

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



دانشگاه صنعتی اصفهان  
دانشکده مهندسی برق و کامپیوتر

## بهبود کارایی الگوریتم یادگیری فدرال برای داده‌های غیرمستقل و غیریکنواخت با در نظر گرفتن میزان شباهت بین شبکه‌های عصبی در دستگاه‌های نهایی

پایان‌نامه کارشناسی ارشد مهندسی کامپیوتر - هوش مصنوعی و رباتیکز

علی بزرگ‌زاد

استاد راهنما

دکتر امیر خورسندی

## فهرست مطالب

<u>صفحه</u>	<u>عنوان</u>
سه	فهرست مطالب
۱	چکیده

### فصل اول: مقدمه

۲	۱-۱ شناخت موضوع
۳	۱-۱-۱ یادگیری متمرکز
۳	۱-۱-۲ یادگیری غیر متمرکز
۳	۱-۱-۳ یادگیری توزیع شده
۴	۲-۱ یادگیری فدرال
۵	۳-۱ تاریخچه یادگیری فدرال
۵	۴-۱ کاربرد یادگیری فدرال
۶	۱-۴-۱ یادگیری فدرال در شهر هوشمند
۷	۲-۴-۱ یادگیری فدرال در بیمارستان
۷	۳-۴-۱ یادگیری فدرال در فروشگاه برنامه‌های کاربردی موبایل
۸	۵-۱ دید کلی از روند موضوع و بیان هدف پژوهش
۸	۶-۱ مروری بر روند ارائه مطالب پایان‌نامه

### فصل دوم: مفاهیم پایه در یادگیری فدرال

۹	۱-۲ مقدمه
۱۰	۲-۲ چالش‌های موجود در یادگیری فدرال
۱۰	۱-۲-۲ تبادل داده بین سرور و کاربران
۱۰	۲-۲-۲ ناهمگنی‌های سیستمی
۱۱	۳-۲-۲ ناهمگنی‌های آماری
۱۱	۴-۲-۲ حریم شخصی
۱۱	۳-۲ نگاه مقالات مرتبط به چالش‌های موجود
۱۱	۱-۳-۲ تبادل داده
۱۲	۲-۳-۲ ناهمگنی سیستمی و آماری
۱۲	۳-۳-۲ حریم شخصی

۴-۲	بیان ریاضی یادگیری فدرال	۱۳
۵-۲	رویکردهای کلی و پایه‌ای در حل چالش‌ها	۱۴
۱-۵-۲	به‌روزرسانی محلی و میانگین‌گیری در سرور	۱۴
۲-۵-۲	FedProx بهینه‌سازی	۱۶
۶-۲	کنترل‌کننده بهینه $H_2$ برای سیستم‌های زمان‌پیوسته	۱۷
۱-۶-۲	تعریف نرم $H_2$ برای سیستم‌های زمان‌پیوسته	۱۸
۲۰	مراجع	

## چکیده

در این چکیده ...

**کلمات کلیدی:** یادگیری فدرال، یادگیری عمیق، ... .

# فصل اول

## مقدمه

### ۱-۱ شناخت موضوع

رشد چشمگیر فناوری به همراه سهولت دسترسی به اینترنت در سال‌های اخیر باعث شده که بیشتر دستگاه‌های اطراف خود را متصل به اینترنت ببینیم. این دنیای جدید که به دنیای اینترنت اشیا<sup>۱</sup> معروف است شامل خانه‌های هوشمند<sup>۲</sup>، دستگاه‌های پوشیدنی<sup>۳</sup>، خودروهای خودران و در صدر آن‌ها تلفن‌های هوشمند<sup>۴</sup> است که همگی زندگی روزمره انسان را تغییر داده‌اند. استفاده از این سیستم‌ها همگی باعث تولید حجم قابل توجهی داده در طول روز می‌شوند که شرکت‌های بزرگ فناوری از این داده‌ها بهره برده و با استفاده از آن‌ها اقدام به انواع سرویس‌دهی به کاربران خود می‌نمایند.

با توجه به گسترش علم هوش مصنوعی و استفاده از روش‌های یادگیری ماشین، می‌توان از این حجم بسیار زیاد داده تولید شده به نحو مطلوبی استفاده نمود و الگوریتم‌های مورد نظرمان جهت رسیدن به اهداف مختلف را بر روی آن‌ها اجرا کرد. حال برای مدیریت و اجرای الگوریتم‌های یادگیری، روش‌های مختلفی وجود دارد که به توضیح هر یک از آن‌ها خواهیم پرداخت.

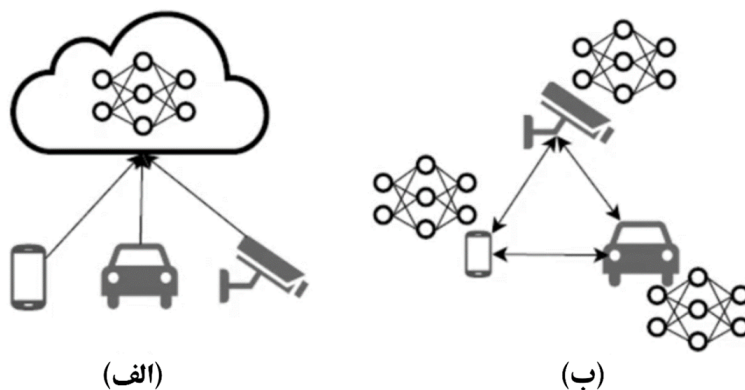
---

<sup>1</sup>Internet of Things

<sup>2</sup>Smart Homes

<sup>3</sup>Wearable Devices

<sup>4</sup>Smart Phones



شکل ۱-۱: (الف) یادگیری متمرکز، (ب) یادگیری غیرمتمرکز [۲].

### ۱-۱-۱ یادگیری متمرکز

روش یادگیری متمرکز<sup>۱</sup> که در اکثر سیستم‌های حال حاضر امروزی مورد استفاده قرار می‌گیرد به این نحو است که تمام گره‌ها<sup>۲</sup> اطلاعات موجود خود را به صورت کامل به سمت سرویس‌دهنده ابری<sup>۳</sup> ارسال می‌نمایند و سرویس‌دهنده ابری در حالی که تمام داده‌ها را در اختیار دارد اقدام به اجرای الگوریتم‌های مورد نظر می‌کند [۱]. در شکل ۱-۱ (الف) این روش به نمایش گذاشته شده است.

### ۲-۱-۱ یادگیری غیرمتمرکز

در روش یادگیری غیرمتمرکز<sup>۴</sup> هر گره به صورت مجزا اقدام به اجرای الگوریتم‌های مورد نظر می‌کند و در واقع پس از اجرای چند مرحله از کد، اطلاعات به‌روز شده را با گره‌های همسایه به اشتراک می‌گذارد، این کار به قدری ادامه پیدا می‌کند تا همگی به مقدار تعیین شده همگرا شوند [۲]. در شکل ۱-۱ (ب) این روش به نمایش گذاشته شده است.

### ۳-۱-۱ یادگیری توزیع شده

روش یادگیری توزیع شده<sup>۵</sup> به این نحو است که مدیریت کل سیستم و تمام داده‌ها در اختیار یک هسته مرکزی قرار دارد ولی به دلیل نیاز به توان پردازشی بالا، این هسته بار پردازشی را بین گره‌های موجود تقسیم می‌کند. در ابتدای راه یادگیری توزیع شده، فرض بر این بوده است که تمام گره‌ها توان پردازشی یکسانی داشته و داده‌ها به میزان مساوی بین گره‌ها پخش خواهند شد. در شکل ۲-۱ این روش به نمایش گذاشته شده است.

<sup>۱</sup>Centralized Learning

<sup>۲</sup>Nodes

<sup>۳</sup>Cloud Server

<sup>۴</sup>Decentralized Learning

<sup>۵</sup>Distributed Learning



شکل ۱-۲: یادگیری توزیع شده [۲].

## ۲-۱ یادگیری فدرال

سیستم‌های متمرکز تا پیش از این، اکثر نیازهای مربوطه را برطرف می‌نمودند ولی در دنیای امروزی و با توجه به زیاد شدن هر روزه دستگاه‌های متصل، موارد دیگری نیز مورد توجه واقع شده است. هزینه‌های مرتبط با ارسال حجم زیاد داده از یک جهت، و افزایش اضطراب در مورد انتقال اطلاعات حساس و شخصی از جهت دیگر، محققان را به سمت بهره‌گیری از الگوریتم‌های غیرمتمرکز و توزیع شده در زمینه یادگیری ماشین هدایت کرده است. یکی از زیر مجموعه‌های روش‌های یادگیری توزیع شده، شاخه جدید و بسیار پراستفاده یادگیری فدرال بوده که بسیار مورد توجه قرار گرفته است.

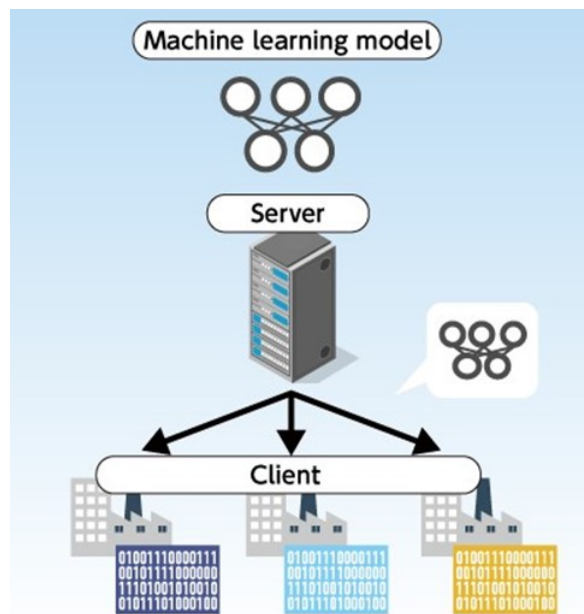
در روش یادگیری فدرال، برخلاف رویکردهای متمرکز یادگیری ماشین، تجزیه و تحلیل داده‌ها به دستگاه‌های لبه<sup>۱</sup> یا سرویس‌گیرنده‌ها<sup>۲</sup> منتقل می‌شود. این روش، به عنوان یک جایگزین مطلوب و نوآورانه برای مدل‌سازی داده‌ها در محیط‌هایی با تعداد زیادی سرویس‌گیرنده معرفی شده است. در این چارچوب، به جای انتقال داده‌های اصلی، پارامترهای مدل‌های محلی در هر مرحله از فرآیند آموزش به سمت سرور منتقل می‌شوند، که این امر توانایی بهبود امنیت و کاهش هزینه‌های ارتباطی را فراهم می‌کند. در شکل ۱-۳ این روش به نمایش گذاشته شده است. سرور در حقیقت نقش رهبری را ایفا می‌کند و با توجه به نوع داده‌ها، یک مدل شبکه عصبی<sup>۳</sup> ایجاد کرده و آن را به سمت کاربران ارسال می‌کند، حال کاربران با توجه به داده‌های خود شبکه را آموزش می‌دهند و بعد از چند بار تکرار، وزن‌های به‌روزرسانی شده را به سمت سرور بر می‌گردانند. همان‌طور که در شکل ۱-۳ مشاهده می‌شود، داده‌ها همگی در سمت کاربران قرار گرفته‌اند و به سمت سرور ارسال نمی‌شوند. عدم اجبار و محدودیت

<sup>۱</sup>Edge Devices

<sup>۲</sup>Clients

<sup>۳</sup>Neural Network





شکل ۱-۳: یادگیری فدرال [۳].

در ارسال اطلاعات گره‌ها در یادگیری فدرال، به حفظ حریم شخصی کاربران کمک می‌کند [۴].

### ۳-۱ تاریخچه یادگیری فدرال

در ابتدای فصل بهار سال ۲۰۱۷ محققین گوگل (Google) طی یک مطلب کوتاه در وبلاگ هوش مصنوعی برای اولین بار موضوع یادگیری فدرال را تحت مطلبی با عنوان ”یادگیری فدرال: یادگیری ماشین اشتراکی، بدون آموزش متمرکز داده‌ها” مطرح نمودند [۵]. در این مطلب به طور کوتاه Google Keyboard یا به اختصار Gboard معرفی شده و نحوه به کاری‌گیری یادگیری فدرال برای پیش‌بینی لغت بعدی را بیان می‌کند. یادگیری فدرال در این کاربرد نیاز به ارسال داده‌های کاربران به سمت سرور را حذف کرده است و به طور محلی مدل را به‌روزرسانی می‌کند. بنابراین، با بهره‌گیری از اطلاعات پنهان بسیار زیاد دستگاه‌ها در فرآیند مدل‌سازی، حریم شخصی سرویس‌گیرنده‌ها به نحوی بیشتر از پیش حفظ می‌شود. در شکل ۱-۴ نحوه استفاده از یادگیری فدرال در این برنامه به نمایش درآمده است.

### ۴-۱ کاربرد یادگیری فدرال

تکنولوژی نسل چهار صنعت<sup>۱</sup>، دامنه ارتباطات نرم‌افزاری و سخت‌افزاری را در انواع مختلف سیستم‌ها گسترش داده است. این هماهنگی فناوری نرم‌افزار و سخت‌افزار، تبدیل به یک پدیده مهم در مجموعه‌ای از محیط‌های

<sup>۱</sup> Industry 4.0



شکل ۱-۴: استفاده از یادگیری فدرال برای پیش‌بینی کلمه بعدی در Gboard [۶].

هوشمند و خودکار شده است. سنسورهای سابق که تنها مسئول اندازه‌گیری وضعیت‌ها بودند، جای خود را به دستگاه‌های هوشمند با قابلیت پردازش و برنامه‌ریزی داده‌ها سپرده‌اند. همچنین، گسترش ارتباطات در بستر اینترنت، امکان انتقال و تبادل داده‌ها بین انواع مختلف سیستم‌ها را ارائه کرده است. این پیشرفت‌ها منجر به کاهش نیاز به مرکزیت در تصمیم‌گیری و توسعه سیستم‌ها شده است و به وجود آورنده کنترل و نظارت پیشرفته و توزیع پردازش شده است. این ویژگی‌ها به همراه حجم بی‌سابقه داده، یادگیری فدرال را به یکی از بهترین روش‌های به‌کارگیری در توسعه سیستم‌های هوشمند تبدیل کرده است [۷]. در اینجا سه نمونه از کاربرد یادگیری فدرال را شرح خواهیم داد.

#### ۱-۴-۱ یادگیری فدرال در شهر هوشمند

در یک شهر هوشمند<sup>۱</sup>، اطلاعات جمع‌آوری شده از سنسورها، دستگاه‌ها و زیرساخت‌های مختلف، از جمله ترافیک، انرژی، پسماند و امنیت، به دلیل ارزش بالایی که دارند، به عنوان منبعی مهم برای بهبود عملکرد و کیفیت زندگی شهروندان محسوب می‌شوند. اما به همراه این ارزش‌ها، حفظ حریم خصوصی و امنیت اطلاعات شهروندان نیز امری بسیار حیاتی است. یادگیری فدرال به عنوان یک رویکرد نوین و مبتنی بر حفظ حریم خصوصی، در اینجا وارد عمل می‌شود.

این روش امکان پردازش داده‌های حساس مانند تصاویر، داده‌های محیطی و اطلاعات مکانی در محیط محلی و توزیع شده را فراهم می‌کند، به‌طوری‌که هر قسمت از شهر می‌تواند به صورت مستقل از سایر قسمت‌ها از این داده‌ها استفاده کند. این رویکرد امکان توسعه مدل‌های هوش مصنوعی و الگوریتم‌های بهبود عملکرد شهر هوشمند را با حفظ حریم خصوصی شهروندان فراهم می‌کند. به‌عنوان مثال، از طریق استفاده از یادگیری فدرال،

<sup>۱</sup> Smart City

می‌توان بهبود در مدیریت ترافیک، بهینه‌سازی مصرف انرژی، کاهش آلودگی هوا و افزایش امنیت شهری را به دست آورد، در حالی که اطلاعات شخصی شهروندان محافظت می‌شود و از نگرانی‌های حریم خصوصی جلوگیری خواهد شد.

#### ۱-۴-۲ یادگیری فدرال در بیمارستان

در یک بیمارستان، اطلاعات پزشکی بسیار حساس و مهم است که باید محفوظ و محرمانه نگهداری شود. اما در عین حال، استفاده از این داده‌ها برای بهبود خدمات بهداشتی و درمانی نیز بسیار ارزشمند است. در اینجا مفهوم یادگیری فدرال وارد عمل می‌شود. با استفاده از روش‌های یادگیری فدرال، بیمارستان می‌تواند از داده‌های پزشکی بیماران خود برای توسعه مدل‌هایی استفاده کند که پیش‌بینی میزان زمان بستری، بهبود در تشخیص بیماری‌ها و حتی افزایش بهره‌وری پزشکان را ایجاد می‌کنند، بدون اینکه این داده‌ها به‌طور مستقیم در اختیار یک مرکز جمع‌آوری اطلاعات واقع شوند.

به عنوان مثال، با استفاده از یادگیری فدرال، مدل‌های هوش مصنوعی می‌توانند روی داده‌های محلی بیماران بیمارستان‌ها آموزش داده شوند تا بیماری‌های مختلف را شناسایی و تشخیص دهند، و اطلاعات مربوط به درمان‌های مؤثرتر را ارائه دهند، در حالی که اطلاعات حساس بیماران محافظت می‌شود. این روش به بیمارستان‌ها امکان می‌دهد که از داده‌های بیماران خود برای بهبود خدمات بهداشتی و درمانی استفاده کنند، در حالی که رعایت مقررات مربوط به حفظ حریم خصوصی و امنیت داده‌ها را به انجام رسانده‌اند. در شکل ۱-۵ یک نمونه استفاده از یادگیری فدرال در سازمان‌ها به نمایش درآمده است.

#### ۱-۴-۳ یادگیری فدرال در فروشگاه برنامه‌های کاربردی موبایل

یک فروشگاه برنامه‌های کاربردی<sup>۱</sup> موبایل را متصور شوید که به کاربران خود امکان می‌دهد برنامه‌های مختلف را دانلود و نصب کنند. این شرکت می‌خواهد با استفاده از داده‌های کاربران خود، الگوریتمی توسعه دهد که به طور دقیق‌تر بتواند پیشنهادات مربوط به برنامه‌هایی که کاربران ممکن است تمایل داشته باشند را ارائه کند. اگر این شرکت از روش‌های متمرکز استفاده کند، باید داده‌های حساس و شخصی کاربران را جمع‌آوری کند و برای آن‌ها تحلیل کند. این ممکن است باعث نگرانی‌های حریم خصوصی کاربران شود و از آن‌ها جلوگیری کند.

در حالی که با استفاده از یادگیری فدرال، این شرکت می‌تواند الگوریتم خود را بر روی داده‌های محلی هر تلفن هوشمند کاربر اجرا کند. به این ترتیب، هیچ داده‌ی حساسی به مرکز جمع‌آوری داده‌ها ارسال نمی‌شود و حریم خصوصی کاربران محفوظ می‌ماند. به عنوان مثال، اگر یک کاربر فقط به برنامه‌های موزیک علاقه‌مند

<sup>۱</sup> App Store



شکل ۱-۵: یادگیری فدرال در یک بیمارستان [۸].

باشد، الگوریتم محلی در تلفن هوشمند او می‌تواند این الگو را تشخیص دهد و پیشنهادات مربوط به برنامه‌های موزیک را به او ارائه دهد، بدون این‌که داده‌های شخصی و حساس او به سرور شرکت ارسال شود. این روش به شرکت امکان می‌دهد از داده‌های کاربران خود برای بهبود خدمات خود استفاده کند، در حالی که حریم خصوصی آن‌ها را محافظت می‌کند.

## ۵-۱ دید کلی از روند موضوع و بیان هدف پژوهش

تکمیل این بخش پس از رسیدن به ساختار کلی پایان‌نامه (چون ممکنه در ادامه تغییر کنه)

چند جمله کلیدی:

به دلیل پراکندگی همگرایی به کندی صورت می‌گیرد

روش جابجایی وزن‌ها بین کاربران نهایی در طول فرایند

چرا جابجایی تصادفی، جابجایی هوشمند بر اساس میزان شباهت

## ۶-۱ مروری بر روند ارائه مطالب پایان‌نامه

تست

## فصل دوم

### مفاهیم پایه در یادگیری فدرال

#### ۱-۲ مقدمه

در جستجوی راه‌حلی برای یادگیری فدرال، لازم است به یک واقعیت مهم توجه کنیم که توزیع فرآیند آموزش بین افراد یا دستگاه‌های مختلف ممکن است به تداخل‌ها و مشکلاتی منجر شود. اگر این چالش‌ها را پیش از شروع فرآیند مدل‌سازی به‌خوبی در نظر نگیریم و راه‌حل‌های مشخصی برای آنها ارائه ندهیم، مدلی که در نهایت تولید می‌شود قطعاً با مشکلاتی از جمله دقت و کارایی مواجه خواهد شد. این مسئله، یکی از بزرگترین معضلاتی است که در مسیر یادگیری فدرال با آن روبرو می‌شویم و برای حل آن، نیازمند توجه دقیق و استفاده از روش‌های مختلف و نوآورانه هستیم.

در این فصل ابتدا چالش‌های موجود در یادگیری فدرال را رصد خواهیم کرد و سپس نگاه مقالات را در هر یک از آن‌ها به صورت کلی بررسی می‌کنیم. در ادامه، بیان ریاضی یادگیری فدرال را توضیح خواهیم داد و در نهایت به رویکردهای کلی و پایه‌ای در حل چالش‌ها اشاره خواهیم داشت.

## ۲-۲ چالش‌های موجود در یادگیری فدرال

با وجود مزیت‌های بسیار زیاد نسبت به روش‌های سنتی یادگیری ماشین، یادگیری فدرال به دلیل ساختار شبکه یادگیری با چالش‌های گوناگونی روبرو است. چالش‌های اصلی یادگیری فدرال عبارتند از:

### ۱-۲-۲ تبادل داده بین سرور و کاربران

تبادل داده بین سرور و کاربران به دلیل مشکلات پهنای باند و ارتباطات شبکه‌ای اصولاً کار پر هزینه‌ای می‌باشد. یکی از دلایل اصلی پرهزینه بودن این ارتباطات، حجم بالای داده‌هایی است که باید بین دستگاه‌های کاربری و سرور منتقل شوند. معمولاً مشکلات ارتباطی به انتقال‌های بسیار زیاد به‌روزرسانی‌های گرادیان بین گره‌های محاسباتی نسبت داده می‌شوند. با افزایش تعداد پارامترها در مدل‌های پیشرفته، اندازه گرادیان‌ها نیز به طور متناسب بزرگ می‌شود [۹].

با این حال، تعداد زیادی از دستگاه‌های کاربر نهایی وجود دارند که در فرآیند آموزش مدل‌ها شرکت می‌کنند، که این موضوع می‌تواند هزینه‌های ارتباطات را به شدت افزایش دهد. علاوه بر این، در بسیاری از مواقع، همه دستگاه‌ها در هر چرخه از فرآیند آموزش شرکت نمی‌کنند، که این نیز به افزایش هزینه‌ها و پیچیدگی‌های مرتبط با انتقال داده‌ها منجر می‌شود.

### ۲-۲-۲ ناهمگنی‌های سیستمی

در دنیای یادگیری فدرال، دستگاه‌ها از نظر حافظه، توان محاسباتی و ارتباطات بسیار با یکدیگر متفاوت هستند. این تفاوت‌ها ممکن است از اختلافاتی مانند تفاوت در پردازنده، نوع حافظه، نوع اتصال شبکه و نیاز به انرژی ناشی شود. محدودیت‌های موجود در شبکه و سیستمی می‌توانند باعث ایجاد وضعیت‌هایی شوند که برخی از دستگاه‌ها در یک زمان معین در دسترس نباشند. برای مثال، اگر تعداد زیادی دستگاه همزمان درخواست ارسال داشته باشند، ممکن است برخی از آن‌ها به دلیل پهنای باند محدود یا محدودیت‌های سخت‌افزاری، قادر به ارسال درخواست نشوند. همچنین، ممکن است یک دستگاه فعال، به دلیل مشکلاتی مانند اختلالات در شبکه یا مصرف اضافی انرژی، از فرآیند یادگیری خارج شود.

این ویژگی‌های سیستمی، جزء اصلی چالش‌های یادگیری فدرال هستند و موجب افزایش تاخیر و اشکالات در سیستم می‌شوند. بنابراین، به منظور حل این مشکلات، روش‌های یادگیری فدرال باید قادر باشند تعداد دقیقی از دستگاه‌هایی که در فرآیند شرکت می‌کنند را پیش‌بینی کنند، همچنین باید در برابر دستگاه‌هایی که در حین عملیات با مشکل روبه‌رو شده‌اند مقاومت مناسبی داشته باشند [۶].

## ۳-۲-۲ ناهمگنی‌های آماری

طریقه تولید و جمع‌آوری داده‌ها بین دستگاه‌ها به شکل گوناگونی انجام می‌شود. این مجموعه داده‌ها اغلب مستقل از یکدیگر نیستند و ارتباطات و اتصالات میان آن‌ها وجود دارد. این الگوی تولید داده‌ها، با فرض استقلال و توزیع یکنواخت داده<sup>۱</sup> (IID) در مسائل بهینه‌سازی متضاد است، که باعث ایجاد پیچیدگی در مدل‌سازی، تجزیه و تحلیل نظری و ارزیابی عملکرد راه‌حل‌ها می‌شود. در نتیجه، هرچند هدف نهایی یادگیری یک مدل سراسری است، اما روش‌های جایگزین مانند آموزش همزمان مدل‌های محلی جداگانه از طریق یادگیری چندوظیفه‌ای<sup>۲</sup> و فرایادگیری<sup>۳</sup>، به عنوان گزینه‌های جایگزین مطرح شده‌اند [۶].

## ۴-۲-۲ حریم شخصی

یکی از چالش‌های اساسی در یادگیری فدرال، حفظ حریم شخصی است که در این روش، داده‌های حساس و شخصی در اختیار بخش‌های مختلفی از شبکه قرار می‌گیرند. در این روش، دستگاه‌های محلی اطلاعاتی از کاربران جمع‌آوری و به سرور ارسال می‌کنند تا مدل‌های یادگیری مشترک را به‌روزرسانی کنند. این ارتباطات می‌توانند حاوی اطلاعات حساسی باشند که می‌توانند به راحتی به شناسایی فرد یا فرآیندهای حیاتی او منجر شوند.

یکی از مشکلات اساسی در اینجا این است که حتی با استفاده از روش‌های رمزنگاری و حفظ امنیت، اطلاعات معینی همچنان ممکن است به سرور ارسال شود که احتمالاً می‌تواند حریم شخصی را نقض کند. به‌طور خاص، اگر داده‌های حساس بدون رمزنگاری به سرور ارسال شوند یا اگر حتی اطلاعاتی که قابلیت شناسایی فرد را دارند به صورت رمزگذاری نشده ارسال شوند، حریم شخصی کاربران مورد تهدید قرار می‌گیرد.

## ۳-۲ نگاه مقالات مرتبط به چالش‌های موجود

### ۱-۳-۲ تبادل داده

با استفاده از فشرده‌سازی داده‌ها، می‌توان هزینه‌های ارتباطی را به طور قابل توجهی کاهش داد. دو روش جهت مدیریت هزینه‌های بالای ارتباطات در فرایند یادگیری فدرال مورد بررسی قرار گرفته است. این روش‌ها به فشرده‌سازی داده‌هایی که از دستگاه‌های کاربری به سرور مرکزی ارسال می‌شوند متمرکز شده‌اند. این فشرده‌سازی اطلاعات ارسالی به گونه‌ای است که حجم داده‌های ارسالی کم شده و در نتیجه، هزینه‌های مربوط به ارتباطات نیز کاهش یابد [۱۰].

<sup>1</sup>Independent and Identically Distributed

<sup>2</sup>Multi-Tasking

<sup>3</sup>Meta Learning

در روشی به نام PCFL<sup>۱</sup> که یک رویکرد حفظ حریم خصوصی و البته بسیار کارآمد از نظر ارتباطی می‌باشد، شامل سه جزء کلیدی است که به ترتیب از فشرده‌سازی دوطرفه، فشرده‌سازی مکانی گرادیان‌ها و یک پروتکل حفظ حریم خصوصی که خود از تقسیم راز<sup>۲</sup> و رمزنگاری همگام<sup>۳</sup> برای محافظت از حریم خصوصی داده‌ها استفاده می‌کند، بهره گرفته است [۱۱].

## ۲-۳-۲ ناهمگنی سیستمی و آماری

برای مقابله با ناهمگنی سیستمی و آماری، روش‌هایی مانند تعادل در به‌روزرسانی مدل مطرح شده است. در این روش، وزن‌دهی به نمونه‌ها بر اساس میزان نیاز به آموزش در هر دستگاه صورت می‌گیرد. این کار باعث می‌شود که دستگاه‌های با حجم داده کمتر، وزن بیشتری در به‌روزرسانی مدل داشته باشند [۱۲]. در رویکرد دیگری به نام یادگیری فعال، دستگاه‌هایی که داده‌های خود را به سرور ارسال می‌کنند، فعالیت خود را به نحوی تنظیم می‌کنند که مدل از داده‌های مهم‌تر و کمتر دیده شده بیشتر یاد می‌گیرد. این روش می‌تواند به تعادل در آموزش مدل کمک کند و از ناهمگنی سیستمی جلوگیری کند [۱۰].

یک روش دیگر برای حل مشکل ناهمگنی سیستمی و آماری در یادگیری فدرال استفاده از رویکرد ترکیبی یا ترکیب روش‌های یادگیری محلی است. در این رویکرد، به جای استفاده از یک الگوریتم یادگیری مشترک برای تمام دستگاه‌ها، از چندین الگوریتم یادگیری محلی با تنوع مدل‌ها و تنظیمات مختلف استفاده می‌شود. سپس، اطلاعات مدل‌های محلی روی سرور یا گره مرکزی جمع‌آوری می‌شود و با استفاده از ترکیب این اطلاعات، یک مدل یادگیری مشترک به‌روزرسانی خواهد شد [۱۲].

## ۲-۳-۳ حریم شخصی

در روش حفظ حریم خصوصی تفاضلی<sup>۴</sup> با افزودن نویز به نتایج محاسبات یا به داده‌های ورودی، اطمینان حاصل می‌کند که حضور یا عدم حضور یک نمونه داده خاص در مجموعه داده‌ها، تأثیر قابل توجهی بر خروجی محاسبات نداشته باشد. این روش به ویژه برای حفظ حریم خصوصی در یادگیری فدرال مفید است زیرا از افشای اطلاعات حساس از طریق پارامترهای مدل جلوگیری می‌کند [۱۳].

رویکرد رمزنگاری همگام امکان محاسبه روی داده‌های رمزنگاری شده را بدون نیاز به رمزگشایی آن‌ها فراهم می‌کند. این تکنیک به ویژه در یادگیری فدرال برای حفظ حریم خصوصی داده‌ها در حین انجام محاسبات مفید است زیرا نیاز به تغییر ماهیت داده نبوده و چون جابجایی در یادگیری فدرال بسیار زیاد رخ می‌دهد، این روش

<sup>1</sup>Privacy Communication efficient Federated Learning

<sup>2</sup>Secret Sharing

<sup>3</sup>Homomorphic Encryption

<sup>4</sup>Differential Privacy



بسیار کارا خواهد بود [۱۴].

## ۴-۲ بیان ریاضی یادگیری فدرال

برای ورود به مباحث ریاضی پایه در یادگیری فدرال، ابتدا باید به تعریف دقیق مسئله بهینه‌سازی مرکزی بپردازیم که در این حوزه مطرح می‌شود. در یادگیری فدرال، هدف اصلی یافتن مجموعه‌ای از پارامترهای مدل است که عملکرد کلی مدل را بر روی داده‌های توزیع‌شده بین تعداد زیادی دستگاه بهینه کند. هر دستگاه دارای داده‌های محلی است و یک تابع هزینه محلی بر اساس این داده‌ها برای آن دستگاه تعریف می‌شود. مسئله بهینه‌سازی کلی در یادگیری فدرال به دنبال کمینه کردن مجموع وزنی این توابع هزینه محلی است تا یک مدل جامع و یکپارچه حاصل شود.

ما یک طرح به‌روزرسانی همزمان را فرض می‌کنیم که به صورت دوره‌های ارتباطی پیش می‌رود. یک مجموعه ثابت از  $K$  مشتری وجود دارد که هر کدام یک مجموعه داده محلی ثابت دارند. در ابتدای هر دوره، یک کسر تصادفی  $C$  از مشتری‌ها انتخاب می‌شوند و سرور وضعیت فعلی پارامترهای مدل جهانی را به هر یک از این مشتری‌ها ارسال می‌کند. هر مشتری انتخاب شده سپس بر اساس وضعیت جهانی و مجموعه داده محلی خود محاسبات محلی را انجام می‌دهد و یک به‌روزرسانی به سرور ارسال می‌کند. سپس سرور این به‌روزرسانی‌ها را به وضعیت جهانی خود اعمال می‌کند و این فرآیند تکرار می‌شود [۱۵].

در حالی که ما بر اهداف شبکه عصبی غیرمحدب<sup>۱</sup> تمرکز داریم، الگوریتمی که بررسی می‌کنیم قابل اعمال به هر هدف مجموع-متناهی<sup>۲</sup> به صورت زیر است.

$$\min_{w \in \mathbb{R}^d} f(w) \quad \text{where} \quad f(w) \stackrel{\text{def}}{=} \frac{1}{n} \sum_{i=1}^n f_i(w) \quad (1-2)$$

برای یک مسئله یادگیری ماشین، معمولاً  $f_i(w) = \ell(x_i, y_i; w)$  در نظر گرفته می‌شود، به این معنی که این تابع نشان‌دهنده خطای پیش‌بینی بر روی نمونه  $(x_i, y_i)$  با استفاده از پارامترهای مدل  $w$  است. فرض می‌کنیم که داده‌ها بین  $K$  مشتری تقسیم شده‌اند، که در آن  $\mathcal{P}_k$  مجموعه‌ای از نقاط داده مربوط به مشتری  $k$  است و  $n_k = |\mathcal{P}_k|$  تعداد این نقاط داده را نشان می‌دهد. بنابراین، می‌توانیم فرمول ۱-۲ را به صورت زیر بازنویسی کنیم:

$$f(w) = \sum_{k=1}^K \frac{n_k}{n} F_k(w) \quad \text{where} \quad F_k(w) = \frac{1}{n_k} \sum_{i \in \mathcal{P}_k} f_i(w) \quad (2-2)$$

اگر مجموعه  $\mathcal{P}_k$  با توزیع یکنواخت تصادفی از مثال‌های آموزشی بین مشتری‌ها تشکیل شده باشد، در آن

<sup>1</sup>Non-Convex

<sup>2</sup>Finite-Sum

صورت  $\mathbb{E}_{\mathcal{P}_k}[F_k(w)] = f(w)$  خواهد بود، که در اینجا امید ریاضی بر روی مجموعه مثال‌های اختصاص داده شده به یک مشتری ثابت گرفته می‌شود. این همان فرض IID (استقلال و توزیع یکسان) است که عموماً توسط الگوریتم‌های بهینه‌سازی توزیع شده استفاده می‌شود، در این جا ما حالتی را که این فرض برقرار نیست (یعنی  $F_k$  می‌تواند تقریباً به هر میزانی از  $f$  فاصله داشته باشد) به عنوان حالت Non-IID (غیرمستقل و غیریکنواخت) می‌شناسیم [۱۵].

## ۲-۵ رویکردهای کلی و پایه‌ای در حل چالش‌ها

روش‌های بهینه‌سازی توزیع شده معمولاً برای حل مسائل بهینه‌سازی در سیستم‌هایی با شبکه‌های محاسباتی بزرگ و توزیع شده استفاده می‌شوند. این روش‌ها بر مبنای تقسیم مسئله بهینه‌سازی به زیرمسائل کوچک‌تر و حل آن‌ها در گره‌های مختلف شبکه استوارند. در این روش‌ها، اغلب فرض می‌شود که داده‌ها به صورت همگن و یکپارچه در سراسر شبکه توزیع شده‌اند و گره‌ها می‌توانند به راحتی با یکدیگر ارتباط برقرار کنند.

فرضیات مطرح شده در یادگیری فدرال به ندرت برقرار است، زیرا در یادگیری فدرال داده‌ها به صورت محلی و ناهمگن در دستگاه‌های مختلف قرار دارند و ارتباطات بین دستگاه‌ها ممکن است محدود و نامنظم باشد. بنابراین روش‌ها و رویکردهای لازم جهت حل این چالش‌ها متفاوت از مسائل بهینه‌سازی توزیع شده هستند. حال سعی می‌کنیم دو رویکرد پایه‌ای برای مسائل یادگیری فدرال را مطرح نماییم.

### ۲-۵-۱ به‌روزرسانی محلی و میانگین‌گیری در سرور

یکی از روش‌های اصلی و پرکاربرد در یادگیری فدرال روش میانگین‌گیری فدرال<sup>۱</sup> (FedAvg) است که توسط محققان گوگل در سال ۲۰۱۷ معرفی شد. این الگوریتم به منظور بهینه‌سازی مدل‌های یادگیری ماشین در یک محیط توزیع شده طراحی شده است، جایی که داده‌ها به صورت محلی در دستگاه‌های کاربران باقی می‌مانند و تنها به‌روزرسانی‌های مدل به اشتراک گذاشته می‌شوند. رویکرد اصلی FedAvg بر مبنای ترکیب به‌روزرسانی‌های محلی از دستگاه‌های مختلف به یک مدل جهانی استوار است.

یکی از مزایای اصلی FedAvg این است که به طور موثری با چالش ناهمگنی داده‌ها مقابله می‌کند. در یادگیری فدرال، داده‌های موجود در دستگاه‌های مختلف ممکن است توزیع‌های متفاوتی داشته باشند. این ناهمگنی می‌تواند به دلیل تفاوت در رفتار کاربران یا حتی محیط‌های مختلف جمع‌آوری داده باشد. میانگین‌گیری وزنی در FedAvg به مدل کمک می‌کند تا به‌روزرسانی‌های مختلف را به گونه‌ای ترکیب کند که این ناهمگنی‌ها را در نظر بگیرد. به عبارت دیگر، اگر یک دستگاه داده‌های بیشتری داشته باشد، تأثیر بیشتری بر مدل نهایی خواهد

<sup>۱</sup> Federated Averaging

داشت. این رویکرد باعث می‌شود که مدل فدرال به تعادل بهتری در یادگیری از داده‌های ناهمگن برسد و کارایی بالاتری داشته باشد. این ویژگی به خصوص در کاربردهایی که کاربران متنوع و داده‌های متنوعی دارند، بسیار مفید است و می‌تواند به بهبود عملکرد مدل در شرایط واقعی کمک کند.

علاوه بر این، FedAvg به کاهش نیاز به ارتباطات مکرر بین دستگاه‌ها و سرور مرکزی کمک می‌کند. در بسیاری از روش‌های بهینه‌سازی توزیع‌شده، نیاز است که دستگاه‌ها به طور مکرر با سرور مرکزی ارتباط برقرار کنند تا به‌روزرسانی‌های خود را ارسال کنند. اما در FedAvg دستگاه‌ها می‌توانند چندین مرحله از بهینه‌سازی را به صورت محلی انجام دهند و سپس تنها به‌روزرسانی نهایی را ارسال کنند. این کاهش در نیاز به ارتباطات نه تنها باعث کاهش پهنای باند مورد نیاز می‌شود، بلکه به حفظ حریم خصوصی کاربران نیز کمک می‌کند، زیرا داده‌ها هرگز از دستگاه‌های محلی خارج نمی‌شوند. بررسی‌ها نشان داده‌اند که متناسب با اندازه داده‌ها پس از رسیدن به تعداد معینی از گره‌ها، اضافه کردن گره‌های بیشتر تأثیری در کاهش هزینه‌های ارتباطی نخواهد داشت. در چنین شرایطی، تمرکز بر افزایش توان محاسباتی محلی یا تعداد مراحل آموزش محلی می‌تواند موجب تسریع فرایند آموزش شود [۱۵].

موفقیت‌های اخیر در کاربردهای یادگیری عمیق تقریباً به‌طور انحصاری به استفاده از انواع الگوریتم نزول گرادیان تصادفی<sup>۱</sup> (SGD) برای بهینه‌سازی متکی بوده‌اند. در واقع، بسیاری از پیشرفت‌ها به تنظیم مدل و بهینه‌سازی تابع خطا با روش‌های ساده گرادیان مربوط می‌شود. بنابراین، طبیعی است که ما الگوریتم‌های بهینه‌سازی فدرال را با شروع از SGD بسازیم.

الگوریتم SGD می‌تواند به سادگی در بهینه‌سازی فدرال استفاده شود، به این صورت که در هر دور ارتباط، گرادیان‌ها بر اساس داده‌های یک مشتری تصادفی انتخاب شده، محاسبه شوند. این رویکرد از نظر محاسباتی کارآمد است، اما نیازمند تعداد بسیار زیادی از دوره‌های آموزش برای تولید مدل‌های خوب است. برای مثال حتی با استفاده از رویکرد پیشرفته‌ای مانند نرمال‌سازی دسته‌ای<sup>۲</sup>، برای آموزش دیتاست معروف MNIST (دیتاستی جهت دسته‌بندی اعداد دستنویس بین صفر تا نه) با دسته‌های کوچکی به اندازه ۶۰ به ۵۰۰۰۰ دور آموزش جهت رسیدن به مدل مطلوب نیاز می‌باشد [۱۶].

در تنظیمات فدرال، مشارکت تعداد زیادتری از مشتریان هزینه‌ای آنچنان بیشتری در زمان واقعی ندارد زیرا همه کاربران می‌توانند به صورت همزمان اقدام به آموزش مدل محلی کنند، بنابراین برای خط مبنای خود از SGD همزمان با دسته‌های بزرگ استفاده می‌کنیم. برای اعمال این رویکرد در تنظیمات فدرال، ما در هر دور یک کسر  $C$  از مشتریان را انتخاب می‌کنیم و گرادیان خطا روی تمام داده‌های نگهداری شده توسط این مشتریان را

<sup>۱</sup>Stochastic Gradient Descent

<sup>۲</sup>Batch Normalization

محاسبه می‌کنیم. بنابراین  $C$  اندازه دسته کلی را کنترل می‌کند، به‌طوری که  $C = 1$  معادل با نزول گرادیان یک دسته کامل است. حال این الگوریتم خط مبنا را FederatedSGD یا FedSGD می‌نامیم.

یک پیاده‌سازی معمول از FedSGD با  $C = 1$  و نرخ یادگیری ثابت  $\eta$  به این صورت است که هر گره  $k$ ، گرادیان  $g_k = \nabla F_k(w_t)$  که میانگین گرادیان روی داده‌های محلی در مدل فعلی  $w_t$  است را محاسبه می‌کند و سرور مرکزی این گرادیان‌ها را جمع‌آوری کرده و به‌روزرسانی  $w_{t+1} \leftarrow w_t - \eta \sum_{k=1}^K \frac{n_k}{n} g_k$  را انجام می‌دهد، در حالی که  $\sum_{k=1}^K \frac{n_k}{n} g_k = \nabla f(w_t)$  خواهد بود. یک به‌روزرسانی معادل به این صورت است که برای هر گره عبارت  $\forall k, w_{t+1}^k \leftarrow w_t - \eta g_k$  محاسبه و سپس  $w_{t+1} \leftarrow \sum_{k=1}^K \frac{n_k}{n} w_{t+1}^k$  انجام شود.

در نتیجه، هر گره به صورت محلی یک گام گرادیان نزولی را روی مدل فعلی با استفاده از داده‌های محلی خود طی می‌کند و سپس سرور میانگین وزنی مدل‌های حاصل را محاسبه می‌کند. وقتی که الگوریتم به این صورت نوشته شود، می‌توانیم با تکرار به‌روزرسانی محلی  $w^k \leftarrow w^k - \eta \nabla F_k(w^k)$ ، چندین بار قبل از مرحله میانگین‌گیری، محاسبات بیشتری به هر گره اضافه کنیم. در نهایت این رویکرد جدید را FederatedAveraging (FedAvg) می‌نامیم.

میزان محاسبات توسط سه پارامتر کلیدی کنترل می‌شود:  $C$ ، کسر گره‌هایی که در هر مرحله محاسبات انجام می‌دهند؛  $E$ ، تعداد مراحل آموزشی که هر گره در هر دور روی مجموعه داده محلی خود انجام می‌دهد؛ و  $B$ ، اندازه دسته محلی که برای به‌روزرسانی‌های هر گره استفاده می‌شود. در اینجا  $B = \infty$  را می‌نویسیم تا نشان دهیم که کل مجموعه داده محلی به عنوان یک دسته واحد در نظر گرفته می‌شود. بنابراین، به عنوان یک نمونه از این الگوریتم گسترده شده جدید، می‌توانیم  $B = \infty$  و  $E = 1$  را انتخاب کنیم که در این حالت دقیقاً با FedSGD برابر خواهد شد. همچنین برای یک گره با  $n_k$  نمونه محلی، تعداد به‌روزرسانی‌های محلی در هر دور با  $u_k = E \frac{n_k}{B}$  نمایش داده می‌شود [۱۵].

## ۲-۵-۲ بهینه‌سازی FedProx

تست

یک سیستم زمان پیوسته در شکل ۱-۲ نشان داده شده است. این سیستم را به فرم استاندارد نیز می توان نمایش داد. در این فرم که در شکل ۲-۲ نشان داده شده است، سیگنال  $w$  ورودی های خارجی سیستم نظیر ورودی مرجع<sup>۱</sup>، نویز و اختلال را شامل می شود.  $z$  سیگنالی است که قرار است کنترل شود و معمولاً خطای سیستم (اختلاف بین خروجی مطلوب و خروجی واقعی) است.  $y$  ورودی کنترل کننده و  $u$  نیز سیگنالی است که کنترل کننده آن را تولید می کند و به سیگنال کنترل معروف است. هم چنین در این فرم به دلیل این که سیستم  $G$  دو ورودی و دو خروجی دارد، می توان آن را به چهار بخش به صورت

$$G = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix}$$

تقسیم کرد. در این صورت روابط

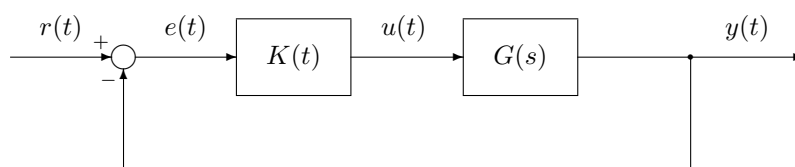
$$\begin{cases} z = G_{11}w + G_{12}u \\ y = G_{21}w + G_{22}u \end{cases}$$

بین ورودی ها و خروجی ها برقرار است.

کنترل کننده بهینه  $H_2$ ، یک کنترل کننده علی و مناسب<sup>۲</sup> است که سیستم را به طور داخلی پایدار کند و هم چنین به وسیله آن نرم  $H_2$  تابع تبدیل از  $z$  به  $w$  ( $T_{zw}$ ) مینیمم شود. به طور معادل می توان گفت کنترل کننده ای است که نرم دو پاسخ ضربه سیگنال  $z$  را مینیمم کند [۱۷]. در صورتی که سیستم متغیر با زمان باشد، تابع تبدیل مفهومی ندارد و از پاسخ ضربه باید استفاده کرد.

## ۲-۶ کنترل کننده بهینه $H_2$ برای سیستم های زمان پیوسته

در این قسمت روش طراحی کنترل کننده بهینه  $H_2$  برای یک سیستم زمان پیوسته بیان می شود. بدین منظور ابتدا به تعریف نرم  $H_2$  برای سیستم های زمان پیوسته می پردازیم و پس از آن روش طراحی کنترل کننده را بیان می کنیم.



شکل ۱-۲: یک سیستم زمان پیوسته

<sup>۱</sup>Reference Input

<sup>۲</sup>Proper

## ۲-۶-۱ تعریف نرم $\mathcal{H}_2$ برای سیستم‌های زمان پیوسته

برای یک سیستم خطی، تغییرناپذیر با زمان و پایدار  $G$  که زمان پیوسته و تک ورودی-تک خروجی است، نرم  $\mathcal{H}_2$  به صورت

$$\|\hat{g}(s)\|_2 = \sqrt{\frac{1}{2\pi} \int_{-\infty}^{\infty} |\hat{g}(j\omega)|^2 d\omega} \quad (۳-۲)$$

تعریف می‌شود. در این رابطه  $\hat{g}(j\omega)$  پاسخ فرکانسی سیستم است. بر اساس خاصیت پارسوال<sup>۱</sup>، نرم  $\mathcal{H}_2$  یک سیستم پایدار، با نرم دو پاسخ ضربه آن برابر است. اگر  $\hat{g}(s)$  تابع تبدیل یک سیستم زمان پیوسته پایدار و  $G\delta(t)$  پاسخ ضربه آن باشد، آنگاه رابطه

$$\|\hat{g}(s)\|_2 = \|G\delta(t)\|_2 = \sqrt{\int_0^{\infty} |G\delta(t)|^2 dt} \quad (۴-۲)$$

برقرار است.

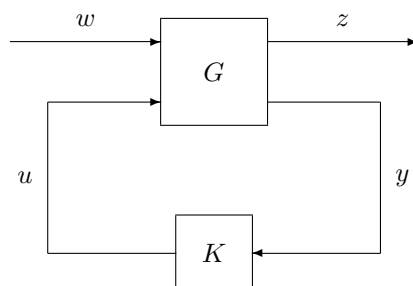
برای سیستم‌های چند ورودی-چند خروجی روابط کمی پیچیده‌تر می‌شوند. فرض کنید سیستم  $G$ ،  $m$  ورودی و  $p$  خروجی داشته باشد. در این صورت ماتریس انتقال آن،  $p$  سطر و  $m$  ستون دارد. نرم  $\mathcal{H}_2$  برای چنین سیستمی به صورت

$$\|\hat{g}(s)\|_2 = \sqrt{\frac{1}{2\pi} \int_{-\infty}^{\infty} \text{trace} [\hat{g}^*(j\omega)\hat{g}(j\omega)] d\theta}$$

تعریف می‌شود. در این رابطه،  $\hat{g}(s)$  ماتریس انتقال سیستم است. هم‌چنین طبق خاصیت پارسوال اگر سیستم پایدار باشد رابطه

$$\|\hat{g}(s)\|_2 = \sum_{i=1}^m \|G\delta(t)e_i\|_2 \quad (۵-۲)$$

نیز برای آن برقرار است. در این رابطه  $e_i$  بردارهای پایه استاندارد در فضای  $\mathbb{R}^m$  هستند.  $\delta(t)e_i$  تابع ضربه‌ای است که به ورودی  $i$  اعمال شده و  $G\delta(t)e_i$  خروجی مربوط به آن است.



شکل ۲-۲: یک سیستم زمان پیوسته در فرم استاندارد

<sup>۱</sup>Parseval

در صورتی که سیستم پایدار باشد، می‌توان از فضای حالت سیستم نیز برای محاسبه نرم  $\mathcal{H}_2$  استفاده کرد. فرض کنید معادلات فضای حالت یک سیستم پایدار به صورت

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases}$$

باشد که  $x$  بردار حالت سیستم،  $u$  بردار ورودی و  $y$  بردار خروجی است.  $A$  نیز یک ماتریس هرویتز<sup>۱</sup> است. همچنین برای محدود بودن نرم  $\mathcal{H}_2$  سیستم زمان‌پیوسته<sup>۲</sup>،  $D$  باید صفر باشد. در این صورت برای محاسبه نرم  $\mathcal{H}_2$  سیستم، می‌توان از روش زیر استفاده کرد [۱۸]:

۱. حل معادله لیاپانوف زمان‌پیوسته<sup>۲</sup> ( $AL + LA^T + BB^T = 0$ ) و یافتن ماتریس نامعلوم  $L$ .

باید دقت شود که در صورت هرویتز بودن ماتریس  $A$ ، معادله لیاپانوف حل یکتا دارد.

۲. محاسبه نرم  $\mathcal{H}_2$  از طریق رابطه  $\|\hat{g}\|_2 = \sqrt{\text{trace}(CLC^T)}$ . [۱۹]

---

<sup>۱</sup>Hurwitz

<sup>۲</sup>Continuous-time Lyapunov Equation

## مراجع

- [1] Elbir, Ahmet M, Coleri, Sinem, Papazafeiropoulos, Anastasios K, Kourtessis, Pandelis, and Chatzinotas, Symeon. A family of hybrid federated and centralized learning architectures in machine learning. *IEEE Transactions on Cognitive Communications and Networking*, 2022.
- [2] Zhou, Zhi, Chen, Xu, Li, En, Zeng, Liekang, Luo, Ke, and Zhang, Junshan. Edge intelligence: Paving the last mile of artificial intelligence with edge computing. *Proceedings of the IEEE*, 107(8):1738–1762, 2019.
- [3] Tomas. Decentralized x: Aggregating heterogeneous and decentralized ais. <https://www.omron.com/global/en/technology/information/dcx>. [Accessed: 17 Apr 2024].
- [4] Smith, Virginia, Chiang, Chao-Kai, Sanjabi, Maziar, and Talwalkar, Ameet S. Federated multi-task learning. *Advances in neural information processing systems*, 30, 2017.
- [5] McMahan, Brendan, Ramage Daniel. Federated learning: Collaborative machine learning without centralized training data. <https://www.omron.com/global/en/technology/information/dcx>, 6 Apr 2017. [Accessed: 18 Apr 2024].
- [6] Li, Tian, Sahu, Anit Kumar, Talwalkar, Ameet, and Smith, Virginia. Federated learning: Challenges, methods, and future directions. *IEEE signal processing magazine*, 37(3):50–60, 2020.
- [7] Talaei, Mahtab. Algorithm development and performance analysis for adaptive differential privacy in federated learning, 21 Aug 2022.
- [8] Kim, Jiyeon, Yang, Inseok, and Lee, Dongik. Control allocation based compensation for faulty blade actuator of wind turbine. *IFAC Proceedings Volumes*, 45(20):355–360, 2012.
- [9] Wang, Hongyi, Sievert, Scott, Liu, Shengchao, Charles, Zachary, Papailiopoulos, Dimitris, and Wright, Stephen. Atomo: Communication-efficient learning via atomic sparsification. *Advances in neural information processing systems*, 31, 2018.
- [10] Konečný, Jakub, McMahan, H Brendan, Yu, Felix X, Richtárik, Peter, Suresh, Ananda Theertha, and Bacon, Dave. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.
- [11] Fang, Chen, Guo, Yuanbo, Hu, Yongjin, Ma, Bowen, Feng, Li, and Yin, Anqi. Privacy-preserving and communication-efficient federated learning in internet of things. *Computers & Security*, 103:102199, 2021.



- [12] Konečný, Jakub, McMahan, Brendan, and Ramage, Daniel. Federated optimization: Distributed optimization beyond the datacenter. *arXiv preprint arXiv:1511.03575*, 2015.
  - [13] Hasan, Jahid. Security and privacy issues of federated learning. *arXiv preprint arXiv:2307.12181*, 2023.
  - [14] Yin, Xuefei, Zhu, Yanming, and Hu, Jiankun. A comprehensive survey of privacy-preserving federated learning: A taxonomy, review, and future directions. *ACM Computing Surveys (CSUR)*, 54(6):1–36, 2021.
  - [15] McMahan, Brendan, Moore, Eider, Ramage, Daniel, Hampson, Seth, and y Arcas, Blaise Aguera. Communication-efficient learning of deep networks from decentralized data. in *Artificial intelligence and statistics*, pp. 1273–1282. PMLR, 2017.
  - [16] Ioffe, Sergey and Szegedy, Christian. Batch normalization: Accelerating deep network training by reducing internal covariate shift. in *International conference on machine learning*, pp. 448–456. pmlr, 2015.
  - [17] Hespanha, Joo P, Naghshtabrizi, Payam, and Xu, Yonggang. A survey of recent results in networked control systems. *Proceedings of the IEEE*, 95(1):138–162, 2007.
  - [18] Zhou, Kemin and Doyle, John Comstock. *Essentials of robust control*, vol. 104. Prentice hall Upper Saddle River, NJ, 1998.
- [۱۹] فیاضی. تحلیل محتوای کتاب فارسی خوانداری و نوشتاری دوره دبستان از منظر معنی‌شناسی قالبی.

جستارهای زبانی، ۱۱(۴):۲۱۱–۲۴۰، ۲۰۲۰.