Vickrey-Clarke-Groves Auction-Based Storage Allocation for Distributed Camera Systems

Martins, Alexandre; Wei, Hung-Yu; Årzén, Karl-Erik

Published in:
Proceedings of the 6th IFAC Conference on Telematics Applications

2022

Document Version:
Publisher's PDF, also known as Version of record

Vickrey-Clarke-Groves Auction-Based Storage Allocation for Distributed Camera Systems ⋆

Alexandre Martins * **Hung-Yu Wei** ** **Karl-Erik Årzén** *

* *Department of Automatic Control, Lund University*
** *Department of Electrical Engineering, National Taiwan University, Taiwan*

Abstract: Video surveillance systems are critical infrastructures and that growing in size and complexity. Storage space is the prime resource in such systems but current surveillance setups are centralized and limited in resources due to security and cost constraints. Allocating the correct amount of storage to each camera considering their large differences in characteristics and video content is challenging. In this paper we propose a game theoretic approach to storage allocation for video surveillance camera systems based on the Vickrey-Clarke-Groves (VCG) auction mechanism.

Keywords: Cameras, Computer control, Constraints, Auction theory, Game theory, VCG auctions, Video

1. INTRODUCTION

The number and size of camera systems used, *e.g.*, in different types of public spaces, are growing due to the Internet of Things (IoT) trend and they are currently one of the major storage and bandwidth consumers. With growing demands on high resolution, high frame rate and level of detail, the amount of storage needed to retain these videos is a growing problem. Surveillance installations are usually critical installations and are mostly running on dedicated infrastructures, storing video in trusted servers owned by systems administrators. Newer installations are usually large scale (commonly hundreds of cameras), heterogeneous and have large differences in resource requirement. (IPVM, 2021).

In this paper the focus is video surveillance systems based on H.264 video cameras, the most prevalent system on the market today. H.264 is a video compression standard based on block-oriented and motion-compensated coding (ITU-T, 2010). A model of the bandwidth generated, and hence, storage needed, by a H.264 video surveillance camera was presented in (Edpalm et al., 2018a,b). The model provides an estimate of the bandwidth needs for a H.264 video, given current scene conditions and specific camera parameters. It allows to calculate the long term resource needs for the camera as long as it keeps the current parameters. Anticipating the amount of storage and bandwidth needed by each camera is difficult due to the uniqueness of each scene, camera characteristics and parameters. The amount of storage available is limited and is one of the main cost of running the system (IPVM, 2021). Furthermore, the cameras compete (or are at the least not explicitly incited to cooperate) for the storage resources available. As a result the system administrators can not trust the devices to provide their real valuation.

This creates a need for strategies to determine the allocation of storage resources that do not rely on trustworthy information being shared between cameras and storage units. We propose the use of auction theory, in particular the Vickrey-Clarke-Groves (VCG) mechanism (Krishna and Perry, 1998), to decide how to allocate these resources. The advantage of VCG auctions is that they provide guarantees, in particular enforcing a fair and envy-free allocation (Pápai, 2003) (an outcome in which each agent does not envy what some other agent has obtained) without requiring control over all devices participating in the auction. Specifically, in this paper we propose a solution to allocate storage resources in a competitive camera system while separating the resource providers (*i.e.*, the storage units) from the resource buyers (*i.e.*, the cameras). The buyers have private information on the amount of resources needed and aim to maximize their valuation (in this case to minimize the compression of their video stream). The storage units enforce the constraint on resource availability by solving a constrained knapsack problem to allocate the resources (see Section 4). This paper focuses on the auction framework and utility determination. The cameras are not explicitly cooperating as the system could be running cameras from different providers which prevents explicit cooperation between devices, and the storage provider acts as a ring buffer as explained in (Martins et al., 2020). We chose to use auction theory to distribute the available resources in the best way possible without relying on the devices truthfulness.

⋆ This work has been partially funded by the Wallenberg AI, Autonomous Systems and Software Program (WASP), the ELLIIT strategic research area on IT and mobile communications, and the Nordforsk university hub on Industrial IoT (HI2OT).

The contributions of this paper are:

- A game theoretic approach based on VCG auctions for storage allocation in camera systems for video surveillance is proposed.
- A utility measure for camera systems based on the video compression value and its variation is proposed.
- Simulation results of the storage allocated for video content is done to validate the game theoretic proposed solution as well as its system resource cost.

2. RELATED WORK

Centralized bandwidth allocation techniques using control theory with maximization of the system-wide visual quality have been proposed in (Seetanadi et al., 2018) and (Silvestre-Blanes et al., 2011). Second price auctions have been applied to video surveillance systems mostly for specific applications such as area overage (Ding et al., 2012; Konda et al., 2016; Dieber et al., 2011), camera placement (Elhamifar and Vidal, 2009; Ermis et al., 2010) and object tracking (Qureshi and Terzopoulos, 2009; Sankaranarayanan et al., 2008). Auction theory has been previously used in various computer science applications such as content delivery delay and cashing cost minimization for large mobile networks involving multiple stakeholders as reported in (Li et al., 2016; Ghosh et al., 2004; Pillai and Rao, 2016). There are also various studies on resource management in cloud computing, mostly focusing on virtual machine resource allocation (Xu and Yu, 2014), some of which are using knapsack optimization to allocate resources, *e.g.*, (Vanderster et al., 2009), or Stackelberg game allocation of CDN resources, *e.g.*, (Li et al., 2016; Hung et al., 2018) or device to device communications, *e.g.*, (Sawyer and Smith, 2019). Auction theory has also been applied to spectrum sharing in mobile networks (Suris et al., 2007; Cramton et al., 2002) and task allocation to mobile devices such as (Wang et al., 2017) where the VCG mechanism is used to allocate computation tasks to mobile devices or using consensus-based auctions (as explained in (Zlot, 2006)) to ensure consensus between mobile robots, *e.g.*, (Brunet et al., 2008; Choi et al., 2009; Hunt et al., 2014) or (Nanjanath and Gini, 2006). However, to the authors' best knowledge, auction theory has not been applied to storage allocation for video surveillance systems before.

3. SYSTEM DESCRIPTION

We consider a simplified camera system with one storage unit $\mathcal{P}$ (physically realized as Network Attached Storage, Cloud storage, or some other technique) and C video cameras indexed by i: $\{c^1, c^2, \ldots, c^C\}$. An overview of the system with $C = 4$ is shown in Figure 1. Typically a video surveillance system is owned by a security department, which buys or rents storage from an IT department or a cloud provider at a fixed rate. In our system, viewing quality is most important. The main system goal is to maximize the overall global video quality given the current system constraints, i.e., the running cost and the video storage size.

All cameras can communicate with the seller and can, e.g., be part of the same virtual network. At the beginning of a

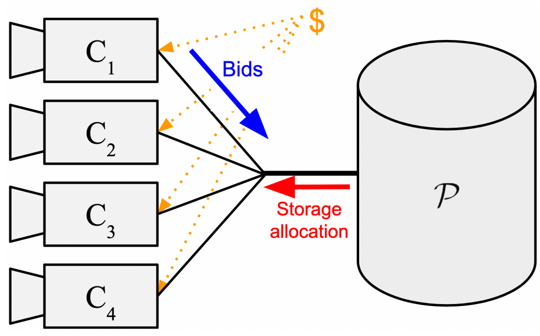

Fig. 1. The simplified system considered in this paper.

predefined period k, e.g., an hour, a day, a week, etc, the cameras can buy storage from the seller to save the video they generate during the coming period, using the money at their disposal. If the cameras run out of storage, they need to wait until the next period to buy more. At each period, k, the cameras obtain an amount of money, m, that they can use to buy resources. The amount they receive depends on the cost of running the system. Each camera has a virtual account holding the money it may use. Any remaining money can be saved for future periods. The amount of money available for camera c^i to buy storage at the beginning of each period k is

$$m^i(k) = m^i(k-1) + m,$$

where $m^i(k-1)$ is the money accumulated and available from previous rounds (after all previous payments have been made).

4. VICKREY–CLARKE–GROVES (VCG) AUCTIONS

Vickrey–Clarke–Groves (VCG) is a combinatorial auction mechanism known to yield efficient outcomes, with desirable properties such as incentive compatibility (players' best interest is to reveal their true valuation) and individual rationality (players will always benefit from entering the auction) provided the resource allocation is optimal, (Maillé and Tuffin, 2007; Nisan and Ronen, 2007). VCG auctions apply to any problem where players have a quasi-linear utility function (the compression values being the linear argument).[1] In our case the seller is the storage unit which also acts as the auctioneer. The players are the cameras who want to buy storage, i.e. they are the buyers. The seller is provided a set of proposals, or bids, A, indicating the amount as well as the value that the buyer is willing to pay for this amount. Each buyer sends multiple bids, typically one per compression level. A subset of A, consisting of maximum one bid from each buyer, satisfying the constraints will be selected by the seller, leading to an outcome a, which can be considered as an allocation vector. The value that buyer i obtains from this outcome is denoted θ^i. The price p^i that the buyer i pays for the decided outcome a is determined by the VCG mechanism, see (3).

The utility of the buyer is the difference between the willingness to pay θ^i and the price p^i it is charged for it:

$$u^i(a, p^i) = \theta^i(a) - p^i \tag{1}$$

The buyers aim at maximizing this utility.

[1] Quasi-linear utility functions are linear in one argument

We assume that buyers are provided with a regular cash inflow in order to be able to buy resources, typically this would be a budget allocated, *e.g.* every auction period, to each camera by the system owner which the buyers can use at their own discretion.

Vickrey–Clarke–Groves auctions work as follows:

(1) Each buyer i is asked to reveal his valuation function $\widetilde{\theta}^i$ which indicates how the buyer values each outcome a. The revealed valuation $\widetilde{\theta}^i$ could differ from the real valuation function θ^i if player i is not truthful.
(2) The auction mechanism computes an outcome $a^*(\widetilde{\theta})$ that maximizes the declared social welfare, *i.e.,* the sum of revealed valuations $\sum_i \widetilde{\theta}^i$, given the constraints, using 0-1 knapsack optimization, see (4).
$$a^*(\widetilde{\theta}) \in \operatorname*{argmax}_{a \in A} \sum_i \widetilde{\theta}^i(a) \tag{2}$$
(3) The price paid by each buyer is given by the loss of declared welfare which the buyer imposes to the others through his presence in the auction, meaning the value that other buyers lose through the difference between the current outcome a^* and an alternate optimal outcome $a' \in a$ without buyer i. For this, we solve one 0-1 knapsack problem per buyer without the items from buyer i present, to find the outcome a' if i was not present, see Eq. 4.
$$p^i = \max_a \sum_{j \neq i} \widetilde{\theta}^j(a') - \sum_{j \neq i} \widetilde{\theta}^j(a^*) \tag{3}$$
The VCG mechanism is a second-price auction mechanism. Each buyer is declaring its real value and the price paid is the loss of declared welfare which the buyer imposes on the other buyers, as such the buyer will always end up paying less than the declared value. This property is enforced thanks to the secondary knapsack problem in Equation (3), which is done to calculate this loss of welfare to other buyers.

The VCG mechanism verifies three properties (Krishna and Perry, 1998):

- **Incentive compatibility.** For each user, bidding truthfully (i.e. declaring $\widetilde{\theta}^i = \theta^i$) is a dominant strategy, meaning it is the strategy which will provide the maximum value.
- **Individual rationality.** Each truthful player obtains a non-negative utility, meaning it is advantageous to be truthful (see Eq. 1).
- **Efficiency.** When players bid truthfully, the social welfare, $\sum_i \theta^i$, is maximized.

The auction mechanism decides on the optimal outcome by solving a 0-1 knapsack problem. It is a problem in combinatorial optimization: We pick a set of items (given by the bids), each with a weight and a value. We want to determine the items to include in a collection so that the total weight is less than or equal to a set limit and the total value is as large as possible. In the 0-1 version of this problem each item is indivisible and cannot be picked more than once. Each buyer $i \in [1..C]$ participating in the VCG mechanism provides n bids, indexed by the letter j. The total number of bids for the whole system is thus $N = n \times C$. Each bid contains an item weight w_j^i and its associated declared value v_j^i (weight and value of bid j from buyer i). The declared value is given by the valuation function $v_j^i = \theta(w_j^i)$.

We add one source constraint per buyer to the classical 0-1 knapsack problem, expressing that the optimization is only allowed to select at most one bid j from each buyer i. There is no guarantee that each buyer will have one of its bids accepted whereas no buyer can have more than one of its bids accepted. There will therefore be at most C items selected by this allocation (as there are C buyers). The total weight possible is W and x_j^i indicates if an item j from buyer i is selected ($x_j^i = 1$ if item j from buyer i is selected and 0 otherwise). The modified 0-1 knapsack problem is

$$\begin{aligned} &\text{maximize } \sum_{i=1}^{C} \sum_{j=1}^{n} v_j^i \cdot x_j^i \\ &\text{subject to} \\ &\quad \sum_{i=1}^{C} \sum_{j=1}^{n} w_j^i \cdot x_j^i \leq W \\ &\quad \forall i \in [1..C] \, (\sum_{j=1}^{n} x_j^i \leq 1) \end{aligned} \tag{4}$$

with $v_j^i = \theta^i(w_j^i)$ and $x_j^i \in \{0, 1\}$.

5. ESTIMATION OF RESOURCE NEEDS

5.1 Storage provider

At each auction period $k = \{1, 2, ...\}$ the storage provider $\mathcal{P}$ sells units of storage (in Gigabyte, Gb). The time between auctions has a duration of T. The total quantity of storage available by storage provider $\mathcal{P}$ is denoted S, the amount of storage for sale at each auction is denoted $s(k)$. $s(k)$ is a subset of S, the total amount of storage available, and the time, R, the data needs to be kept. R is usually determined by the system owner policy. In this paper we define simply $s(k) = \frac{S*T}{R}$, meaning we evenly split the total amount available by the retention time to get the amount to allocation for each auction k of duration T. The storage space allocated to camera c^i is denoted $s^i(k)$. The expected quantities are annotated with a $*$ superscript, *e.g.* the expected allocated storage to camera c^i is denoted $s^{i*}(k)$. Only the storage unit has storage space available, i.e., the cameras are not storage providers.

The storage provider $\mathcal{P}$ acts as a ring buffer, deleting the oldest allocated data in order to accommodate new incoming data. Data stored during the oldest auction period is deleted in order to reuse the allocated storage for the new period k: $\sum_{t=k-R/T}^{k} s(t) \leq S$. The storage provider will then assign the storage following the VCG mechanism's allocation rule, which means that for each auction period k, $\sum_{i=1}^{C} s^i(k) \leq s(k)$ with $s^i(k) \geq 0$ for all cameras c^i.

5.2 Storage buyers (cameras)

In order for a camera to define its storage valuation, θ^i (since VCG is a truthful mechanism, $\widetilde{\theta}^i = \theta^i$), it needs to

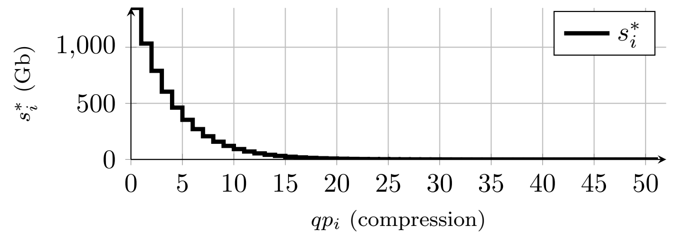

Fig. 2. An example of the expected storage as a function of the compression, *i.e.* $s^{i*}(qp)$.

know how much storage is needed for a video of a certain quality.

For H.264 videos, the compression level is defined by the quantization parameter qp. This value indicates how much data is lost by quantifying the residual data after the transformation step of the encoding process (ITU-T, 2010). qp is an integer between 0 and 51 ($qp \in \{0..51\}$), 0 being lossless and 51 being the highest compression level. The compression level affects both the memory requirements and the visual quality of the video. The higher the compression, the smaller the amount of storage needed per auction period T will be.

The estimated amount of storage that the camera c^i needs for each H.264 quantization level qp is denoted by $s^{i*}(qp)$ and is calculated using the frame size estimation model provided in (Edpalm et al., 2018b). The model returns the expected amount of storage required for a video with constant scene parameters (motion in the scene, light level, etc.) and settings of the specific camera (frame rate, group of picture length, etc.) for a given qp. An example of this function is shown in Fig.2. The estimation model provided in (Edpalm et al., 2018b) is of the form:

$$s^{i*}(qp) = \alpha \times 2^{-qp/6} + \beta \times 5^{-qp/6}, \tag{5}$$

where α and β are positive real numbers defined by a combination of scene and camera parameters. The function $s^*(qp)$ is a positive monotonously decreasing function defined in $\mathbb{R}^+$. This means that it is invertible and we can find $qp^*(s^{i*}) = s^{*-1}(qp)$. In practice the inverse function is found by computing all 52 values of $s^{i*}(qp)$ and inverting the resulting table. Each camera c^i will at the beginning of auction k calculate the $s^{i*}(qp)$ function for all 52 qp values given measurements of the actual scene conditions and, hence, the actual values of the α and β parameters.

6. VALUATION OF RESOURCES

The quality of an H.264 video is directly linked to the compression parameter qp of the video and the variations in qp, see (Cermak et al., 2011; Nemethova et al., 2004; Singh et al., 2012). In these papers the authors correlated the compression parameter and its variation to the perceived video quality using Mean Opinion Score (MOS) testing, a method to assess video quality by collecting the opinions of participants in a controlled environment. The authors found that the perceived degradation in video quality is strongly noticed at higher compression levels but hardly perceptible at lower levels. Moreover, large variations (jumps) in the compression level are easily noticed by viewers. It means that from a viewer perspective, moving away from the high compression levels is more valuable than getting closer to a very low compression level, i.e., the visual quality gain at low compression levels is hardly noticeable while it would be more noticeable at high compression levels. This is shown in Fig. 4 where the derivative/slope of the red curve is higher towards the higher compression levels. The same applies for compression variations over time, avoiding large changes in compression is more valuable than moving towards no variation.

6.1 *Valuation function*

In order to minimize the negative effects of high compression and compression jumps as well as take into account their cost to the viewers, we propose a model of the valuation which embodies the desired characteristics, derived from the simple equation of an ellipse. The valuation function θ^i of buyer C^i is defined as

$$\theta^i(qp^i, m^i) = m \cdot \sqrt{1 - \left(\frac{qp^i + \sigma_n(qp^i)}{2 \cdot 51}\right)^2} \tag{6}$$

where m^i is the money available for the camera c^i, qp^i is the compression value corresponding to the received amount s^i, and $\sigma_n(x)$ is the standard deviation of x over the n last periods.

The function embodies our system objective, i.e., to retain video of the highest possible quality in the system given the available money m^i, where quality is measured by the video compression level, qp^i, and how much it varies. The equation of an ellipse has an interesting characteristic around its vertexes. The derivative of the ellipse is low when approaching the co-vertex (low qp and low $\sigma(qp)$), while it is high close to the vertex (high qp and high $\sigma(qp)$). It is valued more (high derivative) to move away from high qp and high $\sigma(qp)$ values (vertex) than it is to get closer to the low qp and low $\sigma(qp)$ values (co-vertex).

6.2 *Utility and bid choices*

The utility u^i of buyer c^i is then given by

$$u^i(qp^i, m^i, p^i) = \theta^i(qp^i, m^i) - p^i \tag{7}$$

where p^i is the price paid to obtain the amount of storage s^i. The utility is the difference between the value it obtained from the seller and the price it paid to acquire it. An example of $\theta^i(qp^i, m^i)$ with $m^i = 1$ is shown in Fig. 3 and an example with $m^i = 5$ can be seen in Fig. 4 (blue curve). In Fig. 4 one can also see the predicted amount of storage expected per qp value (black curve) as well as the valuation of each qp values with $m^i = 5$ and fixed θ^i (red curve).

To determine the size of the bids in terms of memory, camera c^i calculates $s^{i*}(qp)$. The value of each quantity in $s^{i*}(qp)$ is the associated valuation $\theta^i(qp^i, m^i)$, One example of how the bids are decided is shown in Fig. 4 (blue curve). One can see that the bids b^i (last plot) are a combination of the expected storage s^{i*} (black curve) and the valuation θ^i (red curve). Each bar in the blue curve represent a bid (an amount of storage s^i and the associated value θ^i).

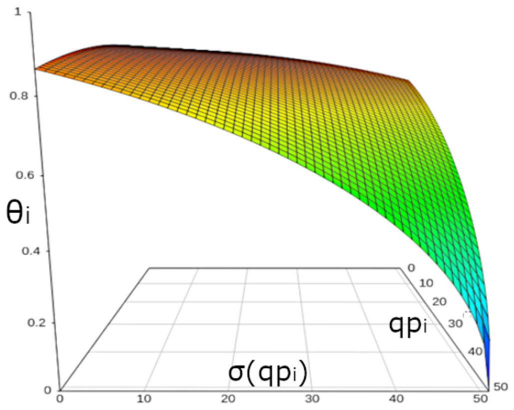

Fig. 3. An illustration of how the valuation of resources, θ^i, depends on the compression and its variation.

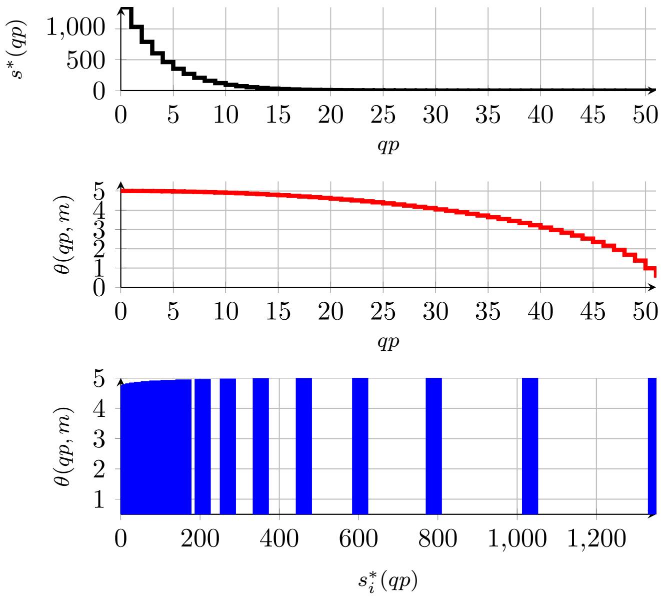

Fig. 4. Bid example (with m = 5). The top plot shows the expected storage amount, $s^{i*}(qp)$. The middle plot shows the valuation of the compression, $\theta^i(qp, m)$. The last plot shows the sent bids b^i.

6.3 Auction steps

The different steps in the VCG mechanism can be summarized as follows:

(1) Camera c^i gets a fixed amount of money m for the new period k and adds it to m^i.
(2) Camera c^i calculates $s^{i*}(qp)$ and $\theta^i(qp^i, m^i)$ given the actual parameters and sends its 52 bids b^i (one per possible qp): (s^{i*}, θ^{i*}) to the storage provider $\mathcal{P}$.
(3) Storage provider $\mathcal{P}$ waits for some time to receive all bids b^i from buyers c^i.
(4) Storage provider $\mathcal{P}$ solves a 0-1 knapsack problem with the received bids b^i to find the optimal allocation of available resources, see Section 4.
(5) Storage provider $\mathcal{P}$ solves one 0-1 knapsack problem per buyer to calculate the payments of cameras c^i defined by Equation (3) of the VCG mechanism.
(6) Storage provider $\mathcal{P}$ sends the allocated storage amount s^i and price p^i to camera c^i.
(7) Camera c^i pays the storage provider $\mathcal{P}$ the required amount p^i and starts streaming data up to the allocated storage amount s^i.

The optimization problem solved is explained in Section 4. The total amount of storage available for sale from the provider $\mathcal{P}$ is S (which is the same as W in Equation 4). x_j^i still indicates if an item j from camera c^i is selected or not ($x_j^i = 1$ if item j from C^i is selected and 0 otherwise).

Each item $j \in [1..n]$ from buyer i has a storage amount s_j^i (denoted w_j^i in Equation 4), and an associated value v_j^i such that $v_j^i = \theta^i(s_j^i)$.

The result of the optimization is the storage amounts s_j^i allocated to each camera c^i, providing the highest possible sum of valuations $\sum_{i=1}^{C} v^i$ depending on the storage resource limitation S of the provider $\mathcal{P}$. As indicated before, $v^i = \theta^i(s^i)$ where $\theta^i(s^i)$ is the declared value of the storage which was given by camera c^i. This value is related to the visual quality of the streamed video as shown in Fig 3.

7. RESULTS

To evaluate the utility function and the use of the VCG mechanism, we implemented a simulator in python and ran multiple simulations with independent players (seller and buyers) communicating via queues.

The 0-1 knapsack problem solving is computationally costly. Because the optimization problem is strongly NP-hard, *i.e.* there is no pseudo-polynomial algorithm to solve it (Google, 2021a). This could prevent the system to scale due to time constraints. In this paper we use the Google® Ortools library (v.9.0.9048) (Google, 2021b) which uses the SCIP mixed integer programming solver (SCI, 2021) on an Apple® MacBook Pro with an octacore M1 ARM®-based chip with 3.2 GHz maximum core frequency (Wikipedia, 2021).

The simulations used the Apple® emulator to convert to ARM instructions which slows down the computation but multi-threading was used with up to 10 simultaneous threads. In order to see how long it would take to run a complete VCG auction period, *i.e.*, with $C + 1$ knapsack optimizations (C being the number of buyers), we ran 5 simulations with random buyer parameters, 100 auctions each and calculated the mean and standard deviation of the time it took to complete all the required optimizations at each period. The results are shown in Fig. 5. As can be seen, the time to complete the VCG auction follows what appears to be a quadratic function of the number of buyers. As such a system with 50 cameras would solve the assignment of each period in approximately 4.5 seconds, a system with 250 cameras would do so in approximately 3 minutes and a system with 450 cameras would take approximately 12 minutes to do the same task. As long as the auction periods are on the order of an hour, this relatively short time required for computing the allocation is acceptable. If the computation time is a limiting factor one could include a delay of one period in the auction.

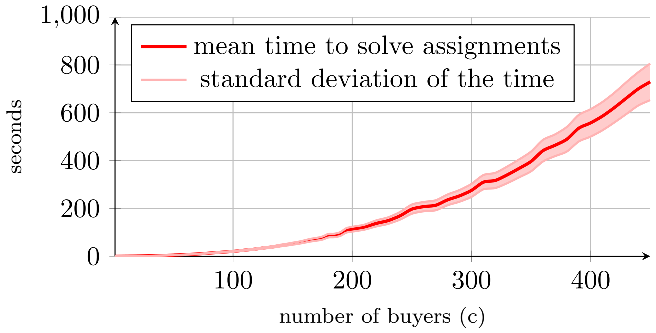

Fig. 5. Mean and deviation of the time (in seconds) to solve the assignments per number of buyers C for each period k.

For clarity of presentation the following simulations only use four cameras. c^1 is a 4K camera and as such requires the most storage quantity, c^2 is a 1080p camera, c^3 is a 720p camera and c^4 is a 480p camera. The simulation parameters used are: $W = 50$ Gb, $R = 10$ hours, i.e., there is $s(k) = 5$ Gb/hour for sale, and $m = 5\$$ is given to each camera at each period of $k = 10$ hours. Each simulation has fixed camera parameters (resolution, size of the Group of Pictures (GOP), and frame rate) and random video parameters (motion level, light level, noise level, etc). The storage needs of the cameras are determined using the model described in (Edpalm et al., 2018b). The cameras have no explicit incentive to cooperate and try to maximize their own utilities. The values of qp^i in the presented figures are the average values over the period k. Random noise is added to the video frame sizes.

As soon as we consider a competitive system, the buyers, c^i, have no incentive to accommodate other players or provide truthful information if it is not in their favor. For these reasons we compare the VCG mechanism approach (which provides truthfulness as a property) with an equal split of the total resources between the buyers, since this does not assume any information from the cameras. Hence, we compare:

(1) Splitting $s(k)$ equally between the buyers c^i.
(2) Using the proposed VCG mechanism.

We run two types of simulations for each assignment:

(1) The video parameters, change significantly (but in a realistic way) every 10 auction periods.
(2) The video parameters are changed randomly every auction period with uniform distributions.

Fig.6 shows the simulation results of the equal allocation (the plots to the left) and VCG allocation (the plots to the right) with low variation of the video parameters (simulation type 1). Fig.7 instead shows the second type of simulation with rapid random changes of the video parameters. The uppermost plots contain the qp^i values of the four cameras, the ones below show the amount of storage s^i allocated and the third plot shows the valuation of the resource acquired, *i.e.,* θ^i, with $m^i = 1$.

We can see in Fig.6 that the VCG approach allocates more storage $s^i(k)$ to c^1 (the 4K camera) which allows c^1 to achieve a lower qp (so better visual quality is perceived by the viewer) by assigning less storage to c^2, c^3 and c^4. The overall visual quality in the VCG allocation case would be more uniform as the camera requiring more storage would be allocated more, which is what the proposed solution aims for.

In Fig.7 we can observe the same behavior as in Fig.6, the VCG mechanism allocates more storage to c^1 allowing it to have a better quality level. Moreover, the variations in qp are also less pronounced than with equal allocation as the cameras c^i react to the parameter changes, modifying the bids b^i accordingly which in turn affects the allocation of storage s^i. By comparing the θ^i curves, we can see the advantage of the VCG mechanism: the video streamed by cameras c^i present closer qp values and less qp deviations which should provide a better system-wide visual quality for the viewer (Cermak et al., 2011; Nemethova et al., 2004; Singh et al., 2012).

One can observe that the VCG approach in this scenario leads to a storage assignment that is very close to a split of the storage resource that is proportional to the camera resolution. However, the latter policy would require the cameras to report this information to the storage, and hence, open up for cameras to provide untruthful information. Using the VCG approach, this assignment is obtained without any need for this information.

8. CONCLUSIONS AND FUTURE WORK

In this work we proposed a storage resource allocation method based on the VCG mechanism with the utility derived from the compression level and its variation. The approach requires limited system knowledge and the results in term of system-wide video quality are encouraging. As discussed in (Maillé and Tuffin, 2007), VCG auctions present at least one prohibitive drawback when compared to simpler allocation, as they need computationally intensive NP-complete optimization problems to be solved. The amount of optimizations grows with the number of cameras in the system as each added camera comes with 52 news bids for the optimization problems and one extra optimization for the payment calculation. This still seems to be computationally feasible for the considered systems (hundreds of cameras with allocations every hour or so). The approach has interesting properties (incentive compatibility, individual rationality and efficiency) for systems with competitive players. Thanks to these properties the seller can be unaware of the camera parameters and efficiently allocate the storage based solely on the cameras' declared values. A logical extension of this paper would be to handle multiple storage providers and incorporate learning in the valuation of resources from the cameras. Also, the video quality is here considered to be correlated with the video compression. An alternative approach would be to use an application-specific metric or a recognized quality metric such as the structural similarity index measure (SSIM), peak signal-to-noise ratio (PSNR) or other metrics as described in (Yang, 2007), but at the expense of additional complexity for the cameras.

REFERENCES

Brunet, L., Choi, H.L., and How, J. (2008). Consensus-based auction approaches for decentralized task assign-

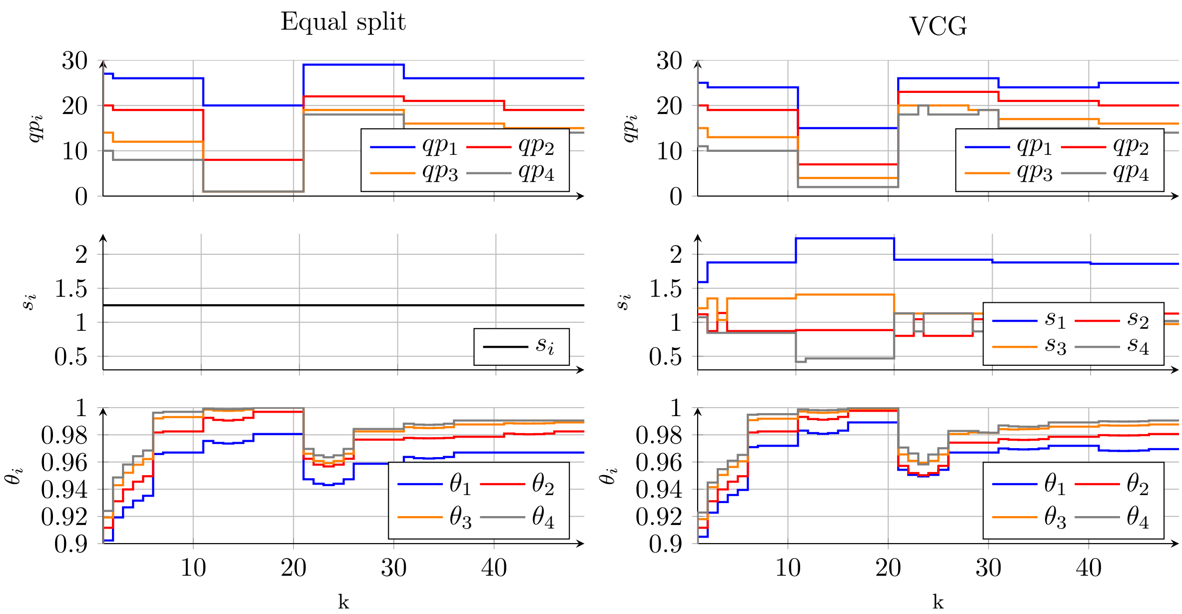

Fig. 6. Simulations with 4 buyers and the same amount of money, and low fixed video parameter changes.

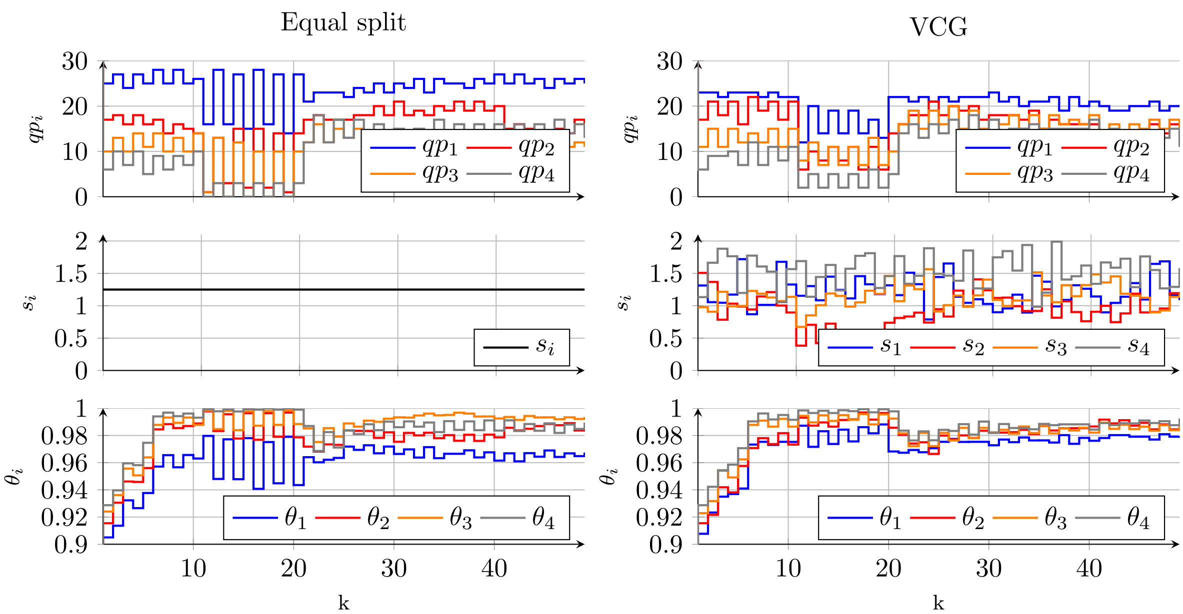

Fig. 7. Simulations with 4 buyers and same amount of money, big random video parameter changes.

ment. In *AIAA guidance, navigation and control conference and exhibit, Honolulu, USA*.

Cermak, G., Pinson, M., and Wolf, S. (2011). The relationship among video quality, screen resolution, and bit rate. *IEEE Transactions on Broadcasting*, 57(2).

Choi, H.L., Brunet, L., and How, J.P. (2009). Consensus-based decentralized auctions for robust task allocation. *IEEE Transactions on Robotics*, 25(4), 912–926.

Cramton, P. et al. (2002). Spectrum auctions. volume 1, 605–639. Elsevier, Amsterdam.

Dieber, B., Micheloni, C., and Rinner, B. (2011). Resource-aware coverage and task assignment in visual sensor networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(10), 1424–1437.

Ding, C., Song, B., Morye, A., Farrell, J.A., and Roy-Chowdhury, A.K. (2012). Collaborative sensing in a distributed PTZ camera network. *IEEE Transactions on Image Processing*, 21(7).

Edpalm, V., Martins, A., Årzén, K.E., and Maggio, M. (2018a). Camera networks dimensioning and scheduling with quasi worst-case transmission time. In *30th Euromicro Conference on Real-Time Systems, ECRTS 2018, Barcelona, Spain*, volume 106.

Edpalm, V., Martins, A., Årzén, K.E., and Maggio, M. (2018b). *H.264 Video Frame Size Estimation*. Technical Report TFRT-7654. Department of Automatic Control, Lund Institute of Technology, Lund University.

Elhamifar, E. and Vidal, R. (2009). Distributed calibration of camera sensor networks. In *2009 Third ACM/IEEE*

International Conference on Distributed Smart Cameras (ICDSC), Como, Italy, 1–7.

Ermis, E.B., Clarot, P., Jodoin, P.M.M., and Saligrama, V. (2010). Activity based matching in distributed camera networks. *IEEE Transactions on Image Processing*, 19, 2595–2613.

Ghosh, P., Roy, N., Das, S., and Basu, K. (2004). A game theory based pricing strategy for job allocation in mobile grids. In *18th International Parallel and Distributed Processing Symposium, 2004. Proceedings, Santa Fe, USA*, 82–.

Google (2021a). OR-Tools Google Developers Github. `https://google.github.io/or-tools/java/classcom_1_1google_1_1ortools_1_1algorithms_1_1KnapsackSolver.html` [Accessed: 2021-08-20].

Google (2021b). OR-Tools Google Developers website. `https://developers.google.com/optimization` [Accessed: 2021-03-25].

Hung, Y.H., Wang, C.Y., and Hwang, R.H. (2018). Combinatorial clock auction for live video streaming in mobile edge computing. In *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications Workshops*, 196–201.

Hunt, S., Meng, Q., Hinde, C., and Huang, T. (2014). A consensus-based grouping algorithm for multi-agent cooperative task allocation with complex requirements. *Cognitive computation*, 6(3).

IPVM (2021). Top 5 problems in video surveillance storage. `https://ipvm.com/reports/problems-video-surveillance-storage` [Accessed: 2021-08-18].

ITU-T (2010). H.264 standard documentation. `https://www.itu.int/rec/T-REC-H.264` [Accessed: 2019-06-13].

Konda, K.R., Conci, N., and De Natale, F. (2016). Global coverage maximization in PTZ-camera networks based on visual quality assessment. *IEEE Sensors Journal*, 16(16), 6317–6332.

Krishna, V. and Perry, M. (1998). Efficient mechanism design. Technical report, University Library of Munich, Germany. `https://EconPapers.repec.org/RePEc:wpa:wuwpga:9703010`.

Li, J., Sun, J., Qian, Y., Shu, F., Xiao, M., and Xiang, W. (2016). A commercial video-caching system for small-cell cellular networks using game theory. *IEEE Access*, 4, 7519–7531.

Li, J., Chen, H., Chen, Y., Lin, Z., Vucetic, B., and Hanzo, L. (2016). Pricing and resource allocation via game theory for a small-cell video caching system. *IEEE Journal on Selected Areas in Communications*, 34(8).

Maillé, P. and Tuffin, B. (2007). Why VCG auctions can hardly be applied to the pricing of inter-domain and ad hoc networks. *NGI 2007: 3rd EuroNGI Conference on Next Generation Internet Networks*, 36–39.

Martins, A., Lindberg, M., Maggio, M., and Årzén, K.E. (2020). Control-based resource management for storage of video streams. In *IFAC World Congress 2020, Berlin*.

Nanjanath, M. and Gini, M.L. (2006). Auctions for task allocation to robots. In *IAS*.

Nemethova, O., Ries, M., Siffel, E., and Rupp, M. (2004). Quality assessment for H.264 coded low-rate and low-resolution video sequences. In *Communications, Internet, and Information Technology (CIIT), Saint Thomas, U.S. Virgin Islands*, 508–512. Citeseer.

Nisan, N. and Ronen, A. (2007). Computationally feasible VCG mechanisms. *Journal of Artificial Intelligence Research*, 29, 19–47.

Pápai, S. (2003). Groves sealed bid auctions of heterogeneous objects with fair prices. *Social choice and Welfare*, 20(3), 371–385.

Pillai, P.S. and Rao, S. (2016). Resource allocation in cloud computing using the uncertainty principle of game theory. *IEEE Systems Journal*, 10(2), 637–648.

Qureshi, F.Z. and Terzopoulos, D. (2009). Planning ahead for PTZ camera assignment and handoff. In *2009 Third ACM/IEEE Int. Conference on Distributed Smart Cameras (ICDSC), Como, Italy*, 1–8. IEEE.

Sankaranarayanan, A., Veeraraghavan, A., and Chellappa, R. (2008). Object detection, tracking and recognition for multiple smart cameras. *Proceedings of the IEEE*, 96, 1606–1624.

Sawyer, N. and Smith, D.B. (2019). Flexible resource allocation in device-to-device communications using Stackelberg game theory. *IEEE Trans. on Communications*, 67(1).

SCI (2021). SCIP mixed integer programming solver. `https://www.scipopt.org/` [Accessed: 2021-08-20].

Seetanadi, G.N., Oliveira, L., Almeida, L., Årzén, K.E., and Maggio, M. (2018). Game-theoretic network bandwidth distribution for self-adaptive cameras. *SIGBED Rev.*, 15(3), 31–36.

Silvestre-Blanes, J., Almeida, L., Marau, R., and Pedreiras, P. (2011). Online QoS management for multimedia real-time transmission in industrial networks. *IEEE Transactions on Industrial Electronics*, 58(3).

Singh, K.D., Hadjadj-Aoul, Y., and Rubino, G. (2012). Quality of experience estimation for adaptive HTTP/TCP video streaming using H.264/AVC. In *2012 IEEE Consumer Communications and Networking Conference (CCNC), Las Vegas, USA*, 127–131.

Suris, J.E., Dasilva, L.A., Zhu, H., and Mackenzie, A.B. (2007). Cooperative game theory for distributed spectrum sharing. In *2007 IEEE International Conference on Communications, ICC'07, Glasgow, Scotland*, 5282–5287.

Vanderster, D.C., Dimopoulos, N.J., Parra-Hernandez, R., and Sobie, R.J. (2009). Resource allocation on computational grids using a utility model and the knapsack problem. *Future Generation computer systems*, 25(1).

Wang, X., Chen, X., and Wu, W. (2017). Towards truthful auction mechanisms for task assignment in mobile device clouds. In *IEEE INFOCOM 2017-IEEE Conference on Computer Communications, Atlanta, USA*.

Wikipedia (2021). Apple M1. `https://en.wikipedia.org/wiki/Apple_M1` [Accessed: 2021-08-20].

Xu, X. and Yu, H. (2014). A game theory approach to fair and efficient resource allocation in cloud computing. *Mathematical Problems in Engineering*.

Yang, K.C. (2007). *Perceptual quality assessment for compressed video*. Ph.D. thesis, UC San Diego.

Zlot, R.M. (2006). *An Auction-Based Approach to Complex Task Allocation for Multirobot Teams*. Ph.D. thesis, Carnegie Mellon University, Pittsburgh, PA.