Collusion in VCG Path Procurement Auctions

Yoram Bachrach[1], Peter Key[1], and Morteza Zadimoghaddam[2]

[1] Microsoft Research, Cambridge, UK yobach,peter.key@microsoft.com
[2] MIT, Cambridge, MA zadimoghaddam@csail.mit.edu

Abstract. We consider collusion in path procurement auctions, where payments are determined using the VCG mechanism. We show that collusion can increase the utility of the agents, and in some cases they can extract any amount the procurer is willing to offer. We show that computing how much a coalition can gain by colluding is NP-complete in general, but that in certain interesting restricted cases, the optimal collusion scheme can be computed in polynomial time. We examine the ways in which the colluders might share their payments, using the core and Shapley value from cooperative game theory. We show that in some cases the collusion game has an empty core, so although beneficial manipulations exist, the colluders would find it hard to form a stable coalition due to inability to decide how to split the rewards. On the other hand, we show that in several common restricted cases the collusion game is convex, so it has a non-empty core, which contains the Shapley value. We also show that in these cases colluders can compute core imputations and the Shapley value in polynomial time.

1 Introduction

Collusion is an agreement between agents to defraud in order to obtain an unfair advantage [22]. We examine collusion in path procurement auctions (PPAs), where a buyer procures a path from a source s to a target t in a graph $G = \langle V, E\rangle$. Each edge $e_i \in E$ is owned by a_i, who incurs a cost c_i when her edge is used. The cost c_i is known only to a_i. The buyer must compensate edges on the chosen path for their costs. Given the private costs, a mechanism can find the minimal cost $s-t$-path. The mechanism can ask each a_i for the minimal amount it would be willing to receive to allow using e_i. If a_i answers (bids) truthfully, this is her cost c_i. However, the costs are the agents' private information and they may bid strategically to increase their payment. VCG mechanisms [23, 10, 13] are used to incentivise agents to reveal their true costs. VCG has desirable properties, but is susceptible to collusion. Though any single agent is incentivised to bid truthfully, *several* agents may *coordinate* bids and split the gains from manipulating. We show how agents might collude and share the gains in VCG PPAs. Our model follows the *collusion game* of [4], but applied to PPAs.

1.1 Preliminaries

In VCG mechanisms we have an agent set $N = \{1, \dots, n\}$. The mechanism chooses an alternative from the set K. Agents report a type $\theta_i \in \Theta_i$, representing her preferences over K, and each agent i has a valuation $w_i(k, \theta_i)$ depending on the chosen $k \in K$. The mechanism uses the choice rule $k : \Theta_1 \times ... \times \Theta_n \to K$, and agent i must also make a payment r_i to the mechanism, according to a payment rule $t_i : \Theta_1 \times ... \times \Theta_n \to \mathbb{R}$. We assume quasi-linear utility $u_i(k, p_i, \theta_i) = w_i(k, \theta_i) - r_i$. An agent i may manipulate and report type $\theta'_i = s_i(\theta_i)$, according to its strategy s_i. Groves mechanisms use $k^*(\theta') = arg\max_{k \in K} \sum_i w_i(k, \theta'_i)$ and payment rule: $r_i(\theta') = h_i(\theta'_{-i}) - \sum_{j \neq i} w_j(k^*, \theta'_j)$, where $h_i : \Theta_{-i} \to \mathbb{R}$ only depends on the reported types of agents other than i. We consider the case of VCG, where: $h_i(\theta'_{-i}) = \sum_{j \neq i} w_j(k^*_{-i}(\theta'_{-i}), \theta'_j)$.

Our collusion analysis uses coalitional game theory. A transferable utility coalitional game is composed of a set N of n agents, and a characteristic function mapping any agent subset (coalition) to a value $v : 2^N \to \mathbb{R}$, indicating the total utility these agents achieve together. The function only defines the gains a coalition achieves, not how to distribute them. An *imputation* $(p_1, \ldots, p_n)$ divides the the gains among the agents, where $p_i \in \mathbb{R}$, such that $\sum_{i=1}^{n} p_i = v(N)$. We call p_i the payoff of agent i, and denote $p(C) = \sum_{i \in C} p_i$. A key issue is choosing the appropriate imputation. A basic imputation requirement is *individual rationality*: for any $i \in N$, $p_i \geq v(\{i\})$. Otherwise, agent i is incentivized to work alone. Similarly, coalition B *blocks* imputation p if $p(B) < v(B)$, since B's members are better off working on their own. A solution concept focusing on this is the *core* [12]: the set of all imputations p not blocked by any coalition, so for any $C \subseteq N$ we have: $p(C) \geq v(C)$.

Another solution concept is the Shapley value [20] which defines a *single* value division. It focuses on *fairness*, rather than stability. The Shapley value fulfills important fairness axioms [20, 25] and has been used to fairly share gains or costs. The Shapley value of an agent depends on its marginal contribution to possible coalition permutations. We denote by π a permutation (ordering) of the agents, and by Π the set of all possible such permutations. Given permutation $\pi \in \Pi = (i_1, \ldots, i_n)$, the marginal worth vector $m^\pi[v] \in \mathbb{R}^n$ is defined as $m^\pi_{i_1} = v(\{i_1\})$ and for $k > 1$ as $m^\pi_{i_k}[v] = v(\{i_1, i_2, \ldots, i_k\}) - v(\{i_1, i_2, \ldots, i_{k-1}\})$. The convex hull of all the marginal vectors is called the *Weber Set*. Weber showed [24] that the Weber set of any game contains its core. The Shapley value is the centroid of the marginal vectors.

Definition 1. *The Shapley value is the payoff vector:* $\phi[v] = \frac{1}{n!} \sum_{\pi \in \Pi} m^\pi[v]$.

Our analysis is based on the notion of convex games. For convex games it is known [21] that the core is non-empty, and that the Weber Set is identical to the core. The Shapley value is a convex combination of the marginal vectors and lies in the Weber Set, so in convex games, the Shapley value lies in the core.

Definition 2. *A game is convex if:* $\forall A, B \subseteq I$, $v(A \cup B) \geq v(A) + v(B) - v(A \cap B)$.

2 Collusion in VCG Path Procurement Auctions

Consider a PPA in a graph $G = \langle V, E \rangle$, where the buyer procures edges $P \subseteq E$ forming an $s-t$-path from a set of agents, each owning an edge in the graph. We identify an agent a_i with her edge $e_i \in E$. Each agent has a cost c_i associated with her edge and the mechanism asks each a_i to provide a bid b_i for using the edge. If the agent is truthful, she would report c_i. Given the edges' true costs, one can find the minimal cost $s-t$-path, but the costs are private information. The canonical solution to induce truthfulness is the VCG mechanism. As discussed in Section 1.1, using VCG prices makes truthful cost revelation the dominant strategy, and results in procuring the cheapest path. Given the edge costs, this path can easily be computed in polynomial time.

Observation 1 (Computing VCG Prices) *Let $G=\langle V,E\rangle$ be a path procurement domain, with cost c_i for edge $e_i\in E$, and let b_i be the bid of e_i. Denote the minimal cost path (according to the declared b_i's) as $(e_{i_1},e_{i_2},\ldots,e_{i_x})$ (of x edges), and let the optimal path not including e_i be $e_{j_1},e_{j_2},\ldots,e_{j_y}$ (of y edges). If e_i is on the chosen path, the payment to e_i's owner is $p_i=\sum_{s=1}^{y} b_{j_s}-\sum_{s=1}^{x} b_{i_s}+b_i$, otherwise $p_i=0$.*

2.1 Colluding in VCG Path Procurement Auctions

We begin with collusion examples. Denote the payment to agent a_i when all the agents bid truthfully (i.e. a_i bids her true cost so $b_i=c_i$) as p_i. Given a set of edges $C\subseteq E$, we denote the VCG payments of all of them under truthful revelation as $p(C)=\sum_{e_i\in C} p_i$.

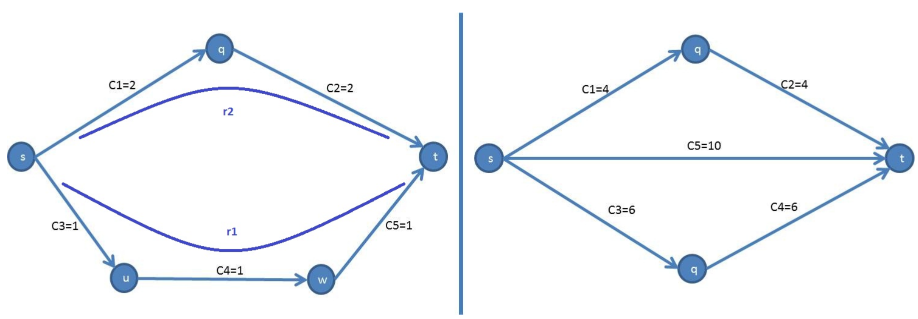

Fig. 1. Left: domain for Examples 1, 2, 3. Right: domain for Example 4.

Example 1 (Collusion on the cheapest path). Consider the graph on the left of Figure 1, with two $s-t$-paths: $r_1=\langle s,u,w,t\rangle$ with costs $c_3=1, c_4=1, c_5=1$ and $r_2=\langle s,q,t\rangle$ with edge costs $c_1=2, c_2=2$. The cheapest path is r_1 with cost $c_{r_1}=1+1+1=3$, and the second cheapest path is r_2 with cost $c_{r_2}=2+2=4$. Consider the agents on r_1: $C=\langle e_3,e_4,e_5\rangle$. If all the edges truthfully declare their costs (so a_i bids b_i where $b_i=c_i$), applying Observation 1 we obtain payments: $p_3=2, p_4=2, p_5=2$. Thus, we have $p(C)=2+2+2=6$. Suppose each of the agents in C reports having no cost, bidding $b'_3=b'_4=b'_5=0$. This manipulation does not change the chosen path, as the cheapest path remains r_1. However, the payments do change. Denote the payments when the agents in C bid untruthfully (so $b'_3=b'_4=b'_5=0$) and the agents in $I\setminus C$ bid truthfully (so $b'_1=c_1=2, b'_2=c_2=2$) as $p'=\langle p'_1,p'_2,\ldots,p'_6\rangle$. Recomputing VCG payments for b' we obtain $p'_3=p'_4=p'_5=4$. Thus each member of C benefits from this manipulation, and the total payments for the C become $p'(C) = \sum_{e_i \in C} p'_i = 12$. Note the actual costs of the agents in C have not changed, but total payments increased by $12 - 6 = 6$. The cost of the coalition C when r_1 is chosen is $c_1 + c_2 + c_3 = 1 + 1 + 1 = 3$, so through this manipulation, the coalition moves from a utility of $p(C) - \sum_{i \in C} c_i = 6 - 3 = 3$ to $p'(C) - \sum_{i \in C} c_i = 12 - 3 = 9$.

Example 2 (Collusion on a s-t cut). Examine the left of Figure 1 again, but consider the case where $C = \langle e_1, e_3 \rangle$ collude, and e_2, e_4, e_5 bid truthfully. Under truthfully declarations, the chosen path is r_1 with payments: $p_1 = p_2 = 0, p_3 = p_4 = p_5 = 2$. We have $p(C) = p_1 + p_3 = 2$, and since r_1 is chosen, e_3 incurs a cost $c_3 = 1$ so the utility of the coalition C is $p(C) - \sum_{i \in C \cap r_1} c_i = 2 - c_3 = 2 - 1 = 1$. Now suppose the colluders in C manipulate and bid $b'_1 = h$ (for a high number $h > 2$, say $h = 100$), and $b'_3 = 0$ [1], while e_2, e_4, e_5 bid truthfully. Again, the manipulation does not change the chosen path which is still r_1, but the payments do change. Again, we denote the payments when the agents in C bid untruthfully ($b'_1 = h, b'_3 = 0$) and the agents in $I \setminus C$ bid truthfully as $p' = \langle p'_1, p'_2, \ldots, p'_5 \rangle$. Recomputing the VCG payments under p' we get $p'_1 = 0, p'_3 = h + 2 - 2 = h$. Thus, $p(C) = h$. Since r_1 is still the chosen path, e_3 still incurs the cost c_3. Thus the new utility of the coalition C is $p(C) - \sum_{i \in C \cap r_1} c_i = h - 1$. Since the payment of the coalition depends on its chosen value for h, its utility is unbounded. One might claim that since a_1 did not increase her utility, she might not be willing to collude (lie for a_3). To get a_1 to cooperate, a_3 can easily *compensate* a_1 via a monetary transfer. Without such a monetary transfers, all the payment goes to e_3. However, using such a transfer, the utility of the coalition of colluders, $p(C) - \sum_{i \in C \cap r_1} c_i = h - 1$, can be shared between e_1 and e_3 in any.

Example 3 (Collusion on the non-optimal path). Consider the left of Figure 1, with the optimal path r_1 and the second cheapest path r_2. Suppose $C = \langle e_1, e_2 \rangle$ collude (edges of a non-optimal path), and e_3, e_4, e_5 bid truthfully. Under truthful declarations the chosen path is r_1, and $p_1 = p_2 = 0$ (as $r_1 = \langle e_3, e_4, e_5 \rangle$ is chosen and not $r_2 = \langle e_1, e_2 \rangle$), so we have $p(C) = 0$, and the utility of C is 0. If C manipulates by bidding $b'_1 = b'_2 = 0$, the chosen path is r_2 rather than r_1, and the payments are $p'_1 = p'_2 = 3$, so we have $p'(C) = 6$. However, since r_2 is chosen, edges e_1, e_2 incur the costs of $c_1 = c_2 = 2$, so the coalition's utility is $p(C) - \sum_{i \in C} c_i = 6 - 4 = 2$. Thus, this manipulation gives C a utility of 2, rather than 0. Without transfers, this utility is shared equally between e_1 and e_2, but it can be shared in any way using transfers.

Example 2 is troublesome, as the colluders achieve unbounded payment from the mechanism[2]. Example 3 shows that even agents on a non-optimal path can manipulate. We now show an example where beneficial manipulations exist, but due to the network structure, the colluders cannot find a stable way to share the gains from manipulating.

Example 4 (Empty Core). Consider Figure 1 on the right. The cheapest path is $r_1 = \langle e_1, e_2 \rangle$ with cost 8, the second cheapest path is $r_2 = \langle e_5 \rangle$ with cost 10, and the third

[1] For this case, e_3 may as well report its true cost. However, if the coalition has other edges on the cheapest path (e.g. e_4 or e_5), this increases their payment as well.

[2] Colluders who can disconnect s and t get any amount the procurer has. This is not surprising as the good sold is $s - t$ connectivity, and the colluders' cartel controls all the supply.

cheapest is $r_3 = \langle e_3, e_4 \rangle$ with cost 12. Under truthful declarations r_1 is chosen, and the payments are $p_1 = p_2 = 6$ (other payments are 0). Coalition $C_t = \langle e_1, e_2 \rangle$ can manipulate similarly to Example 1 by bidding $b'_1 = b'_2 = 0$ to achieve $p'(C_t) = 10 + 10 = 20$. This raises the utility of C_t from $12 - 8 = 4$ to $20 - 8 = 12$. However, Coalition $C_b = \langle e_3, e_4 \rangle$ can manipulate similarly to Example 3 by bidding $b'_3 = b'_4 = 0$ to achieve a $p'(C_b) = 8 + 8 = 16$ [3]. This raises the utility of C_b from 0 to $16 - 12 = 4$.

Consider the case where $C = C_t \cup C_b = \{e_1, e_2, e_3, e_4\}$ collude. C doesn't control e_5 so its payment cannot exceed 10 per edge. Either $\langle e_1, e_2 \rangle$ or $\langle e_3, e_4 \rangle$ or $\langle e_5 \rangle$ is chosen, so the total payment for C cannot exceed 20. The minimal cost C incurs to get any payment is $4+4$ (routing through $\langle e_1, e_2 \rangle$). Thus C's utility is bounded by $20-8 = 12$, similarly to C_t, and achievable the same way. Thus, C_b adds no value to coalition C_t. Consider what happens when $C = \{e_1, e_2, e_3, e_4\}$ try to agree on what to bid and how to share the gains. The optimal collusion bids for them get them a utility of 12. Edges e_1, e_2 (of C_t) might claim they deserve all this utility, as they can achieve this utility on their own. However, e_3, e_4 (of C_b) would claim they deserve at least 4, as they achieve 4 on their own. This results in an unstable coalition and in threats between the coalition members [4]. Section 3 characterizes this as a collusion game with an empty core.

In Example 4 , though the colluders have a beneficial manipulation, they find it hard to form a coalition due to inability to decide how to share the reward. We characterize such situations using the collusion game. Despite hopes of having such instability mitigate collusion, we show that for natural coalitions the colluders can always share the gains in a stable way. We focus on coalitions where all colluders are on the cheapest path (as in Example 1) or a non-optimal path (as in Example 3).

2.2 Collusion Schemes

We consider optimal manipulations in VCG PPAs. Such collusion requires trust among the colluders, as they must coordinate and since in many domains collusive behavior is

[3] These are the payments where only e_3, e_4 collude, so e_1, e_2 truthfully declares their cost, so under the collusion, the VCG mechanism chooses $\langle e_3, e_4 \rangle$ as the "cheapest" path, and computes the payments using the alternative path $\langle e_1, e_3 \rangle$ of cost 8.

[4] Agents e_3, e_4 might threaten to bid $b'_3 = b'_4 = 0$ creating two zero cost paths, so the result would depend on how the mechanism breaks ties. In this case, the agents on the winning path would get a zero payment. If coalition $\{e_1, e_2, e_3, e_4\}$ breaks down into *two* coalitions $\{e_1, e_2\}$ and $\{e_3, e_4\}$ (each pair bidding in a coordinated manner), we have a normal form game. Each pair chooses the total cost of the path, the pair with lower cost winning and obtaining a total reward of the difference between the paths' costs plus its declared cost. A pure strategy Nash equilibrium is where the truly cheap path bids zero, and the truly expensive path bids highly enough to guarantee the cheap path a positive utility: the total payment to the cheap path is $k(h - l) + l$ where k is the number of edges on it and h and l are the declared path prices, so when h is high enough this exceeds the cheap path's true cost. If these are the *only* two paths, there is another Nash equilibrium: the cheap path bids highly, H, and the expensive path bids zero: the expensive path has a positive utility when winning and the cheap path can only win by bidding zero, in which case it would have a negative utility. When analyzing the core of the collusion game, we assume members dropping out do *not* form a new cartel and bid truthfully. Even under this easier assumption, some collusion games have empty cores.

forbidden (the colluders face dire consequences if caught). We first show that in general, given a colluder coalition C, finding the optimal collusion or the utility of a colluder coalition when it optimally manipulates for a coalition is NP-complete.

Theorem 1. *Computing the optimal coalition manipulation in a VCG PPA is NP-Complete.*

Proof. Computing the optimal manipulation value is *in* NP (up to any desired degree of numerical accuracy), since we can non-deterministically choose bids and check if we have a manipulation achieving the target utility. To show NP-hardness, we reduce from LONGEST-PATH (LP), where we are given a graph $G = \langle V, E\rangle$ and are asked to return the length of the longest simple path in it, known to be NP-Complete. Given the LP instance $G = \langle V, E\rangle$, we create a graph $G' = \langle V \cup \{s, t\}, E'\rangle$, which contains a copy of G and two other vertices: s which serves as the source and t which serves as the target of the PPA. All of G's edges are also replicated. Also, the source s is connected to the all the vertices in G, and any vertex is G is connected to the target t. We denote all edges (s, v) where $v \in V$ as S, and all edges (u, t) where $u \in V$ as T. We create an edge e_H, connecting s and t. All edges have a cost of $c_e = 1$ except edges in $S \cup T \cup \{e_H\}$. Edges in $S \cup T$ have zero cost, and e_H has a cost H where H is a very high number (for example $H > |E|^2$). The target coalition for which we find an optimal manipulation is $C = E' \setminus e_H = S \cup T \cup E$, all edges except e_H.

Denote by $L = (l_1, \ldots, l_q)$ the longest simple path in G, and its length by q. Coalition C contains L, and so it can have all the edges in $L \cup S \cup T$ bid zero, and all the other edges in C bid $H + 1$. Then, the cheapest path is $(s, l_1, \ldots, l_q, t)$ with a declared cost of zero, so this path is chosen. Under this manipulation, the second cheapest path is (s, t) with cost H, so each edge is paid H, and the coalition is paid $p(C) = (q+2)H$ (there are q edges on the longest path in G, and the edges (s, l_1) and (l_q, t)). The coalition incurs the true cost of 1 on its q edges in L, so C has a total cost of q. Thus, this manipulation obtains C a utility of $u^*(C) = (q + 2)H - q$. It is easy to see that $u^*(C)$ is the maximal utility C can obtain: the cheapest path must have a total cost of at most H or e_H would be the chosen path, so any edge can be paid at most H, and since L is the longest simple path in G it is impossible to have more than q edges of G on the path the mechanism chooses. Since $u^*(C) = (q + 2)H - q$ and since we choose the value of H in the reduction, given $u^*(C)$ we can extract q, the length of the longest simple path in G. This proves we cannot compute the optimal manipulation bids, since given this manipulation we can compute the chosen path and VCG prices and since we know the true edge costs this allows computing $u^*(C)$.

The hardness result of Theorem 1 forces us to examine restricted cases of the manipulation problem. In the extreme case where *all* the edges collude, they can guarantee any payment the procurer can pay [5]. In typical domains, the set of colluders is unlikely to be all the edges or an arbitrary edge subset. A more reasonable colluder set can be a set of neighboring or close edges, or several edges that are all on a single $s - t$ path. We examine cases where we can tractably compute the optimal manipulation. Example 1 is an example of a simple case, where all colluders are on the cheapest $s - t$ path, and the

[5] We later show that it suffices for the colluders to be able to disconnect s and t.

second cheapest path runs in parallel to the cheapest path. Example 3 shows the second simple case, where all colluders are on a non-optimal $s-t$ path, which runs in parallel to the cheapest path, and can underbid the truly optimal path.

Consider the case where all colluders are on the cheapest path. C is *a simple coalition on the cheapest path* if these hold for all $e_i \in C$: edge e_i is on the cheapest $s-t$ path; when removing e_i, the cheapest $s-t$-path contains no edge $e_j \in C$. Similarly, C is *a simple coalition on a non-optimal path* if the following hold: all edges $e_i \in C$ are a non-optimal $s-t$ path, r; the cheapest path r^* does not intersect r, so $r^* \cap r = \emptyset$; r becomes cheapest when $b'_i = 0$ for all $e_i \in C$: $\sum_{e_i \in r \setminus C} c_i < \sum_{e_i \in r^*} c_i$.

The following theorems are regarding a VCG PPA, where edge e_i bids b_i and has cost c_i, and where C is a simple coalition on the cheapest path r_1. We assume that all non-coalition members bid truthfully, so for $e_i \in I \setminus C$ we have $b_i = c_i$.

Theorem 2 (Simple Cheapest Path Collusion). *Let C be a simple coalition of colluders on the cheapest path r_1. The optimal collusion, maximizing payments p_i of any $e_i \in C$ (and C's payment $p(C) = \sum_{e_i \in C} p_i$) is zero bids: $b_i = 0$ for all $e_i \in C$.*

Proof. Denote the cheapest path under truthful declarations as r_1, and the cheapest path under truthful declarations that does not contain any edge in C as r_2. Consider an edge $e_i \in C$ that increases its bid beyond c_i. This increases the cost of r_1 under declared bids. If several agents in C declare such increased costs so that the cost of r_1 under these modified costs is more than the cost of r_2, the path r_2 will be chosen, resulting in a payment of 0 to all agents in C. Since VCG is individually-rational, this manipulation is not beneficial to the colluders. Thus, it suffices to focus on manipulations where the bids of edges in C are such that the cost of r_1 is at most the cost of r_2, so the procured path is r_1. Eliminating any edge $e_i \in C$ disallows the use of r_1, and for any $e_i \in C$ we denote by r_{-i} the cheapest path when eliminating e_i. Since C is a simple coalition on the cheapest path we have $r_{-i} \cap C = \phi$. Thus for $e_i, e_j \in C$ we have $r_{-i} = r_{-j}$. Since e_i, e_j are arbitrary edges in C, this means that the cheapest path after eliminating any edge in C is the same path r. This path r cannot contain any edge $e_i \in C$, so it is simply the cheapest path that does not contain any edge in C, r_2. Denote the edges in $r = r_2$ as $r_2 = \langle e_{j_1}, e_{j_2}, \ldots, e_{j_y} \rangle$ (y edges). Denote the edges of r_1 as $r_1 = \langle e_{i_1}, e_{i_2}, \ldots, e_{i_x} \rangle$ (x edges, containing the edges of C). We assume all agents in r_2 bid truthfully, and denote the total cost of r_2 as $c(r_2)$. Thus, the formula of Observation 1 can be written as: $p_i = \sum_{s=1}^{y} b_{j_s} - \sum_{s=1}^{x} b_{i_s} + b_i = c(r_2) - \sum_{s=1, e_{i_s} \neq i}^{x} b_{i_s}$. Note that the agents in C control the bids $\{b_i | e_i \in C\}$, and since each b_i must be non-negative (the cost of using edge e_i), each p_i is maximized when the bids are minimal. Thus, the optimal manipulation is bidding $b_i = 0$ for all $e_i \in C$.

Theorem 3 (Simple Non-Optimal Path Collusion). *Let C be a simple coalition of colluders on the non optimal path r. The optimal collusion, which maximizes all the payments p_i of any coalition member (and C total payment $p(C) = \sum_{e_i \in C} p_i$) is zero bids: $b_i = 0$ for all $e_i \in C$.*

Proof. The proof is almost identical to Theorem 3. We denote the non optimal path r which contains all the colluders as $\langle e_{i_1}, e_{i_2}, \ldots, e_{i_x} \rangle$, denote the cheapest path (under true costs) as r^*, and obtain: $p_i = c(r^*) - \sum_{s=1, e_{i_s} \neq i}^{x} b_{i_s}$.

Theorem 4 (Cut Collusion). *Let C be a coalition whose removal disconnects s and t, and $h > 0$ be some value. The colluders can bid so that $\sum_{e_i \in C} p_i > h$.*

Proof. At least one $e_x \in C$ must be used in the chosen $s-t$ path, as C is an $s-t$ cut. VCG is individually rational so if all $e \in C$ bid $b'_i = h$, for e_x we have $p_i > h$.

3 The Collusion Game

Consider a PPA over $G = \langle V, E \rangle$ with source s and target t. We examine a subset $C \subseteq N$, who may decide to collude. Under truthful bidding, VCG chooses path $r_1 = \langle e_{i_1}, e_{i_2}, \ldots, e_{i_x} \rangle$ and payments $p^t_1, \ldots, p^t_n$ [6]. If the agents in C decide to collude, they can form a coalition and use a collusion scheme, such as those of Section 2.1. Denote the chosen path under the optimal manipulation as $r^* = \langle e^*_1, \ldots, e^*_z \rangle$ and the payments under the manipulation $p^*_1, \ldots, p^*_n$. Some manipulations, such as the optimal manipulation for simple collusion on the cheapest path, do not change the chosen path, so $r^* = r_1$, but increase the payments to coalition members so $p^*_i \le p^t_i$ for any $i \in C$. Other schemes, such as collusion on a non-optimal path, change the selected path, so $r^* \neq r_1$. The coalition members gain payments, but the members on the chosen path, $C \cap r^*$, also incur the cost of their edges. Thus, the utility of the colluder coalition C is: $u^*(C) = \sum_{i \in C} p^*_i - \sum_{i \in C \cap r^*} c_i$. Using monetary transfers, the coalition's utility can be distributed among its members in any way they choose. We define a coalitional game, based on the total utility a coalition of colluders generates its members.

Definition 3 (Path Procurement Collusion Game). *Given a VCG PPA, the value $v(C)$ of a coalition $C \subseteq N$ is: $v(C) = u^*(C)$. In order to manipulate the colluders must trust each other, or sign a certain enforceable contract, so the coalition C is typically be restricted to only a certain subset of the agents.*

Given the above definition, Theorem 1 simply says that in general it is hard to even compute the value of a coalition in the collusion game. However, Theorem 2, Theorem 3 and Theorem 4 all show that for important restricted cases, finding the optimal manipulation is trivial. The above definition of the game also allows us to apply solution concepts to decide how the colluders might share their rewards. The core characterizes *stability*, where no subset of the coalition is incetivised to operate on its own. The Shapley value characterizes a *fair* allocation of the reward, reflecting each member's contribution. Having defined the collusion game, the theme of Example 4 is simple — this network structure results in the collusion game having an *empty core* [7].

One might hope that most network structures result in empty cores, so the colluders would not have a stable way of sharing the reward. If this were the case, the problem of collusion would be mitigated since despite the existence of profitable manipulations, the colluders would fight amongst themselves regarding the monetary transfers, and never form a lasting coalition. Unfortunately, we show that for the common cases of

[6] The subscript t stands for truthful.

[7] Example 4 has disjoint C_t and C_b where $v(C_t \cup C_b) = v(C_t)$ but $v(C_b) > 0$ so $p(C_t \cup C_b) = p(C_t) + p(C_b) \le v(C_t)$. One core constraint is $p(C_t) \ge v(C_t)$ so $p(C_t) = v(C_t) = v(C_t \cup C_b)$ and $p(C_b) = 0$. Another is $p(C_b) \ge v(C_b) > 0$, so some core constraints fail.

simple collusion (along the cheapest path or along a non-optimal path), the game always has a nonempty core, and there is a polynomially computable core imputation. Also, regarding *fairness*, we show that the Shapley value, considered "fair", is also in the core and easy to compute. Thus, the colluders can share the gains in a stable and fair manner [8], making collusion a significant problem in such auctions. Our results are based on showing the game is convex. We show convexity by examining the payment of a simple coalition C (on the cheapest path or on a non-optimal path). Denote the cheapest path as r_1 and the cheapest path that contains no edges in C as r_2. Denote the non colluders on the cheapest path r_1 as $T_{r_1} = r_1 \setminus C$. Denote the cost of a path r as $c(r) = \sum_{i \in r} c_i$, and the cost of the edges in T_{r_1} as $c(T_{r_1}) = \sum_{i \in T_{r_1}} c_i$.

Lemma 1 (Shortest Path Collusion Payments). *Let C be a simple coalition of colluders on the cheapest path. The total payment to the colluders under the optimal manipulation is $P^*(C) = |C|(c(r_2) - c(T_{r_1}))$.*

Proof. From Theorem 2, any colluder $i \in C$ would bid $b_i = 0$. Thus, the formula of Observation 1 is simplified to $p_i = c(r_2) - c(T_{r_1})$ (independent of the colluder's identity). Since there are $|C|$ colluders we obtain $P^*(C) = |C|(c(r_2) - c(T_{r_1}))$.

For collusion on a non-optimal path, we denote the optimal (cheapest) path as r_1 and non optimal path that contains C as r. We denote the non colluders on r as $T_r = r \setminus C$. The total cost of the edges in T_r is $c(T_r) = \sum_{i \in T_r} c_i$.

Lemma 2 (Non-Optimal Path Collusion Payments). *Let C be a simple coalition of colluders on a non-optimal path. The total payment to the colluders under the optimal manipulation is $P^*(C) = |C|(c(r_1) - c(T_r)$.*

Proof. The proof is similar to Lemma 1.

Theorem 5 (Convexity of the Collusion Game). *The collusion game is convex for simple coalitions (along the cheapest path or a non-optimal path).*

Proof. We give the proof for a simple coalition along the cheapest path (the other case is almost identical). An alternative definition of convex games is: $\forall S' \subseteq S \subseteq I, \forall i \notin S$: $v(S' \cup \{i\}) - v(S') \leq v(S \cup \{i\}) - v(S)$. We show this for simple coalition on the cheapest path, S. Consider any $S' \subset S$, denote $S \setminus S' = B$, and let a be any agent in $r_1 \setminus S$. Denote $T = r_1 \setminus S \setminus \{a\}$. Denote $|S| = h$ and $|S'| = l$ (where $l \leq h$), and denote $c(r_2) = x$. Using Lemma 1 we can write $v(S), v(S \cup \{a\}), v(S'), v(S' \cup \{a\})$. We have: $v(S \cup \{a\}) = v(S' \cup B \cup \{a\}) = (h+1)(x - c(T)) - c(S') - c(B) - c_a$; $v(S) = v(S' \cup B) = h(x - c_a - c(T)) - c(S') - c(B)$; $v(S' \cup \{a\}) = (l+1)(x - c(B) - c(T)) - C(S') - C_a$; $v(S') = l(x - c(B) - c_a - c(T)) - c(S')$. Opening parentheses and canceling terms we get: $v(S \cup \{a\}) - v(S) = x + h \cdot C_a - c(T) - C_a$; $v(S' \cup \{a\}) - v(S') = x + l \cdot C_a - c(T) - C_a - c(B)$. However, $l \cdot C_a \leq h \cdot C_a$ and $c(B)$ is non-negative, so we have $v(S' \cup \{i\}) - v(S') \leq v(S \cup \{i\}) - v(S)$.

[8] "Fairness" here is for the colluders — the manipulations are devastating for the auctioneer.

Convexity of the collusion game has implications regarding how the colluders can share the gains. Collusion causes the prices paid to the agents to rise, and monetary transfers allow the colluders to share the utility in any way they desire. Under unstable utility distributions the colluders' coalition is likely to disintegrate, but convexity guarantees a *stable* distribution, so the colluders can distribute the gains so no subset of the colluders would benefit from leaving the coalition. The colluders may also want to share the utility in a *fair* manner, using the Shapley value. In general, even if there are *stable* allocations, the Shapley value may be unstable. Unfortunately, for simple coalitions, a stable allocation always exists, and the Shapley value is also stable.

Corollary 1. *For simple coalitions (on the cheapest path or on a non-optimal path), the collusion game has a non-empty core, containing the Shapley value.*

Proof. The collusion game is convex (Theorem 5), so it has a non-empty core coinciding with the Weber set. The Shapley value is in the Weber set so it is in the core.

A final barrier against collusion is computational complexity. Theorem 1 shows that finding the optimal collusion is hard, but it is trivial for simple coalitions. Corollary 1 guarantees the colluders a fair and stable allocation but it might be hard to compute, even for simple coalitions. We show that for simple coalitions, the colluders can tractably compute a simple core imputation or the Shapley value. Since the game is convex, the Weber set is identical to the core. Given a permutation $\pi = \langle \pi_1, \ldots, \pi_n \rangle$ of the agents and an agent e_i, denote the predecessors of i in π as F^i_π. Denote $m^\pi_i = v(F^i_\pi \cup \{e_i\}) - v(F_i)$, and note this can be computed in polynomial time using Lemma 1 (or Lemma 2). The imputation $\langle m^\pi_1, m^\pi_2, \ldots, m^\pi_n \rangle$ is in the Weber set (the Weber set is the convex hull of all these vectors for different permutations π), and so is a core imputation. A naive way of computing the Shapley value, the centroid of such vectors, requires performing this process for all agent permutations π, requiring exponential time. We show a polynomial algorithm to compute Shapley value.

Theorem 6. *For simple coalitions (on the cheapest path or on a non-optimal path), the Shapley value can be computed in polynomial time.*

Proof. The contribution of edge e_i to coalition C (where $e_i \notin C$), $v(C \cup \{e_i\}) - v(C)$ only depends on $|C|$, not on who the specific members of C are. Due to Lemma 1, we have $p^*(C) = |C|(c(r_2) - c(T^C_{r_1}))$ where $T^C_{r_1} = r_1 \setminus C$ are the non-colluders on the cheapest path. Denote $c(r_1) = x$, $c(r_2) = y$ and $\sum_{i \in C} c_i = z$. We have $p^*(C \cup \{e_i\}) - p^*(C) = (|C|+1)(y - c(r_1 \setminus C \setminus \{e_i\})) - |C|(y - c(r_1 \setminus C)) = (|C|+1)(y - x + z + c_i) - |C|(y - x + z) = |C|(y - x + z + c_i - y + x - z) + (y - x + z + c_i) = |C| \cdot c_i + y - x + z + c_i$. Collusion on the cheapest path does not change the chosen path, so we have: $v(C \cup \{e_i\}) - v(C) = -\sum_{j \in C \cup \{e_i\}} c_j + p^*(C \cup \{e_i\}) + \sum_{j \in C} c_j - p^*(C) = -c_i + |C| \cdot c_i + y - x + z + c_i = |C| \cdot c_i + y - x + \sum_{i \in C} c_i$.

Consider computing the Shapley value for e_i, $\phi_i(v) = \frac{1}{n!} \sum_{\pi \in \Pi} v(F^i_\pi \cup \{e_i\}) - v(F^i_\pi)$ (where F^i_π are the predecessors of i in π). Denote $\psi_i(v) = \sum_{\pi \in \Pi} v(F^i_\pi \cup \{e_i\}) - v(F^i_\pi)$. We can compute ψ_i by iterating over the possible *numbers* of predecessors i has in π, $|F^i_\pi|$. Let Π_j be all permutations $\pi \in \Pi$ such that $|F^i_\pi| = j$ (i.e. permutations where i has exactly j predecessors. We can denote the total contribution that i has for coalitions of size j as $M_j = \sum_{\pi \in \Pi_j} v(F_\pi^i \cup \{e_i\}) - v(F_\pi^i)$. Thus we have $\psi_i(v) = \sum_{j=0}^{n-1} M_j$. Thus we only need to compute M_j in polynomial time (for $0 \leq j \leq n-1$).

To compute $M_j = \sum_{\pi \in \Pi_j} v(F_\pi^i \cup \{e_i\}) - v(F_\pi^i)$ we can sum over all possible predecessor sets for i where i has exactly j predecessors, $F = \{F_\pi^i \subseteq N | \pi \in \Pi_j\}$, where $|F| = \binom{n-1}{j}$. Under this notation $M_j = \sum_{C \in F} v(C \cup \{e_i\}) - v(C)$. We've shown that $v(C \cup \{e_i\}) - v(C) = |C| \cdot c_i + y - x + \sum_{i \in C} c_i$, so we have: $M_j = \sum_{C \in F} |C| \cdot c_i + y - x + \sum_{i \in C} c_i$. Since any $C \in F$ have the same size $|C| = j$, we get: $M_j = |F| \cdot (j \cdot c_i + y - x) + \sum_{C \in F} \sum_{i \in C} c_i$. We denote $q = \sum_{C \in F} \sum_{i \in C} c_i$. Thus, $M_j = \binom{n-1}{j} \cdot (j \cdot c_i + y - x) + q$. Consider computing q. Given the coalition C of size $|C| = m$, denote the weights c_i for all $i \in C$ as $W = \langle c_1, \ldots, c_{i-1}, c_{i+1}, \ldots, c_m \rangle$ (W is the set of the costs of all colluders except e_i). Thus, q is simply the sum of weights in all subsets of W of size j, i.e. $q = \sum_{S \subset W || S| = j} \sum_{s \in S} s$. Any weight $w_i \in W$ appears in q exactly $\binom{n-1}{j-1}$ times, so $q = \binom{n-1}{j-1} \sum_{c_x \in W} c_x = \binom{n-1}{j-1} \sum_{e_x \in C \setminus \{e_i\}} c_x$. Given a colluder e_i, we can easily compute $\sum_{e_x \in C \setminus \{e_i\}} c_x$ in polynomial time, and thus compute q in polynomial time. This allows us to compute $M_j = \binom{n-1}{j} \cdot (j \cdot c_i + y - x) + q$ in polynomial time, for any j. We can thus compute $\psi_i(v) = \sum_{j=0}^{n-1} M_j$ in polynomial time, and thus compute the Shapley value of any agent in polynomial time.

4 Related Work

Auctions face untruthful selfish agents, so due to strategic behavior, a mechanism trying to maximize welfare may reach a sub-optimal decision. Proper payment rules incetivize agents to bid truthfully. A prominent scheme for doing so is the VCG mechanism [23, 10, 13]. Despite its advantages, VCG has many shortcomings [3], including vulnerability to collusion [17]. Collusion can occur in many domains and many of its forms are illegal [17]. Our model follows an analysis of multi-unit auctions [4], but for the PPA domain [1]. We examine coalitional deviations, but as opposed to strong Nash equilibrium [2], we convert the normal-form game to a cooperative game. We provide an *internal* model of collusion, as opposed to *external interventions* models [18, 6, 19]. We focus on the core [12] and Shapley value [20]. The Shapley value and other power indices are typically hard to compute [16, 8], so our result for computing it in some collusion games is interesting. For general collusion games, the colluders can *approximate* [7, 16] the Shapley value in order to share their gains. Collusion is also related to shills and false-identity attacks [26, 9, 5, 27], where a single agent pretends to be several agents. A single edge in a VCG PPA can pretend to be several edges to increase payments. Our work is also related to of bidding rings and clubs [15, 14], but we assume the colluders have full information on each other's costs. Core selecting auctions [11] are also related to our work, but we do not consider the auctioneer a participating agent, and the core in our collusion game can be empty.

5 Conclusion

We analyzed collusion in VCG PPAs, and showed that such a domain is vulnerable to collusion. Some questions remain open for future research. First, similar analysis can be done for other auctions, such as combinatorial auctions or sponsored search auctions. Second, due to the drawbacks of VCG for PPAs, what alternative mechanisms should be used? Finally, what bids are likely to occur for domains with an empty core?

References

1. A. Archer and É. Tardos. Frugal path mechanisms. *ACM Transactions on Algorithms (TALG)*, 2007.
2. R. Aumann. Acceptable points in general cooperative n-person games. In *Contributions to the Theory of Games*, volume IV, pages 287–324. 1959.
3. L. M. Ausubel and P. Milgrom. The lovely but lonely vickrey auction. In *Combinatorial Auctions*, chapter 1, pages 17–40. 2006.
4. Y. Bachrach. Honor among thieves: Collusion in multi-unit auctions. In *AAMAS*, 2010.
5. Y. Bachrach and E. Elkind. Divide and conquer: False-name manipulations in weighted voting games. In *AAMAS*, pages 975–982, 2008.
6. Y. Bachrach, E. Elkind, R. Meir, D. Pasechnik, M. Zuckerman, J. Rothe, and J. Rosenschein. The cost of stability in coalitional games. *Algorithmic Game Theory*, pages 122–134, 2009.
7. Y. Bachrach, E. Markakis, E. Resnick, A. Procaccia, J. Rosenschein, and A. Saberi. Approximating power indices: theoretical and empirical analysis. *Autonomous Agents and Multi-Agent Systems*, 20(2):105–122, 2010.
8. Y. Bachrach and J. S. Rosenschein. Computing the Banzhaf power index in network flow games. In *AAMAS*, pages 323–329, Honolulu, Hawaii, May 2007.
9. C. Chen and Y. Tauman. Collusion in one-shot second-price auctions. *Economic Theory*, 28:145–172, 2006.
10. E. H. Clarke. Multipart pricing of public goods. *Public Choice*, 11(1):17–33, 1971.
11. R. Day and P. Milgrom. Core-selecting package auctions. *International Journal of Game Theory*, 36(3):393–407, 2008.
12. D. B. Gillies. Some theorems on n-person games. 1953.
13. T. Groves. Incentives in teams. *Econometrica*, 41:617–663, 1973.
14. K. Leyton-Brown, Y. Shoham, and M. Tennenholtz. Bidding clubs: institutionalized collusion in auctions. In *ACM EC*, 2000.
15. K. Leyton-Brown, Y. Shoham, and M. Tennenholtz. Bidding clubs in first-price auctions. In *AAAI*, 2002.
16. Y. Matsui and T. Matsui. A survey of algorithms for calculating power indices of weighted majority games. *Journal of the Operations Research Society of Japan*, 43, 2000.
17. P. Milgrom. *Putting Auction Theory To Work*. Cambridge University Press, 2004.
18. D. Monderer and M. Tennenholtz. Strong mediated equilibrium. *Artificial Intelligence*, 173(1):180–195, 2009.
19. E. Resnick, Y. Bachrach, R. Meir, and J. Rosenschein. The cost of stability in network flow games. *Mathematical Foundations of Computer Science 2009*, pages 636–650, 2009.
20. L. S. Shapley. A value for n-person games. *Cont. Theory of Games*, 1953.
21. L. S. Shapley. Cores of convex games. *Internat. J. Game Theory*, 1:12–26, 1971.
22. A. Sullivan and S. Sheffrin. *Economics: principles in action*. Prentice Hall, 2003.
23. W. Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of Finance*, 16(1):8–37, 1961.
24. R. J. Weber. Probabilistic values for games. Technical report, Cowles, 1977.
25. E. Winter. The Shapley Value. *Game theory with economic applications*, 2002.
26. M. Yokoo, V. Conitzer, T. Sandholm, N. Ohta, and A. Iwasaki. Coalitional games in open anonymous environments. In *AAAI*, 2005.
27. M. Zuckerman, P. Faliszewski, Y. Bachrach, and E. Elkind. Manipulating the quota in weighted voting games. In *AAAI*, 2008.