Manual for `sgame.sty`

by

Martin J. Osborne
Martin.Osborne@utoronto.ca

For Version 2.11, 2004/5/13

1. Introduction

`sgame.sty` is a LaTeX 2_ε style file for drawing strategic games. The latest version is available at `http://www.economics.utoronto.ca/osborne/latex`. Please notify me of bugs.

2. Installation

- Put `sgame.sty` in a directory from which TeX reads input files. (In MiKTeX the directory might be something like `\miktex\localtexmf\tex\latex\` or a subdirectory thereof.)

- Let TeX know that `sgame` has arrived. (In MiKTeX, "refresh the filename database".)

- Check that you have the `color` package. One way to check is to put `\usepackage{color}` in the preamble of a document and see if you get an error. If you don't have the package, get it from CTAN.

- To use the package in a document, put the lines `\usepackage{sgame}` and `\usepackage{color}` (or `\usepackage{pstcol}`, part of PSTricks, which loads `color`) in the preamble.

- The style is incompatible with `colortbl.sty`.

3. Description

The style file defines two environments, `game` and `game*`. Each environment can be used in four ways, depending on the values of its optional arguments. In each case, the starred version differs from the unstarred versionin that action labels are not accommodated.

3.1 No optional argument

Without any optional arguments, its syntax is

`\begin{game}{`*number-of-rows*`}{`*number-of-columns*`}`*game-body*`\end{game}`

where *number-of-rows* is the number of rows in the game and *number-of-columns* is the number of columns. *game-body* consists of rows in the style of the rows in a `tabular` environment, `&` "tabbing" to a new cell, and `\\` ending a row.

The following example is illustrated in Figure 1.

```
\def\sgtextcolor{blue}%
\def\sglinecolor{red}%
\begin{figure}[htb]\hspace*{\fill}%
\begin{game}{2}{2}
      & $L$    & $R$\\
$T$   &$2,2$   &$100,35$\\
$B$   &$3,0$   &$350,16$
\end{game}\hspace*{\fill}%
\begin{game*}{2}{2}
$2,2$   &$100,35$\\
$3,0$   &$350,16$
\end{game*}\hspace*{\fill}%
\caption[]{A two row, two column strategic game, with and without
action labels.}
\end{figure}
```

	L	*R*
T	2, 2	100, 35
B	3, 0	350, 16

2, 2	100, 35
3, 0	350, 16

Figure 1. A two row, two column strategic game, with and without action labels.

- Note that the width of every column is the same. Further, the content of each cell of the table is centered vertically.

- As in a LaTeX 2_ε `tabular` environment, spaces on either side of the text in each cell is ignored. (Thus in the `tex` file you can, if you wish, align the columns using spaces or tabs.)

- The first `&` **must** be preceded by a space token (space or carriage return). (It may not immediately follow the closing brace of the second argument.)

- If all the text in every cell of the payoff matrix is in math mode, you can avoid typing all the `$`'s by setting `\gamemathtrue`, either by putting this command in your file, or by changing `\gamemathfalse` to `\gamemathtrue` in the style file. Under this setting, all text in every cell of the payoff matrix is automatically in math mode. (Note that even if the text in the cells consists only of comma-separated pairs of numbers, it should still be in math mode, else the spacing is not correct.)

3.2 One optional argument: game label

With one optional argument, the syntax of the `game` environment is

`\begin{game}{`*number-of-rows*`}{`*number-of-columns*`}` `[`*game-label*`]`*game-body*
`\end{game}`

where *number-of-rows* is the number of rows in the game, *number-of-columns* is the number of columns, and *game-label* is a label for the game that is centered beneath the game. (The sense in which it is centered may be changed by setting `\irpawcgltrue` or `\irplwcgltrue`—see Section 6.) This case is useful if two or more games are positioned side-by-side, and each requires a label.

The following example is illustrative.

```
\begin{figure}[htb]\hspace*{\fill}%
\begin{game}{2}{2}[$A$]
      & $L$    & $R$\\
$a$   &$2,2$   &$0,3$\\
$b$   &$3,0$   &$1,1$
\end{game}\hspace*{20mm}%
\begin{game}{2}{2}[$B$]
      & $L$    & $R$\\
$a$   &$2,3$   &$0,4$\\
$b$   &$3,1$   &$1,2$
\end{game}\hspace*{\fill}%
\caption[]{A three player strategic game, in which
player~3 chooses $A$ or $B$.}
\end{figure}
```

	L	*R*
a	2, 2	0, 3
b	3, 0	1, 1

A

	L	*R*
a	2, 3	0, 4
b	3, 1	1, 2

B

Figure 2. A three player strategic game, in which player 3 chooses *A* or *B*.

3.3 Two optional arguments: player labels

In the case of two optional arguments, the syntax of the `game` environment is

`\begin{game}{`*number-of-rows*`}{`*number-of-columns*`}[`*row-player-label*`]`
`[`*column-player-label*`]`*game-body*`\end{game}`

where *number-of-rows* is the number of rows in the game, *number-of-columns* is the number of columns, and *row-player-label* and *column-player-label* are the names of the players.

The following example illustrates this case.

```
\begin{figure}[htb]\hspace*{\fill}%
\begin{game}{2}{3}[Player~1][Player~2]
      & $L$    & $M$    & $R$\\
$T$   &$2,2$   &$2,0$   &$0,3$\\
$B$   &$3,0$   &$0,9$   &$1,1$
\end{game}\hspace*{\fill}%
\caption[]{A two row, three column strategic game with
player labels.}
\end{figure}
```

		Player 2		
		L	*M*	*R*
Player 1	*T*	2, 2	2, 0	0, 3
	B	3, 0	0, 9	1, 1

Figure 3. A two row, three column strategic game with player labels.

3.4 *Three optional arguments: player labels and game label*

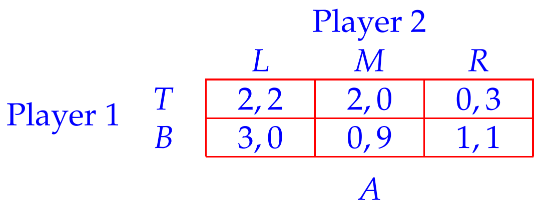
The case of three optional arguments combines the two previous cases: the syntax of the `game` environment is

`\begin{game}{`*number-of-rows*`}{`*number-of-columns*`}[`*row-player-label*`]`
`[`*column-player-label*`] [`*game-label*`]`*game-body*`\end{game}`

where *number-of-rows* is the number of rows in the game, *number-of-columns* is the number of columns, *row-player-label* and *column-player-label* are the names of the players, and *game-label* is a label for the game.

The following example illustrates this case.

```
\begin{figure}[htb]\hspace*{\fill}%
\begin{game}{2}{3}[Player~1][Player~2][$A$]
      & $L$    & $M$    & $R$\\
$T$   &$2,2$   &$2,0$   &$0,3$\\
$B$   &$3,0$   &$0,9$   &$1,1$
\end{game}\hspace*{\fill}%
\caption[]{A two row, three column strategic game with
player labels and a label.}
\end{figure}
```



Player 2

		L	M	R
Player 1	T	$2,2$	$2,0$	$0,3$
	B	$3,0$	$0,9$	$1,1$

A

Figure 4. A two row, three column strategic game with player labels and a label.

4. Cell formatting

You may regard the format of the payoffs in the first column of the game in Figure 5 as unsatisfactory, and the formatting in the left panel of Figure 6 as superior. This latter formatting is implemented by adding "phantom" minus signs before the "2" in the top left cell and before the "3" in the bottom left cell. The top left cell, for example, is entered as `$-2,\phantom{-}2$`.

	L	R
T	$-2, 2$	$0, 3$
B	$3, -4$	$1, 1$

Figure 5. A strategic game.

(`\phantom{<x>}` leaves a space whose size is equal to the size of the object `<x>`.)

You may want to consider other formats. The right panel of Figure 6 shows an alternative. This format is achieved be entering the element of each cell as

```
\begin{array}{c}#1\\[1.6mm]#2\end{array},
```

where `#1` is player 1's payoff and `#2` is player 2's payoff. (Actually, I defined the macro

```
\def\stackedpayoffs#1#2{%
  \begin{array}{c}#1\\[1.6mm]#2\end{array}
}
```

and entered each element as `\stackedpayoffs{#1}{#2}`. I set `\gamestretch` to be 2.1 to improve the appearance of the game.)

	L	R
T	$-2, 2$	$0, 3$
B	$3, -4$	$1, 1$

	L	R
T	-2 2	0 3
B	3 -4	1 1

Figure 6. Two alternative presentations of the game in Figure 5.

Figure 7 shows two more alternatives. The one on the left is achieved by entering the element of each cell as

```
\begin{array}{rl}#1&\\[1.6mm]&#2\end{array},
```

and the one on the left differs only in that there is a `\phantom{-}` in front of player 2's payoff in the top left cell, player 1's payoff in the bottom left cell, and player 2's payoffs in both of the right cells.

	L	R
T	−2 2	0 3
B	3 −4	1 1

	L	R
T	−2 2	0 3
B	3 −4	1 1

Figure 7. Two additional presentations of the game in Figure 5.

5. Vertical position of text within cells

The vertical distance between the lines at the top and bottom of a row in LaTeX's tabular environment is equal to \arraystretch times \baselineskip. The baseline of the text in each cell is placed 30% of the distance from the bottom line. Thus if \arraystretch is large relative to the height of the text, the text is not near the middle of the row. (See the left panel of Figure 8; or try setting \arraystretch equal to 5 and process a document with a tabular environment.) (See the definitions of \strutbox and of \@array in latex.ltx.)

sgame.sty makes the vertical distance between the lines at the top and bottom of a row of a game the same (\arraystretch times \baselineskip), but places the baseline of the text in each cell so that the distances above and below an upper case I placed on the baseline are equal. (See the right panel of Figure 8.) (If you'd like to use another character, define this character to be \sg@alignchar.)

If you want to use sgame's vertical positioning in your tabular environments, as well as your game environments, set \gamevaligntrue before the environment. (If you want to switch back at some point, you can do so by setting \gamevalignfalse.)

If the height of the text in a cell exceeds \arraystretch times \baselineskip, then the height of the row is increased to accommodate the text (just as the distance between lines of text is increased to accommodate large text). The result is that the lines at the top and bottom of the row touch

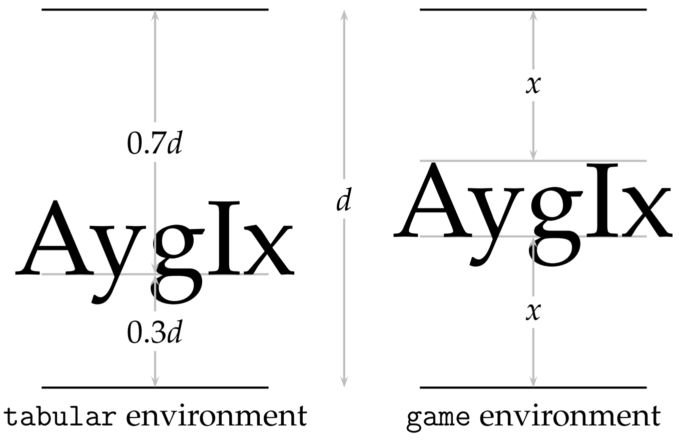
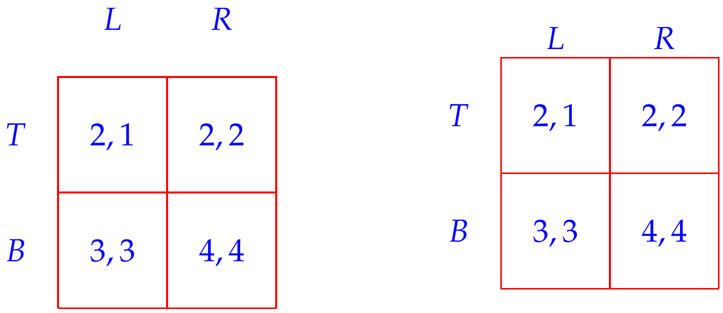


Figure 8. Vertical positioning of text within cells in the `tabular` and `game` environments.

the top and bottom of the text, and this row is taller than the others. To improve the appearance of the table you may want to increase `\gamestretch` (analogous to `\arraystretch`), to put some white space above and below your oversized text.

When `\gamestretch` is increased from 1, the height of *every* row, including the first row (the action labels), is increased. If you don't want to increase the space under the action labels, declare `\ssualfalse`. ("ssual" stands for "stretch space under action labels".) See Figures 10 and 9 for examples (for games with and without player labels).



	L	R
T	$2, 1$	$2, 2$
B	$3, 3$	$4, 4$

	L	R
T	$2, 1$	$2, 2$
B	$3, 3$	$4, 4$

Figure 9. Positioning the action labels of the column player when `gamestretch` is larger than 1, for a game without player labels. In the left game we have `ssualtrue` (the default); in the right game `ssualfalse` is declared.

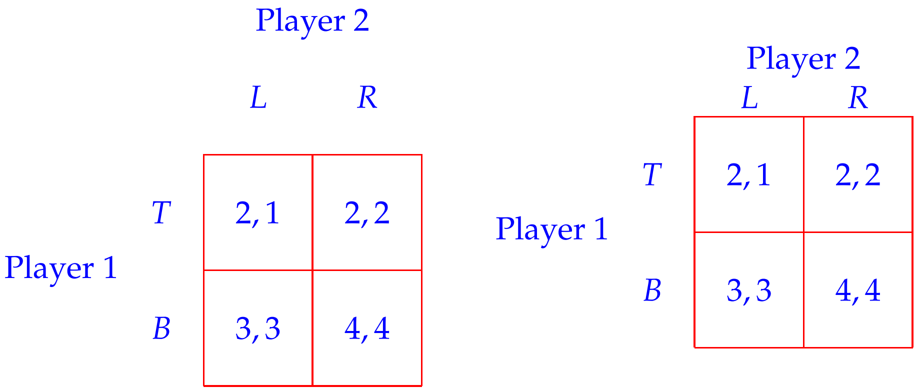


Figure 10. Positioning the action labels of the column player when `gamestretch` is larger than 1, for a game with player labels. In the left game we have `ssualtrue` (the default); in the right game `ssualfalse` is declared.

6. Parameters

`\gamestretch`: a number that controls the spacing between rows, analogous to `\arraystretch`. Possible values: positive real numbers. Default: value of `\arraystretch` when `sgame.sty` is loaded (1 in standard styles). Example: `\renewcommand{\gamestretch}{2}`.

`\sgcolsep`: horizontal padding within cells. Possible values: any dimension. Default: value of `\tabcolsep` when `sgame.sty` is loaded. Example: `\sgcolsep=10pt`.

`\sglabelsep`: vertical space between game and label. Possible values: any dimension. Default: 5pt. Example: `\sglabelsep=10pt`.

`\irpawcgltrue` ("include row player actions when centering game label"): causes game label to be centered under box consisting of row player actions and payoff matrix. Default: `\irpawcglfalse` (i.e. game label is centered under box consisting only of payoff matrix). Example: `\irpawcgltrue`.

`\irplwcgltrue` ("include row player label when centering game label"): causes game label to be centered under box consisting of row player label, row player actions, and payoff matrix. (Is overridden by `\irpawcgltrue`.) Default: `\irplwcglfalse` (i.e. game label is centered under payoff matrix). Example: `\irplwcgltrue`.

`\gamemathtrue`: put all text in cells in payoff matrix in math mode. Default:

\gamemathfalse. Example: \gamemathtrue.

\gamevaligntrue: vertically align the text within cells of a tabular environment as they are aligned in a game environment. Default: \gamevalignfalse. Example: \gamevaligntrue. (See Section 5.)

\ssualtrue: stretch space under action labels when \gamestretch is greater than 1. Default: \ssualtrue. Example: \ssualfalse. (See Section 5.)

\sglinecolor: color of lines around payoffs. Default: black. Example: \def\sglinecolor{lightgray}.

\sgtextcolor: color of text in game (action labels, payoffs, player labels). Default: black. Example: \def\sgtextcolor{blue}.

7. History

Version 1

1993/12 First version.

Version 2

1997/10 Labels allowed (via additional optional argument). Parameters added: \sglabelsep, \sgcolsep, \irplwcgltrue, \irpawcgltrue, and \gamemathtrue.

2002/6 (version 2.1) Modification of LaTeX's \@array changed so that it does not change spacing in tabular environment. Color functionality added. Parameters added: \gamevaligntrue, \ssualtrue (allowing more control over vertical spacing when \gamestretch is greater than 1), \sglinecolor, \sgtextcolor.

2004/5/13 (version 2.11) Minor bug fix.

8. Bugs

For some reason (the fact that the style plays around with catcodes?) you cannot make \begin{game} ... \end{game} the argument of a macro, and if you include a game in a figure with other stuff, the position is not right.

(A game is fine alone in a figure.) The latter problem seems to be solved by putting the game in a minipage of the appropriate width.