

BÁO CÁO PROJECT KTTT

THUẬT TOÁN TỐI ƯU HÓA BẦY ĐÀN (PSO)

Sinh viên thực hiện:

Đinh Xuân Thái Dương 20235689

Đỗ Danh Dũng 20235685

Mục lục

1	Thuật toán Gốc (Original PSO)	2
1.1	Khái niệm và Nguồn gốc	2
1.2	Cơ chế hoạt động cơ bản	2
1.3	Mô hình Toán học (Global Best Version)	2
1.4	Quy trình thực hiện thuật toán	3
1.5	Mã giả (Pseudocode)	3
2	Độ phức tạp tính toán	4
3	Sự hội tụ	5
4	Các Biến thể của PSO (PSO Variants)	5
4.1	Local Best (Biến thể Cục bộ - lbest)	5
4.2	Inertia Weight (Trọng số Quán tính – w)	5
4.3	Các biến thể và khái niệm khác	6
5	Ứng dụng Thực tiễn (Applications)	6
6	SWOT	6
6.1	Điểm mạnh	6
6.2	Điểm yếu	7
6.3	Cơ hội	7
6.4	Các mối đe dọa/Thách thức	7
7	Thực thi thuật toán vào bài toán truyền thông	8
7.1	Mô tả bài toán	8
7.2	Ánh xạ PSO vào bài toán	8
7.3	Triển khai thuật toán PSO Gốc (Original PSO)	8
7.4	Triển khai Biến thể PSO (Inertia Weight PSO)	8
8	Kết quả mô phỏng và So sánh	9
8.1	Thiết lập mô phỏng	9
8.2	Phân tích kết quả	9
8.3	Kết luận về sự cải thiện	10
9	Tham khảo	10

1 Thuật toán Gốc (Original PSO)

1.1 Khái niệm và Nguồn gốc

Particle Swarm Optimization (PSO) là một thuật toán tối ưu hóa dựa trên quần thể, thuộc lĩnh vực Trí tuệ Bầy đàn (Swarm Intelligence) và Đời sống Nhân tạo (Artificial Life). Thuật toán này được giới thiệu lần đầu vào năm 1995 bởi James Kennedy và Russell Eberhart.

Cảm hứng của PSO xuất phát từ hành vi xã hội của các loài động vật trong tự nhiên như đàn chim hoặc đàn cá khi tìm kiếm thức ăn hoặc nơi trú ẩn. Trong tự nhiên, các cá thể trong đàn chia sẻ thông tin để cùng nhau tìm ra vị trí tốt nhất.

1.2 Cơ chế hoạt động cơ bản

Trong không gian tìm kiếm của bài toán (problem space), PSO tạo ra một tập hợp các "hạt" (particles). Mỗi hạt đại diện cho một giải pháp tiềm năng.

- Mỗi hạt có một vị trí (position) và một vận tốc (velocity).
- Các hạt di chuyển trong không gian tìm kiếm, được dẫn dắt bởi kinh nghiệm của chính nó và kinh nghiệm tốt nhất của cả đàn.
- Mục tiêu là tìm ra vị trí tối ưu toàn cục (global optimum) thông qua sự hội tụ của cả đàn.

1.3 Mô hình Toán học (Global Best Version)

Phiên bản gốc thường được gọi là Global Best (gbest), nơi toàn bộ đàn được kết nối hoàn toàn và mọi hạt đều biết vị trí tốt nhất mà bất kỳ hạt nào trong đàn từng tìm thấy. Quá trình cập nhật trạng thái của một hạt i tại bước lặp it bao gồm hai phương trình chính:

a. Cập nhật Vận tốc ($v_{i,d}$): Vận tốc mới được tính dựa trên ba thành phần: đà quán tính, thành phần nhận thức (cá nhân) và thành phần xã hội (cộng đồng).

$$v_{i,d}^{it+1} = v_{i,d}^{it} + C_1 \cdot Rnd(0, 1) \cdot (pb_{i,d}^{it} - x_{i,d}^{it}) + C_2 \cdot Rnd(0, 1) \cdot (gb_d^{it} - x_{i,d}^{it}) \quad (1)$$

Trong đó:

- $v_{i,d}^{it}$: Vận tốc cũ, giúp hạt tiếp tục di chuyển theo hướng trước đó.
- C_1 : Hằng số gia tốc nhận thức, kéo hạt về phía vị trí tốt nhất mà bản thân nó từng tìm thấy ($pb_{i,d}$).
- C_2 : Hằng số gia tốc xã hội, kéo hạt về phía vị trí tốt nhất mà toàn đàn từng tìm thấy (gb_d).
- $Rnd(0, 1)$: Yếu tố ngẫu nhiên từ 0 đến 1 giúp hạt khám phá các vùng không gian lạ.

b. Cập nhật Vị trí ($x_{i,d}$): Vị trí mới đơn giản là vị trí cũ cộng với vận tốc mới.

$$x_{i,d}^{it+1} = x_{i,d}^{it} + v_{i,d}^{it+1} \quad (2)$$

1.4 Quy trình thực hiện thuật toán

1. **Khởi tạo:** Tạo ngẫu nhiên vị trí và vận tốc cho các hạt trong không gian tìm kiếm.
2. **Đánh giá:** Tính toán giá trị thích nghi (fitness) cho từng hạt dựa trên hàm mục tiêu $f(x)$.
3. **Cập nhật P-Best (pb):** Nếu vị trí hiện tại tốt hơn vị trí tốt nhất trước đó của hạt, cập nhật pb .
4. **Cập nhật G-Best (gb):** Nếu vị trí tốt nhất của hạt tốt hơn vị trí tốt nhất toàn cục của đàn, cập nhật gb .
5. **Di chuyển:** Cập nhật vận tốc và vị trí cho tất cả các hạt theo công thức ở mục 1.3.
6. **Lặp lại:** Quay lại bước 2 cho đến khi thỏa mãn điều kiện dừng (ví dụ: số vòng lặp tối đa hoặc đạt độ lỗi cho phép).

1.5 Mã giả (Pseudocode)

Mô tả chi tiết quy trình thực hiện thuật toán PSO:

Algorithm 1 Initialize

```
1: for each particle  $i$  in  $S$  do
2:   for each dimension  $d$  in  $D$  do
3:     //initialize all particles' position and velocity
4:      $x_{i,d} = Rnd(x_{min}, x_{max})$ 
5:      $v_{i,d} = Rnd(-v_{max}/3, v_{max}/3)$ 
6:   end for
7:   //initialize particle's best position
8:    $pb_i = x_i$ 
9:   //update the global best position
10:  if  $f(pb_i) < f(gb)$  then
11:     $gb = pb_i$ 
12:  end if
13: end for
```

Algorithm 2 Particle Swarm Optimization (Global Best)

```
1: //initialize all particles
2: Initialize
3: repeat
4:   for each particle  $i$  in  $S$  do
5:     //update the particle's best position
6:     if  $f(x_i) < f(pb_i)$  then
7:        $pb_i = x_i$ 
8:     end if
9:     //update the global best position
10:    if  $f(pb_i) < f(gb)$  then
11:       $gb = pb_i$ 
12:    end if
13:  end for
14:  //update particle's velocity and position
15:  for each particle  $i$  in  $S$  do
16:    for each dimension  $d$  in  $D$  do
17:       $v_{i,d} = v_{i,d} + C_1 * Rnd(0, 1) * [pb_{i,d} - x_{i,d}] + C_2 * Rnd(0, 1) * [gb_d - x_{i,d}]$ 
18:       $x_{i,d} = x_{i,d} + v_{i,d}$ 
19:    end for
20:  end for
21:  //advance iteration
22:   $it = it + 1$ 
23: until  $it < MAX\_ITERATIONS$ 
```

2 Độ phức tạp tính toán

Độ phức tạp của PSO phụ thuộc vào số lượng hạt, số chiều không gian và số vòng lặp.

Giả sử:

- P : Số lượng hạt (Population size).
- D : Số chiều của bài toán (Dimension).
- T : Số vòng lặp tối đa (Max Iterations).
- C_{cost} : Chi phí tính toán hàm mục tiêu (Fitness function evaluation).

a. Độ phức tạp thời gian (Time Complexity):

- Khởi tạo: $O(P \cdot D)$.
- Cập nhật vị trí và vận tốc: Mỗi hạt cần $O(D)$ phép tính trong mỗi vòng lặp. Với P hạt và T vòng lặp, tổng chi phí là $O(T \cdot P \cdot D)$.
- Đánh giá hàm mục tiêu: $O(T \cdot P \cdot C_{cost})$.
- **Tổng quát:** $O(T \cdot P \cdot D)$ (nếu coi chi phí hàm mục tiêu là hằng số hoặc tỷ lệ với D). PSO được đánh giá là thuật toán có chi phí tính toán thấp, dễ cài đặt song song.

b. Độ phức tạp không gian (Space Complexity):

- Cần lưu trữ vị trí hiện tại (X), vận tốc (V) và vị trí tốt nhất cá nhân ($Pbest$) cho mỗi hạt.
- Tổng quát: $O(P \cdot D)$.

3 Sự hội tụ

- **Điều kiện hội tụ lý thuyết:** Sự hội tụ của PSO phụ thuộc lớn vào các tham số w, c_1, c_2 . Theo phân tích của Clerc và Kennedy (2002), để đảm bảo hạt không dao động hỗn loạn mãi mãi, các tham số cần thỏa mãn các điều kiện nhất định (thường liên quan đến Hệ số co - Constriction Factor).
- **Hiện tượng hội tụ sớm (Premature Convergence):** Một điểm yếu của PSO là khi gbest rơi vào một cực trị địa phương (local optimum), các hạt còn lại sẽ bị "kéo" về đó quá nhanh do thành phần xã hội c_2 làm mất khả năng khám phá các vùng khác.
- **Hiện tượng Stagnation (Sự trì trệ):** Khi các hạt đã hội tụ về gbest nhưng gbest chưa phải là tối ưu toàn cục, vận tốc của các hạt sẽ tiến về 0, khiến quần thể bị "đóng băng" và không thể cải thiện lời giải thêm được nữa.

4 Các Biến thể của PSO (PSO Variants)

4.1 Local Best (Biến thể Cục bộ - lbest)

- **Khác biệt:** Thay vì chia sẻ thông tin với toàn bộ đàn (gbest), trong biến thể này, hạt chỉ chia sẻ thông tin trong một "vùng lân cận" (neighborhood) nhỏ hơn.
- **Cơ chế:** Hạt bị ảnh hưởng bởi vị trí tốt nhất của các hạt hàng xóm thay vì vị trí tốt nhất toàn cục.
- **Ưu/Nhược điểm:**
 - Hội tụ chậm hơn so với phiên bản gốc.
 - Tuy nhiên, nó bao phủ không gian tìm kiếm tốt hơn và ít bị mắc kẹt tại các cực trị địa phương (local minima) hơn.
 - Các vùng lân cận cần chồng lấn nhau để đảm bảo thông tin cuối cùng vẫn lan truyền được đến toàn đàn.

4.2 Inertia Weight (Trọng số Quán tính – w)

Đây là một trong những cải tiến quan trọng nhất, được Shi và Eberhart giới thiệu năm 1998.

- **Cơ chế:** Thêm một tham số w (trọng số quán tính) vào thành phần vận tốc cũ trong phương trình cập nhật vận tốc.
 - Phương trình trở thành:
- $$v_{i,d}^{it+1} = w \cdot v_{i,d}^{it} + C_1 \cdot Rnd(0, 1) \cdot (pb_{i,d}^{it} - x_{i,d}^{it}) + C_2 \cdot Rnd(0, 1) \cdot (gb_d^{it} - x_{i,d}^{it}) \quad (3)$$
- **Mục đích:** Cân bằng giữa khả năng khám phá (exploration - tìm vùng mới) và khai thác (exploitation - tìm kỹ quanh vùng tốt đã biết).
 - Nếu bỏ qua thành phần quán tính ($w = 0$), hạt sẽ di chuyển không có "ký ức" về hướng đi trước đó và chỉ loay quanh vùng giải pháp đã tìm thấy.
 - Giá trị w lớn giúp hạt khám phá diện rộng (global search).
 - Giá trị w nhỏ giúp hạt tinh chỉnh lời giải (local search).
 - **Tham số đề xuất:** Thực nghiệm cho thấy giá trị w nên nằm trong khoảng $[0.9, 1.2]$ hoặc giảm tuyến tính theo thời gian để tối ưu hiệu suất.

4.3 Các biến thể và khái niệm khác

- **Constriction Factor (Hệ số co):** Được Clerc và Kennedy đề xuất năm 2002 để kiểm soát vận tốc mà không cần giới hạn cứng (velocity clamping), giúp đảm bảo sự hội tụ.
- **Velocity Clamping (Kẹp vận tốc):** Đặt một giá trị v_{max} để ngăn các hạt bay ra khỏi không gian tìm kiếm do vận tốc tăng quá nhanh.
- **Topology (Cấu trúc xã hội):** Cách các hạt kết nối với nhau (ví dụ: vòng tròn, sao, lưới) ảnh hưởng lớn đến hiệu suất thuật toán.

5 Ứng dụng Thực tiễn (Applications)

PSO đã có sự gia tăng bùng nổ về số lượng ứng dụng kể từ khi ra đời. Theo khảo sát của Riccardo Poli năm 2007, có khoảng 700 tài liệu ghi nhận các ứng dụng của PSO trong 26 danh mục khác nhau.

- **Mạng Nơ-ron nhân tạo (Artificial Neural Networks):** Dùng để huấn luyện mạng, ví dụ như chẩn đoán bệnh Parkinson.
- **Hệ thống điện (Power Systems):** Điều độ tải điện, kiểm soát điện áp và giảm thiểu tổn thất điện năng.
- **Xử lý Ảnh và Video:** Đây là lĩnh vực có nhiều tài liệu nhất, bao gồm phát hiện và nhận diện khuôn mặt, phân đoạn ảnh, và theo dõi tư thế cơ thể.
- **Ăng-ten:** Thiết kế và điều khiển tối ưu mảng ăng-ten.
- **Lập lịch (Scheduling):** Lập lịch công việc trong các hệ thống máy tính phân tán, lập lịch sản xuất (job-shop scheduling).
- **Điện tử:** Tối ưu hóa cuộn cảm trên chip, pin nhiên liệu và bán dẫn.
- **Điều khiển:** Sử dụng trong các bộ điều khiển PI và PID.

6 SWOT

6.1 Điểm mạnh

- **Dễ triển khai:** Thuật toán tương đối dễ cài đặt so với các thuật toán tối ưu hóa khác.
- **Tối ưu hóa toàn cục:** Có khả năng tìm ra giải pháp tối ưu toàn cục thay vì bị mắc kẹt ở tối ưu cục bộ.
- **Không cần đạo hàm:** PSO không yêu cầu thông tin về gradient (đạo hàm) của hàm mục tiêu, rất hữu ích khi thông tin này không có sẵn hoặc hàm không liên tục.
- **Tính song song:** Dễ dàng cài đặt song song để giải quyết các bài toán quy mô lớn.
- **Mạnh mẽ:** Ít nhạy cảm với việc chọn điều kiện ban đầu và hoạt động tốt trong môi trường nhiễu hoặc không chắc chắn.
- **Linh hoạt:** Áp dụng được cho cả bài toán đơn mục tiêu và đa mục tiêu.

6.2 Điểm yếu

- **Nhạy cảm với tham số:** Hiệu suất thuật toán phụ thuộc nhiều vào việc chọn các siêu tham số (hyperparameters) như hệ số gia tốc và trọng số quán tính.
- **Tốc độ hội tụ:** Có thể hội tụ chậm đối với các bài toán tối ưu phức tạp.
- **Khởi tạo:** Nếu các hạt không được khởi tạo đúng cách, thuật toán có thể không hội tụ hoặc mất nhiều thời gian hơn.
- **Chi phí tính toán:** Có thể tốn kém tài nguyên tính toán đối với các bài toán nhiều chiều do phải đánh giá từng hạt ở mỗi vòng lặp.
- **Ổn định hội tụ:** PSO có thể không ổn định và sự hội tụ bị ảnh hưởng bởi những thay đổi nhỏ trong định nghĩa bài toán.

6.3 Cơ hội

- **Kết hợp thuật toán:** Tiềm năng lớn trong việc kết hợp PSO với các thuật toán khác (như Genetic Algorithms, Gradient Descent) để tận dụng ưu điểm của cả hai.
- **Giải quyết dữ liệu lớn:** Khả năng mở rộng và tính toán song song làm cho PSO trở thành ứng viên sáng giá để xử lý các bài toán tối ưu hóa trong Big Data và khai phá dữ liệu.
- **Tối ưu hóa thời gian thực:** Với tốc độ tính toán nhanh, PSO có cơ hội lớn trong các ứng dụng cần phản hồi tức thì như điều khiển robot tự hành hoặc định tuyến mạng động.

6.4 Các mối đe dọa/Thách thức

- **Không đảm bảo tối ưu toàn cục:** Do bản chất ngẫu nhiên không có sự đảm bảo tuyệt đối rằng thuật toán sẽ luôn tìm thấy tối ưu toàn cục, nó vẫn có thể hội tụ vào tối ưu địa phương.
- **Khó khăn trong tinh chỉnh tham số:** Việc thiết lập các tham số (kích thước quần thể, giới hạn vận tốc, các hệ số trọng số) rất quan trọng; nếu thiết lập sai, thuật toán có thể trở nên vô hiệu.
- **Hội tụ sớm:** Hành vi di chuyển dựa trên hàng xóm có thể dẫn đến việc cả đàn cùng hội tụ vào một tối ưu địa phương thay vì toàn cục.
- **Thiếu cơ sở lý thuyết:** PSO thiếu một cơ sở lý thuyết vững chắc so với một số kỹ thuật khác, dẫn đến khó khăn trong việc đảm bảo tính ổn định.

7 Thực thi thuật toán vào bài toán truyền thông

7.1 Mô tả bài toán

Bài toán được lựa chọn để áp dụng thuật toán là **Tối ưu hóa phân bổ công suất đường xuống (Downlink Power Allocation)** trong hệ thống **Scalable Cell-Free Massive MIMO**. Mục tiêu là tìm ra bộ hệ số công suất $\rho = [\rho_1, \rho_2, \dots, \rho_L]$ cho L trạm phát sao cho tối đa hóa hiệu suất phổ (Spectral Efficiency - SE) của người dùng có chất lượng kết nối kém nhất trong mạng (nguyên lý Max-Min Fairness).

Hàm mục tiêu được định nghĩa như sau:

$$\max_{\rho} \left(\min_{l=1, \dots, L} SE_l(\rho) \right) \quad (4)$$

Với ràng buộc tổng công suất phát tại các điểm truy cập (AP) không vượt quá ngân sách năng lượng cho phép.

7.2 Ánh xạ PSO vào bài toán

Việc áp dụng PSO đòi hỏi sự ánh xạ giữa các khái niệm của thuật toán và thực thể trong mạng truyền thông:

- **Hạt (Particle):** Mỗi hạt đại diện cho một giải pháp phân bổ công suất toàn mạng. Vị trí của hạt thứ i là một vector K chiều: $\mathbf{x}_i = [\rho_{i,1}, \rho_{i,2}, \dots, \rho_{i,K}]$, với $\rho_{i,k} \geq 0$.
- **Không gian tìm kiếm:** Tập hợp các số thực dương, bị giới hạn bởi ràng buộc tổng công suất.
- **Hàm thích nghi (Fitness Function):** Giá trị fitness của một hạt chính là giá trị SE nhỏ nhất ($MinSE$) đạt được với cấu hình công suất đó.
- **Xử lý ràng buộc:** Trong quá trình khởi tạo và cập nhật, nếu vector công suất vi phạm ràng buộc tổng công suất, nó sẽ được chuẩn hóa (normalization) để đưa về vùng khả thi.

7.3 Triển khai thuật toán PSO Gốc (Original PSO)

Trong phiên bản gốc được cài đặt tại file `pso_scf.py`, các tham số được thiết lập cố định:

- Hệ số nhận thức (Cognitive) $C_1 = 1.5$.
- Hệ số xã hội (Social) $C_2 = 1.5$.
- Trọng số quán tính (Inertia Weight) $w = 1.0$ (cố định, không có sự suy giảm).

Cơ chế cập nhật vận tốc chỉ dựa trên sự chênh lệch giữa vị trí hiện tại với P_{best} và G_{best} mà không có sự kiểm soát cân bằng động giữa khám phá và khai thác.

7.4 Triển khai Biến thể PSO (Inertia Weight PSO)

Để cải thiện khả năng hội tụ và tránh mắc kẹt tại cực trị địa phương, biến thể Inertia Weight đã được triển khai với cơ chế giảm tuyến tính theo thời gian:

$$w(t) = w_{max} - \frac{w_{max} - w_{min}}{T_{max}} \times t \quad (5)$$

Trong đó:

- $w_{max} = 0.9$: Giá trị lớn ở giai đoạn đầu giúp hạt di chuyển mạnh, khám phá không gian rộng (Exploration).
- $w_{min} = 0.4$: Giá trị nhỏ ở giai đoạn cuối giúp hạt di chuyển chậm lại, tinh chỉnh quanh vùng tối ưu (Exploitation).
- T_{max} : Tổng số vòng lặp tối đa.

Cải tiến này giúp thuật toán ổn định hơn khi áp dụng cho bài toán tối ưu công suất vốn có không gian tìm kiếm liên tục và phức tạp.

8 Kết quả mô phỏng và So sánh

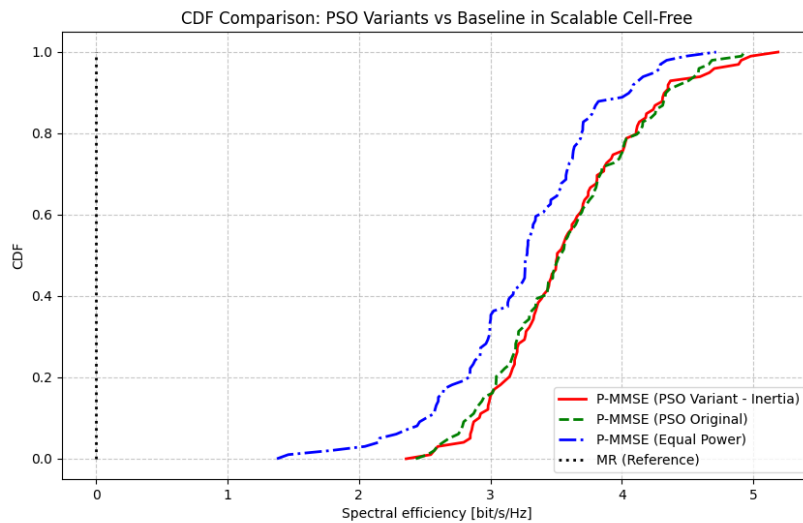
8.1 Thiết lập mô phỏng

Hệ thống được mô phỏng dựa trên các thông số thực tế trong môi trường vô tuyến:

- Số lượng AP (L): 100
- Số lượng người dùng (K): 20
- Mô hình kênh truyền: Kênh Rayleigh có tương quan không gian (Local Scattering Model).
- Phương pháp ước lượng kênh: MMSE.
- Phương pháp kết hợp tín hiệu: P-MMSE (Scalable).

8.2 Phân tích kết quả

Kết quả mô phỏng được đánh giá thông qua biểu đồ Hàm phân phối tích lũy (CDF - Cumulative Distribution Function) của Hiệu suất phổ (SE).



Hình 1: Biểu đồ CDF so sánh hiệu năng giữa các thuật toán

Dựa trên hình 1, ta có các nhận xét sau:

1. **Baseline (Equal Power):** Đường đồ thị nằm xa nhất về phía bên trái, cho thấy xác suất đạt được SE cao là thấp nhất. Đây là phương pháp tham chiếu không có tối ưu hóa.
2. **PSO Gốc (Original):** Đường đồ thị dịch chuyển rõ rệt về phía bên phải so với Baseline. Thuật toán đã tìm được bộ hệ số công suất tốt hơn, cải thiện đáng kể SE cho người dùng ở biên (những người dùng có SE thấp).
3. **PSO Biến thể (Inertia Weight):** Đường đồ thị (màu đỏ) nằm ngoài cùng bên phải, vượt trội hơn so với PSO Gốc. Cơ chế giảm trọng số quán tính giúp thuật toán tìm được điểm tối ưu sâu hơn, nâng cao tính công bằng (Fairness) cho toàn mạng.

8.3 Kết luận về sự cải thiện

So với phương pháp phân bổ công suất đều (Baseline), thuật toán PSO Biến thể đã cải thiện đáng kể hiệu suất phổ của người dùng kém nhất. Việc áp dụng biến thể Inertia Weight không làm tăng độ phức tạp tính toán trong mỗi vòng lặp nhưng mang lại hiệu quả hội tụ tốt hơn, chứng minh tính phù hợp của PSO cho các bài toán tối ưu trong mạng 5G/6G Cell-Free.

9 Tham khảo

- [Particle Swarm Optimization \(PSO\) - An Overview - GeeksforGeeks](#)
- [\(PDF\) Particle Swarm Optimization \(Original Paper 1995\)](#)
- [\(PDF\) Particle Swarm Optimization \(ResearchGate\)](#)