Keio Business School

Excel for Multi-Unit Auction 2 : A Users' Manual

Contents

This material was written by Shuya Abe (Graduate School of Information Science, Osaka University) and Naoki Watanabe (Graduate School of Business Administration, Keio University) and published by Keio Business School. Inquiries about reproducing the case should be referred to Keio Business School (4-1-1 Hiyoshi Kohoku, Yokohama, Kanagawa 223-8526; Phone: +81-45-564-2444; E-Mail: case@kbs.keio.ac.jp) To order copies of the case, go to the website (http://www.kbs.keio.ac.jp).

1 Introduction

This material is a user's manual for a software *Excel for Multi-Unit Auction 2* which runs on a computer to implement the Vickrey-Clarke-Groves (VCG) mechanism and run the Greedy-Based Algorithm (GBA) for determining the allocation of some units of a commodity and the payments of bidders in multiple-unit auctions. The GBA is an approximation algorithm developed by Takahashi and Shigeno (2011) which computes the auction outcomes much faster than the VCG.[1] This section provides an overview of the operating environment for the software, practices in multiple-unit auctions, and some basic notes regarding its use.

1.1 Recommended Operating Environment

As of December 2019, the first version of *Excel for Multi-Unit Auction 2* is released; the filename is vcg-gba.xlsm and may be freely downloaded from the following URL.

http://labs.kbs.keio.ac.jp/naoki50lab/vcg-gba.xlsm

The download may be blocked in some corporate or educational environments, because this software uses the macro function of Excel. In such a case, users are asked to download the program via their personal internet environments.

The screen-shots in this manual were taken in the environment indicated in Table 1. *Excel for Multi-Unit Auction 2* runs on computers that can use Excel VBA (Visual Basic for Applications). The recommended environment is listed in Table 2.

Table 1: Operating environment in which screenshots were captured

OS	Windows 10 Enterprise
Excel	Microsoft Office 2016 Excel
CPU	Intel(R) Core(TM) i7-3770K CPU @ 3.50GHz 3.90GHz
Memory	8 GB
Disk	1 TB

[1] Takahashi, S., Shigeno, M., 2011. Approximation algorithms for a winner determination problem of single-item multi-unit auctions. *JSIAM Letters 3, 29-32*.

Table 2: Recommended environment

OS	Windows 7–, Mac OS X 10.8–
Excel	Microsoft Office 2013–, Microsoft Office for Mac 2016
CPU	no requirement if Excel works
Memory	at least 2GB
Disk	no in particular

1.2 Properties of the VCG mechanism

Consider a situation in which multiple units of a homogeneous commodity are auctioned off to bidders. Upon entering the bids for all units into *Excel for Multi-Unit Auction 2* for the settlement of that competitive bid, the program computes an allocation of those units and the payments of bidders. *Excel for Multi-Unit Auction 2* has two computational methods. One is known as the **VCG mechanism** and the other is its approximation algorithm called the Greedy-Based Algorithm (GBA). These methods are explained with a simple example in the Appendix to this manual. In this subsection, we focus on a brief look at some properties of the VCG mechanism, leaving to the end of this subsection a general setup of multiple-unit auctions.

In Autumn 2017, several media outlets reported that the Japan's Ministry of Internal Affairs and Communications (MIC) was considering the introduction of an auction to allocate usage rights (licenses) for frequency bands of electromagnetic waves. This auction is known as a *spectrum auction*. Organizations or agents active in electronic communications, digital broadcasting, and others require the allocation of the electromagnetic spectrum in order to provide their services. The auction is a mechanism for allocating the right to use given frequency bands, for given periods of time, to various auction participants on the basis of their competitive bids. As of Autumn 2019, the Japan's MIC continued to allocate spectrum on the basis of comparative hearings, although spectrum auctions have previously been introduced in the United States, England, France, and Germany. The aims of introducing spectrum auctions are to enhance the transparency of the allocation process, to improve the efficiency of spectrum allocations, and to increase the revenue derived from usage fees.[2]

In general, auctions have proved useful not only for allocating the electromagnetic spectrum but also for many other allocation problems, including government procurement, issuance of public bonds, awarding of contracts for construction of public facilities, airport takeoff and landing slots, search advertising, naming rights, component procurement, and so on. Thus, recently, national and local governments as well as corporations around the world are increasingly studying the properties of auctions and adopting various types of auction for these and other purposes.

[2] On October 1, 2017, the MIC announced revised fees for the use of electromagnetic spectrum. The details can be seen on the MIC website: http://www.tele.soumu.go.jp/j/sys/fees/sum/money.htm. The total revenue derived from electromagnetic spectrum usage fees in fiscal year 2015 was 74.7 billion JPY, based on the usage fees prior to this revision.

Allocative **efficiency**, in the sense that commodities or services are preferentially allocated to bidders who obtain higher benefits from them, is an important criterion for evaluating allocation systems. The benefit each bidder obtains from consuming a commodity or a service is, however, typically private information known only to the bidder. If we could suppose that a neutral intermediary agent is present, then the intermediary agent would require accurate information regarding the benefit each bidder obtains in order to allocate commodities and services efficiently. In this situation, however, bidders would not necessarily disclose such private information truthfully to the intermediary agent. An allocation system (or a mechanism) is said to be **strategy-proof**, when it is in the best interest of each bidder, irrespective of the other bidders' behavior, to disclose the information regarding his or her benefit truthfully to the intermediary agent under the system.

Designing strategy-proof mechanisms is never an easy task; rather, in practice, there are many situations in which such a mechanism does not exist. In auctions for multiple units of a homogeneous commodity, however, there does exist a mechanism that, given certain assumptions regarding the way in which the benefit is assessed, simultaneously ensures efficiency and strategy-proofness. The mechanism was named after the researchers who found it as the Vickrey-Clarke-Groves (VCG) mechanism.

In the general case of multiple-unit auctions, we seek to allocate k units of a homogeneous commodity among n bidders. Bidders may purchase up to k units of the commodity. Each bidder estimates, for each available unit of the commodity, the monetary benefit he or she would obtain from consuming that unit; this is called as the bidder's *valuation*. A bidder's valuation for each unit is private information known only to that bidder. According to this information, each bidder communicates with a neutral intermediary (an auctioneer) and bid for each unit of the commodity. Strategy-proofness corresponds to the bidding behavior such that every bidder simply states his or her true valuation as the bid for each unit, which is the optimal strategy for him or her.

1.3 Notes

(1) Users need to specify initial conditions for the algorithm, as well as valuations and bids for all bidders for all units, by inputting numerical values in **Arabic Numerals (not double byte characters) without multiple numbers and without missing values**.

(2) *Excel for Multi-Unit Auction 2* uses a program written in Visual Basic. Thus, unlike editing a regular Excel sheet, users cannot restore states by pressing the “backspace” key or by pressing the “Z” key while holding down the “Ctrl” key (Ctrl−Z). The Excel file is restored to its state prior to editing by closing the Excel session without saving the file.

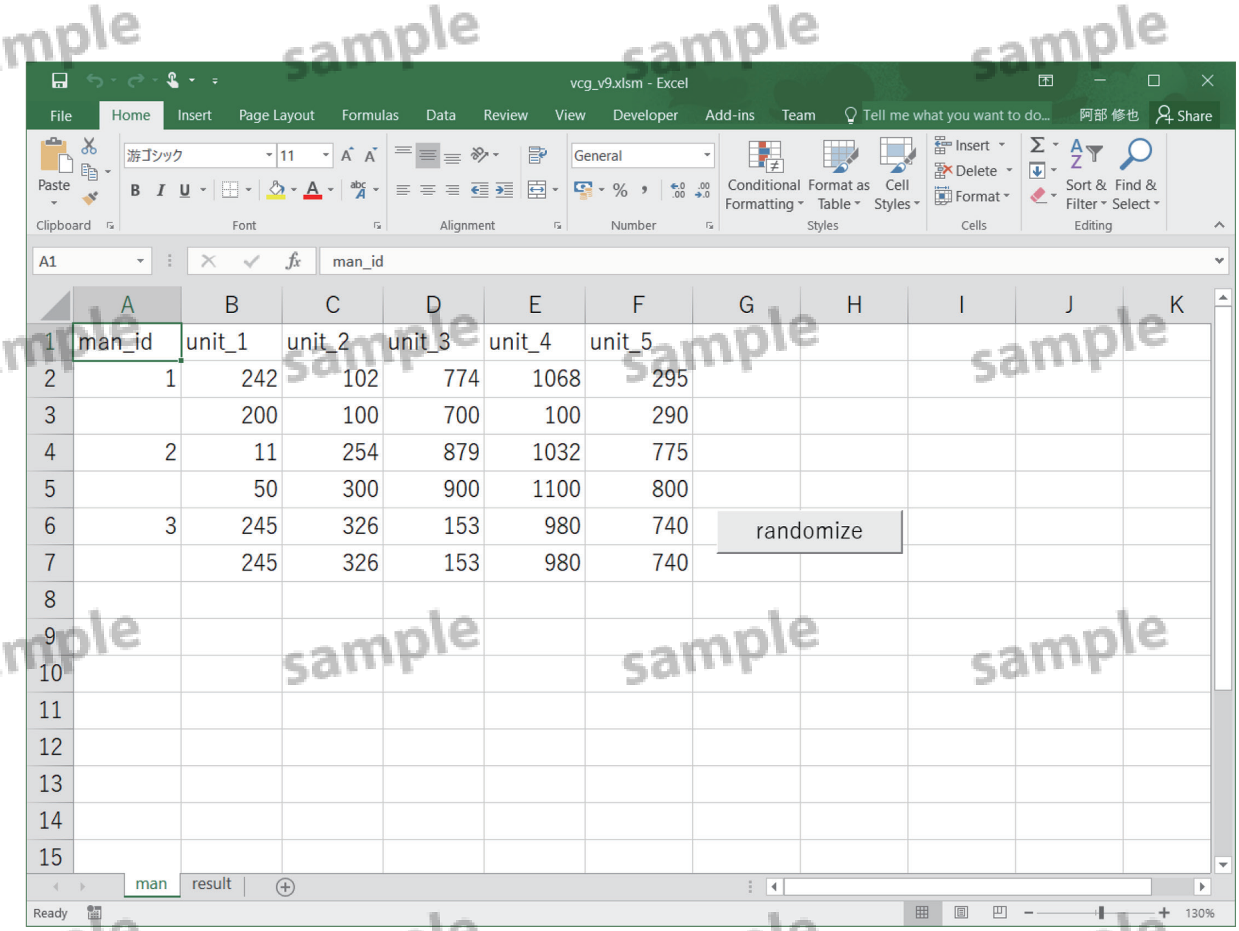
(3) The computation results are overwritten each time the program is exe-cuted. If you wish to save the results of a computation, then copy them to a different file or make separate copies for each Excel file.

2 Structure of the software package

Excel for Multi-Unit Auction 2 consists of two Excel spreadsheets (man and result) and one VBA program. Here we will describe the content of the two spreadsheets; the usage of the VBA program is briefly explained at the beginning of the next section, while for information on reading and writing VBA code, we ask users to read the standard commercially available reference materials. When running the software, to facilitate computer calculations, we assign each bidder a numerical ID to allow bidders to be easily identified in program input and output.

2.1 man sheet

Each bidder has a certain valuation (the amount that he or she is willing to pay) for each possible subset of the k units available, and each bidder enters bids based on these values. The man sheet is used to enter bids in numerical values (not double byte characters) from each bidder for each possible quantity of the commodity. See Figure 1. The entries in this spreadsheet are interpreted as follows.



A1 | man_id

	A	B	C	D	E	F
1	man_id	unit_1	unit_2	unit_3	unit_4	unit_5
2	1	242	102	774	1068	295
3		200	100	700	100	290
4	2	11	254	879	1032	775
5		50	300	900	1100	800
6	3	245	326	153	980	740
7		245	326	153	980	740

Figure 1: man sheet

Column A, man_id: For the bidder with ID *i* (man_id=*i*), his or her ID is entered in row 2*i* of column A in man sheet. The IDs are integers that start with 1 and increase consecutively. No bidders have the same ID, and no ID is omitted. The numbers of bidders and units are set by the user when configuring initial conditions (Section 3.1), and the bidder IDs are automatically entered into man_ sheet accordingly.

Columns B and beyond, unit_j: For each bidder in the man sheet, enter in each cell the valuation or the bid for each possible unit of the commodity. Based on the settings of the initial conditions (Section 3.1), the available units will be automatically entered in man sheet as unit_j (j=1, . . . , *k*). For each bidder and each unit, there is one cell for the valuation and one cell for the bid. For the bidder with ID *i* (man_id=*i*), his or her ID appears in row 2*i* in column A, while valuations are entered in other columns of row 2*i*. The value in row 2*i* in the column corresponding to unit_j is bidder *i*'s valuation for *j* units of the commodity. Similarly, the value in row 2*i* + 1 of the same column is bidder *i*'s bid for *j* units of the commodity. In Figure 1, e.g., for 2 units of the homogeneous commodity (unit_2), the bidder with ID 1 (man_id=1) has his or her valuation of 120 and enters his or her bid of 100. The program requires that values for both the valuation and the bid be entered for all bidders for all units.

randomize **button** Used to generate random numbers for valuations for the purposes of conducting simulations.

See, in Figure 1, bids for all 3 bidders for all units (from 1 to 3) have been entered into the man sheet.

2.2 result sheet

The result sheet displays the units allocated to each bidder, the bidder's valuation and payment for units allocated to him or her, and the points he or she earned (valuation minus payment) under the VCG and the GBA, respectively. The various buttons in the spreadsheet are used to execute the calculations in question (Figure 2).

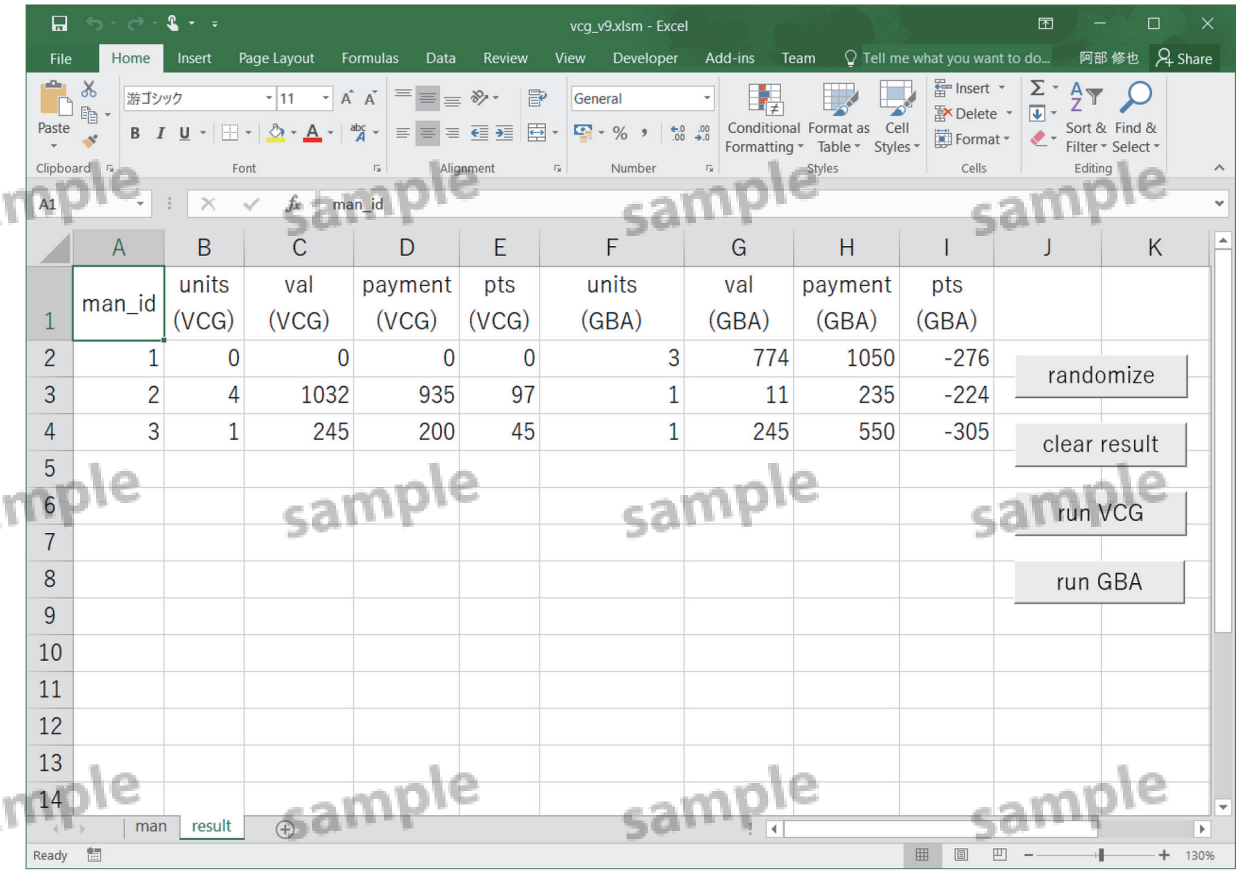
Column A: (man_id): Bidder IDs (in ascending order).

Column B (units (VCG)), **Column F** (units (GBA)): Units allocated to the bidder under the VCG and the GBA, respectively.

Column C (val (VCG)), **Column G** (val (GBA)): Bidder's valuation for the units allocated to him or her under the VCG and the GBA, respectively.

Column D (payment (VCG)), **Column H** (payment (GBA)): Payments of the bidder for the units allocated to him or her under the VCG and the GBA, respectively.

Column E (pts (VCG)), **Column I** (pts (GBA)): Points (the bidder's valuation minus his or her payment) the bidder earned under the VCG and GBA, respectively.



man_id	units (VCG)	val (VCG)	payment (VCG)	pts (VCG)	units (GBA)	val (GBA)	payment (GBA)	pts (GBA)
1	0	0	0	0	3	774	1050	-276
2	4	1032	935	97	1	11	235	-224
3	1	245	200	45	1	245	550	-305

Figure 2: result sheet.

Figure 2. According to the valuations and bids shown in Figure 1, the bidder with ID 2 (man_id=2) was allocated 4 units under the VCG, for which his or her valuation was 1032 and the payment was 935, and thus he or she earned 97 points, whereas under the GBA the bidder was allocated 1 unit, for which his or her valuation was 11 and the payment was 235, and thus he or she earned -224 points. As is shown in the gure, the GBA generates the results which are far from those the VCG generates.

Note that the GBA approximates the results computed by the VCG when all bidders bid their approximately true valuations. The computational procedures of the VCG and GBA are described in the Appendix.

run VCG **button**: Click on this button to run the calculation of commodity allocation and payments of bidders according to the VCG mechanism.

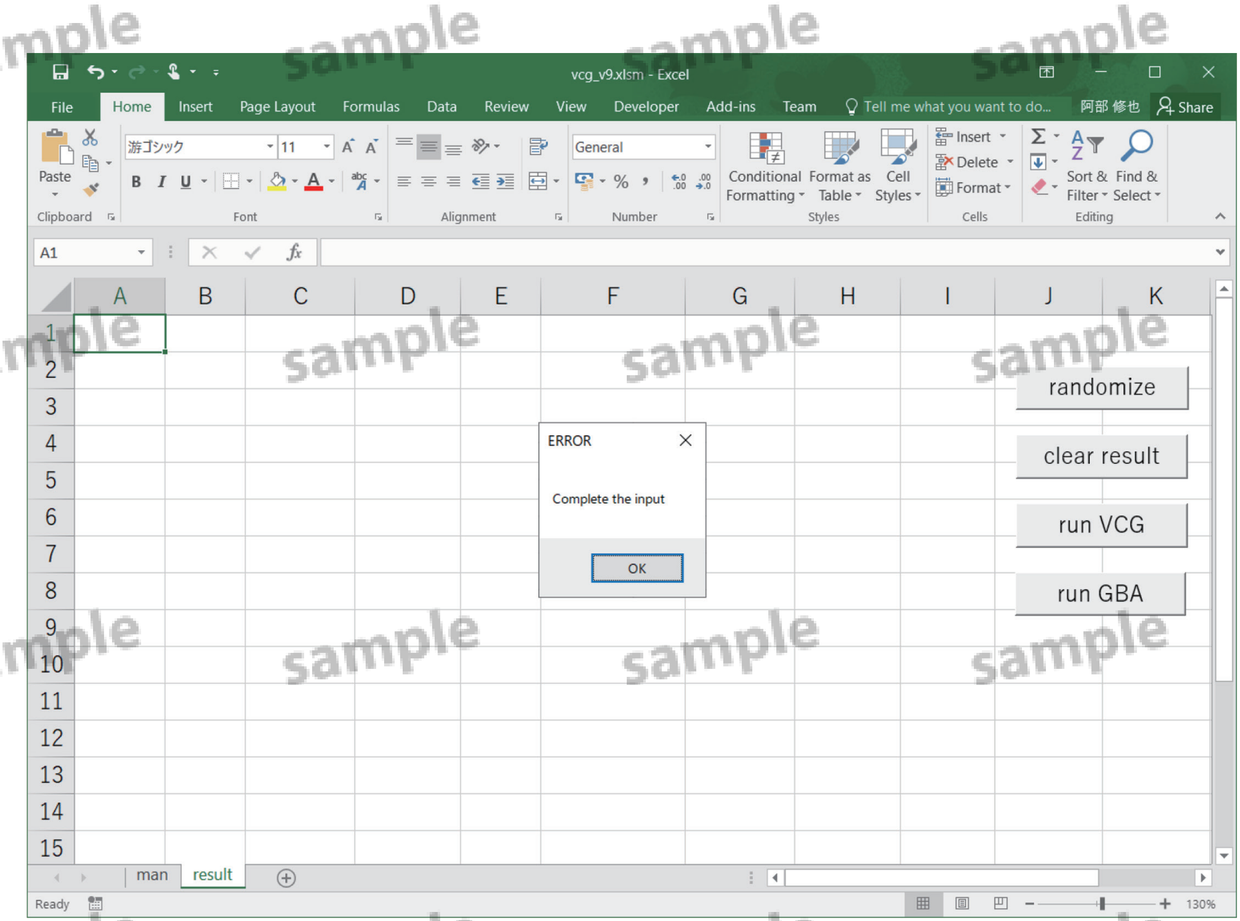
run GBA **button**: Click on this button to run the calculation of commodity allocation and payments of bidders according to the GBA.

When the program is invoked to perform calculations, if some valuations are left unspecified in the man sheet, then the calculation may proceed with those valuations set to 0 in some cases. If some bids are left unspecified, then the calculation will terminate prematurely and an error message (Figure 3) will appear. In such a case, click on the **OK** buttonin the error message and fill in values for all missing valuations and bids.

As demonstrated in the Appendix, under the VCG mechanism, the sum of all bids for all bidders and for all units allocated to bidders is maximized; however, there may be more than one such optimal allocation. In such cases, the program selects an allocation scheme that allocates fewer units to bidders with lower values of IDs. For example, for a case with 3 bidders and 3 units, if there are multiple possible allocation schemes that yield equal values of the total bid, units will be allocated in the order indicated in Table 3.

randomize **button**: Used to generate random numbers for the valuation entries in the man sheet (Section 2.1). This button performs the same operation as the randomize button in the man sheet.

clear result **button**: Used to clear the results shown on the result sheet.



ERROR

Complete the input

Figure 3: Error message presented in the event of one or more bid values being left unspecied

Table 3: Order of precedence for cases with multiple equivalent allocations

priority	assigned to bidder 1	assigned to bidder 2	assigned to bidder 3
1	0	0	3
2	0	1	2
3	0	2	1
4	0	3	0
5	1	0	2
6	1	1	1
7	1	2	0
8	2	0	1
9	2	1	0
10	3	0	0

3 Using *Excel for Multi-Unit Auction 2*

3.1 Getting Started

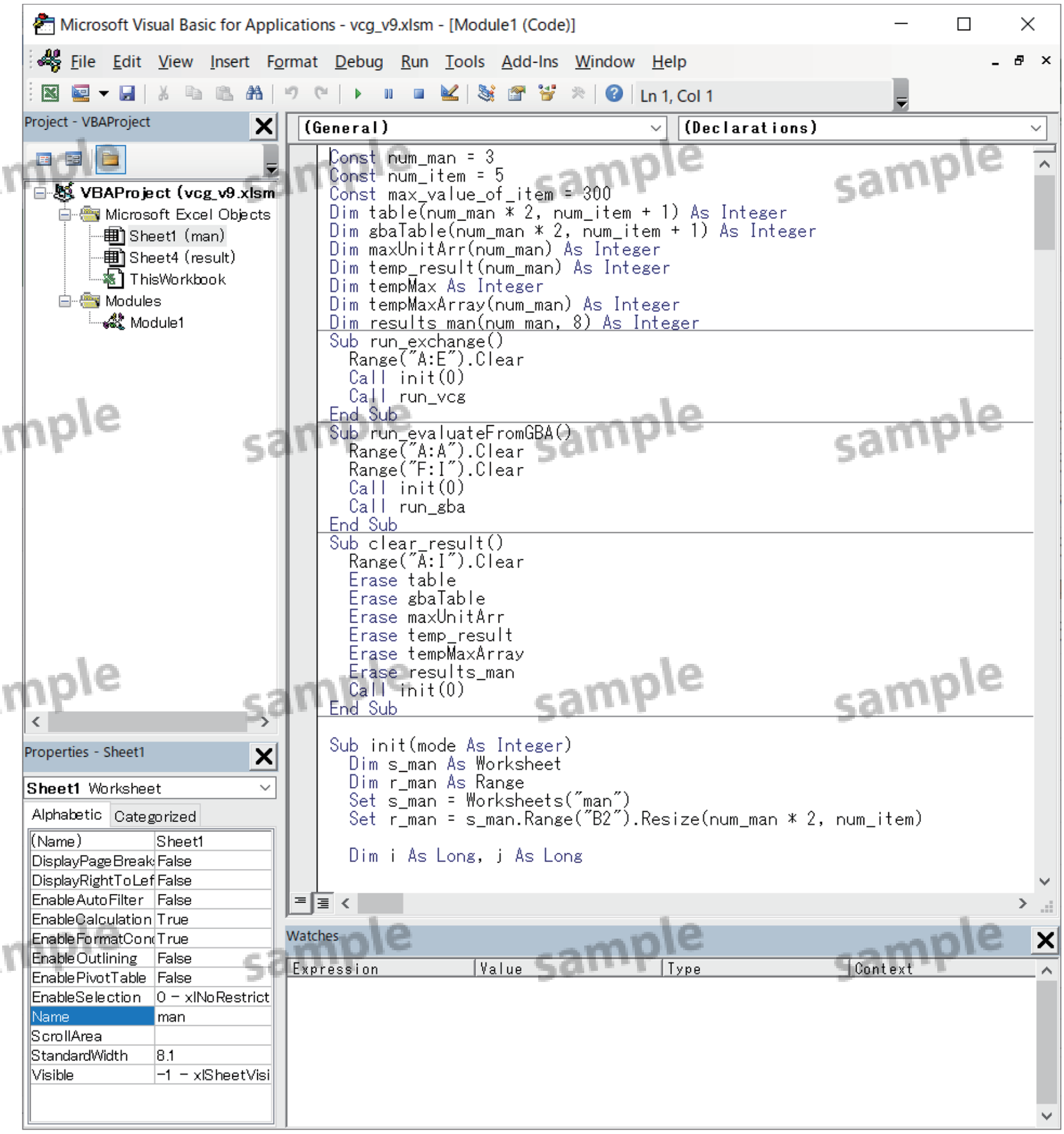
Initially, set the number of bidders n and the number of units k of the homogeneous commodity with the maximum valuation x of the commodity. These values are not set in the spreadsheets but rather within the Visual Basic for Applications (VBA) window. In Figure 4, the default values are 3 bidders (num_man=3), 5 units (num_item=5), and the maximum valuation of the commodity is 300 (max_value_of_item=300).

See Figure 5.

1. From the **Development** ribbon, click on the Visual Basic button.
2. From **Project Explorer**, open **Module 1** within **Standard Modules** inside **VBAProject** (vcg-gba.xlsm).
3. Enter values for Const num_man (number of bidders) and Const num_item (number of units) on the first and second lines, respectively, within **Module 1**. Further, enter the value for Const max_value_of_item (max-imum valuation of the commodity) there.
4. Save the file and close the Visual Basic window.

3.2 Running the program

1. Set the initial conditions for the number of bidders and the number of units within the VBA window (Section 3.1).
2. In the man sheet (Section 2.1), enter valuations and bids for all bidders for all units. In particular, be careful not to leave any bid unspecified.
3. Open the result sheet (Section 2.2) and click on the run VCG or run GBA button.



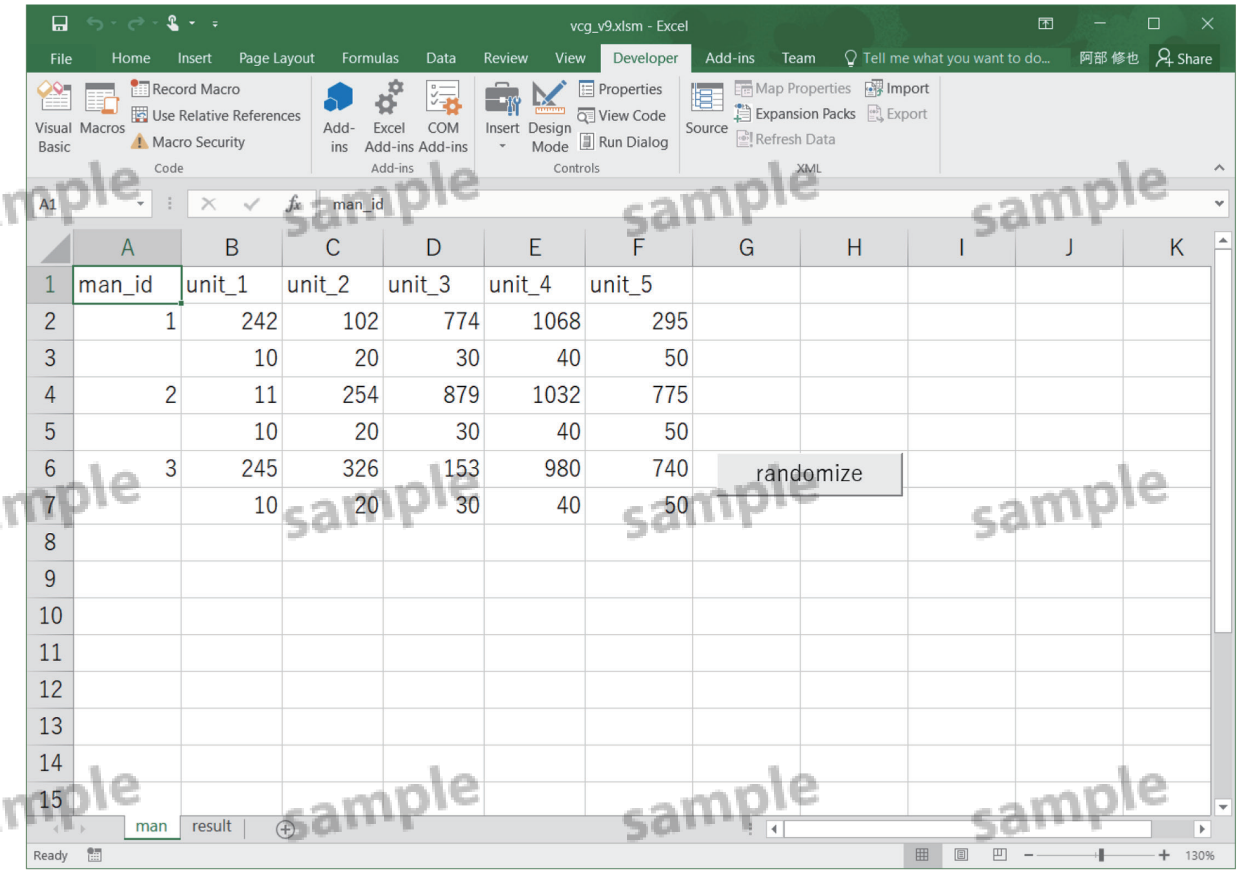
```
Const num_man = 3
Const num_item = 5
Const max_value_of_item = 300
Dim table(num_man * 2, num_item + 1) As Integer
Dim gbaTable(num_man * 2, num_item + 1) As Integer
Dim maxUnitArr(num_man) As Integer
Dim temp_result(num_man) As Integer
Dim tempMax As Integer
Dim tempMaxArray(num_man) As Integer
Dim results_man(num_man, 8) As Integer
Sub run_exchange()
  Range("A:E").Clear
  Call init(0)
  Call run_vcg
End Sub
Sub run_evaluateFromGBA()
  Range("A:A").Clear
  Range("F:I").Clear
  Call init(0)
  Call run_gba
End Sub
Sub clear_result()
  Range("A:I").Clear
  Erase table
  Erase gbaTable
  Erase maxUnitArr
  Erase temp_result
  Erase tempMaxArray
  Erase results_man
  Call init(0)
End Sub

Sub init(mode As Integer)
  Dim s_man As Worksheet
  Dim r_man As Range
  Set s_man = Worksheets("man")
  Set r_man = s_man.Range("B2").Resize(num_man * 2, num_item)

  Dim i As Long, j As Long
```

(Name)	Sheet1
DisplayPageBreak	False
DisplayRightToLef	False
EnableAutoFilter	False
EnableCalculation	True
EnableFormatCon	True
EnableOutlining	False
EnablePivotTable	False
EnableSelection	0 - xlNoRestrict
Name	man
ScrollArea	
StandardWidth	8.1
Visible	-1 - xlSheetVisi

Figure 4: Visual Basic window

3.3 Results

The results computed by the VCG mechanism are shown in Columns A-E within the result sheet. In the same sheet, the results computed by the GBA are shown in Columns F-I (Section 2.2). See Section 2.2 for an explanation of the signicance of each column.



	A	B	C	D	E	F
1	man_id	unit_1	unit_2	unit_3	unit_4	unit_5
2	1	242	102	774	1068	295
3		10	20	30	40	50
4	2	11	254	879	1032	775
5		10	20	30	40	50
6	3	245	326	153	980	740
7		10	20	30	40	50
8						
9						
10						
11						
12						
13						
14						
15						

Figure 5: **Visual Basic** button

3.4 Optional: Generating random input data

This software is equipped with the optional capability to use random numbers to conduct simulations. Click on the randomize button in either the man sheet (Section 2.1) or the result sheet (Section 2.2) to generate random values for the valuations of all bidders for all numbers of units. The numbers of bidders and of units will remain equal to the values specied as initial conditions (Section 3.1).

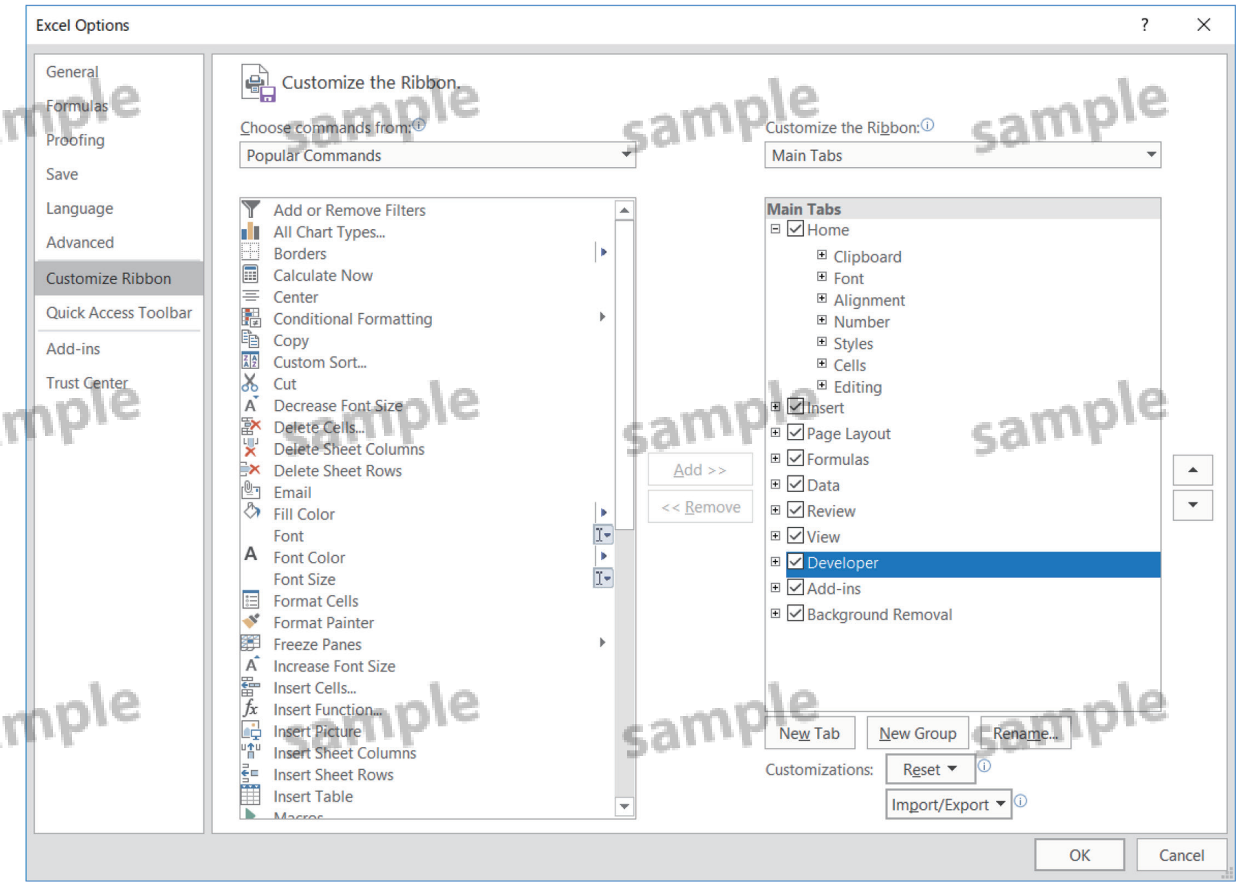
4 Troubleshooting

In what follows, we discuss remedies for some common problems that may prevent proper execution of the program.

4.1 The Development ribbon is not displayed

The initial settings of Microsoft Office Excel are congured to omit display of the **Development** ribbon. To display this ribbon, proceed as follows (for Microsoft Office Excel 2016).

1. Click on the **File** ribbon.
2. Click on the **Options** button.
3. Within **Excel Options**, click on **Ribbon User Settings**.
4. Confirm that the **Ribbon User Settings** field is set to **Main Tab**, and then check the box for **Development** in the list of ribbons for the **Main Tab** (Figure 6).
5. Click on **OK** and close the **Excel Options window**. The **Development** button should be visible. (If not, close and re-open the Excelle.)



Excel Options

Customize the Ribbon.

Choose commands from: Popular Commands

Customize the Ribbon: Main Tabs

Main Tabs

- ⊟ ☑ Home
  - ⊞ Clipboard
  - ⊞ Font
  - ⊞ Alignment
  - ⊞ Number
  - ⊞ Styles
  - ⊞ Cells
  - ⊞ Editing
- ⊞ ☑ Insert
- ⊞ ☑ Page Layout
- ⊞ ☑ Formulas
- ⊞ ☑ Data
- ⊞ ☑ Review
- ⊞ ☑ View
- ⊞ ☑ Developer
- ⊞ ☑ Add-ins
- ⊞ ☑ Background Removal

Figure 6: Ribbon user settings

4.2 An error message appears when the program is run

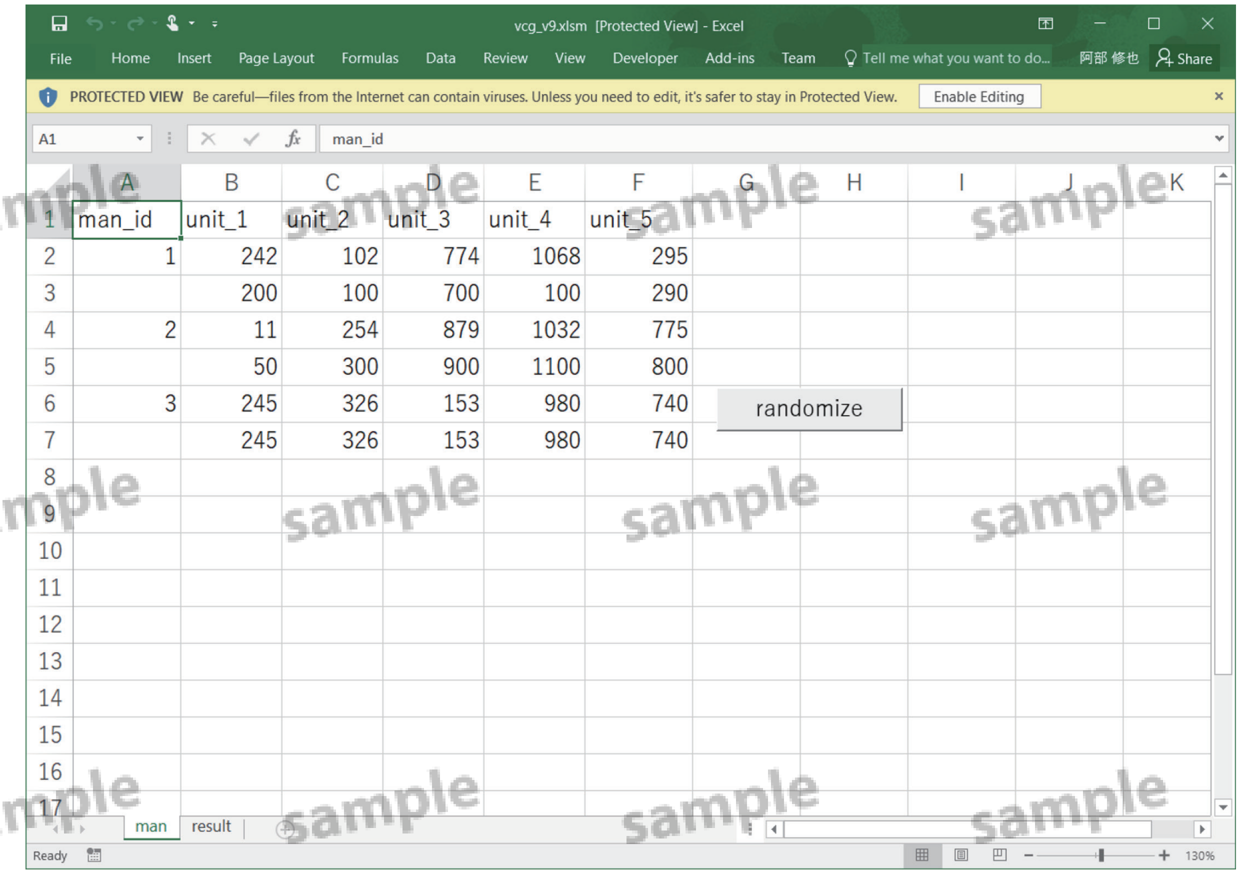
If error messages appear when attempting to run the program, check the following points.

- All numerical input values (numbers of bidders and units in initial conditions (Section 3.1) and valuations and bids in the man sheet (Section 2.1) are specified using double byte characters.
- The numbers of bidders and units specified in the initial conditions (Section 3.1) agree with the numbers of bidders and units listed in the man sheet (Section 2.1). If the numbers do not agree, clicking on the randomize button will automatically set the numbers to the values specified in the initial conditions.
- There is no missing bid entry.

4.3 No error message appears, but the program cannot compute a correct allocation

Check the following points, if no error message appears but the program does not compute correct outcomes.

- Check whether you opened the program in Protected View (Figure 7). If so, click on **Enable editing (E)** to terminate Protected View.
- Check that the numbers of bidders and units specified in the VBA window (Section 3.1) agree with the numbers of bidders and units listed in the man sheet (Section 2.1). If the numbers do not agree, clicking on the randomize button will automatically set the numbers to the values specified in the initial conditions.
- Check that there are no missing valuation entries.



PROTECTED VIEW Be careful—files from the Internet can contain viruses. Unless you need to edit, it's safer to stay in Protected View.

man_id	unit_1	unit_2	unit_3	unit_4	unit_5
1	242	102	774	1068	295
	200	100	700	100	290
2	11	254	879	1032	775
	50	300	900	1100	800
3	245	326	153	980	740
	245	326	153	980	740

Figure 7: Window as displayed when opened in Protected View

Appendix: Examples of multiple-unit auctions

Here we will use an example to explain the underlying principles. Consider a case in which 2 bidders are competing for 3 units of a homogeneous commodity. Bidder 1 estimates a valuation of 80 as the monetary value associated with consumption of 1 unit of this commodity, and valuations of 60 × 2 (= 120)

for the consumption of 2 units and 55 × 3(= 165) for the consumption of 3 units. Meanwhile, bidder 2 assesses the value at 40 for consumption of 1 unit of the commodity, 70 × 2(= 140) for consumption of 2 units, and 65 × 3(= 195) for consumption of 3 units. If the commodity cannot be consumed, then its monetary value to all bidders is 0.

Table 4 lists the bids made by each bidder for each of the available quantities of the commodity. For example, bidder 1 bids 70 for 1 unit, 55 × 2(= 110) for 2 units, and 50 × 3(= 150) for 3 units. Note that in Table 4, in contrast to the notation used in the main text, we have stated valuations and bid values in terms of the cost per unit multiplied by the number of units.

Table 4: Valuations and bids for each available number of units

		1	2	3
Bidder 1	valuation	80 × 1	60 × 2	55 × 3
	bid	70 × 1	55 × 2	50 × 3
Bidder 2	valuation	40 × 1	70 × 2	65 × 3
	bid	40 × 1	60 × 2	65 × 3

VCG mechanism

Allocation

In allocating the commodity via the VCG mechanism, all possible combinations of allocations to bidders are enumerated, and the final allocation is chosen to maximize the total amount of bids. For example, for an allocation assigned 1 unit to bidder 1 and 2 units to bidder 2, the total amount of bids would be 70 × 1 + 60 × 2 = 190. Table 5 lists allocations and corresponding total amount of bids; we see that the allocation that maximizes the total amount of bids is to allocate all three units to bidder 2. In case of ties, one of those allocations is chosen at random.

Table 5: Distributions and corresponding total amount of bids

# of assignments	assigned to bidder 1	assigned to bidder 2	total amount of bids
0	0	0	0
1	1	0	70 × 1 = 70
1	0	1	40 × 1 = 40
2	1	1	70 × 1 + 40 × 1 = 110
2	0	2	60 × 2 = 120
2	2	0	55 × 2 = 110
3	1	2	70 × 1 + 60 × 2 = 190
3	2	1	55 × 2 + 40 × 1 = 150
3	0	3	65 × 3 = 195
3	3	0	50 × 3 = 150

Payments

The payment by each bidder in the VCG mechanism is determined as follows based on the number of units allocated to the bidder.

payment of bidder *i* =
(total amount of bids in the auction that excludes bidder *i*
− (total amount of bids in the original auction)
+ (bidder *i*'s bid for the unit assigned to *i*)

Thus,

payment of bidder 1 = (65 × 3) −(195) + 0 = 0
payment of bidder 2 = (50 × 3) − (195) + (65 × 3) = 150

Note that bidder 2's payment is lower than the actual bid, 195. We have set the bids for bidder 1 corresponding to the number of units allocated to bidder 1 to 0. Then, the benefit for bidder 1 is 0, while the benefit for bidder 2 is the difference between the monetary value realized by consuming 3 units of the commodity and the payment of bidder 2, i.e., 65× 3 − 150 = 45.

GBA: An approximation algorithm for the VCG mechanism

Allocation

The allocation problem of the GBA has 5 steps in total. Unit valuations are given as below. Bidders are asked to bid per unit for each unit.

Table 6: Valuations and bids for each possible number of units

		1	2	3
Bidder 1	valuation	80 × 1	60 × 2	55 × 3
	bid	75 × 1	55 × 2	40 × 3
Bidder 2	valuation	40 × 1	70 × 2	65 × 3
	bid	40 × 1	63 × 2	65 × 3

1. Find the highest unit bid. Give “tentatively” the unit to the highest unit bidder.
2. Update the other unit bids of the highest unit bidder in the following way; The highest unit bid is 75 cast by bidder 1 for 1 unit.

- updated unit bid for 2 units $= \dfrac{55 \times 2 - 75 \times 1}{(2 - 1)} = 35$
- updated unit bid for 3 units $= \dfrac{40 \times 3 - 75 \times 1}{3 - 1} = 22{:}5.$

Table 7: Valuations and bids for each possible number of units

		1	2	3
Bidder 1	valuation	80 × 1	60 × 2	55 × 3
	bid		35 × 2	22.5 × 3
Bidder 2	valuation	40 × 1	70 × 2	65 × 3
	bid	40 × 1	63 × 2	65 × 3

3. Find the highest (updated) unit bid. Give tentatively the corresponding unit to the highest (updated) unit bidder.

   This bidder is also called a “tentative winner”.
4.
   - If all units are just assigned, the assignment is then implemented.
   - If some units are not assigned, go to step 2.
   - If the number of units is less than the sum of assigned units (there is the “excess demand”), then go to step 5.

The highest (updated) unit bid is 65 cast by bidder 2 for 3 units. In the first round, bidder 1 was assigned 1 unit as a tentative winner, and thus there is the excess demand. Thus, go to step 5.

5. Choose such an allocation that maximizes the total amount of bids among the allocations of tentative winners.
   - 1 unit to bidder 1 and 2 units to bidder 2.
     Total amount of bids = 75 1 + 63 2 = 201
   - 0 unit to bidder 1 and 3 units to bidder 3.
     Total amount of bids = 65 3 = 195

Choose Allocation 1.

Once the payment of some bidder exceeds his or her valuation, *Excel for Multi-Unit Auction 2* stops allocating the commodity and assigns some units to bidders which are determined by the stop. This event does not occur, when the bids are not so far from the valuations as the theory expects.

Payments

The payments of bidders are determined in the same way as in Exact VCG.

payment of bidder i (winner) =
(total amount of bids in the auction that excludes bidder i:
− (total amount of bids in the original auction)
+ (bidder i’s bid for the unit assigned to i)

(Allocation 1) … 1 unit to bidder 1 unit and 2 units to bidder 2.
- payment of bidder 1 = $(65\times3) - 201 + (75\times1) = 69$
- payment of bidder 2 = $(75\times1 + 35\times2) - 201 + (63\times2) = 70$

Keio Business School

KCC 2020.1 PDF