SIMPLE MATHEMATICAL MODEL FOR HANGZHOU MAHJONG

ZHAN CHENG

1. Composition

Suppose the player has a characters, b dots and c bamboos with d winds, and each tile of its type belongs to set A, B, C, D respectively. From the rules of Hangzhou Mahjong, we have $a + b + c + d = 14$ and $0 \leq a, b, c, d \leq 14$.

1.1. **Basic Composition.** Define the composition without chows and pungs as set X, then we call set $\mathcal{A}$ is a **Basic composition** which satisfies with $|A \cap X_A| \geq 3$, and we can get set $\mathcal{B}$,$\mathcal{C}$,$\mathcal{D}$ in this way. Also, set K is defined as $|A \cap X_A| \leq 2$, and we can get set L, M corresponding with B, C.

1.2. **State Space Θ and Φ.** Define the **State space** as $\Theta = \{\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}\}$ $(\Theta \neq \emptyset)$, and c_i can be any one basic composition of the state space Θ. **Special state space** Φ is defined as $\Phi = \{K, L, M\}$ $(\Theta = \emptyset)$, and f_i can be any element of the special state space Φ. Actually, Φ can be reduced to $\Phi = \{x, y\}$ since $(\Theta = \emptyset)$, which means there is only one combination way $3 \times 4 + 2$, only 2 tiles left.

1.3. **Decision Function θ, λ and δ.** Define the function $\theta(\mathcal{X}) = \theta_t$ to be the **Decision function**, which implies that action θ_t will be taken in the given basic composition $\mathcal{A}, \mathcal{B}, \mathcal{C}$. Similarly, define the function $\lambda(\mathcal{Y}) = \lambda_t$ to be the **Decision function**, which implies that action λ_t will be taken in the given basic composition $\mathcal{D}$, and $\delta(Z) = \delta_t$ implies the action δ_t will be taken in the given basic composition K, L, M.

1.4. **Decision Space Π.** From 1.2 and 1.3, we can define **Decision space** as $\Pi = \{\theta(\mathcal{A}), \theta(\mathcal{B}), \theta(\mathcal{C}), \lambda(\mathcal{D})\}$, and d_j can be any one decision function of the decision space Π. And **Special decision space** as $\phi = \{\delta(K), \delta(L), \delta(M)\}$, and h_j can be any one decision of the decision space ϕ.

1.5. **Revenue Function q.** It is defined as $q(c_i, d_j)$ or $q(f_i, h_j)$, which implies the value of the decision d_j for given state c_i or the decision h_j for given state f_i.

2. Permutation

Denote pair as p, chow as c and pung as p. We have winning composition $W_1 = p + N \times c + M \times p$, if we have N chows and M pungs. It also can be seen as $W_2 = P \times o + Q \times b + N \times c + M \times p + R \times r$, which implies we have P other permutations, Q basic permutations, N chows, M pungs and R residues. The conception of the basic and other permutations will be given in the next section. Compare the equation W_1 with W_2, we can find that we should reduce the number of o, b, r to 0 to win the game, which means we should discard one tile from these permutations or residues.

2.1. **Basic Permutation.** For given certain type of composition, tiles can be expressed as permutations by letters $A, B, C, D, E \cdot \cdots$. For example ABC represents all permutations that are not a chow for given certain type of composition. And we define the permutation with up to five letters plus permutation $ABCDEF$ as the **Basic permutation**. All of the basic permutation are listed as follows:

$$ABC, AAB, ABCD, AABC, AABB, ABCDE, AABCD, AABBC, ABCDEF$$

2.2. **Other Permutation.** It is defined as the combination of the basic permutation and chows and pungs are NOT allowed to appear in the **other permutation** with up to 9 letters. All of the other permutation are listed as follows:

$$AABCDE, AABBCD, AABBCC, AABCDEF, AABBCDE, AABBCCD$$

$$AABBCDEF, AABBCCDE, AABBCCDD, AABBCCDEF, AABBCCDDE$$

It is clear that some of the permutations are ignored, that's because there are chows appearing. (Theorem 2.1)

Theorem 2.1. *There must be a chow in the composition if there are at least seven tiles of a certain type.*

Proof. The problem can be seen as: There must be three balls sitting together if we put 7 balls into 9 boxes.

We first consider the results of putting 6 balls into 9 boxes

$$6 = 1 \times 6 \text{ or } 2 \times 1 + 1 \times 4 \text{ or } 2 \times 2 + 1 \times 2 \text{ or } 2 \times 3$$

In this equation, it shows all the possible situation of the distribution. And we only need to check that for 6 balls into 9 boxes, 5 balls into 8 boxes, 4 balls into 7 boxes and 3 balls into 6 boxes, all of the ball are not adjacent to each other.

Consider the formula C^m_{n-m+1}, and it holds if and only if $n + 1 \geq 2m$, which implies that m balls into n boxes with none of the balls is adjacent to each other.

Through checking, we find the 4 into 7 and 3 into 6 satisfy the condition. If we add 1 ball, it will become the situation of 7 into 9, and it doesn't hold any more, which implies there must be three balls sitting together if we put 7 balls into 9 boxes. $\square$

3. Value

3.1. Value For Basic.

3.1.1. *Value Factor.* Value Factors are set to each tile of a given basic permutation, with f_1 : the defence value factor, f_2 : the neighbor factor, f_3 : the remote eat factor, f_4 : the normal eat factor, f_5 : the meet factor, f_6 : the abnormal eat factor and f_7 : the pair factor. In these factors, f_1 is already set, and other factors need to be determined by the given composition in $\mathcal{A}, \mathcal{B}, \mathcal{C}$. Similarly, f_{L1} and f_{L2} are defined as the pair factor and the meet factor respectively by the given composition in $\mathcal{D}$. In particular, f_{L1} and f_{L2} of the null tile are defined as $+\infty$ since it is a wild tile. f_D as the defence factor for K, L, M.

3.1.2. *Value Vector and Probability vector.* All the value factors form the **Value vector** for one tile of a given basic permutation in $\mathcal{A}, \mathcal{B}, \mathcal{C}$.

$$\overrightarrow{\theta_i} = (f_{1i}, f_{2i}, f_{3i}, f_{4i}, f_{5i}, f_{6i}, f_{7i})$$

Similarly,

$$\overrightarrow{\lambda_i} = (f_{L1i}, f_{L2i})$$

For K, L, M

$$\overrightarrow{\delta_x} = (f_{Dx})$$

$$\overrightarrow{\delta_y} = (f_{Dy})$$

for given permutation $\mathcal{D}$. **Probability vector** is defined by the probability of getting the remaining same tile, and it expands to the vector form to join in the vector calculation.

For $\mathcal{A}, \mathcal{B}, \mathcal{C}$,

$$\overrightarrow{R} = (r_1, r_2, r_3, r_4, r_5, r_6, r_7)^T$$

For $\mathcal{D}$

$$\overrightarrow{R} = (r_{L1}, r_{L2})^T$$

FOR K, L, M

$$\overrightarrow{R} = (r_x, r_y)$$

3.1.3. *Value Matrix.* If the current composition M in $\mathcal{A}, \mathcal{B}, \mathcal{C}$ has m tiles, then its **Value matrix** V can be expressed as

$$V = \begin{pmatrix} \overrightarrow{\theta_1} \\ \vdots \\ \overrightarrow{\theta_m} \end{pmatrix}$$

To be more specific,

$$V = \begin{pmatrix} f_{11} & \cdots & f_{71} \\ \vdots & \ddots & \vdots \\ f_{1m} & \cdots & f_{7m} \end{pmatrix}$$

Similarly, if the current composition N in $\mathcal{D}$ has n tiles, it can be expressed as:

$$V_L = \begin{pmatrix} \overrightarrow{\lambda_1} \\ \vdots \\ \overrightarrow{\lambda_n} \end{pmatrix}$$

To be more specific,

$$V_L = \begin{pmatrix} f_{L11}, f_{L21} \\ \vdots \\ f_{L1n}, f_{L2n} \end{pmatrix}$$

For K, L, M

$$V_D = (\overrightarrow{\delta_x}, \overrightarrow{\delta_y})^T$$

3.1.4. *Total Value Vector.* From 3.2 and 3.3, we define **Total value vector**

$$\overrightarrow{A} = \overrightarrow{V} \cdot \overrightarrow{R} = (s_1, s_2, \cdots s_m)^T$$

$$\overrightarrow{B} = \overrightarrow{V_L} \cdot \overrightarrow{R} = (s_1, \cdots s_n)^T$$

$$\overrightarrow{C} = \overrightarrow{R} \cdot \overrightarrow{V_D} = (s_x, s_y)$$

In this formula, we can easily get the minimum value s_{min} and its value vector $\overrightarrow{\theta_{min}}$ or $\overrightarrow{\lambda_{min}}$ or $\overrightarrow{\delta_{min}}$

3.2. **Value For Other.**

3.2.1. *Split.* As mentioned above, the other permutation can be regard as the combination of the basic permutation. Let's say (C_{1i}, C_{2i}) is the ith split of permutation M in $\mathcal{A}, \mathcal{B}, \mathcal{C}$, which satisfies the following condition:

$$\begin{cases} C_{1i} \cup C_{2i} = M \\ C_{1i} \cap C_{2i} = \emptyset \\ (C_{1i}, C_{2i}) \Leftrightarrow (C_{2i}, C_{1i}) \end{cases}$$

3.2.2. *Split Set.* We define the collection of all the split of a given permutation M as a **Split set** C,

$$C = \{(C_{11}, C_{21}), (C_{12}, C_{22}), \cdots, (C_{1k}, C_{2k})\}$$

3.2.3. *Minimum Value and its Value Vector.* As mentioned above, for each ith split, the minimum value and its value vector can be expressed as

$$V_{Ci} = min \left\{ \theta(C_{1i}) \cdot \overrightarrow{R}, \theta(C_{2i}) \cdot \overrightarrow{R} \right\} = \theta(C_{ni}) \cdot \overrightarrow{R}$$

where $n = 1$ or 2.

Then the minimum value of the split set can be determined through

$$V_{Cmin} = min \{V_{C1}, V_{C2}, \cdots, V_{Ck}\}$$

and its value vector can be obtained by $V_{Cmin} = \theta(C_{nmin}) \cdot \overrightarrow{R}$

The statement of the expression of the split of **three elements** is skipped, because it is similar to the two elements case.

4. Probability

As mentioned above, the probability vector is defined by the probability of getting the remaining same tile, so how to determine its expression?

4.1. **Rules.** There are 136 tiles in Hangzhou Mahjong with 36 characters, 36 dots, 36 bamboos and 28 winds, each player will have 13 tiles when the game starts, and the banker have an extra tile. The playing order is anticlockwise and starts from the banker, each player should draw a tile and discard a tile to keep 13 tiles. As soon as the player draw a tile and form a winning composition, he wins the game.

Actually, there will be 20 tiles left in the table to increase its complexity of Hangzhou Mahjong.

4.2. **Probability Vector.** The probability of getting a certain tile will be expressed as

$$P_\theta = \frac{4 - e_{\theta i} - g_{\theta i} - \eta_{\theta i}}{63 - n - 4m + \alpha + \sum_{k=1}^{3} \beta_k \left| n_{pk} - n_{qk} \right|}$$

Then $r_i = P_\theta$ when $i = 1, 2, \cdots, 7$ for $\mathcal{A}, \mathcal{B}, \mathcal{C}$.

As for the $\mathcal{D}$, θ_i is changed to λ_i, and the parameter α is reduced since the winds tiles cannot be taken the action eat.

$$P_\lambda = \frac{4 - e_{\lambda i} - g_{\lambda i} - \eta_{\lambda i}}{63 - n - 4m + \sum_{k=1}^{3} \beta_k \left| n_{pk} - n_{qk} \right|}$$

Then $r_{L1}, r_{L2} = P_\lambda$.

Also, for K, L, M, we cannot determine the type of 2 tiles, so we use t to control the parameter α,

$$P_\delta = \frac{4 - e_{\delta i} - g_{\delta i} - \eta_{\delta i}}{63 - n - 4m + \alpha \times t + \sum_{k=1}^{3} \beta_k \left| n_{pk} - n_{qk} \right|}$$

If the tile is not wind, then $t = 1$, else $t = 0$. $r_x, r_y = P_\delta$.

In these formulas, m stands for the mth round and in the first round $m = 0$; n represents the order of the player, each player is marked with $0, 1, 2, 3$ anticlockwise from the banker.

$e_{\theta i}$ represents the number of tiles i the player has; $g_{\theta i}$ represents the number of tiles i which can be seen from the player's view on the table; $\eta_{\theta i}$ represents the number of tiles i which can not be seen, and it is waited to be determined as a variable.

α means the times that the action eat has occurred in this game, and β_k means the times that the action meet has occurred by player k from the current player's view. The next player's order is $k = 1$, then $k = 2$, finally $k = 3$. n_{pk}, n_{qk} represents the order of the player respectively who does the action meet.

5. Strategy

Finally we can get the conclusion with constraint condition, that is

$$\begin{cases} \overrightarrow{\lambda_{j1}} \cdot (1,0)^T \leq S_i \\ \overrightarrow{\lambda_{j2}} \cdot (0,1)^T \geq S_i \\ \min q(c_i, d_j) \\ \Theta \neq \emptyset \end{cases}$$

$$\begin{cases} \min q(f_i, h_j) \\ \Theta = \emptyset \end{cases}$$

where $(i = 1, 2, \cdots, m)$ and $(j = 1, 2, \cdots, n)$.

We should take the decision d_j or h_j as the best strategy.