

The mahjong package*

Daniel Schmitz

`schmytzi@gmail.com`

April 15, 2021

Contents

1	Introduction	2
2	Mahjong Tiles	2
2.1	Suited Tiles	2
2.2	Honor Tiles	2
3	MPSZ Algebraic Notation	3
3.1	Standard Notation	3
3.2	Extensions	3
4	Typesetting Mahjong Tiles in Your Document	4
5	Acknowledgments	4

Abstract

The `mahjong` package provides a $\text{\LaTeX} 2_{\epsilon}$ and $\text{\LaTeX} 3$ interface for typesetting mahjong tiles using an extended version of MPSZ algebraic notation. Features include spaces, rotated, blank, and concealed tiles, as well as red fives. The size of the mahjong tiles can be controlled using a package option and an optional argument of `\mahjong`. It is primarily aimed at Riichi (aka. Japanese) Mahjong but can be used to typeset any style of mahjong. However, flower tiles and jokers are currently missing.

*This document corresponds to `mahjong` v1.0, dated 2021/04/14




























1 Introduction


Mahjong is a tile-based game originating from China which is popular in East and South-East Asia and has since spread throughout the world. The `mahjong` package provides an interface for typesetting mahjong tiles and hands using MPSZ algebraic notation. This documentation assumes familiarity with the game in general but none of its many styles. Nonetheless, some basic terms will be defined because of differing vocabulary among players.

2 Mahjong Tiles

2.1 Suited Tiles

The suited tiles are referred to as follows:

Suit	Tiles
Bamboo	        
Dots	        
Character	        

Suited tiles are referred to using the pattern $\langle value \rangle \langle suit \rangle$. For instance,  is called *4 Bamboo*.

2.2 Honor Tiles

This documentation refers to the seven honor tiles as follows:








Winds			
			
East Wind (E)	South Wind (S)	West Wind (W)	North Wind (N)
Dragons			
			
White Dragon	Green Dragon	Red Dragon	

Table 1: MPSZ notation reference. Each tile is identified by its column's number and its row's letter.

	0	1	2	3	4	5	6	7	8	9
s										
p										
m										
z										

3 MPSZ Algebraic Notation

3.1 Standard Notation

MPSZ notation assigns each tile an identifier consisting of a digit and a letter (table 1). For suited tiles, the digit corresponds to the tile's value and the letter to its suit, Bamboo

(s), Dots (p) or Character (m). For instance, 2m identifies (2 Character). The only exception of this rule are red fives, whose numeric value is 0. Red 5 Bamboo, for example, has identifier 0s. Honor tiles are assigned the "suit" z, with 1z – 4z corresponding to E, S, W and N, and 5z – 7z to the white, green and red dragon, respectively.

Collections of tiles, such as melds or hands, are represented by concatenating the identifiers of the tiles they comprise. For instance, 3s4s5s corresponds to . Groups of tiles sharing the same suit can be abbreviated by omitting all but the last suit identifier. The above example can also be expressed as 345s. Spaces are ignored and the notation is case-insensitive.


3.2 Extensions

Spaces. Spaces can be inserted using -: 444s-567s produces .


Concealed Tiles. Concealed (or face-down) tiles are represented by X. They don't need a suit identifier and don't act as one. 123s X 456s and 123 X 456s are therefore equivalent.

Blank Tiles. Blank or unknown tiles are represented by ?. Just like concealed tiles, they don't change the current suit. 123s ? 456s and 123 ? 456s are equivalent, for instance.

Rotation. Inserting an apostrophe (') rotates the *preceeding* tile counter-clockwise.

For instance, 6'66m is rendered as . This can only be done once per tile, i.e. it is not possible to rotate them 180° or 270°. When you want to rotate the last tile of a group, it doesn't matter whether the apostrophe appears before or after the suit, so 77'm and 77m' are equivalent.

Rotation and Stacking. Quotes (") cause the *preceeding* tile to be rendered as two

rotated and stacked tiles. For instance, 77"7z produces .

4 Typesetting Mahjong Tiles in Your Document

`\mahjong` The main interface is `\mahjong [<height>] {<hand>}`. *<hand>* refers to a tile sequence in MPSZ notation as discussed above and *<height>* specifies the height of the rendered mahjong tiles. If *<height>* is not specified, the default height is used.

`\mahjong_typeset_hand:n` The L^AT_EX 3 interface for rendering mahjong tiles are `\mahjong_typeset_hand:n` and its variants. This macro accepts the hand to be rendered in MPSZ notation. The height can be specified by setting `\l_mahjong_tile_height` and the default height is saved in `\g_mahjong_default_height`.

The default height can be set using the package's `height` parameter. For instance, `\usepackage[height=2\baselineskip]{mahjong}` sets the default size of mahjong tiles to double the value of `\baselineskip` in the context they are rendered in. The parameter defaults to `\baselineskip`.

5 Acknowledgments

The mahjong tiles used in this package were created by GitHub user [FluffyStuff](#). The original repository is [FluffyStuff/riichi-mahjong-tiles](#), used under CC-BY Version 4.0.

```

1 <*pkg>
2 <@@=mahjong>
3 \NeedsTeXFormat{LaTeX2e}[2019/10/01]
4 \RequirePackage{expl3}
5 \ProvidesExplPackage{mahjong}{2021/04/14}{1.0}{Typeset Mahjong Hands}
6 \RequirePackage{xparse}
7 \RequirePackage{l3keys2e}
8 \RequirePackage{graphicx}
9 \RequirePackage{stackengine}

10 \msg_new:nnnn {mahjong} {invalid_token}
11 {Token ~ #1 ~ is ~ not ~ valid ~ in ~ MPSZ ~ notation}
12 {Valid ~ tokens ~ are ~ digits ~ 0-9, ~ m, ~ p, ~ s, ~ z, ~ x, ~ -, ~ ?, ~ ', ~ and ~ " }
13 \msg_new:nnnn {mahjong} {unknown_tile}
14 {I~don't~know~tile~#1.}
15 {Please~check~the~documentation~for~recognized~tiles.}

```

```

16 \msg_new:nnnn {mahjong} {unknown_orientation}
17 {Orientation ~ #1 ~ is ~ unknown}
18 {This ~ should ~ not ~ happen.~ Please ~ create ~ a ~ bug ~ report.}
19
20 \keys_define:nn {mahjong} {
21     height .dim_gset:N = \g_mahjong_default_height
22 }
23
24 % Identifiers for all suits
25 \cs_new:Npn \c__mahjong_suits_tl {mpsz}
26 % Allowed tokens
27 \cs_new:Npn \c__mahjong_allowed_tokens_tl {0123456789mpsz-?x''}
28
29 % Variables have to be declared globally
30 \tl_new:N \l__mahjong_suit_tl
31 \tl_new:N \l__mahjong_tiles_tl
32 \tl_new:N \l__mahjong_reversed_tl
33 \tl_new:N \l__mahjong_hand_tl
34 \tl_new:N \l__mahjong_current_suit_tl
35 \tl_new:N \l__mahjong_current_group_tl
36 \tl_new:N \l__mahjong_current_char
37
38
39 \dim_set:Nn \g_mahjong_default_height {\baselineskip}
40 \dim_new:N \l_mahjong_tile_height
41
42 \int_new:N \l__mahjong_tile_orientation_int
43 \seq_new:N \l__mahjong_tile_maps_seq
44 \str_new:N \l__mahjong_file_path_str
45
46
47 \ProcessKeysPackageOptions{mahjong}

```

`__mahjong_make_tile:nn` Inserts a mahjong tile into the input stream. This functions only handles that use the front background and have a foreground, i.e. regular and blank tiles.
`__mahjong_make_tile:VV`
`__mahjong_make_tile:xV`
`__mahjong_make_tile:nV`

```

48 \cs_set:Npn \__mahjong_make_tile:nn #1#2 {
49     \file_if_exist:nTF {#1} {
50         \int_case:nnF {#2} {
51             {0} {
52                 \stackinset{c}{0pt}{c}{0pt}{
53                     \includegraphics[
54                         angle=0,
55                         height=.85\l_mahjong_tile_height]
56                     {#1}
57                 }{
58                     \includegraphics[
59                         angle=0,
60                         height=\l_mahjong_tile_height]
61                     {tiles/Front.pdf}
62                 }
63             } {1} {
64                 \stackinset{c}{0pt}{c}{0pt}{
65                     \includegraphics[

```

```

66         angle=90,
67         width=.85\l_mahjong_tile_height]
68     {#1}
69 }{
70     \includegraphics[
71         angle=90,
72         width=\l_mahjong_tile_height]
73     {tiles/Front.pdf}
74 }
75 } {2} {
76     % Stack 2 rotated tiles on top of each other.
77     \stackon [0pt] {
78         \__mahjong_make_tile:nn {#1} {1}
79     } {
80         \__mahjong_make_tile:nn {#1} {1}
81     }
82 }
83 } {
84     \msg_fatal:nx {mahjong} {unknown_orientation} {#2}
85 }
86 } {
87     \msg_error:nx {mahjong} {unknown_tile} {#1}
88 }
89 }
90
91 \cs_generate_variant:Nn \__mahjong_make_tile:nn {VV, xV, nV}

```

(End definition for __mahjong_make_tile:nn.)

\mahjong_typeset_hand:n Parseses and typesets a mahjong hand in MPSZ notation. Set \l_mahjong_tile_height to control the tiles' size.

```

92 % Parses a full hand
93 \cs_set:Npn \mahjong_typeset_hand:n #1 {
94     \tl_set:Nx \l__mahjong_hand_tl {\text_lowercase:n {#1}}
95     % MPSZ notation is easier to parse right-to-left, so reverse string
96     % There is no string reversal function but we can reverse token lists
97     % Token lists and strings are freely convertible between each other
98     \tl_set:Nx \l__mahjong_reversed_tl {\tl_reverse:V \l__mahjong_hand_tl}
99     \tl_map_variable:NNn \l__mahjong_reversed_tl \l__mahjong_current_char {
100         % Check if we recognize the current token
101         \exp_args:NVV \tl_if_in:nnF \c__mahjong_allowed_tokens_tl \l__mahjong_current_char {
102             \msg_error:nx {mahjong} {invalid_token} {\l__mahjong_current_char}
103         }
104         \exp_args:NVV \tl_if_in:nnTF \c__mahjong_suits_tl \l__mahjong_current_char {
105             % If we've found a suit identifier, change the current suit
106             \tl_set:NV \l__mahjong_current_suit_tl \l__mahjong_current_char
107         } {
108             \str_case:VnF \l__mahjong_current_char {
109                 {'} {
110                     % An apostrophe indicates that the next tile is rotated
111                     \int_set:Nn \l__mahjong_tile_orientation_int {1}
112                 }
113                 {"} {

```

```

114         % Quotes mean the next tile is actually 2 rotated tiles stacked on top of
115         \int_set:Nn \l__mahjong_tile_orientation_int {2}
116     }
117 } {
118     % Default case: We've got a complete tile identifier
119     % Turn it into a property list
120     \prop_clear:N \l_tmpa_prop
121     \prop_put:NnV \l_tmpa_prop {suit} \l__mahjong_current_suit_tl
122     \prop_put:NnV \l_tmpa_prop {face} \l__mahjong_current_char
123     \prop_put:NnV \l_tmpa_prop {orientation} \l__mahjong_tile_orientation_int
124     % Add it to the beginning of the sequence (we are parsing in reverse)
125     \seq_put_left:NV \l__mahjong_tile_maps_seq \l_tmpa_prop
126     \int_set:Nn \l__mahjong_tile_orientation_int {0}
127 }
128 }
129 }
130 % Typesetting begins here. Sequence is in correct order
131 \seq_map_variable:NNn \l__mahjong_tile_maps_seq \l_tmpa_prop {
132     \prop_get:NnV \l_tmpa_prop {face} \l_tmpa_tl
133     \prop_get:NnV \l_tmpa_prop {orientation} \l_tmpa_int
134     \str_case:VnF \l_tmpa_tl {
135         {-} {
136             % If the current face is a dash, insert a space
137             \includegraphics[height=\l_mahjong_tile_height]{tiles/space.pdf}
138         } {x} {
139             % Insert a flipped tile
140             \int_case:nn {\l_tmpa_int} {
141                 {0} { % Upright
142                     \includegraphics[
143                         angle=0,
144                         height=\l_mahjong_tile_height]
145                     {tiles/Back.pdf}
146                 } {1} { % Rotated
147                     \includegraphics[
148                         angle=90,
149                         width=\l_mahjong_tile_height]
150                     {tiles/Back.pdf}
151                 } {2} { % Rotated and stacked
152                     \stackon [Opt] {
153                         \includegraphics[
154                             angle=90,
155                             width=\l_mahjong_tile_height]
156                         {tiles/Back.pdf}
157                     } {
158                         \includegraphics[angle=90,
159                             width=\l_mahjong_tile_height]
160                         {tiles/Back.pdf}
161                     }
162                 }
163             }
164         } {?} { % Blank tile
165             \__mahjong_make_tile:nV {tiles/Blank.pdf} \l_tmpa_int
166         }
167     } { % Any other tile identified by a code.

```

```

168         \_mahjong_make_tile:xV {tiles/\l_tmpa_tl\prop_item:Nn \l_tmpa_prop {suit}.pdf} \
169     }
170 }
171 % Clear the list for the next invocation
172 \seq_clear:N \l__mahjong_tile_maps_seq
173 }
174
175 % Have TeX automatically expand the argument for us
176 \cs_generate_variant:Nn \_mahjong_typeset_hand:n {x}

```

(End definition for \mahjong_typeset_hand:n. This function is documented on page ??.)

\mahjong This is the only L^AT_EX 2_ε macro in this package. It typesets a mahjong hand.

```

177 \NewDocumentCommand{\mahjong}{0{\g_mahjong_default_height} m}{
178     \dim_set:Nn \l_mahjong_tile_height {#1}
179     \mahjong_typeset_hand:n {#2}
180 }
181 </pkg>

```

(End definition for \mahjong. This function is documented on page ??.)