associated with S_{11} and reward $r_{2,i}$ and satisfies the equation as follows:

$$A_i = \sum_{j=1}^{n} S_j \tag{17}$$

$$r_{2,i} = \sum_{j=1}^{n} S_j * \beta_j \tag{18}$$

S_j in the above equation is the split part of A_i in the overlap region. When the location of the S_j area is close to the centre frequency and the S_{11} curve is close to negative infinity, the weighted β_j is larger, giving a higher positive reward. Similarly, R'' is the difference between the current and previous states. Then, R'' is calculated as follows:

$$R'' = r_{2,i} - r_{2,i} \tag{19}$$

3.3.3 Stepwise training

A database is created, and it stores information regarding different frequencies. The initial state is selected based on the current reward R' in state s_i. The formula for R' is calculated by Formula 16. The smaller the distance from the target frequency in state s_i is, the larger the reward.

Consists of two parts. (1) The agent adjusts the circuit grid parameter matrix to transform the centre frequency and collects a dataset with different centre frequencies. (2) The circuit grid parameter matrix with the closest target frequency is extracted from the dataset and set to the initial state. Subsequently, the agent learns to adjust the matrix to complete the filter design.

In designing filters of different frequencies, the stepwise training method avoids repetitive detection of central frequencies, reducing the agent's exploration space.

3.3.4 Pseudocode of PAAC-K

The pseudocode for the PAAC-K is given in algorithm 1.

Algorithm 1 Parallel Advantage Actor-Critic with K-means

Select c samples as the initial clustering centers $C_i = \{a_1, a_2, a_3, \ldots, a_c\}$
Get n vectors $x_i\,(i = 1, 2, ..., n)$
while $C_i = \{a_1, a_2, a_3, \ldots, a_c\}$ is changing **do**
  for x_1 to x_n **do**
    for a_1 to a_c **do**
      Calculate the distance from x_i to $C_i = \{a_1, a_2, a_3, \ldots, a_c\}$
      $z_{ik} = \begin{cases} 1 & \text{if } \|x_i - a_k\|^2 = \min_{l \leqslant k \leqslant c} \|x_i - a_k\|^2 \\ 0 & \text{otherwise} \end{cases}$
      $J(G_i, C_i) = \sum_{i=1}^{n} \sum_{k=1}^{c} z_{ik} \|x_i - a_k\|^2$
    end for
  end for
  for a_1 to a_c **do**
    Recalculate its clustering center $C_i = \{a_1, a_2, a_3, \ldots, a_c\}$
    Update C_i using $a_k = \frac{\sum_{i=1}^{n} z_{ik} x_{ij}}{\sum_{i=1}^{n} z_{ik}}$
  end for
end while
Instantiate set n of $work_n$ environments
Initialize step counter $epoches = 0$ and network weights θ, θ_v
Collects data from clustering results as agent's action
Get the state s_1 using the Stepwise Training method
while $epoches < epoches_{\max}$ **do**
  Set initial state as s_1
  while $(t < T_{\max})$ and (the agent not selects "end" action) **do**
    Sample $\boldsymbol{a}_t$ from $\pi(\boldsymbol{a}_t \mid \boldsymbol{s}_t; \theta)$
    Calculate $\boldsymbol{v}_t$ from $V(\boldsymbol{s}_t; \theta_v)$
    for parallel $work_1$ to $work_n$ **do**
      Perform action $a_{t,n}$ in environment $work_n$
      Observe new state $s_{t+1,n}$ and reward $r_{t+1,n}$
    end for
    save data(rewards,states,actions)
  end while
  $\boldsymbol{R}_{t_{\max}+1} = \begin{cases} 0 & \text{for terminal } \boldsymbol{s}_t \\ V(s_{t_{\max}+1}; \theta) & \text{for non-terminal } \boldsymbol{s}_t \end{cases}$
  for $t = 1 \rightarrow T_{\max}$ **do**
    $\boldsymbol{R}_t = \boldsymbol{r}_t + \gamma \boldsymbol{R}_{t-1}$
  end for
  $d\theta = \frac{1}{N \cdot T_{\max}} \sum_{n=1}^{N} \sum_{t=1}^{T_{\max}} (R_{t,n} - v_{t,n}) \nabla_\theta \log \pi (a_{t,n} \mid s_{t,n}; \theta) + \beta \nabla_\theta H(\pi(s_{n,t}; \theta))$
  $d\theta_v = \frac{1}{N \cdot T_{\max}} \sum_{n=1}^{N} \sum_{t=1}^{T_{\max}} \nabla_{\theta_v} (R_{t,n} - V(s_{t,n}; \theta_v))^2$

  Update θ using $d\theta$ and θ_v using $d\theta_v$
end while

4 Application example

4.1 Background of the microstrip filter design

In the traditional method, any filter can be designed from the original low-pass prototype based on the frequency. A suitable filter structure is obtained by combining performance indicators through frequency variations and the transformation of individual components. In this process, a low-pass filter prototype is converted into a high-pass, bandpass, or bandstop filter. Then, a two-port network [46] analysis and electromagnetic wave theory are used to optimize the filter and complete the design.

4.2 Hairpin bandpass filter

The hairpin structure [47] is adopted to complete the design of the bandpass filters. This structure uses a polystyrene dielectric substrate material [48] with a dielectric constant of 2.6. A hairpin resonator with a double-barrelled step impedance resonator with a cross-finger structure is an improved half-wavelength coupled microstrip line filter with a more compact structure.