**ویژگی‌های اصلی ارتباط در مغز که باید شبیه‌سازی کنیم**

1. **ارتباط غیرمتمرکز و توزیع‌شده**
   * هر نورون به تعداد زیادی نورون دیگه متصل است، بدون وجود مرکز کنترل واحد که همه چیز رو مدیریت کنه.
2. **ارتباط ناهمزمان و پراکنده**
   * هر نورون به طور مستقل پیام‌ها (پتانسیل‌های عمل) رو در لحظه ارسال و دریافت می‌کنه.
3. **سیناپس‌های پویا و قابل تغییر**
   * قدرت اتصال بین نورون‌ها (وزن سیناپسی) به مرور زمان تغییر می‌کند (تقویت یا تضعیف) که روی مسیر سیگنال‌ها تأثیر می‌گذاره.
4. **انتقال تک‌جهته سیگنال‌ها**
   * سیگنال‌ها از آکسون یک نورون به دندریت‌های نورون بعدی منتقل می‌شوند (مسیرهای مشخص و جهت‌دار).
5. **پیام‌های جمع‌شونده و شرطی (Thresholding)**
   * نورون‌ها با دریافت سیگنال‌ها به صورت تجمعی و رسیدن به آستانه‌ای، تصمیم به ارسال سیگنال بعدی می‌گیرند.
6. **زمان‌بندی و تأخیر انتقال**
   * بین ارسال و دریافت پیام‌ها تاخیر وجود دارد و این نقش مهمی در پردازش اطلاعات دارد.
7. **ارتباطات چندلایه و شبکه‌ای**
   * شبکه از نورون‌های متعددی در لایه‌های مختلف تشکیل شده است که مسیرهای پیچیده و چندگانه‌ای برای عبور سیگنال‌ها دارد.

**پیشنهاد طراحی شبکه عصبی نرم‌افزاری بر اساس این ویژگی‌ها**

**۱. نود (Node) و نورون (Neuron)**

* هر نود متشکل از چند نورون است.
* هر نورون به صورت مستقل فعالیت می‌کند.
* هر نورون دارای لیست اتصالات به نورون‌های دیگر (با وزن سیناپسی) است.

**۲. پیام‌رسانی**

* پیام‌ها **تک‌جهته** و از نورون فرستنده به نورون گیرنده ارسال می‌شوند.
* پیام‌ها در صف‌های انتظار (Queue) ذخیره می‌شوند و با تأخیر زمانی مشخص پردازش می‌شوند.
* هر نورون پیام‌های ورودی را جمع می‌کند و در صورت رسیدن به آستانه (Threshold)، سیگنال خروجی ارسال می‌کند.

**۳. ساختار ارتباطات**

* **سیناپس‌ها** شامل:
  + شناسه نورون هدف
  + وزن اتصال (وزن سیناپسی)
  + تأخیر انتقال (ms یا سیکل‌های زمانی)
* نورون‌ها بر اساس وزن و تجمع پیام‌ها تصمیم‌گیری می‌کنند.

**۴. هماهنگی بین نودها**

* هر نود فقط مسئول نورون‌های خودش است.
* ارتباط بین نودها از طریق پیام‌رسانی (مثل MQTT یا پروتکل دیگر) با پیام‌های نورونی انجام می‌شود.
* پیام‌ها باید شامل اطلاعات وزن، شناسه نورون فرستنده و گیرنده، و زمان ارسال باشند.

**معماری پیشنهادی فنی**

| **لایه/مولفه** | **وظیفه** | **توضیح** |
| --- | --- | --- |
| نورون (Neuron) | دریافت پیام، جمع‌بندی، تصمیم به ارسال | شامل آستانه و وزن سیناپسی |
| نود (Node) | مجموعه‌ای از نورون‌ها و مدیریت آنها | اجرای چرخه پردازش پیام‌ها و هماهنگی |
| سیستم پیام‌رسانی | ارسال پیام‌ها بین نورون‌ها و نودها | MQTT با تاپیک‌های مشخص برای هر سیناپس |
| GodNode (خدای مرکزی) | مدیریت کلی قوانین و نقشه شبکه | فقط کنترل، بدون دخالت مستقیم در پیام‌ها |

**نکات مهم**

* **تاخیرها و زمان‌بندی** باید در مدل لحاظ شوند (مثلاً با صف پیام‌های ورودی و زمان‌بندی ارسال)
* **وزن‌های سیناپسی** باید قابلیت تغییر داشته باشند (برای یادگیری و گسترش شبکه)
* **هر پیام باید شامل:**
  + شناسه نورون فرستنده
  + شناسه نورون گیرنده
  + وزن سیناپسی (یا در metadata)
  + زمان ارسال و دریافت

**گام بعدی: پیاده‌سازی مدل پایه نورون با تجمع پیام، وزن و آستانه**

**طراحی شبکه‌ٔ عصبی توزیع‌شده بود رو اصلاح می‌کنم تا هر نود چند نورون داشته باشه و مدیریت نورون‌ها (ایجاد، حذف، و ارتباط) بر اساس الگویی شبیه سیستم عصبی واقعی باشه.**

**طراحی اصلاح‌شده سیستم عصبی توزیع‌شده**

**الگو:**

* **GodNode → مثل مغز مرکزی (کورتکس پیشانی) که قوانین، اهداف کلان و هماهنگی کلی رو تعیین می‌کنه.**
* **Node → مثل یک ناحیه از مغز که چند نورون داره. هر نود مستقل پردازش می‌کنه ولی گهگاهی با GodNode همگام می‌شه.**
* **Neuron → واحد پردازش که سیگنال می‌گیره، پتانسیلش رو جمع می‌کنه و در صورت عبور از آستانه، سیگنال بعدی رو می‌فرسته.**

**ویژگی‌های اصلاح‌شده**

1. **هر نود یک NeuronManager داره که چند نورون رو مدیریت می‌کنه.**
2. **ارتباط داخلی نورون‌ها در یک نود → از asyncio.Queue (مثل سیناپس محلی).**
3. **ارتباط بین نودها و GodNode → از MQTT (مثل آکسون‌های طولانی).**
4. **Autoscale نورون‌ها: اگر ورودی زیاد بشه، نورون جدید ایجاد می‌شه؛ اگر ترافیک کم بشه، خاموش می‌شن.**
5. **نورون‌ها می‌تونن نقش متفاوت داشته باشن (sensory، integrator، decision، memory).**
6. **GodNode فقط دستورالعمل‌ها و وزن‌های کلی رو می‌فرسته؛ پردازش اصلی محلی انجام می‌شه → شبیه واقعیت مغز.**