q,t_1)) \\ P_2 &= (SP(s_2,q) \cup SP(p,q)^R \cup SP(p,t_2)) \\ P_3 &= P_3''. \end{aligned}$$

Then, given the locations of p and q, S^* = (P_1'', P_2, P_3) which can be obtained in $O(1)$ time.

Case ii) - b) - (3) $K(B_{121},P_3)$ = $K(B_{122},P_3)$ = 0.

In this case, $S^* = (S_{12}, SP(s_3,t_3))$ which is the same as for case ii) - a).

Case ii) - c) $K(P_1,P_3)$ = $K(P_2,P_3)$ = 1.

Case ii) - c) - (1) $K(B_{121},P_3)$ = $K(B_{122},P_3)$ = 1.

It is shown in [7] that this case is contained in case ii) - b) - 2).

Case ii) - c) - (2) $K(B_{211},P_3) = K(B_{212},P_3) = 1$.

By symmetry, this case is the same as case ii) - c) - (1).

Case ii) - c) - (3) $K(B_{121},P_3)$ + $K(B_{122},P_3)$ = $K(B_{211},P_3)$ + $K(B_{212},P_3)$ = 1.

Without loss of generality, we assume that $K(B_{121},P_3)$ = 1.

Case ii) - c) - 3) - 1) $K(B_{211},P_3) = 1$ and $K(B_{212},P_3)$ = 0.

Let p = $rc(s_1,t_2)$, and q = $rc(t_1,s_2)$, r = $rc(s_2,t_3)$, u = $rc(t_2,s_3)$. We use $S''_{121,3}$ = (B'_{121}, P''_3) to denote the optimal cost solution for (s_1,p) and (r,t_3) obtained by Theorem 3. As shown in Fig. 10, for given locations of p, q, u, and r, S^* can be obtained in $O(1)$ time by Theorem 3.

Case ii) - c) - 3) - 2) $K(B_{211},P_3) = 0$ and $K(B_{212},P_3)$ = 1.

If $r(P_3,P_1)$ and $r(P_3,P_2)$ are continuous along P_3 as shown in Fig. 11, then $r(P_2,P_3)$ = $SP(p,u)$, and $r(P_1,P_3)$ = $SP(v,p)$. Since S^* is the optimal solution, the locations of u and v must satisfy $\forall y \in \mathcal{V}$,

$$\begin{aligned} d(t_2,u) + d(s_3,u) + d(p,u) &\leqslant d(t_2,y) + d(s_3,y) + d(p,y), \\ d(p,v) + d(s_1,v) + d(t_3,v) &\leqslant d(p,y) + d(s_1,y) + d(t_3,y). \end{aligned}$$

In this case, as shown in Fig. 11, given the locations of p and q, the locations of u and v can be found in $O(1)$ time by Theorem 2.

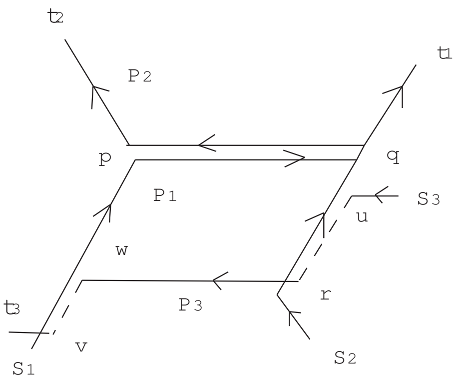

Fig. 10. Form 4 of Theorem 5 (case ii) - c) - 3) - 1) in the proof of Theorem 5) : Given locations of p, q, r, and u, we find S^* in $O(1)$ time. We use $S''_{121,3}$ = (B'_{121}, P''_3) to denote optimal cost solution obtained by applying Theorem 3 to (s_1, p) and (r, t_3). Then, S^* = (P''_1, P_2, P''_3) where P''_1 = $(B'_{121} \cup SP(p,q) \cup SP(q,t_1))$, P_2 = $(SP(s_2,r) \cup SP(r,u) \cup SP(u,q) \cup (SP(p,q))^R \cup SP(p,t_2)$, and P''_3 = $(SP(s_3,u) \cup (SP(r,u))^R \cup P''_3)$.

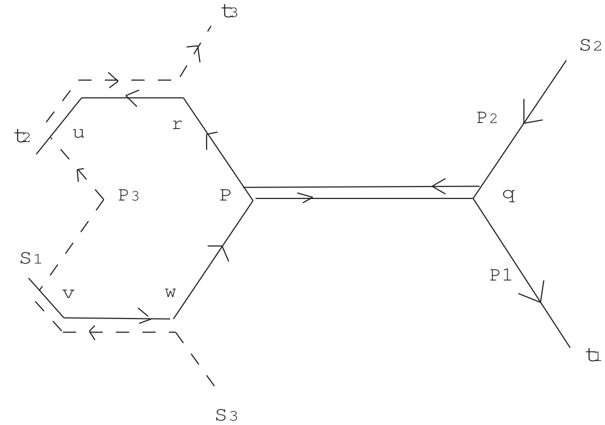

Fig. 12. Form 6 of Theorem 5 (case ii) - c) - 3) - 2) in the proof of Theorem 5) : $K(B_{121}, P_3)$ = $K(B_{212}, P_3)$ = 1 ,and $r(P_3, P_1)$ and $r(P_3, P_2)$ are not continuous along P_3. Given the locations of p, q, u, and v, the locations of r and w can be found in $O(1)$ time by Theorem 2. Then, S^* = (P_1, P_2, P_3) where P_1 = $(SP(s_1,v) \cup SP(v,w) \cup SP(w,p) \cup SP(p,q) \cup SP(q,t_1))$, P_2 = $(SP(s_2,q) \cup (SP(p,q))^R \cup SP(p,r) \cup SP(r,u) \cup SP(u,t_2)$, and P_3 = $(SP(s_3,w) \cup (SP(v,w))^R \cup SP(v,u) \cup (SP(r,u))^R \cup SP(r,t_3))$.

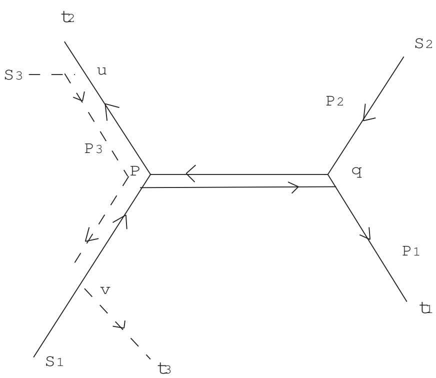

Fig. 11. Form 5 of Theorem 5 (case ii) - c) - 3) - 2) in the proof of Theorem 5) : $K(B_{121}, P_3)$ = $K(B_{212}, P_3)$ = 1, and $r(P_3, P_1)$ and $r(P_3, P_2)$ are continuous along P_3. Given the locations of p and q, the locations of u and v can be found in $O(1)$ time by Theorem 2. Then, S^* = (P_1, P_2, P_3) where P_1 = $(SP(s_1,v) \cup SP(v,p) \cup SP(p,q) \cup SP(q,t_1))$, P_2 = $(SP(s_2,q) \cup (SP(p,q))^R \cup SP(p,u) \cup SP(u,t_2)$, and P_3 = $(SP(s_3,u) \cup (SP(p,u))^R \cup (SP(v,p))^R \cup SP(v,t_3))$.

Otherwise, $r(P_2, P_3)$ = $SP(r,u) \subset B_{212}$ and $r(P_1, P_3)$ = $SP(v,w) \subset B_{121}$, as shown in Fig. 12. Since S^* is the optimal solution, the locations of r and w must satisfy $\forall y \in \mathcal{V}$,

$$\begin{aligned} d(t_3,r) + d(u,r) + d(p,r) &\leqslant d(t_3,y) + d(u,y) + d(p,y), \\ d(p,w) + d(s_3,w) + d(v,w) &\leqslant d(p,y) + d(s_3,y) + d(v,y). \end{aligned}$$

In this case, as shown in Fig. 12, given the locations of p, q, u, and v, the locations of r and w can be found in $O(1)$ time by Theorem 2.

Theorem 6: Given a three unicast problem U = $\{(s_1,t_1),(s_2,t_2),(s_3,t_3)\}$, we can find a minimal cost solution S^* = (P_1, P_2, P_3) in $O(n^4)$ time where n is the number of vertices in the convex hull of $(s_1, t_1, s_2, t_2, s_3, t_3)$.

Proof: For each of the six possible forms given in Theorem 5, we can calculate the optimal cost solution in $O(n^4)$ time by considering all $O(n^4)$ possible locations for the fixed points. We then compare their costs and choose the minimal cost solution S^*. ■

V. General multiple unicast sessions problem

In this section, we generalize our problem to general multiple unicasts problem and introduce a simple greedy algorithm to obtain an approximate solution. In Section III, we described a polynomial time algorithm which finds an optimal cost solution for the two unicast sessions problem. Based on this algorithm, we present a greedy algorithm to obtain an approximate solution for n-unicast sessions (s_1,t_1),...,(s_n,t_n) on the triangular grid.

We define a metric m_{ij},$(i, j \in \{1,..n\}, i \neq j)$ and a selection function I. We use $S_{(i,j)}$ to denote an optimal cost solution obtained by applying Theorem 3 to two unicast sessions (s_i,t_i) and (s_j,t_j). Let m_{ij} = $d(s_i,t_i)$ + $d(s_j,t_j)$ − $C(S_{ij})$. Given (s_i,t_i) and (s_j,t_j), m_{ij} denotes the difference between the cost of the shortest paths solution and the optimal cost. The selection function $I : N \subset (1,2,..n) \rightarrow N\text{x}N$ chooses a pair of indices in N which maximizes the metric m_{ij}. Precisely,

$$I(N) = \arg\max_{i,j \in N} \{m_{ij}\}.$$

We use CS to denote the current solution and $N \subset (1,2,..n)$ to denote a set of indices which are not used in the current solution at each step. In each step of the algorithm, we update two sets CS and N using the selection function I. Then, we remove $I(N)$ from N and add $S_{I(N)}$ to CS at each step.

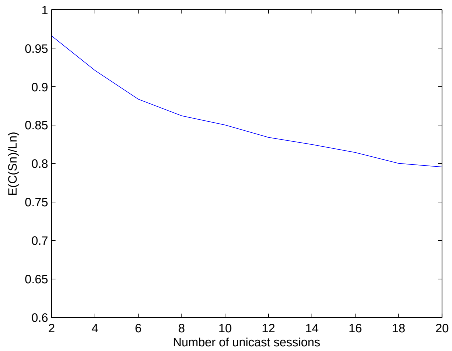

Fig. 13. Simulation result: As n increases, $E(\frac{C(S_n)}{L_n})$ decreases.

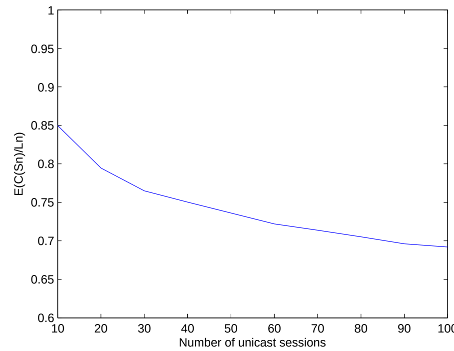

Fig. 14. Simulation result: We extend the result in Fig. 13. When 100 unicast sessions are chosen uniformly at random on the grid, $E(\frac{C(S_n)}{L_n})$ is 0.69

Finally, at the end of the algorithm, we obtain a suboptimal solution $CS = S = (P_1,..,P_n)$.

$N \leftarrow \{1, 2, ..., n\}$
$CS \leftarrow \emptyset$
While $|N|>1$
$N = N - I(N)$
$CS = CS \cup S_{I(N)}$
endwhile
If $N = \{k\}$ $(1 \leqslant k \leqslant n)$
return $CS = CS \cup SP(s_k, t_k)$
else
return CS
endif

Theorem 7: The time complexity of the above greedy algorithm is $O(n^3)$.

Proof: Please see [7]. ■

VI. Simulation

In order to determine the effectiveness of reverse carpooling, we constructed a simulation environment which models operation on the wireless triangular grid. Given a wireless triangular grid, we choose the locations of unicast sessions uniformly at random on the grid and compare the average network cost between a suboptimal solution obtained by the greedy algorithm of Section V and the shortest paths solution without network coding. The simulations were done using MATLAB.

A. metric

Given n-unicast problem $U = ((s_1, t_1), .., (s_n, t_n))$, we use S_n to denote a suboptimal solution obtained by our greedy algorithm. Let $L_n = \sum_{i=1}^{n} l(SP(s_i, t_i))$ denote the sum of the cost of the shortest path solution for each unicast session. Our evaluation uses the performance metric $E(\frac{C(S_n)}{L_n})$ which is the average ratio between $C(S_n)$ and L_n.

B. Simulation Results

We use a *triangular grid* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ as the set of vertices $\mathcal{V} = \{a(1,0) + b(\frac{1}{2}, \frac{\sqrt{3}}{2}) : -5 \leqslant a \leqslant 5, -10 \leqslant b \leqslant 10\}$ and randomly choose the locations of a given number of unicast sessions. Fig. 13 indicates that when 20 unicasts are chosen uniformly at random on the graph G, the average cost of the greedy reverse carpooling solution is 0.79 times that of the shortest paths solution. As the number of unicast sessions increases, $E(\frac{C(S_n)}{L_n})$ decreases. This result agrees with our intuition that the number of opportunities to apply reverse carpooling increases with the number of unicast sessions in a given network. As shown in Fig. 14, when the number of unicast session is 100, $E(\frac{C(S_n)}{L_n})$ is 0.69. When n is sufficiently large, $E(\frac{C(S_n)}{L_n})$ converges to an optimal value of 0.5.

VII. Conclusion

We have presented two polynomial time algorithms to obtain approximately optimal solutions which minimize the number of transmissions for two and three unicast sessions, respectively. By extending the algorithm for two unicasts problem, we presented a greedy algorithm to obtain an approximate solution for n-unicast sessions problem. From simulations, we have shown that the algorithm reduces the power consumption significantly for multiple unicasts on a triangular grid.

References

[1] R. Ahlswede, N. Cai, S.-Y.R.Li,and R. W. Yeung, "Network information flow," *IEEE Trans. Inform. Theory*, vol 46, no. 4, pp. 1204-1216, July 2000.

[2] M. Efrros, T. Ho, and S. Kim, "A Tiling Approach to Network Code Design for Wireless Networks," in *Proc. of Information Theory Workshop(ITW)*, Punta del Este, Uruguay, Mar. 2006, pp. 62-66.

[3] Y.Wu, P.A. Chou, and S.-Y. Kung. "Information exchange in wireless networks with network coding and physical layer broadcast," *Technical Report MSR-TR-2004-78*, Microsoft Research, Redmond, WA, Aug. 2004.

[4] Y.Wu, P.A. Chou, and S.-Y. Kung. "Minimun-energy multicast in mobile ad-hoc networks using network coding ," *IEEE Trans. Commun.*, ,vol 53, no. 11, pp. 1906-1918, Nov. 2005.

[5] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard and J. Crowcroft, "XORs in the Air: Practical Wireless Network Coding," in *Proc. of ACM SIGCOMM 2006*.

[6] J. Widmer, C. Fragouli,and J.-Y. L. Boudec "Low-complexity energy-efficient broadcasting in wireless ad-hoc networks using network coding," in *Proc. of Workshop on Network Coding, Theory, and Applications (NetCod 2005)*, Riva del Garda, Italy, Apr. 2005.

[7] S. Kim, M. Effros, and T. Ho, "On Low-Power Multiple Unicast Network Coding Over a Wireless Triangular Grid," available at https://www.its.caltech.edu/˜ sukwon/paper.pdf