

# BombBuster: Formal Game Rules

## 1 Formal Definition

Let

$$W = \{w_1, w_2, \dots, w_m\} m = 48$$

be the **multiset of wires**, where each wire  $w \in W$  has a **type/value**

$$\text{val}(w) \in \{1, \dots, K\}. K = 12$$

Each value  $k \in \{1, \dots, K\}$  appears  $r_k$  times in  $W$ . All wires together form the bomb.

$$r_k = 4$$

## 2 Players and Hands

Let there be  $N$  players  $P = \{p_1, p_2, \dots, p_N\}$ .

- The wires are **partitioned** into disjoint subsets

$$W = \bigsqcup_{i=1}^N W_i, \quad W_i = \{w_{i,1}, \dots, w_{i,n_i}\}$$

such that each  $p_i$  holds  $n_i$  wires.  $n_i = m/N$

- Each player orders their subset  $W_i$  in **ascending order of value**, i.e.

$$\text{val}(w_{i,1}) < \text{val}(w_{i,2}) < \dots < \text{val}(w_{i,n_i}).$$

The ordering is visible to the player but the values are **hidden to all others**.

## 3 Game Objective

Reveal (cut) every wire in  $W$  according to the rules below. A **cut** means revealing the wire's value, marking it as defused.

## 4 Turn Mechanics

A **turn** is played by one active player  $p_a$ .

### 4.1 Action Declaration

Player  $p_a$  selects:

- one of their own value of its own wires  $\text{val}(w_{a,i})$ , and
- one target wire  $w_{b,j}$  belonging to another player  $p_b$ ,  $b \neq a$ ,

and **declares**:

$$\text{call}(p_a, i, p_b, j, v)$$

which reads as:

“I cut wire in position  $j$  of player  $p_b$ , declaring it is of value  $v$ .”

### 4.2 Validity Condition

The declaration is **valid** if and only if:

$$\text{val}(w_{a,i}) = \text{val}(w_{b,j}) = v.$$

If this holds, both wires are **revealed** (cut). Otherwise, the bomb counter increases (a strike is recorded).

### 4.3 Special Case

If a player  $p_i$  holds *all remaining wires* of a certain value  $v$ , then  $p_i$  may **cut all of them directly**, without needing to make a call.

## 5 Knowledge Representation

Each player ( $p_i$ ) maintains a **belief system**

$$\mathcal{B}_i(t)$$

representing their probabilistic knowledge about the hidden wire configuration of all players at time ( $t$ ).

Formally,  $(\mathcal{B}_i(t))$  is a joint probability distribution over the random variables

$$\{X_{k,j} \mid k \in \{1, \dots, N\}, ; j \in \{1, \dots, n_k\}\},$$

where each ( $X_{k,j} \in \{1, \dots, K\}$ ) denotes the (unknown) value of the wire in position ( $j$ ) of player ( $p_k$ ).

Thus,

$$\mathcal{B}_i(t) : \Pr_i(X_{1,1}, \dots, X_{N,n_N} \mid \text{information available up to } t).$$

## Local Marginals and Derived Beliefs

From  $(\mathcal{B}_i(t))$ , player  $(p_i)$  can derive **marginal** and **conditional** beliefs such as:

- **Single-wire belief**

$$\Pr_i(X_{k,j} = v),$$

the probability that wire  $(j)$  of player  $(p_k)$  has value  $(v)$ .

- **Pairwise belief**

$$\Pr_i(X_{k,j} = v, X_{k,j+1} = w),$$

capturing uncertainty about relative ordering or joint constraints between adjacent wires.

- **Cross-player belief**

$$\Pr_i(X_{a,j} = v, X_{b,\ell} = w),$$

encoding hypotheses about which player holds which copies of the same wire value.

These allow statements such as:

- “Wire in position  $(j)$  of player  $(p_k)$  can be of value  $(v)$  or  $(w)$ .”
- “Either (or both) wires in positions  $(j)$  and  $(j+1)$  of player  $(p_k)$  could be of value  $(v)$ .”
- “Wire of value  $(v)$  is most likely held by player  $(p_a)$  or  $(p_b)$ .”

## Initialization

At  $(t = 0)$ , each player’s belief system is initialized as:

$$\mathcal{B}_i(0) = \Pr(X_{1,1}, \dots, X_{N,n_N})$$

subject to:

1. **Partition constraint:** each value  $(v \in \{1, \dots, K\})$  appears exactly  $(r_v)$  times across all variables.
2. **Ordering constraint:** for each player  $(p_k)$ ,

$$X_{k,1} < X_{k,2} < \dots < X_{k,n_k}.$$

In practice,  $(\mathcal{B}_i(0))$  is uniform over all valid configurations consistent with these constraints.

## Belief Update

After each action (call, reveal, or strike), player ( $p_i$ ) updates ( $\mathcal{B}_i(t)$ ) using Bayes' rule:

$$\mathcal{B}_i(t+1) \propto \mathcal{B}_i(t) \cdot \Pr(\text{new observation} \mid X_{1,1}, \dots, X_{N,n_N}).$$

Observations include:

- Successful or failed calls (revealed equalities or inequalities),
- Newly revealed wire values,
- Inferred eliminations of impossible configurations.

This representation allows each player's knowledge to encode not just *what each wire might be*, but also *how these possibilities are correlated* across positions and players.

## 6 Deduction Updates

Whenever a valid call occurs:

$$\text{call}(p_a, i, p_b, j, v)$$

and the result is correct, the following updates apply:

- $\text{val}(w_{a,i}) = \text{val}(w_{b,j}) = v$  become **known**.
- $v$  is removed from all other  $\mathcal{B}_{i',j'}(t)$  sets (since each copy of  $v$  has one fewer remaining instance).
- The relative order constraints on other wires in  $W_a$  and  $W_b$  tighten the possible intervals of their remaining  $\mathcal{B}_{i,j}$ .

Even if the call fails, the information is updated: the claimed equality was **false**, so  $v \notin \mathcal{B}_{b,j}$  or  $\mathcal{B}_{a,i}$ .

## 7 Game State Summary

A complete game state at time  $t$  is represented by:

$$G_t = (\{\mathcal{B}_{i,j}(t)\}_{i,j}, R_t, H_t, \text{strikes}_t)$$

where:

- $\mathcal{B}_{i,j}(t)$ : possible values of each hidden wire,
- $R_t$ : revealed wires,
- $H_t$ : hidden wires,
- $\text{strikes}_t$ : number of errors made so far.

## 8 Implementation via Belief Sets

To implement the reasoning process, we can move from the formal joint probability distribution  $\mathcal{B}_i(t)$  to a more concrete representation. Each player  $p_i$  maintains a collection of sets, where each set represents the possible values for a single unknown wire.

### 8.1 Belief Representation

Let  $C_{i,k,j}(t)$  be the **candidate set** of possible values for wire  $w_{k,j}$  (the  $j$ -th wire of player  $p_k$ ) from the perspective of player  $p_i$  at time  $t$ .

$$C_{i,k,j}(t) \subseteq \{1, \dots, K\}$$

Player  $p_i$  has full knowledge of the value of wire  $w_{k,j}$  if and only if  $|C_{i,k,j}(t)| = 1$ .

### 8.2 Initialization ( $t = 0$ )

At the start of the game, player  $p_i$  initializes these sets based on their own private information and the public rules of the game.

1. **Own Wires:** Player  $p_i$  knows the values of their own wires. For each of their wires  $j \in \{1, \dots, n_i\}$ :

$$C_{i,i,j}(0) = \{\text{val}(w_{i,j})\}$$

2. **Other Players' Wires:** For any other player  $p_k$  (where  $k \neq i$ ), the initial sets  $C_{i,k,j}(0)$  are constructed by applying all known constraints.

- Initially, every set contains all possible values:  $C_{i,k,j}(0) = \{1, \dots, K\}$ .
- **Partition Constraint:** Player  $p_i$  knows which values they hold. Let the multiset of all values in the game be  $M$ , and the multiset of values held by  $p_i$  be  $M_i$ . The remaining values  $M_{\text{other}} = M \setminus M_i$  must be partitioned among all other players. A value  $v$  can be pruned from  $C_{i,k,j}(0)$  if there is no consistent assignment of values in  $M_{\text{other}}$  to the other players' wires that respects the ordering constraint.
- **Ordering Constraint:** The values within any player's hand are strictly increasing. This powerful constraint allows for significant initial pruning. For any wire  $w_{k,j}$ , its value must be greater than at least  $j - 1$  other values and smaller than at least  $n_k - j$  other values in player  $p_k$ 's hand. A practical way to enforce this is to iteratively prune the sets until they are consistent. For example, a value  $v$  can be removed from  $C_{i,k,j}(t)$  if:

$$\left| \bigcup_{l=1}^{j-1} \{u \in C_{i,k,l}(t) \mid u < v\} \right| < j - 1$$

or

$$\left| \bigcup_{l=j+1}^{n_k} \{u \in C_{i,k,l}(t) \mid u > v\} \right| < n_k - j$$

This process is repeated for all sets until no more values can be removed.

### 8.3 Belief Update via Constraint Propagation

After any action, new information  $\mathcal{O}$  becomes publicly available. Every player  $p_i$  updates their candidate sets  $C_{i,k,j}(t)$  to  $C_{i,k,j}(t+1)$  by incorporating  $\mathcal{O}$  and propagating the new constraints until a stable state (a fixed point) is reached.

#### 8.3.1 Successful Call

A valid call,  $\text{call}(p_a, i, p_b, j, v)$ , reveals the public information  $\mathcal{O} = \{\text{val}(w_{a,i}) = v, \text{val}(w_{b,j}) = v\}$ . The update proceeds as follows for every player:

1. **Direct Update:** The values of the two wires are now known with certainty.

$$C_{i,a,i}(t+1) := \{v\} \quad \text{and} \quad C_{i,b,j}(t+1) := \{v\}$$

2. **Partition Update:** Let  $R(v, t)$  be the count of publicly revealed wires with value  $v$  at time  $t$ . If  $R(v, t+1) = r_v$ , then all instances of value  $v$  have been found. For any other wire  $w_{k,l}$  whose value is not yet certain,  $v$  is removed:

$$\text{If } |C_{i,k,l}(t)| > 1, \quad C_{i,k,l}(t+1) := C_{i,k,l}(t) \setminus \{v\}$$

3. **Ordering Update:** The new knowledge about  $\text{val}(w_{a,i})$  and  $\text{val}(w_{b,j})$  tightens the bounds on adjacent wires for players  $p_a$  and  $p_b$ .

- For player  $p_a$ , all wires  $l < i$  must have values less than  $v$ :

$$C_{i,a,l}(t+1) := C_{i,a,l}(t) \cap \{1, 2, \dots, v-1\} \quad \forall l < i$$

- For player  $p_a$ , all wires  $l > i$  must have values greater than  $v$ :

$$C_{i,a,l}(t+1) := C_{i,a,l}(t) \cap \{v+1, \dots, K\} \quad \forall l > i$$

The same logic is applied to the wires of player  $p_b$  around position  $j$ .

4. **Propagation:** Any reduction in a set  $C_{i,k,l}$  may trigger further reductions in other sets due to the ordering and partition constraints. The pruning process described during initialization must be re-applied iteratively until no more changes occur.

### 8.3.2 Failed Call

Assuming the active player  $p_a$  makes a call based on their known wire value  $\text{val}(w_{a,i}) = v$ , a failed call implies  $\text{val}(w_{b,j}) \neq v$ . This provides the public information  $\mathcal{O} = \{\text{val}(w_{b,j}) \neq v\}$ .

1. **Direct Update:** The value  $v$  is eliminated as a possibility for wire  $w_{b,j}$ .

$$C_{i,b,j}(t+1) := C_{i,b,j}(t) \setminus \{v\}$$

2. **Propagation:** This reduction in  $C_{i,b,j}$  may now trigger a cascade of further deductions through the iterative application of ordering and partition constraints, as described above.

### 8.3.3 Special Case Declaration

If player  $p_k$  declares and reveals all their  $c$  remaining (uncut) wires of value  $v$ , the update is similar to a successful call. For each revealed wire  $w_{k,j_1}, \dots, w_{k,j_c}$ :

1. The sets are updated to singletons:  $C_{i,k,j_m}(t+1) := \{v\}$  for  $m \in \{1, \dots, c\}$ .
2. Partition and ordering constraints are propagated from this new information.