

# Representação e Resolução de *Rīchi Mahjong* como um Problema de Planejamento

Anny G. N. Figueira e Gabriel G. dos Santos


<sup>1</sup>Escola Politécnica - Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS)  
Porto Alegre – RS – Brasil

{anny.figueira, gabriel.giordani}@acad.pucrs.br

## Abstract.

**Resumo.** Este trabalho apresenta uma versão do jogo Mahjong, chamada *Rīchi Mahjong* e trata da implementação de uma inteligência artificial capaz de jogar, bem como de um simulador para o jogo. São utilizadas técnicas de planejamento para o desenvolvimento desta inteligência artificial.

## 1. Introdução

Mahjong é um jogo baseado em estratégia e sorte, amplamente difundido na Ásia e presente em diversas formas, sendo uma delas a variante japonesa clássica, também conhecida como *Rīchi Mahjong*  utilizada neste trabalho. Por ser um jogo de informação incompleta e que contém elementos de sorte, bem como ações que alteram o fluxo de jogo, trata-se de um problema relativamente complexo de se resolver.

Espera-se ao final do desenvolvimento que o sistema seja capaz de realizar um planejamento de jogo de acordo com alguma estratégia, bem como replanejar e mudar de estratégia conforme o seu andamento.

## 2. Regras e fluxo de jogo

*Rīchi Mahjong*, referenciado a partir daqui como simplesmente Mahjong, é um jogo de turno com 4 jogadores, todos contra todos, cujo objetivo principal é possuir a maior quantidade de pontos ao final. É jogado com um total de 136 peças compostas por números e símbolos e são apresentadas a seguir.

### 2.1. Peças

As peças são divididas entre peças numeradas (de três diferentes naipes) e honras (ventos e dragões). Cada peça possui 4 cópias no jogo.

As peças numeradas são divididas em 3 naipes: círculos (Figura 1), bambus (Figura 2) e caracteres (Figura 3). São numeradas de 1 a 9. As peças de 2 a 8 são chamadas simples, enquanto as peças de valor 1 e 9 são chamadas terminais.



Figura 1. Círculos



Figura 2. Bambus



Figura 3. Caracteres

As honras são divididas em dois tipos: ventos (Figura 4) e dragões (Figura 5). Os ventos são leste, sul, oeste e norte, enquanto os dragões são branco, verde e vermelho.



Figura 4. Ventos



Figura 5. Dragões

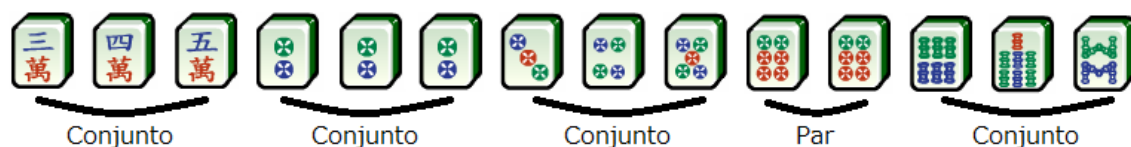
## 2.2. Jogo

As 136 peças disponíveis são arranjadas de forma que 70 peças são reservadas em uma chamada muralha morta, utilizada para fins especiais. As demais peças são utilizadas para distribuição das mãos iniciais e o restante disponibilizado para compra pelos jogadores.

Cada jogador inicia o jogo com 13 peças e o objetivo é formar uma mão válida composta de 4 conjuntos e um par (com exceção das mãos especiais, que serão abordadas em 2.2.4). O conceito de mão válida é atrelado à contagem de pontos desta mão e será apresentado em detalhes na seção 2.2.3.

Um conjunto é composto de uma sequência (*chī*), trinca (*pon*) ou quádrupla (*kan*), enquanto o par é composto de duas peças iguais. Uma sequência só pode ser formada por peças numeradas e de mesmo naipe. Os demais conjuntos podem ser compostos por quaisquer tipos de peças, contanto que sejam iguais.

A figura 6 apresenta uma mão válida composta de peças simples. Ela é formada por um *chī* de caracteres 3, 4 e 5, *pon* de círculos 2, *chī* de círculos 3, 4 e 5, par de círculos 6 e *chī* de bambus 6, 7 e 8.



**Figura 6. Exemplo de mão válida**

Uma partida de Mahjong consiste em 8 mãos, valor que pode variar a depender de algumas condições que serão abordadas em 2.2.2. Após o decorrer destas mãos, o jogador com maior número de pontos é considerado vencedor. Cada mão completa garante uma quantidade específica de pontos, cujo cálculo é apresentado em 2.2.3.

### 2.2.1. Fluxo de Jogo

O jogo é baseado em turnos, sendo executado em sentido anti-horário. Em sua vez, o jogador realiza uma compra e um descarte.

Ao início do jogo, é realizado um sorteio e cada jogador é assinalado a um vento. Cada jogador então senta em sua respectiva posição (chamada ordem dos ventos) na mesa de acordo com as posições de um compasso invertido horizontalmente: leste é o *dealer*, à sua direita está o sul, do outro lado o oeste e à sua esquerda, o norte.

O jogador leste inicia o jogo com 14 peças, diferentemente dos demais que iniciam com 13. Caso ele inicie com uma mão válida, pode declarar *tsumo* e encerrar a mão, iniciando a contagem de pontos. Caso ele inicie com 4 peças iguais, ele pode declarar um *kan*, que será explicado a seguir.

Caso contrário, ele irá apenas descartar uma peça. Os descartes são realizados no centro da mesa, com a peça virada pra cima.

Qualquer jogador pode requisitar a peça através de uma chamada *pon* ou *kan* no momento em que um descarte é realizado, pegando para si a peça descartada a fim de completar seu conjunto. Conjuntos completos desta maneira são apresentados à direita do jogador com as peças lado a lado e virados para cima, à mostra para todos os jogadores. Isso é feito de forma a sinalizar que este conjunto foi concluído a partir de um descarte.

Qualquer jogador pode também realizar uma chamada de *ron*, completando sua mão com a peça descartada e iniciando a contagem de pontos.

Caso nenhuma destas chamadas tenha sido realizada, o próximo jogador da direita pode requisitar a peça através de uma chamada de *chī*, completando uma sequência com a peça descartada. Esta sequência também é colocada à sua direita e virada para cima, à mostra para todos jogadores.

No momento em que uma chamada é realizada, o fluxo de jogo é alterado. O jogador que realizou a chamada obrigatoriamente realiza um descarte e então finaliza seu turno, passando a vez para o jogador à sua direita (exceto, é claro, caso outra chamada seja realizada em seu descarte).

Jogadores que possuam quatro peças iguais podem declarar um *kan* em seu turno. Neste caso, as quatro peças são posicionadas lado a lado à sua direita, com as peças das

extremidades viradas para baixo e as do meio viradas para cima. Isto é feito de forma a indicar que este *kan* não foi completo a partir de um descarte. Ao declarar um *kan*, o jogador compra uma peça da muralha morta e pode encerrar sua mão com esta peça. Caso isso não ocorra, ele descarta normalmente e encerra seu turno.

No momento em que um jogador compra uma peça que completa uma mão válida, ele pode declarar *tsumo* e iniciar a contagem de pontos. Além disso, caso o jogador chegue no ponto em que falta apenas uma peça para completar sua mão, ele pode declarar *rīchi* sinalizando este fato para os demais jogadores. A partir deste ponto, ele não pode mais fazer alterações na sua mão. Esta declaração é levada em consideração na contagem de pontos.

### 2.2.2. Término do Jogo

Uma partida de Mahjong consiste de duas rodadas: rodada leste e rodada sul. O vento predominante de cada rodada influencia na contagem de pontos, conforme será apresentado em 2.2.3.

Quando uma mão é finalizada, alternam-se os ventos dos jogadores em sentido anti-horário de acordo com a ordem dos ventos apresentada em 2.2.1. Ou seja, cada jogador assume o vento do jogador à sua esquerda.

Ao final da primeira troca de ventos completa (em que todos jogadores já assumiram a posição leste), a rodada troca e o vento predominante se torna o sul. Quando o *dealer* completa uma mão, não há alteração nos ventos. Por isso, uma partida pode durar mais do que as 8 mãos estipuladas inicialmente.

Além disso, caso a pontuação de algum jogador chegue a zero, o jogo acaba imediatamente.

Caso esgotem-se as peças disponíveis para compra sem que um jogador tenha completado uma mão, é declarado empate e o jogo avança para a próxima mão. Pela configuração de jogo, são reservadas 14 peças em uma chamada “muralha morta”, indisponível para compra. Ou seja, no momento em que um jogador comprar a 15ª peça restante, a mão atual é encerrada.



### 2.2.3. Contagem de Pontos

### 2.2.4. Mãos Especiais

## 3. Tecnologias Empregadas

Esta seção apresenta uma breve descrição das tecnologias empregadas na implementação do jogador artificial. Foram realizadas provas de conceito com a linguagem PDDL, mais abrangente, e a linguagem GDL, específica para modelagem de jogos.

### 3.1. Planning Domain Definition Language

A *Planning Domain Definition Language* (PDDL)  é a linguagem para a especificação de problemas de planejamento.  Ela tem como objetivo descrever os predicados de um determinado domínio, assim como as ações possíveis e os efeitos de cada ação.

A primeira versão da linguagem [McDermott et al. 1998] suporta as seguintes estruturas:

- Condicionais
- Quantificação universal em universos dinâmicos
- Axiomas
- Restrições de segurança
- Hierarquia de ações compostas por sub-ações e sub-objetivos

As ações são descritas com base em pré-condições e efeitos (pós-condições).

O objetivo pode ser descrito como um problema de otimização (maximização ou minimização) ou como atingir um predicado ou conjunto de predicados em específico.

É dado um estado inicial para o problema de planejamento que, então, elabora uma sequência de ações a fim de atingir o objetivo.

A abordagem através de planejamento foi pensada como uma forma simples de formalizar o domínio do problema do ponto de vista do jogador, uma vez que trata as restrições e consequências de cada ação e permite abstrair o fluxo de jogo, que é abordada no simulador, mas não a nível de jogador.

### 3.2. Lógica Temporal de Ações

A *Temporal Logic of Actions* (TLA), produzido por Leslie Lamport[Lamport a], introduz o conceito de tempo para fórmulas lógicas. Para que isto seja possível, são introduzidos operadores para representar “sempre”, “em algum momento” e “em seguida”. Tais operadores são:

- $\Box$  : Sempre
- $\Diamond$  : Em algum momento
- $\bigcirc$  : Em seguida


Com a utilização da TLA é possível transformar afirmações como “Para qualquer estado, se um requerimento de um recurso acontece, então ele irá, em algum momento, ser atendido” na fórmula:  $\Box(request \rightarrow \Diamond acknowledged)$ , onde *request* representa o ato de requerer um recurso e *akwonledged* representa o ato de atender.

Como é possível especificar a ação de um estado em um momento do tempo, a TLA proporciona uma maneira relativamente simples para formalizar o fluxo do jogo. Por este motivo, é desejado utilizar a TLA para a formalização do programa que irá simular o fluxo de jogo e permitir a interação das IAs e do jogador.


Será utilizada a TLA Toolbox[Lamport b] para a escrita e verificação da formalização em TLA.


### 3.3. Game Description Language with Incomplete Information

A *Game Description Language* (GDL) é uma linguagem utilizada para criar “jogadores gerais”, capazes de receber a descrição de um jogo e jogar de acordo com as regras deste, sem a necessidade de algoritmos específicos. O funcionamento da linguagem, assim como a estrutura dos jogadores, pode ser visto em [sta ].

O principal problema que foi encontrado inicialmente ao tentar utilizar a GDL foi que ela não permite a descrição de um jogo com informação incompleta  e é o caso do *Rīchi Mahjong*. Para contornar este problema foi decidido utilizar a GDL-II (*Game Description Language with Incomplete Information*)[Thielscher 2010] que introduz o conceito de informação incompleta através de novos comandos que permitem definir compras randômicas de peças e como a informação a respeito dos movimentos possíveis é distribuída entre os jogadores.

## Referências

General game playing. <http://ggp.stanford.edu/>. 

Lamport, L. The tla home page. <https://lamport.azurewebsites.net/tla/tla.html>. 

Lamport, L. The tla toolbox. <https://lamport.azurewebsites.net/tla/toolbox.html>.

McDermott, D., Ghallab, M., Howe, A., Knoblock, C., Ram, A., Veloso, M., Weld, D., and Wilkins, D. (1998). Pddl-the planning domain definition language.

Thielscher, M. (2010). A general game description language for incomplete information games. In *AAAI*, volume 10, pages 994–999.