

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



دانشگاه شاهرود

دانشکده علوم ریاضی

گروه ریاضی کاربردی

پایان نامه

برای دریافت درجه کارشناسی ارشد در رشته

ریاضی کاربردی، گرایش تحقیق در عملیات

عنوان

یک کاربرد از عملگر تصویر و سیستمهای
دینامیکی برای حل رده ای از مسائل بهینه
سازی غیرخطی

استاد راهنما

دکتر علیرضا ناظمی

دانشجو

حسن فرخی

۱۳۹۴

پروردگارا...!

نه می توانم مویشان را که در راه عزت من سفید شد، سیاه کنم و نه برای دست های
پینه بسته شان که شمره تلاش برای افتخار من است، مرهمی دارم. پس توفیقم ده که
هر لحظه سگرگزارشان باشم و ثانیه های عمرم را در عصبای دست بودنشان بگذارم.

تقدیم بہ

پدر و مادر عزیزم

سپاس گزارمی...

نخستین سپاس و ستایش از آن خداوندی است که بنده کوچکش را در دریای بی‌کران اندیشه، قطره‌ای ساخت تا وسعت آن را از دریچه اندیشه‌های ناب آموزگارانی بزرگ به تماشا نشیند. لذا اکنون که در سایه‌سار بنده نوازی‌هایش پایان‌نامه حاضر به انجام رسیده است، برخورد لازم می‌دانم تا مراتب سپاس را از بزرگوارانی به‌جا آورم که اگر دست یاری‌گرشان نبود، هرگز این پایان‌نامه به انجام نمی‌رسید. ابتدا از استاد گرانقدرم جناب آقای دکتر علی‌رضاناظمی که زحمت راهنمایی این پایان‌نامه را برعهده داشتند، کمال سپاس را دارم. سپاس آخر را به مهربان‌ترین همراهان زندگی‌ام به پدر و مادرم و تمامی دوستانی که در طول انجام این پروژه کمک‌رسان من بوده‌اند تقدیم می‌کنم.

حسن فرخی
۱۳۹۴

تعمدنامه

اینجانب حسن فرخی دانشجوی کارشناسی ارشد رشته ریاضی کاربردی دانشکده علوم ریاضی دانشگاه شاهرود، نویسنده پایان‌نامه با عنوان یک کاربرد از عملگر تصویر سیستم‌های دینامیکی برای حل رده‌ای از مسائل بهینه‌سازی غیرخطی، تحت راهنمایی دکتر علیرضا ناظمی متعهد می‌شوم:

- تحقیقات در این پایان‌نامه توسط اینجانب انجام شده است و از صحت و اصالت برخوردار است.
- در استفاده از نتایج پژوهش‌های دیگر پژوهش‌گران، به مرجع مورد استفاده استناد شده است.
- مطالب این پایان‌نامه، تاکنون توسط خود، یا فرد دیگری برای دریافت هیچ نوع مدرک یا امتیازی در هیچ‌جا ارایه نشده است.
- حقوق معنوی این اثر، به دانشگاه شاهرود متعلق دارد، و مقالات مستخرج با نام “ دانشگاه شاهرود “ یا “ Shahrood University “ به چاپ خواهد رسید.
- حقوق معنوی تمام افرادی که در به‌دست آوردن نتایج اصلی پایان‌نامه تاثیرگذار بوده‌اند، در مقالات مستخرج از پایان‌نامه رعایت می‌گردد.
- در تمام مراحل انجام این پایان‌نامه، در مواردی که از موجود زنده (یا بافت‌های آنها) استفاده شده است، ضوابط و اصول اخلاقی رعایت شده است.
- در تمام مراحل انجام این پایان‌نامه، در مواردی که به حوزه اطلاعات شخصی افراد دسترسی یافته (یا استفاده) شده است، اصل رازداری و اصول اخلاق انسانی رعایت شده است.

حسن فرخی

۱۳۹۴

مالکیت نتایج و حق نشر

- تمام حقوق معنوی این اثر و محصولات آن (مقالات مستخرج، کتاب، برنامه‌های رایانه‌ای، نرم‌افزارها و تجهیزات ساخته شده) متعلق به دانشگاه شاهرود می‌باشد. این مطلب باید به نحو مقتضی، در تولیدات علمی مربوطه ذکر شود.
- استفاده از اطلاعات و نتایج موجود در این پایان‌نامه بدون ذکر منبع مجاز نمی‌باشد.

چکیده

طیف گسترده ای از مسائل علوم و مهندسی می توانند به صورت مسائل بهینه سازی غیرخطی فرمول بندی شوند. یک رهیافت امیدوارکننده برای حل مسائل بهینه سازی غیرخطی با بعد بالا، به کارگیری شبکه های عصبی مصنوعی است. در این پایان نامه دو مدل شبکه عصبی برای حل مسائل بهینه سازی غیرخطی معرفی می کنیم. اولین مدل می تواند مسائل برنامه ریزی غیرخطی محدب با محدودیت های تساوی و نامساوی را حل نماید. مدل دوم بر پایه تابع تصویر برای حل مسائل برنامه ریزی غیرخطی محدب به دست آمده است. علاوه بر این، وجود و همگرایی و ویژگیهای پایداری مدل های شبکه عصبی ارائه شده را تحلیل می کنیم. کارایی شبکه های عصبی معرفی شده با استفاده از چندین مثال عددی نشان داده شده است.

کلمات کلیدی: بهینه سازی محدب، شبکه عصبی، شرایط کاروش-کان-تاگر، پایداری

فهرست مطالب

د	لیست تصاویر	
ر	لیست جداول	
۱	مقدمات و مفاهیم اساسی	۱
۱	مقدمه‌ای بر شبکه‌های عصبی	۱.۱
۱	شبکه‌های عصبی بیولوژیکی	۱.۱.۱
۲	شبکه‌های عصبی مصنوعی	۱.۱.۲
۶	تاریخچه تکامل شبکه عصبی	۱.۱.۳
۷	تاریخچه حل مسائل بهینه‌سازی با استفاده از شبکه‌های عصبی	۱.۱.۴
۹	مقدمه‌ای بر بهینه‌سازی غیرخطی	۱.۲
۱۰	سیستم‌های دینامیکی	۱.۳
۱۵	یک مدل شبکه عصبی لاگرانژی کارا برای حل رده‌ای از مسائل بهینه‌سازی محدب	۲
۱۵	مقدمه	۲.۱
۱۶	مدل شبکه عصبی	۲.۲
۱۷	تحلیل همگرایی و پایداری	۲.۳
۱۹	مثال‌های عددی	۲.۴
۲۵	استفاده از یک تابع شایسته برای حل رده‌ای از مسائل بهینه‌سازی غیرخطی محدب	۳
۲۵	بیان مسئله	۳.۱
۲۶	فرمول بندی و مدل گرادیان	۳.۲
۲۸	تجزیه و تحلیل پایداری	۳.۳
۳۰	مثال‌های عددی	۳.۴
۳۵	استفاده از تابع تصویر برای حل مسائل بهینه‌سازی غیرخطی محدب در حالت کلی	۴

۳۵	مقدمه	۴.۱
۳۶	مدل شبکه عصبی	۴.۲
۳۷	تجزیه و تحلیل پایداری	۴.۳
۴۱	مثال‌های عددی	۴.۴
۴۸	نتیجه‌گیری	۴.۵
۵۰		آ کد مثال‌های فصل ۴	
۵۰	مثال (۱.۴.۴)	۱.آ
۵۲		مراجع	
۵۷		واژه‌نامه فارسی به انگلیسی	
۶۱		واژه‌نامه انگلیسی به فارسی	

لیست تصاویر

۲	۱.۱	ساختار یک نرون.
۳	۲.۱	مدل ریاضی یک نرون.
۴	۳.۱	ساختار یک نرون با لایه ورودی، لایه پنهان و لایه خروجی.
۶	۴.۱	شبکه‌های عصبی دو لایه بازگشتی.
۱۱	۵.۱	پایداری لیاپانوف.
۱۲	۶.۱	پایداری مجانبی.
	رفتارهای منحنی $x(t)$ در شبکه عصبی (۷.۲) با ۱۰ نقطه شروع مختلف در مثال	۱.۲	
۲۰	۱.۴.۲	
۲۰	۲.۲	رفتار همگرایی $\ x(t) - x^*\ _2$ در مثال ۱.۴.۲
۲۱	۳.۲	نمودار فاز شبکه عصبی (۷.۲) با ۱۱ نقطه شروع مختلف در مثال ۲.۴.۲
	رفتارهای منحنی $x(t)$ در شبکه عصبی (۷.۲) با ۳۰ نقطه شروع مختلف در مثال	۴.۲	
۲۲	۳.۴.۲	
۲۲	۵.۲	نمودار فاز شبکه عصبی (۷.۲) با ۸ نقطه شروع مختلف در مثال ۴.۴.۲
	رفتارهای منحنی $x(t)$ در شبکه عصبی (۷.۲) با ۳۰ نقطه شروع مختلف در مثال	۶.۲	
۲۳	۵.۴.۲	
	رفتارهای منحنی $x(t)$ در شبکه عصبی (۷.۲) با ۵۰ نقطه شروع مختلف در مثال	۷.۲	
۲۴	۶.۴.۲	
۲۸	۱.۳	یک نمودار بلوکی ساده برای شبکه عصبی (۹.۳) و (۱۰.۳)
	نمودار فاز شبکه عصبی (۹.۳) و (۱۰.۳) با ۶ نقطه شروع مختلف و $k = 100$ در مثال	۲.۳	
۳۱	۱.۴.۳	
	رفتار همگرایی $\ y(t) - y^*\ _2$ با $y_0 = (1, -1, 1, -1, 1, -1, 1, -1)^T$ در مثال	۳.۳	
۳۲	۱.۴.۳	
۳۲	۴.۳	رفتار همگرایی $\ y(t) - y^*\ _2$ با ۱۰ نقطه شروع مختلف و $k = 20$ در مثال ۱.۴.۳
۳۳	۵.۳	رفتار همگرایی $\ y(t) - y^*\ _2$ با $y_0 = (3, -3, 3, -3)^T$ در مثال ۲.۴.۳

۳۳	رفتارگذرای $y(t)$ با ۳۰ نقطه شروع مختلف و $k = 20$ در مثال ۲.۴.۳	۶.۳
	رفتارهای منحنی $x(t)$ در شبکه عصبی (۷.۴) با ۲۰ نقطه شروع مختلف در مثال	۱.۴
۴۲	مسیرهای شبکه عصبی (۷.۴) برای مثال ۲.۴.۴	۱.۴.۴
۴۳	مسیرهای شبکه عصبی (۶.۴) برای مثال ۲.۴.۴	۲.۴
۴۳	رفتارهای گذرای $x_1(t), x_2(t), x_5(t), x_6(t)$ در شبکه عصبی (۷.۴) بانقطه شروع	۳.۴
۴۴	در مثال ۳.۴.۴ $x_0 = (1, -1, 1, -1, 1, -1, 1)^T$	۴.۴
	رفتارهای گذرای $x_1(t), x_2(t), x_6(t)$ در شبکه عصبی (۷.۴) بانقطه شروع $x_0 =$	۵.۴
۴۵	در مثال ۳.۴.۴ $x_0 = (1, -1, 1, -1, 1, -1, 1)^T$	۶.۴
	رفتارواگرای شبکه عصبی (۶.۴) بانقطه شروع $x_0 = (1, -1, 1, -1, 1, -1, 1)^T$	۶.۴
۴۵	در مثال ۳.۴.۴	۷.۴
	رفتارواگرای شبکه عصبی (۷.۲) بانقطه شروع $x_0 = (1, -1, 1, -1, 1, -1, 1)^T$	۷.۴
۴۶	در مثال ۳.۴.۴	۸.۴
۴۷	رفتارهای منحنی $x(t)$ در شبکه عصبی (۷.۴) با ۵۰ نقطه شروع مختلف در مثال ۴.۴.۴	۸.۴
۴۷	رفتار همگرایی $\ x(t) - x^*\ _2$ در مثال ۴.۴.۴	۹.۴
	رفتارهای منحنی $x(t)$ در شبکه عصبی (۷.۴) با ۱۰۰ نقطه شروع تصادفی در مثال	۱۰.۴
۴۸	در مثال ۵.۴.۴	۵.۴.۴

لیست جداول

۵ ۱.۱ توابع فعال سازی
---	---------------------------

فصل ۱

مقدمات و مفاهیم اساسی

۱.۱ مقدمه‌ای بر شبکه‌های عصبی

۱.۱.۱ شبکه‌های عصبی بیولوژیکی

واحد اصلی دستگاه عصبی، سلولی خاص به نام نرون^۱ می‌باشد و بدون شک نحوه کار مغز و راز شعور آدمی در آن نهفته است. نرون‌ها با وجود تفاوت‌های زیاد از نظر اندازه و شکل ظاهری، مشخصه‌های مشترکی دارند. همان‌طور که در شکل ۱.۱ نشان داده شده است از تنه سلولی نرون تعدادی شاخک کوتاه خارج می‌شود که دندریت^۲ نام دارند. دندریت‌ها و تنه سلولی، سیگنال‌ها را از نرون‌های مجاور دریافت می‌کنند و از طریق یک لوله باریک به نام آکسون^۳ به نرون‌های دیگر منتقل می‌گردد. آکسون در انتهای خود به تعدادی رشته جانبی باریک تقسیم می‌گردد که پایانه‌های آکسونی نام دارد و با دندریت‌های سایر نرون‌ها مرتبط است. ارتباط بین آکسون یک نرون و دندریت نرون دیگر را سیناپس^۴ می‌نامند. یک سیناپس مرکب از پایانه قبل از سیناپسی^۵، شکاف سیناپسی^۶ و پایانه بعد از سیناپسی^۷ می‌باشد. همه سیگنال‌های جمع‌بندی شده، در تنه نرون ترکیب می‌شوند و اگر وسعت سیگنال‌های ترکیب شده به آستانه نرون برسد، مرحله تحریک شدن^۸ فعال می‌شود و یک سیگنال خروجی تولید می‌شود. این سیگنال به صورت یک پالس منفرد یا بخشی از پالس‌ها در یک میزان خاص به موازات آکسون به پایانه‌های سیناپسی انتقال می‌یابد و موجب ترشح موادی به نام عصب-رسانه می‌شود. عصب-رسانه در داخل

^۱Neuron

^۲Dendrite

^۳Axon

^۴Synapse

^۵Presynaptic Terminal

^۶Synaptic Cleft

^۷Postsynaptic Terminal

^۸Firing

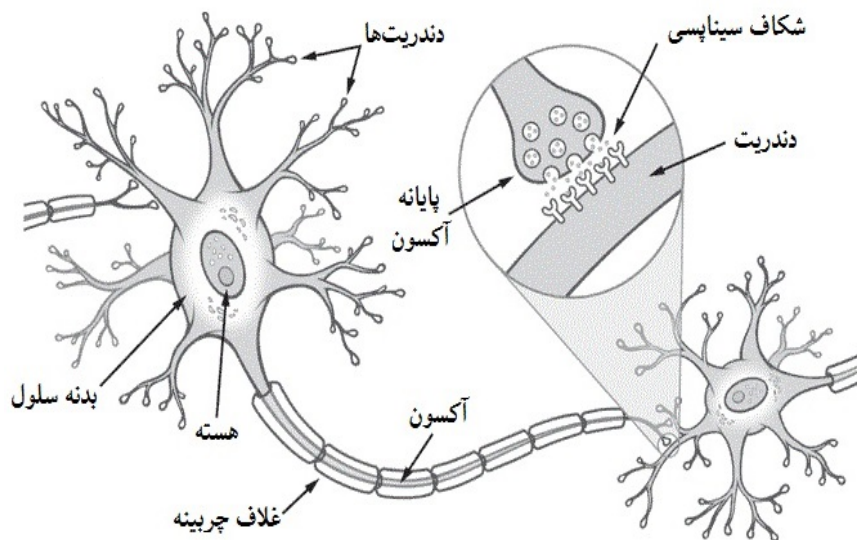
شکاف سیناپسی پخش می‌شود و نرون بعدی را تحریک می‌کند و به این ترتیب یک سیگنال از یک نرون به دیگری انتقال می‌یابد. تعداد بسیار زیادی آکسون از نرون‌های مختلف با دندریته‌های یک نرون به این صورت ارتباط برقرار می‌کنند.
نرون‌ها با توجه به عملکردشان به سه دسته تقسیم می‌شوند:

(۱) **نرون‌های حسی:** این نرون‌ها پیام‌ها را از گیرنده‌ها به دستگاه عصبی مرکزی منتقل می‌کنند. گیرنده سلول خاصی است که در اندام‌های حسی، عضلات، پوست و ... قرار دارد و تغییرات فیزیکی یا شیمیایی را در می‌یابد، این تغییرات را به سیگنال‌های عصبی تبدیل می‌کند.

(۲) **نرون‌های حرکتی:** این نرون‌ها پیام‌ها را از مغز یا طناب نخاعی به اندام‌های عمل کننده می‌رسانند.

(۳) **نرون‌های رابط:** این نرون‌ها پیام‌ها را از نرون‌های حسی می‌گیرند و به یک نرون رابط دیگر انتقال می‌دهند و یا پیام‌ها را به یک نرون حرکتی می‌رسانند. نرون‌های رابط فقط در مغز، چشم و طناب نخاعی یافت می‌شوند.

حال منظور از یادگیری عبارت است از تغییری نسبتاً ساختاری در سیناپس (بعد از تحریک)، که این تغییر ساختاری باعث افزایش کارایی سیناپس می‌شود.

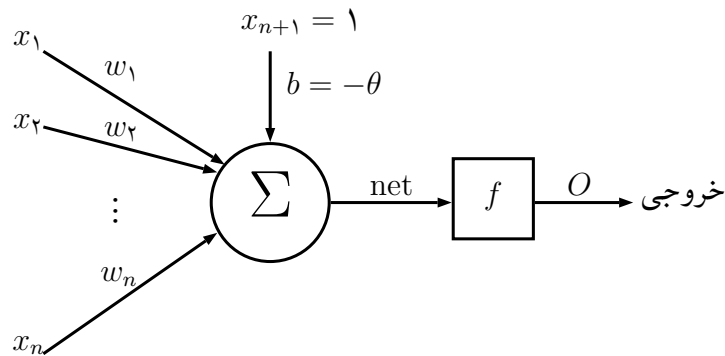


شکل ۱.۱: ساختار یک نرون.

۲.۱.۱ شبکه‌های عصبی مصنوعی

بعد از اشاره‌ای گذرا بر شبکه عصبی بیولوژیکی و نشان دادن گوشه‌ای بسیار کوچک از این جهان مرموز و فوق‌العاده پیچیده به موضوع اصلی یعنی شبکه‌های عصبی مصنوعی می‌پردازیم. شبکه‌های عصبی

مصنوعی^۹ مدل‌های سخت افزاری یا نرم افزاری الهام گرفته از ساختار و رفتار نرون‌های بیولوژیکی و سیستم‌های عصبی انسان هستند که شامل عناصر پردازشی (به نام نرون‌ها) و ارتباطات میان آنها با ضرایب (وزن‌ها) اختصاص یافته به ارتباطات است، که ساختار شبکه عصبی را تشکیل می‌دهند. در این پایان‌نامه ما شبکه‌های عصبی مصنوعی را یک مدل ریاضی فرض خواهیم کرد و به‌جای استفاده از عبارت شبکه‌های عصبی مصنوعی به‌طور خلاصه عبارت شبکه‌های عصبی را به‌کار خواهیم برد.



شکل ۲.۱: مدل ریاضی یک نرون.

مدل شبیه‌سازی شده یک نرون، در شکل ۲.۱ نمایش داده شده است. یک نرون در حالت کلی یک مجموعه از $n + 1$ ورودی x_j ($j = 1, \dots, n + 1$) دارد. این ورودی‌ها همان سیگنال‌های عصبی می‌باشند که از محیط یا از یک نرون دیگر فرستاده می‌شوند. w_j ($j = 1, \dots, n + 1$) که روی آکسون‌ها واقع هستند نقش ساختار سیناپسی را ایفا می‌کنند. این‌ها هستند که باید تغییر کنند تا فرآیند یادگیری اتفاق افتد حال این‌که این تغییر بر چه قاعده‌ای استوار باشد بستگی به الگوریتم آموزش یا تعلیم دارد. وزن آخرین ورودی یعنی θ را با b نشان می‌دهند و آن را بایاس می‌نامند. سیگنال‌های عصبی x_1 و x_2 و ... و x_{n+1} به ترتیب در w_1 و w_2 و ... و w_{n+1} ضرب شده و مجموع این حاصل ضرب‌ها که در شکل آن را با net نشان می‌دهیم، وارد نرون می‌شوند. در این مرحله نرون همانند یک تابع روی net عمل می‌کند و حاصل کار خود (برد تابع) را از طریق آکسون‌های طرف راست به نرون‌های دیگر می‌فرستد. این تابع را تابع فعال‌سازی^{۱۰} می‌نامند.

البته خاطر نشان می‌کنیم که اگر نرون به‌عنوان تابع فرض شود این تابع از \mathbb{R}^n به \mathbb{R} تعریف می‌گردد. یعنی سیگنال خروجی از نرون در تمام آکسون‌های طرف راست یکسان می‌باشد که روی شکل با O نمایش داده‌ایم. همان‌طور که قبلاً اشاره شد نرون‌ها می‌توانند حسی، ارتباطی و یا حرکتی باشند. اطلاعات از طریق نرون‌های حسی دریافت می‌شود و از طریق نرون‌های ارتباطی انتقال می‌یابد و بالاخره نتیجه کار شبکه عصبی و یا واکنش شبکه عصبی به اطلاعات دریافت شده از طریق نرون‌های حرکتی به خارج

^۹Artificial Neural Networks

^{۱۰}Activation Function

منتقل می‌گردد (شکل ۳.۱).

نرون‌های حسی را لایه ورودی، نرون‌های حرکتی را لایه خروجی و نرون‌های ارتباطی را لایه پنهان نیز می‌گویند. لایه مخفی می‌تواند از چندین لایه تشکیل گردد البته در شکل ۳.۱ لایه مخفی را تک لایه در نظر گرفته است.

مشخصه‌های اصلی یک شبکه عصبی که در طراحی آن باید مدنظر قرار گیرند عبارت‌اند از :

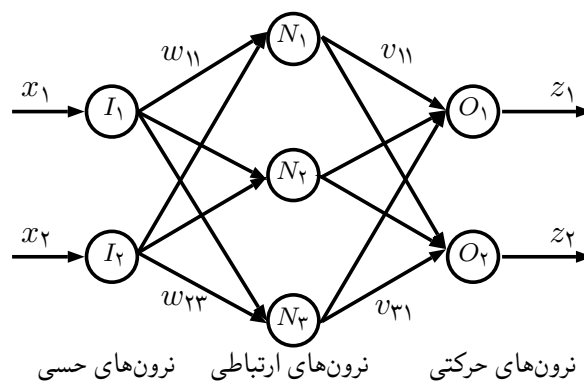
معماری شبکه عصبی: نحوه اتصالات بین نرون‌ها، تعداد آن‌ها و تعداد لایه‌های تشکیل دهنده بخش نرون‌های ارتباطی را معماری شبکه عصبی می‌گوییم.

تابع‌های فعال‌سازی: این‌که چه تابعی روی نرون قرار بگیرد تا بر ورودی‌های نرون اثر کرده و خروجی نرون را تولید کند تابع فعال‌سازی آن نرون گویند. در جدول ۱.۱ تعدادی از مهم‌ترین توابع فعال‌سازی نشان داده شده است.

الگوریتم آموزش: روشی که وزن‌های روی آکسون‌ها بر اساس آن تغییر می‌کنند (w_{ij} ها و v_{jk} ها در شکل ۳.۱) تا خروجی شبکه عصبی به حالت مطلوب درآید را الگوریتم آموزش می‌گوییم.

با توجه به مطالب قبل می‌توان برای شبکه‌های عصبی مصنوعی خصوصیات عمومی زیر را نسبت داد:

۱. پردازش اطلاعات در عناصر بسیار ساده به نام نرون (سلول عصبی) انجام می‌گیرد.
۲. سیگنال‌های عصبی بین سلول‌های عصبی از طریق اتصالات بین آن‌ها مبادله می‌شوند.
۳. به هر خط اتصال (آکسون) یک وزن نسبت داده می‌شود که در شبکه‌های عصبی معمولی این وزن‌ها در سیگنال‌های منتقل شده ضرب می‌گردند.
۴. هر نرون دارای یک تابع فعال‌سازی می‌باشد که این تابع در اکثر موارد غیرخطی است.



شکل ۳.۱: ساختار یک نرون با لایه ورودی، لایه پنهان و لایه خروجی.

جدول ۱.۱: توابع فعال‌سازی

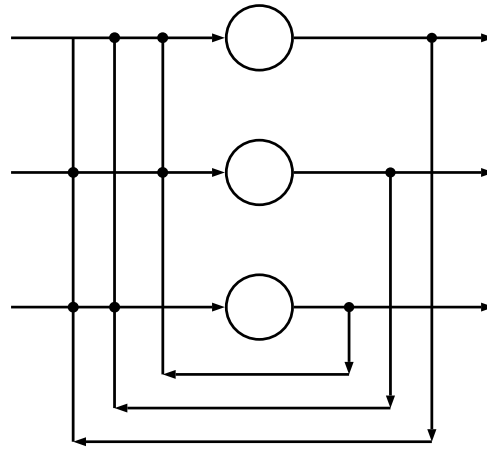
شکل تابع	تعریف تابع	نام	ردیف
	$f(x) = \begin{cases} 0 & x < 0, \\ 1 & x \geq 0. \end{cases}$	آستانه‌ای دو مقداره	۱
	$f(x) = \begin{cases} -1 & x < 0, \\ 1 & x \geq 0. \end{cases}$	آستانه‌ای دو مقداره متقارن	۲
	$f(x) = x$	خطی	۳
	$f(x) = \begin{cases} -1, & x < -1, \\ x, & -1 \leq x \leq 1, \\ 1, & x > 1. \end{cases}$	آستانه‌ای خطی متقارن	۴
	$f(x) = \begin{cases} 0, & x < 0, \\ x & 0 \leq x \leq 1, \\ 1, & x > 1. \end{cases}$	آستانه‌ای خطی	۵
	$f(x) = \frac{1}{1+e^{-x}}$	سیگموئید	۶

شبکه‌های عصبی بازگشتی

در شبکه‌های عصبی بازگشتی^{۱۱}، حداقل یک سیگنال برگشتی از یک نرون به همان نرون یا نرونهاي همان لایه و یا لایه قبل وجود دارد. چنین شبکه‌ای حافظه‌ای را نگه می‌دارد و حالت بعد نه تنها به

^{۱۱} Feedback (Recurrent) Neural Network

سیگنالهای ورودی بلکه به حالات قبل شبکه نیز وابسته است. این شبکه‌های عصبی معمولاً در مدل ریاضی به صورت معادلات دیفرانسیل مرتبه اول ظاهر می‌شوند.



شکل ۴.۱: شبکه‌های عصبی دو لایه بازگشتی.

۳.۱.۱ تاریخچه تکامل شبکه عصبی

در سال ۱۹۴۳ یک نرولوژیست به نام وارن مک کلاچ^{۱۲} به همراه یک متخصص آمار به نام والتر پیتز^{۱۳} اولین مدل ریاضی نرون را ارائه دادند [۴۴]. این نرون ساده یک عنصر محاسبه‌گر بود که عناصری به نام ورودی را در مقادیر ثابتی به نام وزن ضرب می‌کرد و آنها را از یک عملگر خطی عبور می‌داد و حاصل خروجی نرون را تشکیل می‌داد. در این مدل وزنها مقادیر ثابتی بودند. در سال ۱۹۴۹ دونالد هب^{۱۴} [۱۶] کتابی درباره نحوه یادگیری در مغز انسان نوشت. بر این مبنا قانون یادگیری برای نرون را ارائه داد که به قانون هب معروف است و بعدها در تعلیم شبکه‌های عصبی از آن استفاده شد. در سال ۱۹۵۴ فارلی و کلارک^{۱۵} [۱] مدلی از نرون‌ها را به‌طور تصادفی به هم وصل کردند و قانون هب را پیاده‌سازی نمودند و نشان دادند که می‌توان دو الگوی ورودی را با این قانون تعلیم از هم تشخیص داد. با پیشرفت علم کامپیوتر اولین مدل از نرون عصبی مصنوعی توسط ناتانیال روچستر^{۱۶} محقق شرکت IBM در سال ۱۹۵۸ با استفاده از کامپیوتر شبیه‌سازی شد. روزنبلات^{۱۷} [۲۱، ۲۲] در سال ۱۹۵۸ نرون ساده مک‌کلاچ پیتز را اصلاح کرد و به آن قابلیت یادگیری و سازگاری را اضافه کرد و این نرون را پرسپترون نامید. قانون تعلیم پرسپترون^{۱۸} اولین قانون رسمی برای شبکه‌های عصبی است.

^{۱۲}Warren McCulloch

^{۱۳}Walter Pitts

^{۱۴}Donald Hebb

^{۱۵}Farley and Clark

^{۱۶}Nathanial Rochester

^{۱۷}Frank Rosenblatt

^{۱۸}Perceptron

برنارد ویدرو^{۱۹} و مارسین هاف^{۲۰} و همکارانشان یک مدل سه سطحی از نرون‌ها را ابداع کردند که آدلین^{۲۱} نام‌گذاری شد و مختصر شده حروف "عنصر خطی تطبیقی"^{۲۲} بود [۹، ۱۰، ۱۱، ۱۲]. و برای آدلین یک قانون تعلیم بر مبنای مشتق‌گیری بیان نمودند. این قانون تعلیم به $\alpha - LMS$ معروف است. از متصل کردن چند آدلین به یکدیگر مادالین^{۲۳} ساخته شد. ویدرو و هاف از شبکه عصبی مادالین برای حذف اکو از خطوط تلفن استفاده نمودند.

سال ۱۹۶۹ آغاز یک دوره رکود برای شبکه عصبی بود. پاپرت و مینسکی^{۲۴} [۳۵] کتابی به نام پرسپترون نوشتند و نشان دادند که اگرچه گیت‌های OR و AND را می‌توان با استفاده از پرسپترون پیاده‌سازی کرد ولی گیت XOR قابل پیاده‌سازی نمی‌باشد و لذا نمی‌توان از شبکه عصبی برای پردازش اطلاعات استفاده کرد. الگوریتم پس انتشار^{۲۵} پال ورباس^{۲۶} [۳۸] در سال ۱۹۷۴ ارائه شد و بعدها توسط رامل هارت^{۲۷} [۱۷] به‌طور مستقل کشف گردید. این الگوریتم از زمان پیدایش به‌طور گسترده به‌عنوان یک الگوریتم آموزش در شبکه‌های عصبی پیش‌خورد^{۲۸} مورد استفاده قرار گرفته است. سال ۱۹۸۲ تا ۱۹۸۶ را می‌توان تولد دوباره شبکه عصبی دانست. در این سال‌ها دو اتفاق مهم در این زمینه رخ داد. اول ارائه شبکه عصبی بازگشتی و مفهوم تابع انرژی توسط جان هاپفیلد^{۲۹} [۲۷] بود که در سال ۱۹۸۲ منتشر شد. دومین اتفاق، حل مسأله فروشده دوره‌گرد (TSP) توسط شبکه هاپفیلد بود [۲۸]. هاپفیلد نشان داد که از این شبکه بازگشتی می‌توان برای حل مسائل بهینه‌سازی استفاده کرد و افق جدیدی را در شبکه‌های عصبی مصنوعی گشود.

۴.۱.۱ تاریخچه حل مسائل بهینه‌سازی با استفاده از شبکه‌های عصبی

طیف گسترده‌ای از مسائل مهم در رشته‌های علوم پایه و مهندسی از جمله کنترل بهینه، طراحی ساختمان، پردازش تصویر، تقریب توابع، تحلیل رگرسیون و... قابل تبدیل به مسائل بهینه‌سازی خطی و غیرخطی هستند. در دهه‌های گذشته الگوریتم‌های بهینه‌سازی عددی فراوانی برای حل مسائل بهینه‌سازی خطی و غیرخطی ارائه شده‌اند. از آن‌جا که زمان مورد نیاز برای حل مسائل بهینه‌سازی و به‌خصوص مسائل بهینه‌سازی غیرخطی بسیار وابسته به بعد و ساختار مسأله است بنابراین الگوریتم‌های عددی کارایی کمی

^{۱۹}Bernard Widrow

^{۲۰}Marcian Hoff

^{۲۱}Adeline

^{۲۲}Adaptive Linear Elements

^{۲۳}Madaline

^{۲۴}Papert and Minsky

^{۲۵}Back Propagation

^{۲۶}Paul Werbos

^{۲۷}Rumelhart

^{۲۸}Feed Forward

^{۲۹}John Hopfield

از خود نشان می‌دهند. یک رهیافت امیدوارکننده برای حل مسائل بهینه‌سازی با ابعاد بالا استفاده از شبکه‌های عصبی مصنوعی می‌باشد.

در سال ۱۹۸۶ تانک^{۳۰} و هاپفیلد [۴۳] اولین شبکه عصبی را برای حل مسائل برنامه‌ریزی خطی معرفی کردند و نشان دادند که وضعیت شبکه در هر تکرار به‌گونه‌ای تغییر می‌کند که تابع انرژی متناظر با آن به‌طور یکنواخت کاهش می‌یابد تا جایی که به نقطه مینیمم خود می‌رسد و این نقطه مینیمم متناظر با نقطه تعادل شبکه عصبی می‌باشد. آن‌ها این شبکه عصبی را توسط یک مدار الکترونیکی پیاده‌سازی کردند. هم‌چنین از این شبکه برای حل مسأله فروشنده دوره‌گرد با ۳۰ شهر استفاده نمودند. این شبکه دارای نقص‌هایی بود به‌خصوص این‌که نقطه تعادل شبکه در شرایط KKT^{۳۱} صدق نمی‌کرد و لذا جواب مطلوبی از مسأله حاصل نمی‌شد. با این وجود کارهای هاپفیلد انگیزه بسیار خوبی را برای محققین به‌وجود آورد تا در این زمینه فعالیت کنند. کندی و چاو^{۳۲} [۳۱] با افزودن یک پارامتر جریمه متناهی به شبکه هاپفیلد آن را توسعه دادند و از آن برای حل مسأله برنامه‌ریزی غیرخطی محدب استفاده نمودند. با پارامترهای جریمه این شبکه در یافتن نقطه بهینه دقیق ناتوان است به‌خصوص اگر پارامتر جریمه بزرگ باشد این شبکه به‌سختی عمل می‌کند.

برای جلوگیری از بکار بردن پارامتر جریمه یک شبکه عصبی سوئیچ-خازن^{۳۳} توسط رودریگز-وازکز^{۳۴} و همکاران [۳۹] معرفی شد. هم‌چنین ما و شانبلات^{۳۵} [۱۴] یک شبکه عصبی دو فازی ارائه کردند که فاز اول شبکه مشابه شبکه کندی و چاو بود و در فاز دوم آن مسیر شبکه به جواب دقیق مسأله همگرا می‌شد بنابراین این روش جواب‌های دقیق‌تری نسبت به شبکه کندی و چاو ارائه می‌داد. مشکل این شبکه در این بود که پایداری فاز دوم شبکه بستگی به انتخاب یک مقدار بزرگ از پارامتر جریمه داشت و لذا اگرچه تأثیر پارامتر جریمه در این روش کاهش یافته بود و جواب‌های حاصل دقیق‌تر بودند اما هنوز مستقل از پارامتر نبود.

ژانگ و کنستانتینیدس^{۳۶} [۴۰] بر مبنای روش لاگرانژ یک شبکه عصبی ارائه کردند که کاملاً مستقل از پارامتر جریمه و قادر به حل مسائل غیرخطی بود. نقطه تعادل این شبکه عصبی در شرایط بهینگی مرتبه اول و دوم صدق می‌کرد و هم‌چنین شبکه حاصل همگرا بود. در سال ۱۹۹۳ بوزردوم و پتیسسن^{۳۷} [۵] شبکه‌ای را بر مبنای گرادیان و روش تصویر و مستقل از پارامتر جریمه ابداع کردند که تنها قادر به حل مسائل درجه دوم با متغیرهای کران‌دار بود. این روش در عمل کارایی بود اما نمی‌توانست مسائل کلی برنامه‌ریزی خطی و درجه دوم را حل کند.

^{۳۰} Tank

^{۳۱} Karush-Kuhn-Tucker

^{۳۲} Kennedy and Chua

^{۳۳} Switched-Capacitor

^{۳۴} Rodriguez-Vazquez

^{۳۵} Maa and Shanblatt

^{۳۶} Zhang and Constantinides

^{۳۷} Bouzerdoum and Pattison

زیا^{۳۸} و همکاران [۴۹، ۵۰، ۵۳، ۵۴] چندین مدل را برای حل مسائل بهینه‌سازی غیرخطی با قيود خطی و غیرخطی ارائه کردند. هم‌چنین چندین مدل شبکه عصبی برای حل مسائل بهینه‌سازی غیرخطی توسط عفتی^{۳۹} و همکاران [۲، ۳، ۱۸، ۳۶، ۴۱، ۴۲] ارائه شده است.

۲.۱ مقدمه‌ای بر بهینه‌سازی غیرخطی

تعریف ۱.۲.۱ ([۳۷]). تابع $f : \mathbb{R}^n \rightarrow \mathbb{R}$ روی مجموعه محدب و ناتهی $S \subseteq \mathbb{R}^n$ محدب نامیده می‌شود اگر برای هر $x, y \in S$ و $\alpha \in (0, 1)$ داشته باشیم

$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y)$$

تابع f روی S اکیداً محدب نامیده می‌شود اگر نامساوی بالا زمانی که $x \neq y$ است، به‌طور اکید برقرار باشد.

یک مسأله برنامه‌ریزی غیرخطی در حالت کلی به‌صورت زیر در نظر گرفته می‌شود:

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & g_i(x) \leq 0, \\ & h_j(x) = 0, \\ & x \in X \subset \mathbb{R}^n, \end{aligned} \tag{۱.۱}$$

که $f : \mathbb{R}^n \rightarrow \mathbb{R}$ ، $g_i : \mathbb{R}^n \rightarrow \mathbb{R}$ ($i = 1, \dots, m$) و $h_j : \mathbb{R}^n \rightarrow \mathbb{R}$ ($j = 1, \dots, p$) توابع دو بار مشتق پذیر پیوسته هستند. برای تسهیل در نمادگذاری، توابع برداری $\mathbf{g} = (g_1, g_2, \dots, g_m)$ و $\mathbf{h} = (h_1, h_2, \dots, h_p)$ را معرفی می‌کنیم و (۱.۱) را به‌صورت زیر می‌نویسیم:

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & \mathbf{g}(x) \leq 0, \\ & \mathbf{h}(x) = 0, \\ & x \in X \subset \mathbb{R}^n, \end{aligned} \tag{۲.۱}$$

که محدودیت‌های $\mathbf{g}(x) \leq 0$ و $\mathbf{h}(x) = 0$ ، محدودیت‌های تابعی و محدودیت $x \in X \subset \mathbb{R}^n$ ، محدودیت مجموعه‌ای نامیده می‌شود. هرگاه توابع f و \mathbf{g} روی X محدب باشند و \mathbf{h} آفین باشد یعنی $\mathbf{h} = A\mathbf{x} - \mathbf{b}$ ، $A \in \mathbb{R}^{p \times n}$ ، $\text{rank}(A) = p$ ($0 \leq p < n$) و $\mathbf{b} \in \mathbb{R}^p$ باشد، آن‌گاه (۲.۱) یک مسأله بهینه‌سازی محدب^{۴۰} و در غیر این صورت یک مسأله بهینه‌سازی نامحدب^{۴۱} نامیده می‌شود.

^{۳۸}Xia

^{۳۹}Effati

^{۴۰}Convex Optimization

^{۴۱}Nonconvex Optimization

قضیه ۲.۲.۱ (شرایط کافی مرتبه دوم^{۴۲} [۱۵]). فرض کنید x^* یک نقطه شدنی برای مسأله (۲.۱) باشد. اگر $\mu \in \mathbb{R}^m$ و $\lambda \in \mathbb{R}^p$ وجود داشته باشند به طوری که

$$\mu \geq 0, \quad (۳.۱)$$

$$\mu^T g(x^*) = 0, \quad (۴.۱)$$

$$\nabla f(x^*) + \lambda^T \nabla h(x^*) + \mu^T \nabla g(x^*) = 0, \quad (۵.۱)$$

و ماتریس هسین^{۴۳}

$$\nabla^2 L(x^*, \lambda, \mu) = \nabla^2 f(x^*) + \sum_{i=1}^m \mu_i \nabla^2 g_i(x^*) + \sum_{j=1}^p \lambda_j \nabla^2 h_j(x^*) \quad (۶.۱)$$

بر زیرفضای

$$M(x^*) = \{d \mid \nabla h(x^*)d = 0, \nabla g_j(x^*)d = 0, \forall j \in J(x^*)\},$$

که در آن

$$J(x^*) = \{j \mid g_j(x^*) = 0, \mu_j > 0\}$$

معین مثبت باشد آن‌گاه x^* یک نقطه مینیمم محلی اکید^{۴۴} برای مسأله (۲.۱) است.

۳.۱ سیستم‌های دینامیکی

تعریف ۱.۳.۱ ([۲]). سیستم دینامیکی زیر را در نظر بگیرید:

$$\dot{x} = f(x(t)), \quad x(t_0) = x_0 \in \mathbb{R}^n, \quad (۷.۱)$$

که در آن f یک تابع از \mathbb{R}^n به \mathbb{R}^n می‌باشد. x^* یک نقطه تعادل^{۴۵} (۷.۱) نامیده می‌شود اگر $f(x^*) = 0$.

تعریف ۲.۳.۱ (پایداری به مفهوم لیاپانوف^{۴۶} [۲]). فرض می‌کنیم که $x(t)$ یک جواب (۷.۱) باشد، نقطه تعادل تنها x^* پایدار به مفهوم لیاپانوف است اگر برای هر $x_0 = x(t_0)$ و هر $\varepsilon > 0$ ، یک $\delta > 0$ وجود داشته باشد به طوری که اگر $\|x(t_0) - x^*\| < \delta$ آن‌گاه

$$\|x(t) - x^*\| < \varepsilon, \quad \forall t \geq t_0.$$

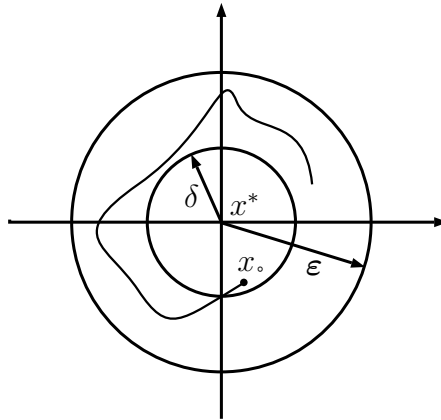
^{۴۲}Second-Order Sufficient Conditions

^{۴۳}Hessian Matrix

^{۴۴}Strict Local Minimum Point

^{۴۵}Equilibrium Point

^{۴۶}Stability in the Sense of Lyapunov



شکل ۵.۱: پایداری لیاپانوف.

تعریف ۳.۳.۱ ([۲]). سیستم دینامیکی (۷.۱) همگرای سراسری^{۴۷} به مجموعه جواب‌های (۷.۱) که با Ω^* نمایش داده می‌شود، گفته می‌شود اگر برای هر نقطه آغازین دلخواه، مسیر $x(t)$ در رابطه زیر صدق کند:

$$\lim_{t \rightarrow \infty} \text{dist}(x(t), \Omega^*) = 0,$$

که در آن

$$\text{dist}(x, \Omega^*) = \inf_{y \in \Omega^*} \|x - y\|,$$

و $\|\cdot\|$ نرم l_2 را نشان می‌دهد.

تعریف ۴.۳.۱ ([۲]). سیستم دینامیکی (۷.۱) در نقطه تعادل x^* که یکتا می‌باشد، پایدار مجانبی سراسری^{۴۸} نامیده می‌شود اگر x^* به مفهوم لیاپانوف پایدار باشد و در شرط زیر صدق کند:

$$\lim_{t \rightarrow \infty} x(t) = x^*.$$

تعریف ۵.۳.۱ ([۲]). مجموعه $M \subseteq \mathbb{R}^n$ یک مجموعه تغییر ناپذیر^{۴۹} نسبت به سیستم (۷.۱) گفته می‌شود اگر $x(t_0) \in M$ و $t_0 \geq 0$ ، به ازای هر $t \geq t_0$ ، $x(t) \in M$ باشد.

تعریف ۶.۳.۱ ([۲]). فرض می‌کنیم که $\Omega \subset \mathbb{R}^l$ یک مجموعه بسته و محدب باشد که ممکن است بیکران باشد. $P_\Omega(x): \mathbb{R}^l \rightarrow \Omega$ یک عملگر تصویر^{۵۰} نامیده می‌شود و به صورت زیر تعریف می‌شود:

$$P_\Omega(x) = \arg \min_{v \in \Omega} \|x - v\|, \quad (۸.۱)$$

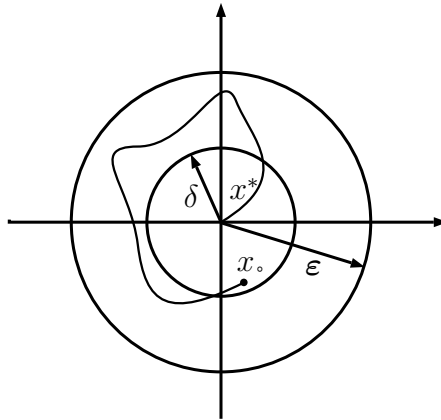
که $\|\cdot\|$ نرم l_2 را نشان می‌دهد.

^{۴۷}Globally Convergent

^{۴۸}Globally Asymptotically Stable

^{۴۹}Invariant Set

^{۵۰}Projection Operator



شکل ۶.۱: پایداری مجانبی.

تعریف ۷.۳.۱ ([۲]) تابع $F : E \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$ روی E در شرط لیپ شیتز صدق می‌کند (پیوسته لیپ شیتز^{۵۱} است) اگر ثابت L وجود داشته باشد به طوری که برای هر دو نقطه $x, y \in E$ داشته باشیم:

$$\|F(x) - F(y)\| \leq L \|x - y\|.$$

تابع F پیوسته لیپ شیتز محلی^{۵۲} روی E نامیده می‌شود اگر برای نقطه $x_0 \in E$ یک ε -همسایگی مانند $N_\varepsilon(x_0) \subset E$ و یک ثابت L_0 وجود داشته باشد به طوری که برای هر دو نقطه $x, y \in N_\varepsilon(x_0)$ داشته باشیم:

$$\|F(x) - F(y)\| \leq L_0 \|x - y\|.$$

قضیه ۸.۳.۱. تابع $F : E \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$ را در نظر بگیرید. اگر $F \in C^1(E)$ ، آنگاه F روی E لیپ شیتز محلی است.

□

برهان. به [۳۳] مراجعه شود.

تعریف ۹.۳.۱ ([۲]). نگاشت $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$

(آ) یکنوا نامیده می‌شود اگر برای هر دو نقطه $x, y \in \mathbb{R}^n$ داشته باشیم:

$$\langle x - y, F(x) - F(y) \rangle \geq 0.$$

(ب) اکیداً یکنوا نامیده می‌شود اگر برای هر دو نقطه متمایز $x, y \in \mathbb{R}^n$ داشته باشیم:

$$\langle x - y, F(x) - F(y) \rangle > 0.$$

که $\langle \cdot, \cdot \rangle$ ضرب داخلی را نشان می‌دهد.

^{۵۱}Lipschitz Continuous

^{۵۲}Locally Lipschitz Continuous

تعریف ۱۰.۳.۱ ([۲]). اگر $g : \Omega_1 \subset \mathbb{R}^l \rightarrow \mathbb{R}$ ، آن‌گاه هر مجموعه غیر تهی به شکل زیر

$$L(r) = \{u \in \Omega_1 | g(u) \leq r, r \in \mathbb{R}\}$$

یک مجموعه سطح^{۵۳} از g نامیده می‌شود.

تعریف ۱۱.۳.۱ ([۲]). یک ماتریس $n \times n$ ، $M(x)$ ، که عناصر آن m_{ij} ، $i = 1, \dots, n$ و $j = 1, \dots, n$ ، توابعی هستند که روی مجموعه $S \subset \mathbb{R}^n$ تعریف شده‌اند، روی S نیمه معین مثبت^{۵۴} نامیده می‌شود اگر

$$v^T M(x) v \geq 0, \forall v \in \mathbb{R}^n, \forall x \in S.$$

تعریف ۱۲.۳.۱ ([۲]). فرض می‌کنیم $V : \mathbb{R}^n \rightarrow \mathbb{R}$ یک تابع به‌طور پیوسته مشتق پذیر باشد. $V(x)$ به‌طور شعاعی بی‌کران گفته می‌شود اگر

$$\|x\| \rightarrow \infty \Rightarrow V(x) \rightarrow \infty.$$

لم ۱۳.۳.۱ ([۲۹]). فرض کنیم که $g : \Omega_1 \subset \mathbb{R}^l \rightarrow \mathbb{R}$ باشد که در آن Ω_1 بی‌کران است. آن‌گاه همه مجموعه‌های سطح g کراندار می‌باشند اگر و فقط اگر

$$\lim_{k \rightarrow \infty} g(u^k) = +\infty,$$

که در آن $\{u^k\} \subset \Omega_1$ و $\lim_{k \rightarrow \infty} \|u^k\| = +\infty$.

لم ۱۴.۳.۱ ([۲۹]). فرض کنیم که $g : \Omega_1 \subset \mathbb{R}^l \rightarrow \mathbb{R}$ روی مجموعه بسته Ω_1 پیوسته باشد آن‌گاه همه مجموعه‌های سطح g کراندار می‌باشند اگر و فقط اگر مجموعه مینیمم‌کننده‌های g غیرتهی و کراندار باشند.

قضیه ۱۵.۳.۱ ([۲۹]). فرض کنیم که تابع $F : K \subset \mathbb{R}^l \rightarrow \mathbb{R}^l$ روی K به‌طور پیوسته مشتق‌پذیر باشد، آن‌گاه F روی K یکنوا (اکیداً) است اگر و فقط اگر ماتریس ژاکوبین آن $\nabla F(x)$ برای هر $x \in K$ نیمه‌معین مثبت (معین مثبت) باشد.

قضیه ۱۶.۳.۱ ([۳۳]). در سیستم دینامیکی (۷.۱) فرض کنیم که f یک تابع پیوسته از \mathbb{R}^n به \mathbb{R}^n باشد، آن‌گاه برای هر $t_0 \geq 0$ و $x_0 \in \mathbb{R}^n$ ، یک جواب محلی $x(t)$ به ازای $t \in [t_0, \tau)$ که $\tau > t_0$ وجود دارد. علاوه بر این اگر f در x_0 پیوسته لیپ شیتز محلی باشد آن‌گاه جواب یکتاست و اگر f در \mathbb{R}^n پیوسته لیپ شیتز باشد آن‌گاه τ می‌تواند تا $+\infty$ توسعه داده شود.

تعریف ۱۷.۳.۱ ([۲]). اگر یک جواب محلی تعریف شده در بازه $[t_0, \tau)$ نتواند به یک جواب محلی روی یک بازه بزرگتر $[t_0, \tau_1)$ که $\tau_1 > \tau$ می‌باشد گسترش یابد، آن‌گاه یک جواب ماکسیمال نامیده می‌شود و بازه $[t_0, \tau)$ ، بازه ماکسیمال وجود جواب نامیده می‌شود. بازه ماکسیمال وجود جواب مربوط به x_0 اغلب به‌صورت $[t_0, \tau(x_0))$ تعریف می‌شود.

^{۵۳}Level Set

^{۵۴}Positive Semidefinite

قضیه ۱۸.۳.۱ ([۳۰]). فرض کنیم $f: \mathbb{R}^n \rightarrow \mathbb{R}^n$ یک تابع پیوسته باشد. اگر $x(t)$ ، $t \in [t_0, \tau(x_0))$ ، یک جواب ماکسیمال باشد و $\tau(x_0) < +\infty$ آن‌گاه

$$\lim_{t \rightarrow \tau(x_0)} \|x(t)\| = +\infty.$$

قضیه ۱۹.۳.۱ (قضیه پایداری لیاپانوف [۲۵]). فرض کنید که $x = 0$ یک نقطه تعادل سیستم (۷.۱) باشد و $V: \mathbb{R}^n \rightarrow \mathbb{R}$ یک تابع به‌طور پیوسته مشتق پذیر باشد به‌طوری‌که:

$$V(0) = 0. \quad ۱.$$

$$V(x) > 0, \quad x \in \mathbb{R}^n \setminus \{0\} \quad ۲.$$

$$\dot{V}(x) \leq 0, \quad x \in \mathbb{R}^n \setminus \{0\} \quad ۳.$$

آن‌گاه $x = 0$ نقطه پایداری سیستم خواهد بود.

قضیه ۲۰.۳.۱ (قضیه پایداری مجانبی [۲۵]). تحت شرایط قضیه ۱۹.۳.۱ اگر $V(\cdot)$ در شرایط زیر صدق کند:

$$V(0) = 0. \quad ۱.$$

$$V(x) > 0, \quad x \in \mathbb{R}^n \setminus \{0\} \quad ۲.$$

$$\dot{V}(x) < 0, \quad x \in \mathbb{R}^n \setminus \{0\} \quad ۳.$$

آن‌گاه $x = 0$ پایدار مجانبی خواهد بود.

قضیه ۲۱.۳.۱ (قضیه پایداری مجانبی سراسری [۲۵]). تحت شرایط قضیه ۲۰.۳.۱ اگر $V(\cdot)$ به‌طور شعاعی بی‌کران باشد آن‌گاه نقطه $x = 0$ پایدار مجانبی سراسری است.

قضیه ۲۲.۳.۱ (اصل تغییرناپذیری لِسال^{۵۵} [۲۵]). فرض کنیم V یک تابع به‌طور پیوسته مشتق پذیر باشد، همچنین فرض کنیم:

۱. $M \subset \mathbb{R}^n$ یک مجموعه فشرده و پایدار نسبت به جواب سیستم (۷.۱) باشد.

$$\dot{V}(x) \leq 0, \quad x \in M \quad ۲.$$

$$\dot{V}(x) = 0 \quad \text{در} \quad E \quad ۳.$$

۴. N بزرگترین مجموعه پایدار در E باشد.

آن‌گاه به ازای هر $x(t_0) \in M$ که $t_0 \geq 0$ می‌باشد داریم:

$$\lim_{t \rightarrow \infty} \text{dist}(x(t), N) = 0.$$

فصل ۲

یک مدل شبکه عصبی لاگرانژی کارا برای حل ردهای از مسائل بهینه‌سازی محدب

۱.۲ مقدمه

تعداد بسیار زیادی از مسائل در رشته‌های علوم و مهندسی از جمله کنترل بهینه، طراحی ساختمان، پردازش سیگنال، تقریب توابع، تحلیل رگرسیون و ... می‌توانند به صورت مسائل بهینه‌سازی غیرخطی مقید فرمول‌بندی شوند [۷، ۸]. در دهه‌های گذشته الگوریتم‌های عددی فراوانی برای حل مسائل بهینه‌سازی غیرخطی ارائه شدند. اما از آن‌جا که زمان محاسبه مورد نیاز برای یافتن جواب‌ها بسیار وابسته به بعد و ساختار مسائل است، این الگوریتم‌ها در مسائل بهینه‌سازی با مقیاس بزرگ ناکارا هستند. در مقایسه با الگوریتم‌های عددی، شبکه‌های عصبی مصنوعی دارای قابلیت انجام محاسبات موازی و همگرایی سریع هستند. تاکنون چندین شبکه عصبی برای حل مسائل بهینه‌سازی غیرخطی محدب ارائه شدند. از جمله کِنْدی و چاو [۳۱] با توسعه شبکه عصبی تانک و هاپفیلد [۴۳] یک شبکه عصبی شامل یک پارامتر جریمه متناهی برای حل مسائل برنامه‌ریزی غیرخطی محدب ارائه دادند. یک شبکه عصبی سوئیچ خازن توسط رودریگز-وازکز و همکاران [۳۹] برای حل یک دسته از مسائل بهینه‌سازی غیرخطی محدب معرفی شد. زیا و همکاران [۴۵، ۴۶، ۴۷، ۴۸، ۴۹، ۵۰] چندین مدل را برای حل مسائل بهینه‌سازی غیرخطی با قيود خطی و غیرخطی ارائه کردند. هم‌چنین چندین مدل شبکه عصبی برای حل مسائل بهینه‌سازی غیرخطی توسط عفتی و همکاران [۱۸، ۱۹] ارائه شده است. در این فصل در بخش دوم یک شبکه عصبی برای حل مسائل بهینه‌سازی غیرخطی محدب تحت شرایط خاص معرفی شده است. در بخش سوم پایداری و همگرایی این شبکه عصبی را مورد بررسی قرار داده شده است. در بخش چهارم مثال‌هایی برای نشان دادن کارایی و عملکرد شبکه عصبی ارائه شده است.

۲.۲ مدل شبکه عصبی

مسئله بهینه‌سازی غیرخطی محدب را به صورت زیر در نظر می‌گیریم:

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & g(x) \leq 0, \\ & h(x) = 0, \end{aligned} \quad (1.2)$$

که در آن $f: \mathbb{R}^n \rightarrow \mathbb{R}$, $x \in \mathbb{R}^n$, $g(x) = (g_1(x), g_2(x), \dots, g_m(x))^T$ یک تابع برداری پیوسته m -بعدی است، و توابع f, g_1, \dots, g_m محدب و دوبار مشتق‌پذیر فرض شده‌اند. همچنین $h(x) = Ax - b$ ، $A \in \mathbb{R}^{l \times n}$ ، $\text{rank}(A) = l$ ($0 \leq l < n$) و $b \in \mathbb{R}^l$. تا پایان این فصل فرض می‌کنیم مسئله (۱.۲) یک جواب بهینه یکتا دارد.

تابع لاگرانژ متناظر با (۱.۲) را مشابه با [۲۶] به صورت زیر تعریف می‌کنیم:

$$L(x, u, v) = f(x) + \frac{1}{2} \sum_{k=1}^m u_k^2 g_k(x) + \sum_{p=1}^l v_p h_p(x), \quad (2.2)$$

که در آن $u \in \mathbb{R}^m$ و $v \in \mathbb{R}^l$ ضرب‌گرهای لاگرانژ هستند.

قضیه ۱.۲.۲. $x^* \in \mathbb{R}^n$ یک جواب بهینه برای (۱.۲) است اگر و تنها اگر $u^* \in \mathbb{R}^m$ و $v^* \in \mathbb{R}^l$ وجود داشته باشند به قسمی که $(x^{*T}, u^{*T}, v^{*T})^T$ در شرایط KKT زیر صدق کند:

$$\begin{cases} u^* \geq 0, \quad g(x^*) \leq 0, \quad u^{*T} g(x^*) = 0, \\ \nabla f(x^*) + \nabla g(x^*)^T u^* + \nabla h(x^*)^T v^* = 0, \\ h(x^*) = 0. \end{cases} \quad (3.2)$$

اکنون یک شبکه عصبی برای حل (۱.۲) و دوگان آن به صورت زیر تعریف می‌کنیم:

$$\frac{dx_i}{dt} = -\nabla_{x_i} L(x, u, v) = -\left(\frac{\partial f}{\partial x_i} + \frac{1}{2} \sum_{k=1}^m u_k^2 \frac{\partial g_k}{\partial x_i} + \sum_{p=1}^l v_p \frac{\partial h_p}{\partial x_i} \right), \quad i = 1, \dots, n, \quad (4.2)$$

$$\frac{du_k}{dt} = \nabla_{u_k} L(x, u, v) = u_k g_k, \quad k = 1, \dots, m, \quad (5.2)$$

$$\frac{dv_p}{dt} = \nabla_{v_p} L(x, u, v) = h_p(x), \quad p = 1, \dots, l, \quad (6.2)$$

با نقطه آغازین $(x_0^T, u_0^T, v_0^T)^T$ و $(k = 1, \dots, m) u_k(t_0) > 0$

حال با در نظر گرفتن $w = (x^T, u^T, v^T)^T \in \mathbb{R}^{n+m+l}$ به عنوان مجموعه نقاط بهینه (۱.۲) و دوگان آن و

$$\Psi(x, u, v) = \begin{bmatrix} - \left(\nabla f(x) + \frac{1}{2} \nabla g(x)^T u + \nabla h(x)^T v \right) \\ \text{diag}(u_1, \dots, u_m) g(x) \\ h(x) \end{bmatrix},$$

شبکه عصبی (۴.۲)-(۶.۲) را می‌توان به صورت زیر نوشت:

$$\frac{dw}{dt} = \Psi(w), \quad (7.2)$$

$$w(t_0) = w_0, \quad u(t_0) > 0.$$

۳.۲ تحلیل همگرایی و پایداری

در این بخش پایداری و همگرایی شبکه عصبی (۷.۲) را مورد بررسی قرار می‌دهیم.

قضیه ۱.۳.۲. فرض کنید $w^* = (x^{*T}, u^{*T}, v^{*T})^T$ نقطه تعادل شبکه عصبی (۷.۲) باشد. آنگاه x^* نقطه KKT مسأله (۱.۲) است.

برهان. با مقایسه شرایط KKT (۳.۲) با سیستم دینامیکی (۷.۲) می‌بینیم آن چه باید اثبات شود این است که $g_k(x^*) \leq 0$ و $u_k^* g_k(x^*) = 0$ ($k = 1, \dots, m$). جواب سیستم (۵.۲) به صورت زیر است:

$$u_k(t) = u_k(t_0) e^{\int_{t_0}^t g_k(x) dt}, \quad k = 1, \dots, m,$$

که $u_k(t_0) > 0$ یک مقدار آغازین ناصفر برای $u_k(t)$ است. واضح است که اگر x متعلق به ناحیه شدنی نباشد یعنی حداقل یک k وجود داشته باشد که $g_k(x) > 0$ ، ضرب‌گرهای متناظر $u_k(t)$ به صورت نمایی با زمان افزایش می‌یابد، و در نتیجه عبارت $\frac{1}{2} u_k^* g_k(x)$ در تابع لاگرانژ (۲.۲) با زمان افزایش می‌یابد. این بدان معناست که با افزایش زمان t جریمه نیز بدون وقفه افزایش می‌یابد تا منحنی $x(t)$ به درون ناحیه شدنی میل کند. بنابراین بعد از یک زمان متناهی منحنی $x(t)$ به سمت مجموعه شدنی حرکت می‌کند. این روند به‌طور پیوسته تکرار خواهد شد مگر آن‌که تمام محدودیت‌ها برقرار شوند. بنابراین $(x^T, u^T, v^T)^T$ سرانجام به نقطه KKT، $(x^{*T}, u^{*T}, v^{*T})^T$ میل می‌کند در حالی که ضرب‌گرهای $u_k(t)$ متناظر با محدودیت‌های غیرفعال ($g(x) < 0$) به سمت صفر میل می‌کند، و باقیمانده ضرب‌گرها متناظر با محدودیت‌های فعال به اعداد ثابت میل می‌کنند. بنابراین برهان کامل است. \square

لم ۲.۳.۲. برای هر نقطه آغازین $w(t_0) = (x(t_0)^T, u(t_0)^T, v(t_0)^T)^T$ ، یک جواب یکتا و پیوسته $w(t) = (x(t)^T, u(t)^T, v(t)^T)^T$ برای سیستم دینامیکی (۷.۲) وجود دارد.

برهان. چون $\nabla f(x)$ ، $\nabla g_k(x)$ ($k = 1, \dots, m$) و $\nabla h_p(x)$ ($p = 1, \dots, l$) روی مجموعه محدب و باز $D \subseteq \mathbb{R}^{n+m+l}$ به‌طور پیوسته مشتق‌پذیر هستند، بنابراین $\nabla f(x) + \frac{1}{2} \nabla g(x)^T u + \nabla h(x)^T v$

قضیه ۸.۳.۱ پیوسته لیپ شیتز محلی هستند و طبق قضیه ۱۶.۳.۱، شبکه عصبی (۷.۲) دارای جواب پیوسته یکتا $w(t)$ ، برای $t \in [t_0, \tau)$ ، $\tau > t_0$ ، $\tau \rightarrow \infty$ می‌باشد. □

قضیه ۳.۳.۲. اگر $\nabla^2 f(x)$ معین مثبت و $\nabla^2 g_k(x)$ (که $k = 1, \dots, m$) نیمه‌معین مثبت باشد یا اگر $\nabla^2 f(x)$ نیمه‌معین مثبت و $\nabla^2 g_k(x)$ (که $k = 1, \dots, m$) معین مثبت باشد، آنگاه شبکه عصبی (۷.۲) پایدار سراسری به مفهوم لیاپانوف و همگرایی سراسری به یک نقطه KKT، $w^* = (x^{*T}, u^{*T}, v^{*T})^T$ می‌باشد به طوری که x^* جواب بهینه مسأله (۱.۲) و $(u^{*T}, v^{*T})^T$ جواب بهینه دوگان آن است.

برهان. تابع لیاپانوف زیر را در نظر می‌گیریم:

$$E(w) = \|\Psi(w)\|^2 + \frac{1}{\gamma} \|w - w^*\|^2. \quad (۸.۲)$$

بنابراین

$$\frac{d\Psi}{dt} = \frac{\partial \Psi}{\partial w} \frac{dw}{dt} = \nabla \Psi(w) \Psi(w),$$

که ماتریس ژاکوبین نگاشت Ψ به صورت زیر به دست می‌آید:

$$\nabla \Psi(w) = \begin{bmatrix} -\nabla^2 f(x) - \frac{1}{\gamma} \sum_{k=1}^m u_k \nabla^2 g_k(x) & -\nabla g(x)^T A & -\nabla h(x)^T \\ A^T \nabla g(x) & B & O_{m \times l} \\ \nabla h(x) & O_{l \times m} & O_{l \times l} \end{bmatrix}, \quad (۹.۲)$$

که O ماتریس صفر، $A = \text{diag}(u_1, \dots, u_m)$ و $B = \text{diag}(g_1(x), \dots, g_m(m))$. حال با استفاده از (۹.۲) داریم:

$$\begin{aligned} \frac{dE(w(t))}{dt} &= \left(\frac{d\Psi}{dt} \right)^T \Psi + \Psi^T \left(\frac{d\Psi}{dt} \right) + (w - w^*)^T \frac{dw(t)}{dt} \\ &= \Psi^T \left(\nabla \Psi(w)^T + \nabla \Psi(w) \right) \Psi + (w - w^*)^T \Psi(w) \\ &= -2 \left[\left(\nabla f(x) + \frac{1}{\gamma} \nabla g(x)^T u + \nabla h(x)^T v \right)^T \left(\nabla^2 f(x) + \frac{1}{\gamma} \sum_{k=1}^m u_k \nabla^2 g_k(x) \right) \right. \\ &\quad \left. \left(\nabla f(x) + \frac{1}{\gamma} \nabla g(x)^T u + \nabla h(x)^T v \right) \right] + 2 \sum_{k=1}^m (u_k g_k) \nabla g_k + (w - w^*)^T \Psi(w) \\ &\leq 2 \sum_{k=1}^m (u_k g_k) \nabla g_k + (w - w^*)^T \Psi(w), \end{aligned}$$

طبق فرض قضیه $\left(\nabla^2 f(x) + \frac{1}{\gamma} \sum_{k=1}^m u_k \nabla^2 g_k(x) \right) -$ معین منفی است.

حال اگر $g_k(x) \leq 0$ ، $(k = 1, \dots, m)$ ، در نتیجه $\sum_{k=1}^m (u_k g_k) \nabla g_k \leq 0$. به علاوه در این حالت $\nabla \Psi(w)$ معین

منفی است و بنا به قضیه ۱۵.۳.۱ $-\Psi(w)$ اکیداً یکنواست یعنی

$$(w - w^*)^T (\Psi(w) - \Psi(w^*)) = (w - w^*)^T \Psi(w) \leq 0,$$

که $\Psi(w^*) = 0$. بنابراین

$$\frac{dE(w(t))}{dt} \leq 0. \quad (10.2)$$

از طرف دیگر اگر $g_k(x(t)) > 0$ برای بعضی $k \in \{1, \dots, m\}$ ، آن‌گاه طبق اثبات قضیه ۱.۳.۲ می‌بینیم که $x(t)$ بعد از یک زمان متناهی به درون ناحیه شدنی میل می‌کند، درحالی‌که ضرب‌گرهای متناظر u_k تا زمانی که $u_k \geq 0$ افزایش می‌یابد. در نتیجه بعد از یک زمان متناهی و با روندی مشابه حالت قبل به دست می‌آید $\frac{dE(w(t))}{dt} \leq 0$. بنابراین شبکه عصبی (۷.۲) همگرای سراسری به مفهوم لیاپانوف است. حال چون $E(w) \geq \frac{1}{2}\|w - w^*\|^2$ برای هر نقطه آغازین $w_0 \in \mathbb{R}^{n+m+l}$ ، منحنی جواب $\{w(t)\}$ کراندار است.

فرض کنید $\Gamma = \{w(t) \mid \frac{dE(w(t))}{dt} = 0\}$ ، $\Lambda = \{w \in \mathbb{R}^{n+m+l} \mid V(w) \leq V(w_0)\}$ و Π بزرگترین مجموعه تغییر ناپذیر در Γ باشد بنابراین با استفاده از اصل تغییر ناپذیری لسال ۲.۳.۱، منحنی‌های $w(t)$ به Π همگرا خواهند شد، یعنی

$$\lim_{t \rightarrow \infty} \text{dist}(w(t), \Pi) = 0.$$

اما از (۷.۲) و (۱۰.۲)، $\frac{dw}{dt} = 0$ اگر و تنها اگر $\frac{dE}{dt} = 0$ ، بنابراین $W^* = \{w \in \Lambda \mid \frac{dw}{dt} = 0\} = \Gamma$ ، در نتیجه شبکه عصبی (۷.۲) همگرای سراسری به جواب‌های بهینه مسأله (۱.۲) است. بالاخص اگر $W^* = \{(x^{*T}, u^{*T}, v^{*T})^T\}$ ، شبکه عصبی (۷.۲) برای حل (۱.۲) پایدار مجانبی سراسری به نقطه تعادل $w^* = (x^{*T}, u^{*T}, v^{*T})^T$ است. \square

۴.۲ مثال‌های عددی

در این بخش مثال‌هایی ارائه خواهد شد تا نتایجی را که در بخش‌های قبل حاصل شد نشان دهند. شبیه‌سازی این مثال‌ها با استفاده از نرم افزار Matlab 7، و حل‌کننده معادلات دیفرانسیل معمولی ode45 انجام شده است.

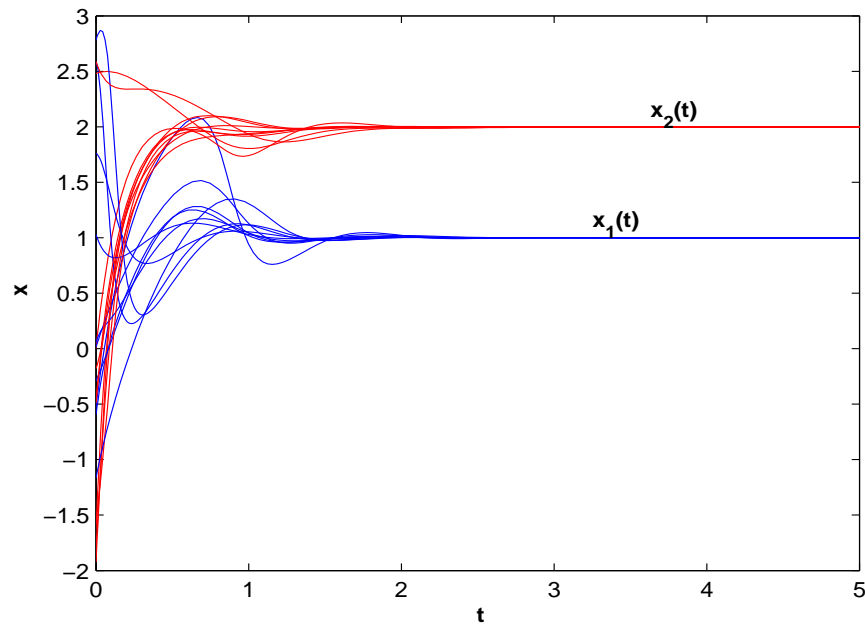
مثال ۱.۴.۲ ([۱۸]). مسأله بهینه‌سازی غیرخطی زیر را در نظر بگیرید:

$$\begin{aligned} \min \quad & f(x) = x_1^2 + 2x_1^2 + 2x_1x_2 - 10x_1 - 12x_2 \\ \text{s.t.} \quad & \begin{cases} g_1(x) = x_1 + 3x_2 - 8 \leq 0, \\ g_2(x) = x_1^2 + x_2^2 + 2x_1 - 2x_2 - 3 \leq 0. \end{cases} \end{aligned}$$

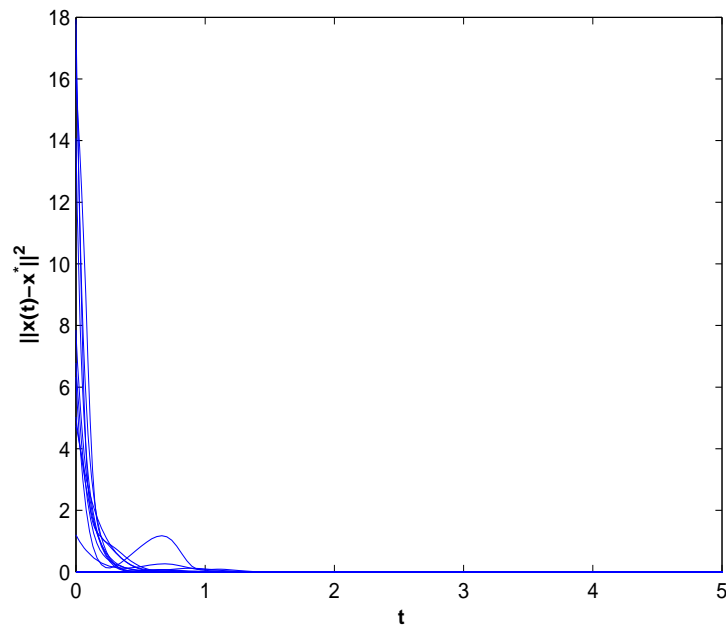
جواب بهینه این مسأله $x^* = (1, 2)^T$ است. از شبکه عصبی (۷.۲) برای حل این مسئله استفاده می‌کنیم. نتایج شبیه‌سازی نشان می‌دهد که شبکه عصبی (۷.۲) با هر نقطه شروع همواره به $w^* = (x^{*T}, u^{*T})^T$ همگراست. برای مثال شکل ۱.۲ رفتارهای منحنی $x(t)$ را براساس شبکه عصبی (۷.۲) با ۱۰ نقطه شروع تصادفی نشان می‌دهد. شکل ۲.۲ خطای نرم l_2 را بین x و x^* نشان می‌دهد.

مثال ۲.۴.۲ ([۵۰]). مسأله بهینه‌سازی غیرخطی زیر را در نظر بگیرید:

$$\begin{aligned} \min \quad & f(x) = x_1^4 + x_2^4 + 2x_1x_2 + (x_1 - 1)^4 + (x_2 - 3)^4 \\ \text{s.t.} \quad & \begin{cases} g_1(x) = x_1^2 + x_2^2 - 64 \leq 0, \\ g_2(x) = (x_1 + 3)^2 + (x_2 + 4)^2 - 36 \leq 0, \\ g_3(x) = (x_1 - 3)^2 + (x_2 + 4)^2 - 36 \leq 0, \\ x_1 \geq 0, x_2 \geq 0. \end{cases} \end{aligned}$$



شکل ۱.۲: رفتارهای منحنی $x(t)$ در شبکه عصبی (۷.۲) با ۱۰ نقطه شروع مختلف در مثال ۱.۴.۲.

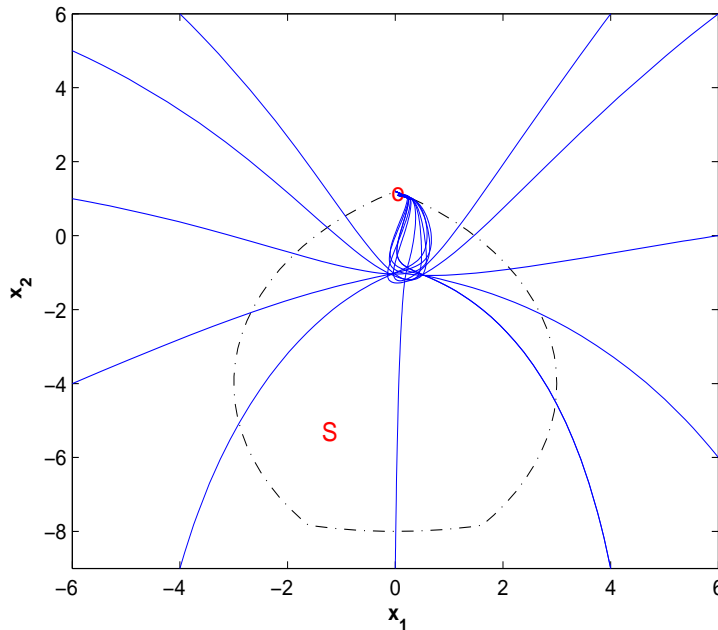


شکل ۲.۲: رفتار همگرایی $\|x(t) - x^*\|_2$ در مثال ۱.۴.۲.

این مسأله فقط یک جواب $x^* = (0, 1/196)^T$ دارد. قضیه ۳.۳.۲ تضمین می‌کند که شبکه عصبی (۷.۲) به x^* همگرایی سراسری است. شکل ۳.۲ نمودار فزاینده‌های حالت $(x_1(t), x_2(t))^T$ را با ۱۱ نقطه شروع مختلف نشان می‌دهد. هم‌چنین S ناحیه شدنی را مشخص می‌کند. این بردارها به جواب دقیق x^* همگرایی سراسری هستند. در مقایسه، مسئله بالا را با شبکه عصبی کندی-چاوداده شده با

$$\frac{dx}{dt} = -(\nabla f(x) + s[\nabla g(x)g^+(x) + \nabla h(x)h(x)])$$

باهفت نقطه شروع مختلف و پارامتر جریمه $s = 5000$ در [۵۰] حل شده است. از حل آن نتیجه می‌گیریم که شبکه عصبی کندی-چاوبه x^* همگرای سراسری نیست.



شکل ۳.۲: نمودار فاز شبکه عصبی (۷.۲) با ۱۱ نقطه شروع مختلف در مثال ۲.۴.۲.

مثال ۳.۴.۲ [۵۱] مسأله بهینه‌سازی غیرخطی زیر را در نظر بگیرید:

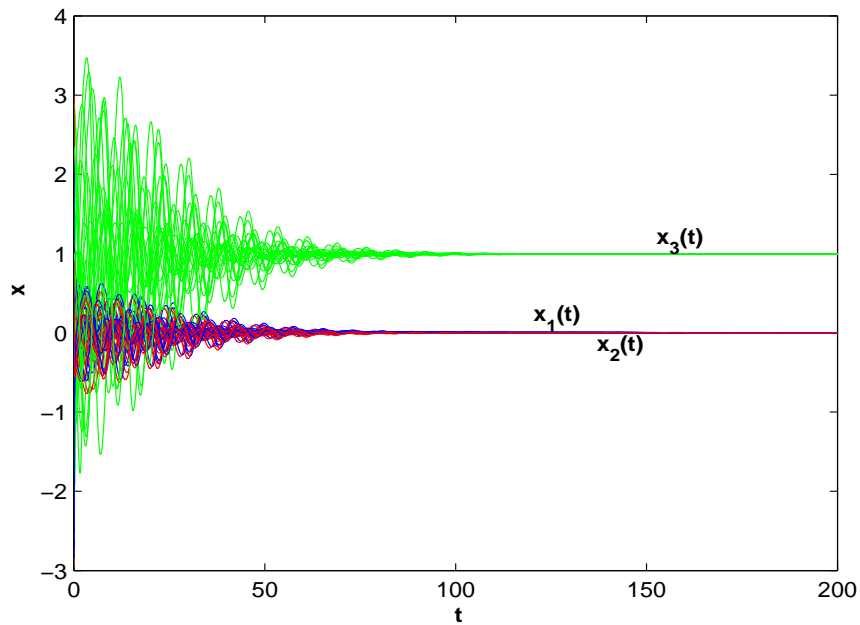
$$\begin{aligned} \min \quad & f(x) = (x_1 + 3x_2 + x_3)^2 + 4(x_1 - x_2)^2 \\ \text{s.t.} \quad & \begin{cases} g(x) = x_1^3 - 6x_2 - 4x_3 - 3 \leq 0, \\ h(x) = x_1 + x_2 + x_3 - 1 = 0, \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0. \end{cases} \end{aligned}$$

جواب بهینه این مسأله $x^* = (0, 0, 1)^T$ است. از شبکه عصبی (۷.۲) برای حل این مسأله استفاده می‌کنیم. شکل ۴.۲ رفتارهای منحنی $x(t)$ را براساس شبکه عصبی (۷.۲) با ۳۰ نقطه شروع تصادفی مختلف نشان می‌دهد.

مثال ۴.۴.۲ ([۵۲]). مسأله بهینه‌سازی غیرخطی زیر را در نظر بگیرید:

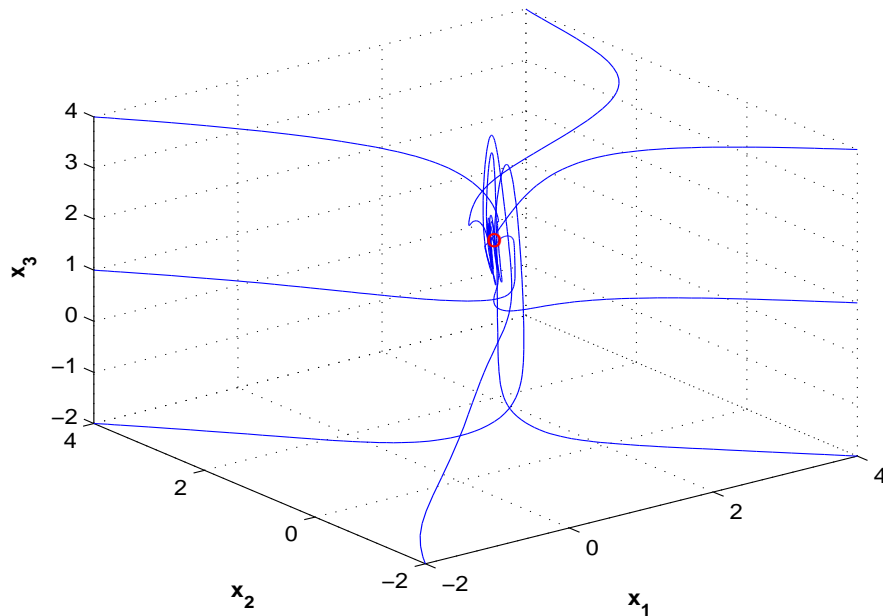
$$\begin{aligned} \min \quad & \frac{1}{4} [(x_1 - x_2)^4 + (x_2 + x_3)^2 + (x_1 + x_3)^2] \\ \text{s.t.} \quad & \begin{cases} x_1^2 + x_2^4 - x_3 \leq 0, \\ (2 - x_1)^2 + (2 - x_2)^2 - x_3 \leq 0, \\ 2e^{-x_1 + x_2} - x_3 \leq 0, \\ x_1^2 + x_2^2 - 2x_1 + x_2 - 4 \leq 0, \\ |x_1| \leq 2, |x_2| \leq 2, x_3 \geq 0. \end{cases} \end{aligned}$$

جواب بهینه این مسأله $x^* = (1/109, 0/922, 1/954)^T$ است. از شبکه عصبی (۷.۲) برای حل این مسأله استفاده می‌کنیم. نتایج شبیه‌سازی نشان می‌دهد که شبکه عصبی (۷.۲) با هر نقطه شروع



شکل ۴.۲: رفتارهای منحنی $x(t)$ در شبکه عصبی (۷.۲) با ۳۰ نقطه شروع مختلف در مثال ۳.۴.۲.

همواره به $w^* = (x^{*T}, u^{*T})^T$ همگراست. شکل ۵.۲ نمودار فاز شبکه عصبی را با ۸ نقطه شروع مختلف نشان می‌دهد.



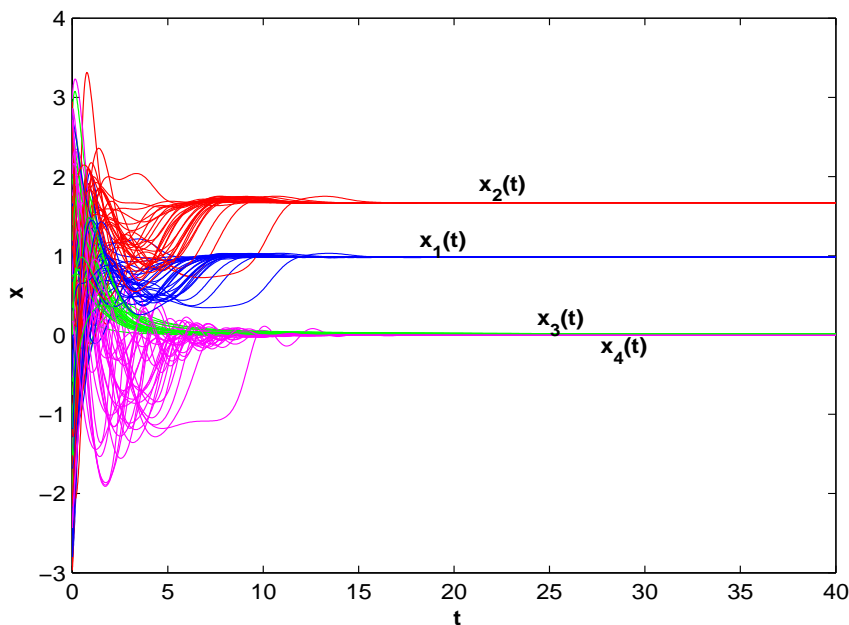
شکل ۵.۲: نمودار فاز شبکه عصبی (۷.۲) با ۸ نقطه شروع مختلف در مثال ۴.۴.۲.

مثال ۵.۴.۲ ([۴]). مسأله بهینه‌سازی غیرخطی زیر را در نظر بگیرید:

$$\min \quad 0.4x_1 + x_1^2 + x_2^2 - x_1x_2 + 0.5x_3^2 + 0.5x_4^2 + \frac{1}{3}x_1^3$$

$$\text{s.t.} \quad \begin{cases} -x_1 + x_2 - x_3 \leq 2, \\ 3x_1 + x_2 - x_3 - x_4 \leq 18, \\ \frac{1}{3}x_1 + x_2 - x_4 = 2, \\ x \geq 0. \end{cases}$$

جواب بهینه این مسأله $x^* = (0.982, 1.672, 0, 0)^T$ است. از شبکه عصبی (۷.۲) برای حل این مسأله استفاده می‌کنیم. همه مسیرهای شبکه به x^* همگرا هستند. علاوه بر این هنگامی که نقطه شروع به عنوان یک نقطه شدنی انتخاب می‌شود، مسیر شبکه باز هم به x^* همگراست. شکل ۶.۲ رفتارهای منحنی $x(t)$ را بر اساس شبکه عصبی (۷.۲) با ۳۰ نقطه شروع تصادفی مختلف نشان می‌دهد.



شکل ۶.۲: رفتارهای منحنی $x(t)$ در شبکه عصبی (۷.۲) با ۳۰ نقطه شروع مختلف در مثال ۵.۴.۲.

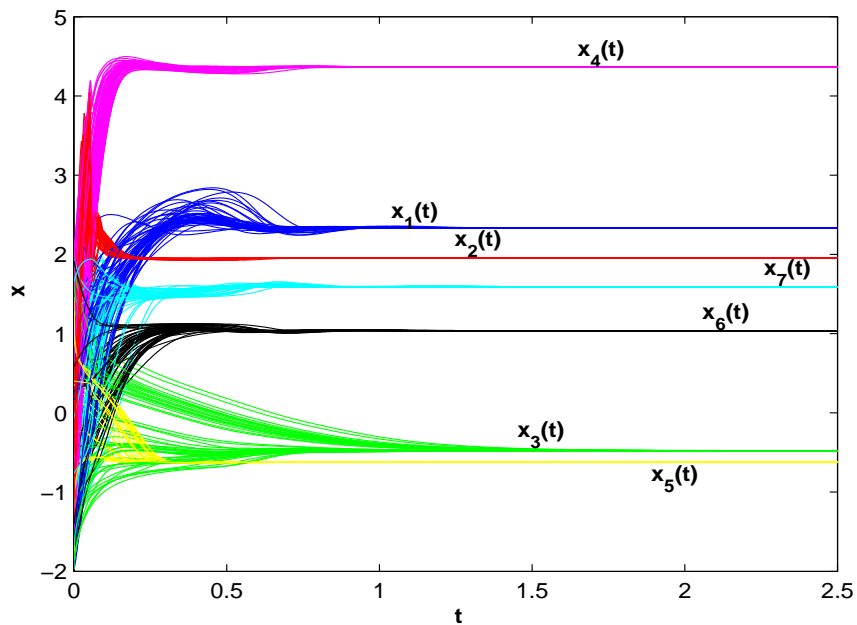
مثال ۶.۴.۲ ([۴۸]). مسأله بهینه‌سازی غیرخطی زیر را در نظر بگیرید:

$$\min \quad (x_1 - 10)^2 + 5(x_2 - 12)^2 + x_3^4 + 3(x_4 - 11)^2 + 10x_5^6 + 7x_6^2 + x_7^4 - 4x_6x_7 - 10x_6 - 8x_7$$

$$\text{s.t.} \quad \begin{cases} 2x_1^2 + 3x_2^4 + x_3 + 4x_4^2 + 5x_5 - 127 \leq 0, \\ 7x_1 + 3x_2 + 10x_3^2 + x_4 - x_5 - 282 \leq 0, \\ 23x_1 + x_2^2 + 6x_6^2 - 8x_7 - 196 \leq 0, \\ 4x_1^2 + x_2^2 + 2x_3^2 - 3x_1x_2 + 5x_6 - 11x_7 \leq 0. \end{cases}$$

جواب بهینه این مسأله $x^* = (2.330, 1.951, -0.4775, 4.366, -0.625, 1.038, 1.594)^T$ است. از شبکه عصبی (۷.۲) برای حل این مسأله استفاده می‌کنیم. نتایج شبیه‌سازی نشان می‌دهد

که شبکه عصبی (۷.۲) با هر نقطه شروع همواره به $w^* = (x^{*T}, u^{*T})^T$ همگراست. شکل ۷.۲ رفتارهای منحنی $x(t)$ را بر اساس شبکه عصبی (۷.۲) با ۵۰ نقطه شروع تصادفی مختلف نشان می‌دهد.



شکل ۷.۲: رفتارهای منحنی $x(t)$ در شبکه عصبی (۷.۲) با ۵۰ نقطه شروع مختلف در مثال ۶.۴.۲.

فصل ۳

استفاده از یک تابع شایسته برای حل رده ای از مسائل بهینه سازی غیرخطی محدب

۱.۳ بیان مسئله

مسئله برنامه ریزی غیرخطی محدب زیر را در نظر بگیرید:

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & g(x) \leq 0, \\ & h(x) = 0, \end{aligned} \tag{۱.۳}$$

که $f: \mathbb{R}^n \rightarrow \mathbb{R}$ ، $x \in \mathbb{R}^n$ و $g(x) = (g_1(x), g_2(x), \dots, g_m(x))^T$ یک تابع پیوسته برداری m بعدی با n متغیر می باشد. توابع f, g_1, \dots, g_m محدب و دوبار مشتق پذیر فرض شده اند و $h(x) = Ax - b$ ، $A \in \mathbb{R}^{l \times n}$ و $b \in \mathbb{R}^l$.

فرض ۱.۱.۳. در طول این فصل مفروضات زیر را در نظر می گیریم:

- مسئله (۱.۳) یک جواب بهینه منحصر به فرد دارد و در شرایط اسلاتر صدق می کند. یعنی یک $x^* \in \mathbb{R}^n$ وجود دارد به طوریکه

$$g(x^*) < 0, \quad Ax^* - b = 0.$$

- گرادیان $\{\nabla g_k(x) | k = 1, \dots, m\} \cup \{\nabla h_p(x) | p = 1, \dots, l\}$ مستقل خطی هستند.
- ماتریس $(\nabla f(x) + \sum_{k=1}^m u_k \nabla g_k(x))$ روی فضای پوچ گرادیان های $\{\nabla h_p(x) | p = 1, \dots, l\}$ ماتریس معین مثبت است.

به خوبی می دانیم که سه تایی $(x^{*T}, u^{*T}, v^{*T})^T \in \mathbb{R}^{n+m+l}$ یک جواب بهینه (۱.۳) است اگر و فقط اگر $(x^{*T}, u^{*T}, v^{*T})^T$ در شرایط KKT زیر صدق کند

$$\begin{cases} u^* \geq 0, & g(x^*) \leq 0, & u^{*T}g(x^*) = 0, \\ \nabla f(x^*) + \nabla h(x^*)^T v^* + \nabla g(x^*)^T u^* = 0, \\ h(x^*) = 0. \end{cases} \quad (2.3)$$

لم ۲.۱.۳. اگر A یک ماتریس غیرمنفرد $n \times n$ باشد آنگاه دستگاه $AX = 0$ فقط جواب بدیهی $X = 0$ دارد.

۲.۳ فرمول بندی و مدل گرادیان

می توانیم بین جواب مسئله (۱.۳) و جواب یک مسئله مینیم سازی نامقید هم ارز توسط یک تابع شایسته ارتباط ایجاد کنیم. تابع شایسته تابعی است که مینیم سراسری اش با جوابهای NCP مطابقت می کند. رده توابع NCP که در زیر تعریف می شود برای ساختن یک تابع شایسته استفاده می شود.

تعریف ۱.۲.۳. تابع $\phi: \mathbb{R}^2 \rightarrow \mathbb{R}$ یک تابع NCP نامیده می شود اگر در شرایط زیر صدق کند:

$$\phi(a, b) = 0 \iff a \geq 0, b \geq 0, ab = 0.$$

یک تابع NCP متداول، تابع فیشر برمیستر^۱ است که قویاً نیمه هموار است و به صورت زیر تعریف می شود:

$$\phi_{\text{FB}}(a, b) = (a + b) - \sqrt{a^2 + b^2}.$$

تابع شایسته $\psi_{\text{FB}}: \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}_+$ به وسیله مجذور ϕ_{FB} به دست می آید یعنی

$$\psi_{\text{FB}}(a, b) = \frac{1}{4} \|\phi_{\text{FB}}(a, b)\|^2.$$

تابع فیشر برمیستر آشفته نیز به صورت زیر داده شده است:

$$\phi_{\text{FB}}^\varepsilon(a, b) = (a + b) - \sqrt{a^2 + b^2} + \varepsilon, \quad \varepsilon \rightarrow 0_+.$$

گزاره ۲.۲.۳. برای هر $\varepsilon \in \mathbb{R}$ داریم:

$$\phi_{\text{FB}}^\varepsilon(a, b) = 0 \iff a > 0, b > 0, ab = \frac{\varepsilon}{4}.$$

گزاره بالا بدیهی است و برهان حذف شده است. قابل توجه است که $\phi_{\text{FB}}^\varepsilon(a, b)$ با توجه به a, b برای هر $\varepsilon > 0$ همواری باشد.

^۱Fisher Burmister

لم ۳.۲.۳. $y^* = (x^{*T}, u^{*T}, v^{*T})^T$ در معادله زیر صدق می کند

$$\eta(y) = \begin{bmatrix} \nabla f(x) + \nabla h(x)^T v + \nabla g(x)^T u \\ -h(x) \\ \phi_{\text{FB}}^\varepsilon(u, -g(x)) \end{bmatrix} = \circ, \quad (۳.۳)$$

اگر فقط اگر x^* برای هر $\varepsilon \rightarrow \circ_+$ یک نقطه KKT برای (۱.۳) باشد.

برهان. فرض کنید $\eta(y^*) = \circ$ باشد، آنگاه:

$$\nabla f(x^*) + \nabla h(x^*)^T v^* + \nabla g(x^*)^T u^* = \circ, \quad (۴.۳)$$

$$h(x^*) = \circ, \quad (۵.۳)$$

$$\phi_{\text{FB}}^\varepsilon(u^*, -g(x^*)) = \circ, \quad \text{for } \varepsilon \rightarrow \circ_+. \quad (۶.۳)$$

از گزاره ۲.۲.۳ به آسانی درمی یابیم که $\phi_{\text{FB}}^\varepsilon(u^*, -g(x^*)) = \circ$ اگر فقط اگر

$$u^* \geq \circ, \quad g(x^*) \leq \circ, \quad u^{*T} g(x^*) = \circ, \quad \text{for } \varepsilon \rightarrow \circ_+. \quad (۷.۳)$$

از (۴.۳)، (۵.۳) و (۷.۳) درمی یابیم که $y^* = (x^{*T}, u^{*T}, v^{*T})^T$ در شرایط KKT صدق می کند. عکس این قضیه بدیهی است.

□

اکنون با استفاده از تعریف ۱.۲.۳ و لم ۳.۲.۳ به راحتی درمی یابیم که جواب سیستم (۲.۳) معادل مسئله مینیم سازی هموار نامقید زیر است:

$$\min E(y) = \frac{1}{\gamma} \|\eta(y)\|_\gamma^2. \quad (۸.۳)$$

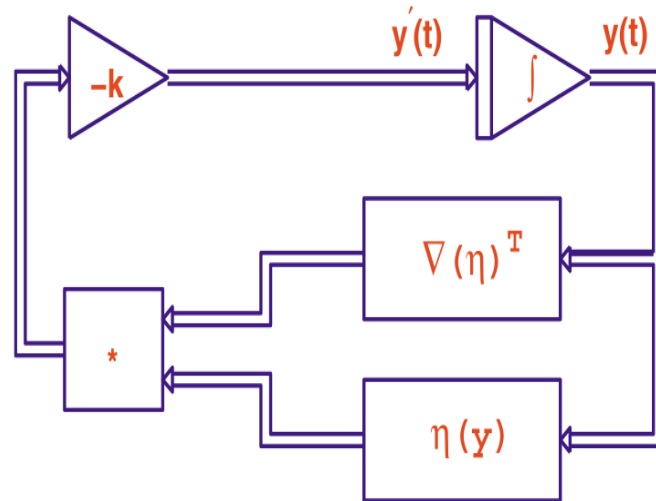
روشن است که $E(y)$ یک تابع شایسته هموار برای سیستم (۲.۳) است. تابع شایسته E در (۸.۳) برای هر $y \in \mathbb{R}^{m+n+l}$ به طور پیوسته مشتق پذیر است.

اکنون فرض کنید $x(\cdot)$ ، $u(\cdot)$ ، $v(\cdot)$ متغیرهای وابسته به زمان باشند. هدف ما طراحی یک شبکه عصبی است که یک نقطه تعادل رامعین می کند که همچنین یک نقطه ایستای تابع انرژی $E(y)$ می باشد. از این رومی توانیم روش تندترین کاهش را برای ساخت مدل شبکه عصبی زیر برای حل مسئله (۱.۳) به صورت زیر استفاده کنیم:

$$\frac{dy(t)}{dt} = -k \nabla E(y(t)), \quad k > \circ, \quad (۹.۳)$$

$$y(\circ) = y_\circ, \quad (۱۰.۳)$$

که k عامل مقیاس است و سرعت همگرایی شبکه عصبی (۹.۳) و (۱۰.۳) را نشان می دهد. اینکه چطور شبکه عصبی (۹.۳) و (۱۰.۳) روی سخت افزار قابل اجراست در شکل ۱.۳ نشان داده شده است.



شکل ۱.۳: یک نمودار بلوکی ساده برای شبکه عصبی (۹.۳) و (۱۰.۳)

۳.۳ تجزیه و تحلیل پایداری

در این بخش همگرایی مسیر شبکه عصبی (۹.۳) و (۱۰.۳) را بررسی می کنیم. ابتدا همگرایی را اثبات می کنیم.

لم ۱.۳.۳. ماتریس ژاکوبین $\nabla\eta(y)$ از نگاشت تعریف شده η در (۳.۳) غیرمنفرد است.

برهان. بایک محاسبه ساده و با استفاده از گزاره ۳.۲ در [۵۵] و گزاره ۳.۳ در [۵۶] به وضوح داریم:

$$\nabla\eta(y) = \begin{bmatrix} (\nabla^2 f(x) + \sum_{k=1}^m u_k \nabla^2 g_k(x)) & \nabla h(x)^T & \nabla g(x)^T \\ -\nabla h(x) & O_{l \times l} & O_{l \times m} \\ (-\text{diag}\{\nabla_{g_k(x)} \phi_{\text{FB}}^{\varepsilon}(u_k, -g_k(x))\}_{k=1}^m) \nabla g(x) & O_{m \times l} & \text{diag}\{\nabla_{u_k} \phi_{\text{FB}}^{\varepsilon}(u_k, -g_k(x))\}_{k=1}^m \end{bmatrix}$$

از فرض ۱.۱.۳ و قضیه ۳.۱ در [۵۶] درمی یابیم که $\nabla\eta(y)$ غیرمنفرد است و این برهان را کامل می

کند. \square

اکنون ارتباط بین نقطه تعادل (۹.۳) و (۱۰.۳) را با جواب مسئله (۱.۳) بررسی می کنیم.

قضیه ۲.۳.۳. فرض کنید x^* نقطه KKT (۱.۳) باشد آنگاه y^* نقطه تعادل شبکه عصبی (۹.۳) و (۱۰.۳) است. از سوی دیگر اگر $y^* = (x^{*T}, u^{*T}, v^{*T})^T$ نقطه تعادل شبکه عصبی (۹.۳) و (۱۰.۳) باشد و ماتریس ژاکوبین $\eta(y)$ در (۳.۳) غیرمنفرد باشد آنگاه x^* نقطه KKT مسئله (۱.۳) می باشد.

برهان. فرض کنید x^* جواب بهینه (۱.۳) باشد. ازلم ۳.۲.۳ به روشنی درمی یابیم که $\eta(y^*) = 0$. بایک محاسبه ساده به وضوح داریم:

$$\nabla E(y) = (\nabla\eta(y))^T \eta(y), \quad (11.3)$$

که $\nabla \eta(y)$ ماتریس ژاکوبین $\eta(y)$ است. با استفاده از (۱۱.۳) داریم $\nabla E(y^*) = 0$ یعنی y^* نقطه تعادل مدل دینامیکی (۹.۳) و (۱۰.۳) است. با استفاده از (۱.۳.۳) و (۲.۱.۳) عکس قضیه بدیهی است. □
 لم ۳.۳.۳. نقطه تعادل مدل شبکه عصبی ارائه شده (۹.۳) و (۱۰.۳) منحصر به فرد است.

برهان. چون مسئله (۱.۳) دارای جواب بهینه منحصر به فرد x^* ، است شرایط KKT (۲.۳) برای داشتن جواب منحصر به فرد $y^* = (x^{*T}, u^{*T}, v^{*T})^T$ لازم و کافی است. علاوه بر آن از قضیه ۲.۳.۳ درمی یابیم که جواب سیستم (۲.۳)، نقطه تعادل مدل شبکه عصبی ارائه شده (۹.۳) و (۱۰.۳) است. بنابراین نقطه تعادل شبکه (۹.۳) و (۱۰.۳) منحصر به فرد است. □

قضیه ۴.۳.۳. فرض کنید y^* یک نقطه تعادل تنها برای (۹.۳) و (۱۰.۳) باشد آنگاه y^* برای (۹.۳) و (۱۰.۳) پایدارمجانبی است.

برهان. ابتدا ملاحظه می کنیم که $E(y) \geq 0$ و $E(y^*) = 0$. علاوه بر این چون y^* یک نقطه تعادل تنها برای (۹.۳) و (۱۰.۳) است پس یک همسایگی $\Omega_* \subseteq \mathbb{R}^{n+m+l}$ برای y^* وجود دارد به طوری که:
 $\nabla E(y^*) = 0$, and $\nabla E(y) \neq 0, \forall y \in \Omega_* \setminus \{y^*\}$.

ادعای کنیم که برای هر $y \in \Omega_* \setminus \{y^*\}$ داریم $E(y) > 0$. در غیر این صورت اگر $y \in \Omega_* \setminus \{y^*\}$ وجود داشته باشد به طوری که $E(y) = 0$ شود آنگاه با استفاده از (۸.۳) و (۱۱.۳) داریم $\nabla E(y) = 0$. یعنی y یک نقطه تعادل (۱۰.۳) می باشد که به وضوح با این فرض که y یک نقطه تعادل تنها در Ω_* است در تناقض است. علاوه بر این

$$\frac{dE(y(t))}{dt} = [\nabla E(y(t))]^T \frac{dy(t)}{dt} = -k \|\nabla E(y(t))\|^2 \leq 0, \quad (12.3)$$

و

$$\frac{dE(y(t))}{dt} < 0, \quad \forall y(t) \in \Omega_* \text{ and } y(t) \neq y^*.$$

با استفاده از (۲.۳) در [۶] درمی یابیم که y^* پایدارمجانبی است. □

قضیه ۵.۳.۳. فرض کنید $y = y(t, y_0)$ با نقطه شروع $y_0 = y(0, y_0)$ یک مسیر برای (۹.۳) و (۱۰.۳) باشد و مجموعه سطح $L(y_0) = \{y \in \mathbb{R}^{n+m+l} : E(y) \leq E(y_0)\}$ کراندار باشد آنگاه:
 (a) $\gamma^+(y_0) = \{y(t, y_0) | t \geq 0\}$ کراندار است.
 (b) وجود دارد \bar{y} به طوری که $\lim_{t \rightarrow \infty} y(t, y_0) = \bar{y}$.

برهان. (a) فرض کنید y^* یک نقطه تعادل شبکه (۹.۳) و (۱۰.۳) باشد آنگاه می گوئیم $\nabla E(y^*) = 0$. با محاسبه مشتق $E(y)$ در طول مسیر $(t, y_0), (t \geq 0)$ داریم:

$$\frac{dE(y(t))}{dt} = [\nabla E(y(t))]^T \frac{dy}{dt} = -k \|\nabla E(y(t))\|^2 \leq 0. \quad (13.3)$$

بنابراین در طول مسیر $(t \geq 0)$ ، $y = y(t, y_0)$ ، $E(y)$ به طور یکنواخت غیر صعودی است. بنابراین $\gamma^+(y_0) \subseteq L(y_0)$ است یعنی $\gamma^+(y_0) = \{y(t, y_0) | t \geq 0\}$ کراندار است.

(b) با استفاده از (a)، $\gamma^+(y_0) = \{y(t, y_0) | t \geq 0\}$ مجموعه ای کراندار از نقاط است. در مسیر مستقیم دنباله صعودی یکنواخت $\{\bar{t}_n\}$ ، $0 \leq \bar{t}_1 \leq \dots \leq \bar{t}_n \leq \dots$ ، $\bar{t}_n \rightarrow +\infty$ یک زیر دنباله کراندار ساخته شده از تعداد زیادی نقاط نامتناهی می باشد. بنابراین نقطه حدی \bar{y} وجود دارد یعنی اینکه یک زیر دنباله $\{t_n\} \subseteq \{\bar{t}_n\}$ ، $t_n \rightarrow +\infty$ وجود دارد به طوریکه:

$$\lim_{n \rightarrow +\infty} y(t_n, y_0) = \bar{y},$$

که \bar{y} در رابطه زیر صدق می کند

$$\frac{dE(y(t))}{dt} = 0,$$

که نشان می دهد \bar{y} نقطه ω -limit است. با استفاده از قضیه مجموعه تغییرناپذیر لسل داریم:

$$y(t, y_0) \rightarrow \bar{y} \in M \quad \text{as } t \rightarrow \infty,$$

که M بزرگترین مجموعه تغییرناپذیر در $\{y(t, y_0) | \frac{dE(y(t, y_0))}{dt} = 0\}$ است. از (۹.۳)، (۱۰.۳) و (۱۳.۳) نتیجه می شود که $\frac{dx}{dt} = 0$ ، $\frac{du}{dt} = 0$ و $\frac{dv}{dt} = 0 \Leftrightarrow \frac{dE(y(t))}{dt} = 0$. بنابراین با استفاده از $M \subseteq K \subseteq D^*$ داریم $\bar{y} \in D^*$. بنابراین از هر y_0 حالت شروع، مسیر $y(t, y_0)$ از (۹.۳) و (۱۰.۳) به \bar{y} میل می کند و این برهان را کامل می کند.

□

نتیجه ۶.۳.۳. اگر $D^* = \{(x^{*T}, u^{*T}, v^{*T})^T\}$ آنگاه شبکه عصبی (۹.۳) و (۱۰.۳) برای حل (۱.۳) به نقطه تعادل منحصر به فرد $y^* = (x^{*T}, u^{*T}, v^{*T})^T$ پایدار مجانبی سراسری است که D^* به عنوان مجموعه نقاط بهینه (۱.۳) تعریف می شود.

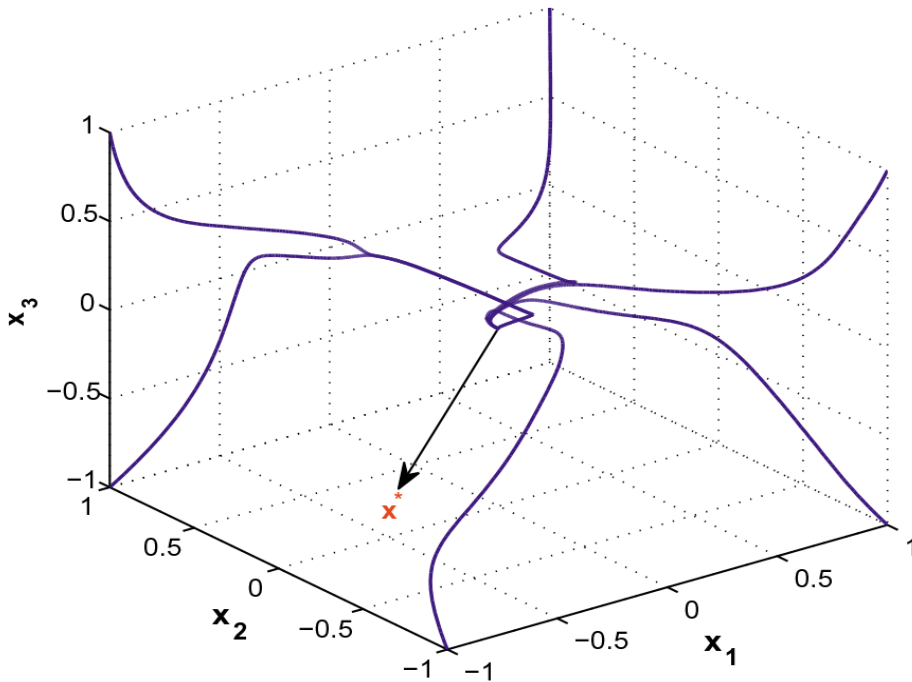
۴.۳ مثال های عددی

مثال ۱.۴.۳ ([۲۴]). مسأله بهینه سازی غیرخطی زیر را در نظر بگیرید:

$$\begin{aligned} \min \quad & x_1^2 + 2x_2^2 + e^{2x_3} \\ \text{s.t.} \quad & \begin{cases} x_1 x_2 + x_2 - 1 \leq 0, \\ x_2 + x_3^2 - 2 \leq 0, \\ x_3 \geq 0. \end{cases} \end{aligned}$$

جواب بهینه این مسئله و دوگان آن به ترتیب $x^* = (0, 0, 0)^T$ و $u^* = (0, 0, 0, 0, 1)^T$ می باشند. قضایای ۴.۳.۳ و ۵.۳.۳ و نتیجه ۶.۳.۳ تضمین می دهند که مدل بیان شده در (۹.۳) و (۱۰.۳) به x^* همگرای سراسری است. شکل ۲.۳ نمودار فاز متغیرهای حالت $(x_1(t), x_2(t), x_3(t))^T$ را بآه نقطه

شروع مختلف و عامل مقیاس $k = 100$ نشان می دهد. تاثیر پارامتر k را روی مقدار $\|y(t) - y^*\|^2$ امتحان می کنیم. از شکل ۳.۳ درمی یابیم که شبکه عصبی (۹.۳) و (۱۰.۳) بانقطه شروع $y_0 = (1, -1, 1, -1, 1, -1, 1, -1)^T$ وقتی $k = 0.1$ است کمترین کاهش و وقتی $k = 10$ است سریع ترین کاهش را برای $\|y(t) - y^*\|^2$ ایجاد می کند. شکل ۴.۳ نیز رفتار همگرایی $\|y(t) - y^*\|^2$ را با $k = 10$ حالت شروع مختلف و $k = 20$ شرح می دهد.

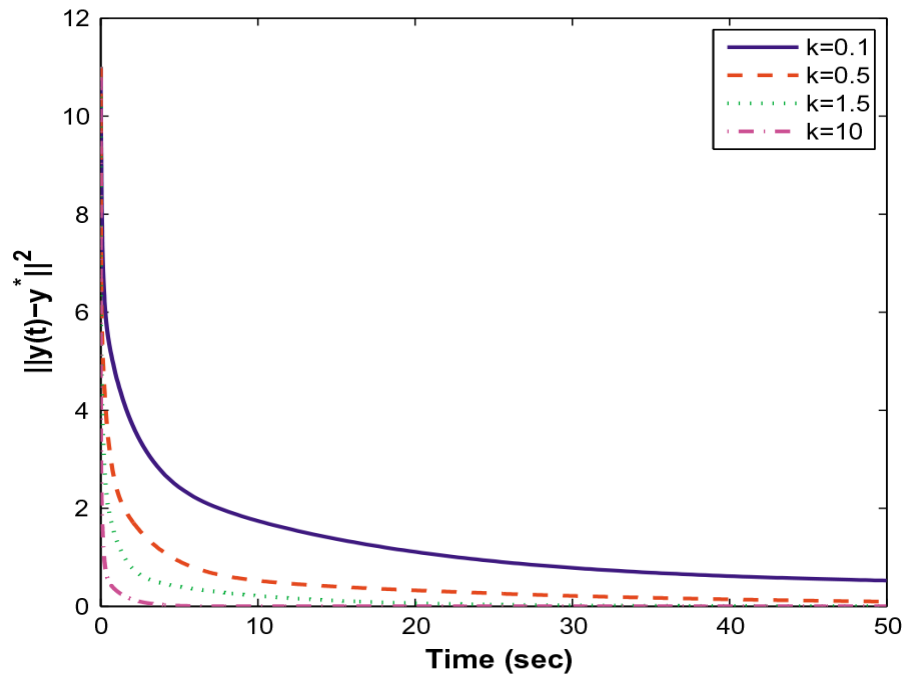


شکل ۲.۳: نمودار فاز شبکه عصبی (۹.۳) و (۱۰.۳) با ۶ نقطه شروع مختلف و $k = 100$ در مثال ۱.۴.۳.

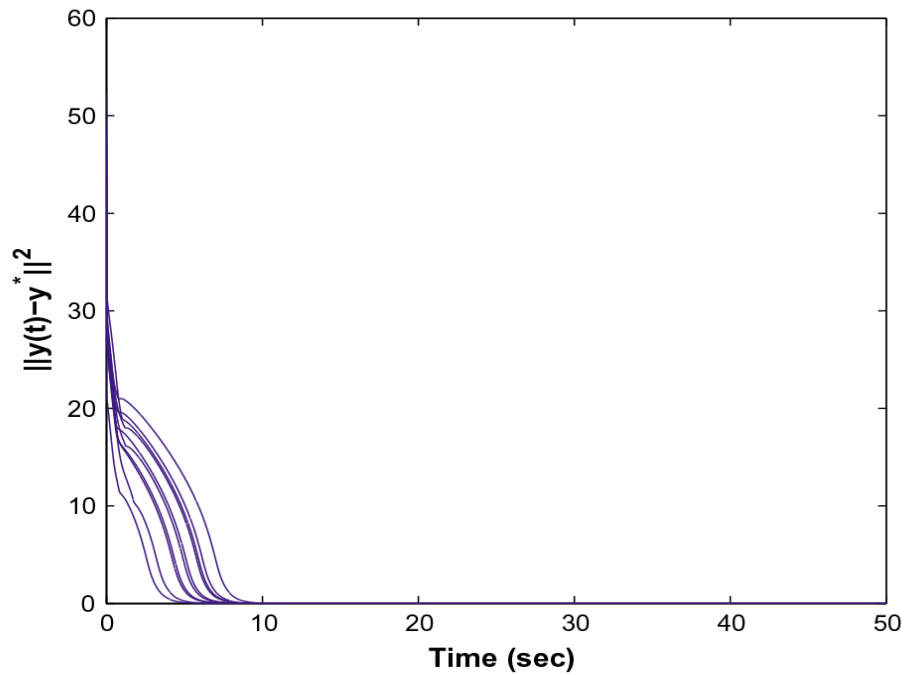
مثال ۲.۴.۳ ([۴۱]). مسأله بهینه سازی غیرخطی زیر را در نظر بگیرید:

$$\begin{aligned} \min \quad & -2x_1 - x_2 \\ \text{s.t.} \quad & \begin{cases} x_1 + x_2 - 1 \leq 0, \\ x_1 + 2x_2 - 4 \leq 0. \end{cases} \end{aligned}$$

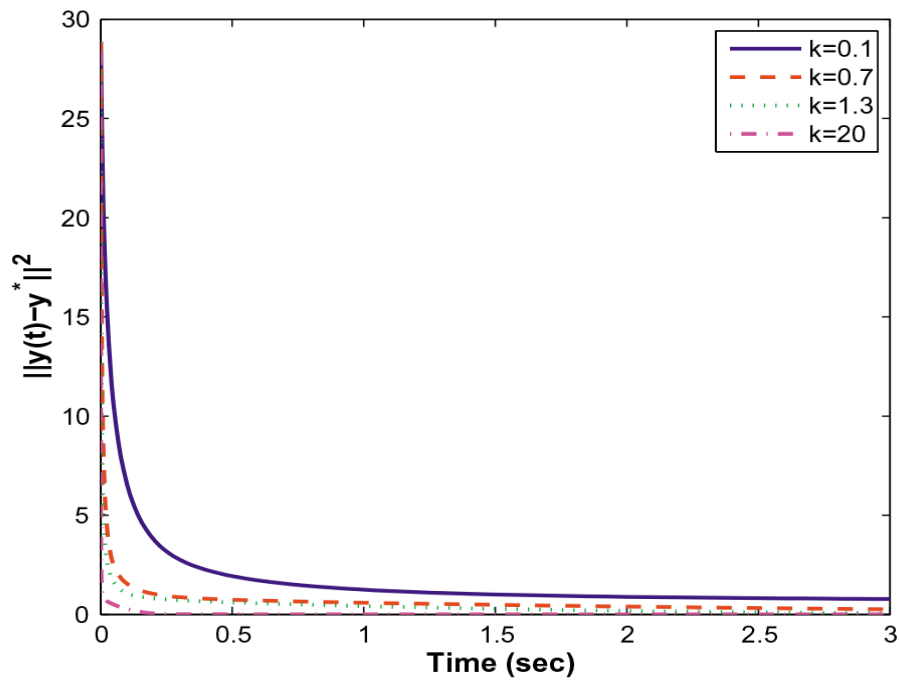
این مسئله و دوگان آن فقط یک جواب $y^* = (0.8944, 0.4472, 1.118, 0)^T$ دارند. شکل ۵.۳ تغییرات $\|y(t) - y^*\|^2$ را با k و $y_0 = (3, -3, 3, -3)^T$ حالت شروع شرح می دهد. همه نتایج شبیه سازی نشان می دهند که مسیر (۹.۳) و (۱۰.۳) با هر نقطه شروع همواره به $y^* = (x^{*T}, u^{*T})^T$ همگراست. برای مثال شکل ۶.۳ رفتار گذرای $y(t)$ را بر اساس (۹.۳) و (۱۰.۳) با ۳۰ نقطه شروع تصادفی و $k = 20$ نشان می دهد. روشن است که تغییر نقاط شروع تاثیر زیادی روی همگرایی مدل شبکه عصبی ندارد.



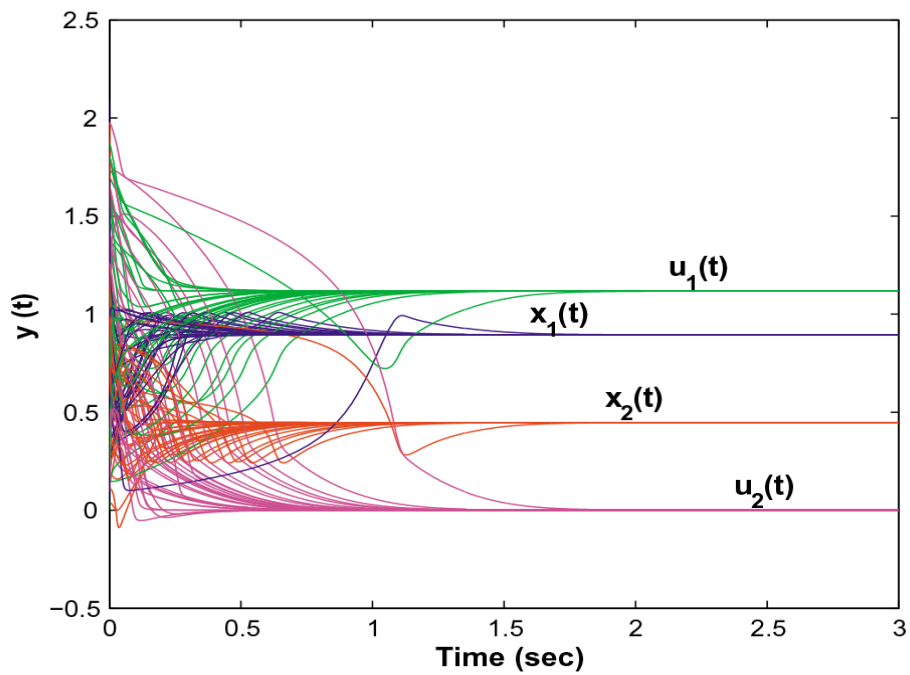
شکل ۳.۳: رفتار همگرایی $\|y(t) - y^*\|^2$ با $y_0 = (1, -1, 1, -1, 1, -1, 1, -1)^T$ در مثال ۱.۴.۳.



شکل ۴.۳: رفتار همگرایی $\|y(t) - y^*\|^2$ با نقطه شروع مختلف و $k = 20$ در مثال ۱.۴.۳.



شکل ۵.۳: رفتار همگرایی $\|y(t) - y^*\|^2$ با $y_0 = (3, -3, 3, -3)^T$ در مثال ۲.۴.۳.



شکل ۶.۳: رفتار گذرای $y(t)$ با ۳۰ نقطه شروع مختلف و $k = 20$ در مثال ۲.۴.۳.

فصل ۴

استفاده از تابع تصویر برای حل مسائل بهینه‌سازی غیرخطی محدب در حالت کلی

۱.۴ مقدمه

مسائل برنامه ریزی محدب اغلب در تحقیقات علمی و کاربردمهندسی به وجود می‌آیند. روشهای عددی سنتی برای حل مسائل برنامه ریزی محدب شامل یک فرآیند تکراری پیچیده زمان محاسباتی هستند که ممکن است استفاده از آنها را در مقیاس بزرگ یا بهینه سازی مسائل زمان حقیقی از جمله در تجزیه و تحلیل رگرسیون، پردازش تصویر و سیگنال، برآورد پارامتر، طراحی فیلتر، کنترل روبات و... محدود کند. به خوبی اثبات شده که شبکه عصبی می‌تواند مسائل بهینه سازی را در زمان واقعی^۱ حل کند. اخیراً مطالعات برای ساخت شبکه های عصبی جدید به منظور حل مسائل بهینه سازی انجام شده است. برخی شبکه های عصبی برای حل بهینه سازی محدب بر اساس روش گرادیان، قضیه دوگان و روش تصویر پیشنهاد شد [۲۰، ۲۳]. کندی و چاو [۳۱] یک شبکه عصبی برای برنامه ریزی غیرخطی مطرح کردند. شبکه شامل یک پارامتر جریمه متناهی است. بنابراین فقط به یک جواب بهینه تقریبی همگراست. چن و همکاران [۱۳] یک شبکه عصبی برای حل مسائل برنامه ریزی غیرخطی محدب بر اساس روش اولیه دوگان مطرح کردند. این مدلی است که می‌تواند مسائل اولیه و دوگان را همزمان حل کند. اما در این حالت تعداد متغیرهای حالت بیشتر می‌شود که باعث بزرگ شدن ابعاد شبکه می‌شود. بر اساس روش تصویر شرایط بهینگی K.K.T برای برنامه ریزی محدب، فریزو همکاران [۲۰]، زیاو وانگ [۵۰]، مدل‌های شبکه عصبی تصویری مطرح کردند. هر چند برای برخی مسائل برنامه ریزی محدب، پایداری شبکه عصبی فریزو تضمین نمی‌شود. شبکه عصبی تصویری جدید ارائه شده در این فصل، شبکه تصویر فریزو را بهبود می‌دهد. پایداری و همگرایی سراسری با استفاده از قضیه پایداری لیاپانوف و اصل تغییرناپذیری لسل ثابت می‌شود. در این فصل در بخش دوم یک مدل شبکه عصبی بر اساس قضیه تصویر و شرایط K.K.T تولید می‌

^۱ Real-Time

کنیم. دربخش سوم پایداری وهمگرایی سراسری ثابت خواهد شد. دربخش چهارم مثالهایی برای نشان دادن کارایی شبکه موردنظر ارائه شده است.

۲.۴ مدل شبکه عصبی

برنامه ریزی غیرخطی محدب زیرابا محدودیت های نامساوی غیرخطی درنظرمی گیریم:

$$\begin{aligned} \min_{x \in \Omega} \quad & f(x) \\ \text{s.t.} \quad & g_j(x) \leq 0, \quad j = 1, 2, \dots, m, \end{aligned} \quad (1.4)$$

که $f(x), g_j(x), (j = 1, 2, \dots, m)$ محدب و به طورپیوسته مشتق پذیرهستند. فرض کنید

$$S = \{x \in \mathbb{R}^n : g_j(x) \leq 0; j = 1, 2, \dots, m\}$$

به خوبی می دانیم $x^* \in S$ جواب بهینه (۱.۴) است اگر و تنها اگر $\lambda^* = (\lambda_1^*, \lambda_2^*, \dots, \lambda_m^*)^T \in \mathbb{R}^m$ وجود داشته باشد به طوری که $(x^{*T}, \lambda^{*T})^T$ در شرایط K.K.T زیر صدق کند:

$$\begin{cases} (x - x^*)^T (\nabla f(x^*) + \nabla g(x^*)^T \lambda^*) \geq 0, & \forall x \in \Omega, \\ \lambda_j^* g_j(x^*) = 0, \quad \lambda_j^* \geq 0, \quad g_j(x^*) \leq 0, & j = 1, \dots, m. \end{cases} \quad (2.4)$$

لم ۱.۲.۴ [۵۰]. x^* جواب بهینه (۱.۴) است اگر و تنها اگر $\lambda^* = (\lambda_1^*, \lambda_2^*, \dots, \lambda_m^*)^T \in \mathbb{R}^m$ وجود داشته باشد به طوری که:

$$\begin{cases} x^* - P_\Omega[x^* - \nabla f(x^*) - \nabla g(x^*)^T \lambda^*] = 0, \\ \lambda^* - [\lambda^* + g(x^*)]^+ = 0. \end{cases} \quad (3.4)$$

که

$$P_\Omega(x_i) = \begin{cases} l_i, & x_i < l_i, \\ x_i, & l_i \leq x_i \leq h_i, \\ h_i, & x_i > h_i. \end{cases} \quad (x_i)^+ = \begin{cases} x_i, & x_i \geq 0, \\ 0, & x_i < 0. \end{cases}$$

لم ۱.۲.۴ نشان می دهد که جواب بهینه (۱.۴) باحل (۳.۴) به دست می آید.

لم ۲.۲.۴ [۳۲]. فرض کنید مجموعه $\Omega \subset \mathbb{R}^n$ یک مجموعه محدب بسته است. آنگاه

$$(v - P_\Omega(v))^T (P_\Omega(v) - u) \geq 0, \quad v \in \mathbb{R}^n, u \in \Omega, \quad (4.4)$$

$$\|P_\Omega(u) - P_\Omega(v)\| \leq \|u - v\|, \quad u, v \in \mathbb{R}^n. \quad (5.4)$$

شبکه عصبی تصویرفریز^۲ برای حل (۱.۴) به صورت زیر بیان می شود:

$$\begin{cases} \frac{dx}{dt} = -(x - P_{\Omega}[x - \nabla f(x) - \nabla g(x)^T \lambda]), \\ \frac{d\lambda}{dt} = -(\lambda - [\lambda + g(x)]^+). \end{cases} \quad (۶.۴)$$

متأسفانه شبکه عصبی (۶.۴) برای برخی مسائل برنامه ریزی محدب ناپایدار است. برای سادگی فرض می کنیم:

$$\begin{aligned} u(t) &= (x^T, \lambda^T)^T \in \mathbb{R}^{n+m}, \\ D &= \Omega \times \mathbb{R}_+^m, \\ \bar{\lambda} &= [\lambda + g(x)]^+, \\ \bar{x} &= P_{\Omega}[x - \nabla f(x) - \nabla g(x)^T \lambda], \end{aligned}$$

و D^* را مجموعه نقاط بهینه (۱.۴) در نظر می گیریم.

بر اساس نتایج فوق مدل شبکه عصبی جدید زیر معرفی می کنیم:

$$\begin{cases} \frac{dx}{dt} = -(x - P_{\Omega}[x - \nabla f(x) - \nabla g(x)^T \bar{\lambda}]), \\ \frac{d\lambda}{dt} = -(\lambda - [\lambda + g(x)]^+). \end{cases} \quad (۷.۴)$$

فرض کنید D^e مجموعه نقاط تعادل شبکه عصبی (۷.۴) باشد. طبق لم ۱.۲.۴، $x \in D^e$ یک نقطه تعادل شبکه عصبی (۷.۴) است اگر و تنها اگر $x \in D^*$ یک جواب بهینه (۱.۴) باشد. در این صورت $D^* = D^e$.

۳.۴ تجزیه و تحلیل پایداری

در این بخش پایداری نقطه تعادل وهمگرایی جواب بهینه شبکه عصبی (۷.۴) را مطالعه می کنیم.

قضیه ۱.۳.۴. برای هر نقطه اولیه $u(t_0) = (x(t_0)^T, \lambda(t_0)^T)^T \in \mathbb{R}^{n+m}$ یک جواب پیوسته منحصر به

فرد $u(t) = (x(t)^T, \lambda(t)^T)^T$ برای سیستم (۷.۴) وجود دارد. علاوه بر این $x(t) \in \Omega$ و $\lambda(t_0) \geq 0$.

برهان. $\nabla f(x)$ و $\nabla g_j(x)$ روی مجموعه محدب باز $D \subseteq \mathbb{R}^{n+m}$ شامل $\Omega \times \mathbb{R}_+^m$ به طور پیوسته مشتق

پذیر هستند. پس $x - P_{\Omega}[x - \nabla f(x) - \nabla g(x)^T \lambda]$ و $\lambda - [\lambda + g(x)]^+$ به طور موضعی پیوسته

لیپ شیتز هستند. طبق قضیه وجود و یکتایی جواب برای معادلات دیفرانسیل معمولی ۱.۶.۳.۱، مسئله

مقدار اولیه سیستم (۷.۴) یک جواب منحصر به فرد دارد.

فرض کنید نقطه اولیه $x_0 = x(t_0) \in \Omega$ و $\lambda_0 = \lambda(t_0) \geq 0$ داده شده است. چون:

$$\begin{cases} \frac{dx}{dt} + x = P_{\Omega}[x - \nabla f(x) - \nabla g(x)^T \bar{\lambda}], \\ \frac{d\lambda}{dt} + \lambda = [\lambda + g(x)]^+. \end{cases} \quad (۸.۴)$$

^۲ Friesz

در نتیجه

$$\begin{cases} \int_{t_0}^t (\frac{dx}{dt} + x) e^t dt = \int_{t_0}^t P_{\Omega}[x - \nabla f(x) - \nabla g(x)^T \bar{\lambda}] e^t dt, \\ \int_{t_0}^t (\frac{d\lambda}{dt} + \lambda) e^t dt = \int_{t_0}^t [\lambda + g(x)]^+ e^t dt. \end{cases} \quad (9.4)$$

آنگاه

$$\begin{cases} x(t) = e^{-(t-t_0)} x_0 + e^{-t} \int_{t_0}^t e^t P_{\Omega}[x - \nabla f(x) - \nabla g(x)^T \bar{\lambda}] dt, \\ \lambda(t) = e^{-(t-t_0)} \lambda_0 + e^{-t} \int_{t_0}^t e^t [\lambda + g(x)]^+ dt. \end{cases} \quad (10.4)$$

با استفاده از قضیه مقدار میانگین برای انتگرال‌ها داریم:

$$\begin{cases} x(t) = e^{-(t-t_0)} x_0 + (1 - e^{-(t-t_0)}) P_{\Omega}[\hat{x} - \nabla f(\hat{x}) - \nabla g(\hat{x})^T \hat{\lambda}], \\ \lambda(t) = e^{-(t-t_0)} \lambda_0 + (1 - e^{-(t-t_0)}) [\hat{\lambda} + g(\hat{x})]^+. \end{cases} \quad (11.4)$$

□ چون $x(t_0) \in \Omega$ و $\lambda(t_0) \geq 0$ در نتیجه $x(t) \in \Omega$ و $\lambda(t) \geq 0$ این اثبات را کامل می‌کند.

ملاحظه ۲.۳.۴. نتایج قضیه ۱.۳.۴ نشان می‌دهد که شبکه عصبی (۷.۴) خوش تعریف است. در اثبات قضیه ۱.۳.۴ فرض کردیم $(x_0^T, \lambda_0^T)^T \in \Omega \times \mathbb{R}_+^m$ یعنی نقطه اولیه باید شدنی باشد. اما اگر $(x_0^T, \lambda_0^T)^T \notin \Omega \times \mathbb{R}_+^m$ وقتی $t \rightarrow +\infty$ $(x(t, x_0)^T, \lambda(t, x_0)^T)^T \in \Omega \times \mathbb{R}_+^m$ می‌شود. بنابراین یک لحظه t_0 وجود دارد به طوری که به ازای $t > t_0$ $(x(t, x_0)^T, \lambda(t, x_0)^T)^T$ در ناحیه شدنی قرار می‌گیرد و $(x(t, x_0)^T, \lambda(t, x_0)^T)^T$ در $\Omega \times \mathbb{R}_+^m$ می‌ماند.

قضیه ۳.۳.۴. فرض کنید $f(x)$ و $g_j(x)$ ، $j = 1, 2, \dots, m$ ، روی مجموعه محدب باز $N \subseteq \mathbb{R}^{n+m}$ شامل $\Omega \times \mathbb{R}_+^m$ محدب و مشتق پذیر باشند. آنگاه شبکه عصبی (۷.۴) به مفهوم لیاپانوف پایدار است و برای هر نقطه اولیه $(x(t_0)^T, \lambda(t_0)^T)^T \in \mathbb{R}^{n+m}$ مسیر جواب (۷.۴) به یک نقطه در D^* همگرا خواهد بود. به خصوص شبکه عصبی (۷.۴) هنگامی که D^* فقط یک نقطه دارد پایدار مجانبی سراسری است.

برهان. با استفاده از قضیه ۱.۳.۴، $\forall (x_0^T, \lambda_0^T)^T \in \Omega \times \mathbb{R}_+^m$ یک جواب پیوسته منحصر به فرد $(x(t)^T, \lambda(t)^T)^T \in \Omega \times \mathbb{R}_+^m$ برای سیستم (۷.۴) وجود دارد. یک تابع لیاپانوف به صورت زیر تعریف می‌کنیم:

$$\begin{aligned} V(x, \lambda) = & f(x) - f(x^*) + \frac{1}{\nu} [(\bar{\lambda})^\nu - (\lambda^*)^\nu] - (x - x^*)^T (\nabla f(x^*) + \nabla g(x^*)^T \lambda^*) \\ & - (\lambda - \lambda^*)^T \lambda^* + \frac{1}{\nu} \|x - x^*\|^\nu + \frac{1}{\nu} \|\lambda - \lambda^*\|^\nu. \end{aligned} \quad (12.4)$$

باتوجه به اینکه $\|\bar{\lambda}\|^2 = \sum_{j=1}^m [(\lambda_j + g_j(x))^+]^2$

$$[(\lambda_j + g_j(x))^+]^2 = \begin{cases} [(\lambda_j + g_j(x))]^2, & \lambda_j + g_j(x) \geq 0, \\ 0, & o.w. \end{cases} \quad (13.4)$$

داریم:

$$\nabla \|\bar{\lambda}\|^2 = \nabla \left(\sum_{j=1}^m [(\lambda_j + g_j(x))^+]^2 \right) = \begin{pmatrix} 2 \nabla g(x)^T \bar{\lambda} \\ 2 \bar{\lambda} \end{pmatrix}. \quad (14.4)$$

بامحاسبه مشتق $V(t)$ در طول مسیر سیستم (۷.۴) با $-\lambda + \bar{\lambda} = g(x) - (\lambda + g(x))^-$ داریم:

$$\begin{aligned} \frac{dV(x, \lambda)}{dt} &= [\nabla f(x) + \nabla g(x)^T \bar{\lambda} - \nabla f(x^*) - \nabla g(x^*)^T \lambda^* + x - x^*]^T (-x + \bar{x}) \\ &\quad + (\bar{\lambda} + \lambda - 2\lambda^*)^T (-\lambda + \bar{\lambda})/2 \\ &= -\|x - \bar{x}\|^2 + [\nabla f(x) + \nabla g(x)^T \bar{\lambda} - \nabla f(x^*) - \nabla g(x^*)^T \lambda^* + \bar{x} - x^*]^T (-x + \bar{x}) \\ &\quad - \frac{1}{2} \|\lambda - \bar{\lambda}\|^2 + (\bar{\lambda} - \lambda^*)^T (-\lambda + \bar{\lambda}) \\ &= -\|x - \bar{x}\|^2 - [\nabla f(x) - \nabla f(x^*)]^T (x - x^*) - (\nabla f(x^*) + \nabla g(x^*)^T \lambda^*)^T (\bar{x} - x^*) \\ &\quad - [x - \nabla f(x) - \nabla g(x)^T \bar{\lambda} - \bar{x}]^T (\bar{x} - x^*) - (\nabla g(x)^T \bar{\lambda} - \nabla g(x^*)^T \lambda^*)^T (\bar{x} - x^*) \\ &\quad - \frac{1}{2} \|\lambda - \bar{\lambda}\|^2 + (\bar{\lambda} - \lambda^*)^T [g(x) - (\lambda + g(x))^-] \\ &= -\|x - \bar{x}\|^2 - \frac{1}{2} \|\lambda - \bar{\lambda}\|^2 - [\nabla f(x) - \nabla f(x^*)]^T (x - x^*) \\ &\quad - [\nabla f(x^*) + \nabla g(x^*)^T \lambda^*]^T (\bar{x} - x^*) - [x - \nabla f(x) - \nabla g(x)^T \bar{\lambda} - \bar{x}]^T (\bar{x} - x^*) \\ &\quad - \bar{\lambda}^T [-\nabla g(x)(x^* - x) - g(x) + g(x^*)] + \bar{\lambda}^T g(x^*) \\ &\quad - \bar{\lambda}(\lambda + g(x))^- + (\lambda^*)^T [\nabla g(x^*)(x - x^*) - g(x) + g(x^*)] \\ &\quad - (\lambda^*)^T g(x^*) + (\lambda^*)^T (\lambda + g(x))^- \\ &= -\|x - \bar{x}\|^2 - \frac{1}{2} \|\lambda - \bar{\lambda}\|^2 - [\nabla f(x) - \nabla f(x^*)]^T (x - x^*) \\ &\quad - [\nabla f(x^*) + \nabla g(x^*)^T \lambda^*]^T (\bar{x} - x^*) - [x - \nabla f(x) - \nabla g(x)^T \bar{\lambda} - \bar{x}]^T (\bar{x} - x^*) \\ &\quad - \bar{\lambda}^T [g(x^*) - g(x) - \nabla g(x)(x^* - x)] + \bar{\lambda}^T g(x^*) \\ &\quad - (\lambda^*)^T [g(x) - g(x^*) - \nabla g(x^*)(x - x^*)] + (\lambda^*)^T (\lambda + g(x))^- \\ &\leq -\|x - \bar{x}\|^2 - \frac{1}{2} \|\lambda - \bar{\lambda}\|^2 - [\nabla f(x) - \nabla f(x^*)]^T (x - x^*) \\ &\quad - [\nabla f(x^*) + \nabla g(x^*)^T \lambda^*]^T (\bar{x} - x^*) - [x - \nabla f(x) - \nabla g(x)^T \bar{\lambda} - \bar{x}]^T (\bar{x} - x^*) \\ &\quad - \bar{\lambda}^T [g(x^*) - g(x) - \nabla g(x)(x^* - x)] - (\lambda^*)^T [g(x) - g(x^*) - \nabla g(x^*)(x - x^*)] \end{aligned} \quad (15.4)$$

در نامساوی لم ۲.۲.۴ فرض کنید $\bar{\lambda} = \nabla g(x) - \nabla f(x)$ و $v = x - \nabla f(x) - \nabla g(x)^T \bar{\lambda}$ و $u = x^*$ لذا داریم:

$$(x - \nabla f(x) - \nabla g(x)^T \bar{\lambda} - \bar{x})^T (\bar{x} - x^*) \geq 0. \quad (16.4)$$

به واسطه مشتق پذیری و تحدب $f(x)$ و $g(x)$ در $\forall x \in \Omega$ داریم:

$$\begin{cases} [\nabla f(x) - \nabla f(x^*)]^T (x - x^*) \geq 0, \\ g(x^*) - g(x) - \nabla g(x)^T (x^* - x) \geq 0, \\ g(x) - g(x^*) - \nabla g(x^*)^T (x - x^*) \geq 0. \end{cases} \quad (17.4)$$

با جایگذاری (۲.۴)، (۱۶.۴) و (۱۷.۴) در (۱۵.۴) داریم:

$$\frac{dV(x, \lambda)}{dt} \leq -\|x - \bar{x}\|^2 - \frac{1}{\gamma} \|\lambda - \bar{\lambda}\|^2 \leq 0. \quad (18.4)$$

این به این معناست که شبکه عصبی (۷.۴) به مفهوم لیاپانوف پایدار است.

چون

$$V(x, \lambda) \geq \frac{1}{\gamma} (\|x - x^*\|^2 + \|\lambda - \lambda^*\|^2).$$

بنابراین $V(x, \lambda)$ معین مثبت و شعاعی بی کران است. بنابراین یک زیر دنباله همگرا به صورت زیر وجود دارد.

$$\{(x(t_k)^T, \lambda(t_k)^T)^T \mid t_0 < t_1 < \dots < t_k < t_{k+1} < \dots\}, \quad t_k \rightarrow \infty, \quad k \rightarrow \infty,$$

به طوریکه

$$\lim_{k \rightarrow \infty} (x(t_k)^T, \lambda(t_k)^T)^T = (\hat{x}^T, \hat{\lambda}^T)^T.$$

از قضیه لساال داریم:

$\{(x(t)^T, \lambda(t)^T)^T \rightarrow M\}$ وقتی $t \rightarrow \infty$ که M بزرگترین مجموعه تغییرناپذیر در

$$K = \{(x(t)^T, \lambda(t)^T)^T \mid \frac{dV(x, \lambda)}{dt} = 0\}$$

است.

از طرفی (۷.۴) و (۱۸.۴) نشان می دهند که

$$\frac{d\lambda}{dt} = 0, \quad \frac{dx}{dt} = 0 \iff \frac{dV(x, \lambda)}{dt} = 0.$$

بنابراین $(\hat{x}^T, \hat{\lambda}^T)^T \in D^*$ با $M \subseteq K \subseteq D^*$. با جایگذاری $x^* = \hat{x}$ و $\lambda^* = \hat{\lambda}$ در (۱۲.۴) تابع لیاپانوف دیگری را به صورت زیر تعریف می کنیم:

$$\begin{aligned} \hat{V}(x, \lambda) = & f(x) - f(\hat{x}) + \frac{1}{\gamma} [(\bar{\lambda})^2 - (\hat{\lambda})^2] - (x - \hat{x})^T (\nabla f(\hat{x}) + \nabla g(\hat{x}) \hat{\lambda}) \\ & - (\lambda - \hat{\lambda})^T \hat{\lambda} + \frac{1}{\gamma} \|x - \hat{x}\|^2 + \frac{1}{\gamma} \|\lambda - \hat{\lambda}\|^2. \end{aligned} \quad (19.4)$$

پس $\hat{V}(x, \lambda)$ به طور پیوسته مشتق پذیر است و $\hat{V}(\hat{x}, \hat{\lambda}) = 0$. با توجه به اینکه

$$\lim_{k \rightarrow \infty} (x(t_k)^T, \lambda(t_k)^T)^T = (\hat{x}^T, \hat{\lambda}^T)^T,$$

بنابراین

$$\lim_{k \rightarrow \infty} \hat{V}(x(t_k)^T, \lambda(t_k)^T)^T = \hat{V}(\hat{x}, \hat{\lambda}) = 0.$$

بنابراین $\forall \epsilon > 0$ وجود دارد $q > 0$ به طوری که برای هر $t > t_q$ داریم $\hat{V}(x, \lambda) < \epsilon$. به طور مشابه با تجزیه و تحلیل بالایی توانیم ثابت کنیم که تابع $\frac{d\hat{V}(x, \lambda)}{dt} \leq 0$ در نتیجه برای $t \geq t_q$

$$\|x(t) - \hat{x}\|^2 / 2 + \|\lambda(t) - \hat{\lambda}\|^2 / 2 \leq \hat{V}(x, \lambda) \leq \epsilon.$$

این یعنی $\lim_{t \rightarrow \infty} x(t) = \hat{x}$ و $\lim_{t \rightarrow \infty} \lambda(t) = \hat{\lambda}$ است. بنابراین مسیر جواب شبکه عصبی (۷.۴) به نقطه تعادل $(\hat{x}^T, \hat{\lambda}^T)^T$ همگرای سراسری است. همچنین $(\hat{x}^T, \hat{\lambda}^T)^T$ یک جواب بهینه (۱.۴) است. به خصوص اگر $D^* = \{((x^*)^T, (\lambda^*)^T)^T\}$ آنگاه برای هر $x_0 \in \Omega$ و $\lambda_0 \geq 0$ جواب $(x^T, \lambda^T)^T$ با نقطه اولیه $(x_0^T, \lambda_0^T)^T$ با تجزیه و تحلیل بالا به $((x^*)^T, (\lambda^*)^T)^T$ همگرای می‌شود. این یعنی شبکه عصبی (۷.۴) پایدار مجانبی سراسری است و این اثبات را کامل می‌کند. \square

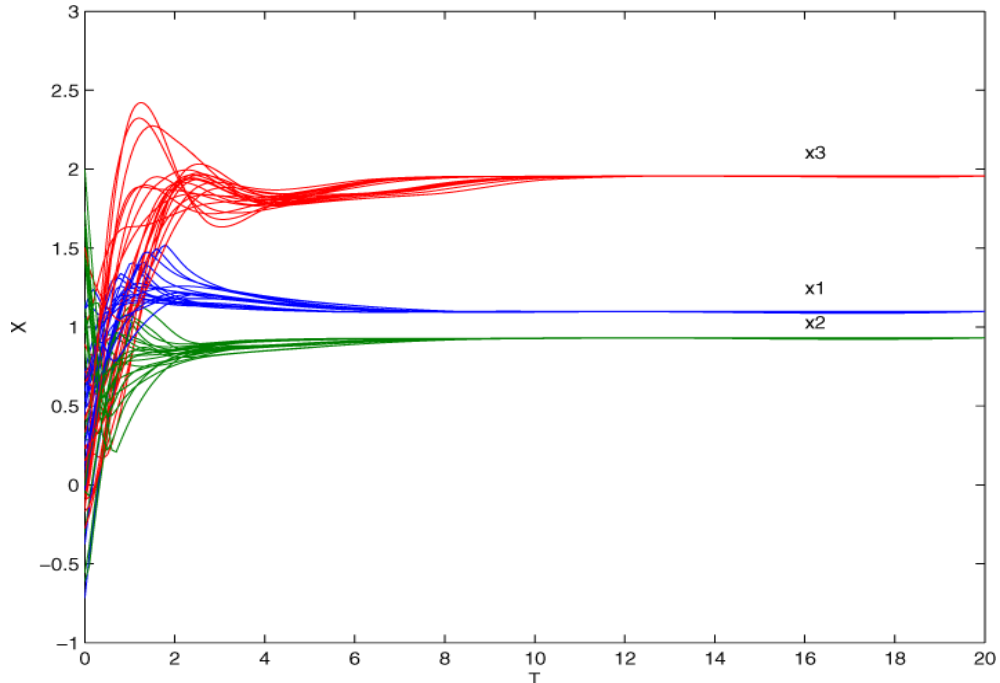
۴.۴ مثال‌های عددی

در این بخش شبیه سازی چندین مثال داده شده تاکارایی شبکه عصبی ارائه شده را برای حل مسائل برنامه ریزی غیرخطی محدب نشان دهیم. شبیه سازی در Matlab انجام شده و معادله دیفرانسیل معمولی باروش رانگ کوتا مرتبه ۴ حل شده است.

مثال ۱۰.۴.۴ ([۵۲]). مسئله برنامه ریزی غیرخطی زیر را در نظر بگیرید:

$$\begin{aligned} \min & \frac{1}{\gamma} [(x_1 - x_2)^4 + (x_2 + x_3)^2 + (x_1 + x_3)^2] \\ \text{s.t.} & \begin{cases} x_1^2 + x_2^2 - x_3 \leq 0, \\ (2 - x_1)^2 + (2 - x_2)^2 - x_3 \leq 0, \\ 2e^{-x_1 + x_2} - x_3 \leq 0, \\ x_1^2 + x_2^2 - 2x_1 + x_2 \leq 4, \\ |x_1| \leq 2, \quad |x_2| \leq 2, \quad x_3 \geq 0. \end{cases} \end{aligned}$$

جواب بهینه مسئله، $x^* = (1/0.983, 0/9037, 1/9565)^T$ است. با استفاده از شبکه عصبی (۷.۴) برای حل این مسئله، همه نتایج شبیه سازی نشان می دهد مسیرهای شبکه عصبی (۷.۴) به جواب بهینه x^* همگراست. رفتار گذرای متناظر در شکل ۱.۴ نشان داده شده است.

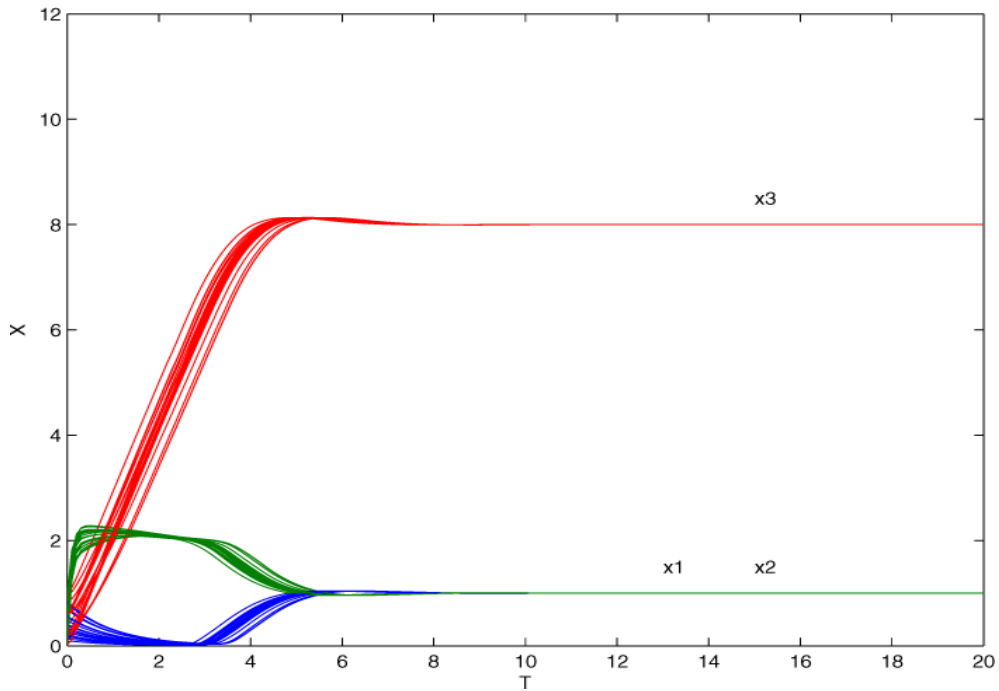


شکل ۱.۴: رفتارهای منحنی $x(t)$ در شبکه عصبی (۷.۴) با ۲۰ نقطه شروع مختلف در مثال ۱.۴.۴.

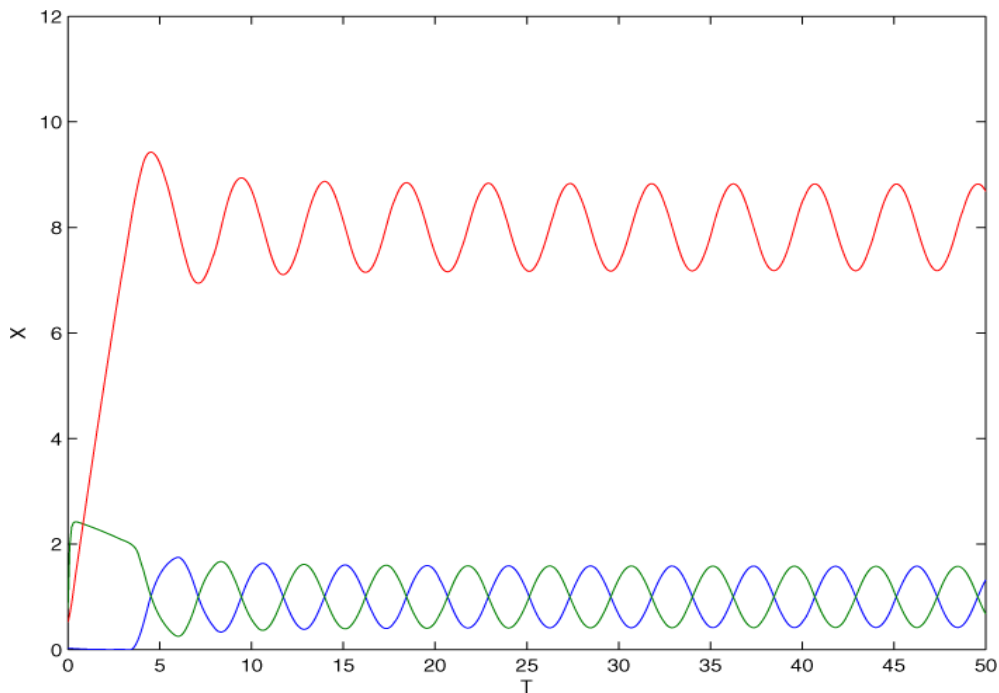
مثال ۲.۴.۴ ([۵۲]). مسئله برنامه ریزی غیرخطی زیر را در نظر بگیرید:

$$\begin{aligned} \min \quad & \frac{1}{4}(x_1 + x_2)^4 - 16x_2 \\ \text{s.t.} \quad & \begin{cases} -x_1 + x_2 \leq 0, \\ x \geq 0. \end{cases} \end{aligned}$$

جواب بهینه مسئله برنامه ریزی غیرخطی $x^* = (1, 1)^T$ است. با استفاده از شبکه عصبی (۷.۴) برای حل این مسئله، همه نتایج شبیه سازی نشان می دهد که مسیرهای شبکه عصبی (۷.۴) به جواب بهینه این مسئله همگراست. برای مقایسه، این مسئله را با استفاده از شبکه عصبی (۶.۴) حل می کنیم. نتیجه شبیه سازی نشان می دهد مسیر شبکه عصبی (۶.۴) پایدار نیست. رفتار گذرای متناظر در شکل‌های ۲.۴ و ۳.۴ نشان داده شده است.



شکل ۲.۴: مسیرهای شبکه عصبی (۷.۴) برای مثال ۲.۴.۴



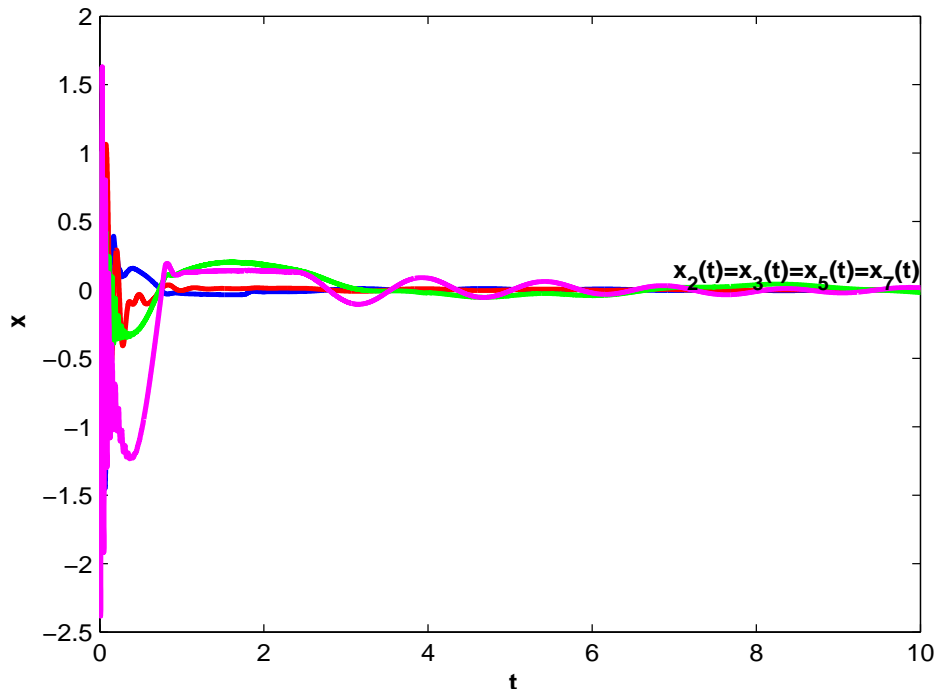
شکل ۳.۴: مسیرهای شبکه عصبی (۶.۴) برای مثال ۲.۴.۴

مثال ۳.۴.۴ ([۷]). مسئله برنامه ریزی خطی زیر را در نظر بگیرید:

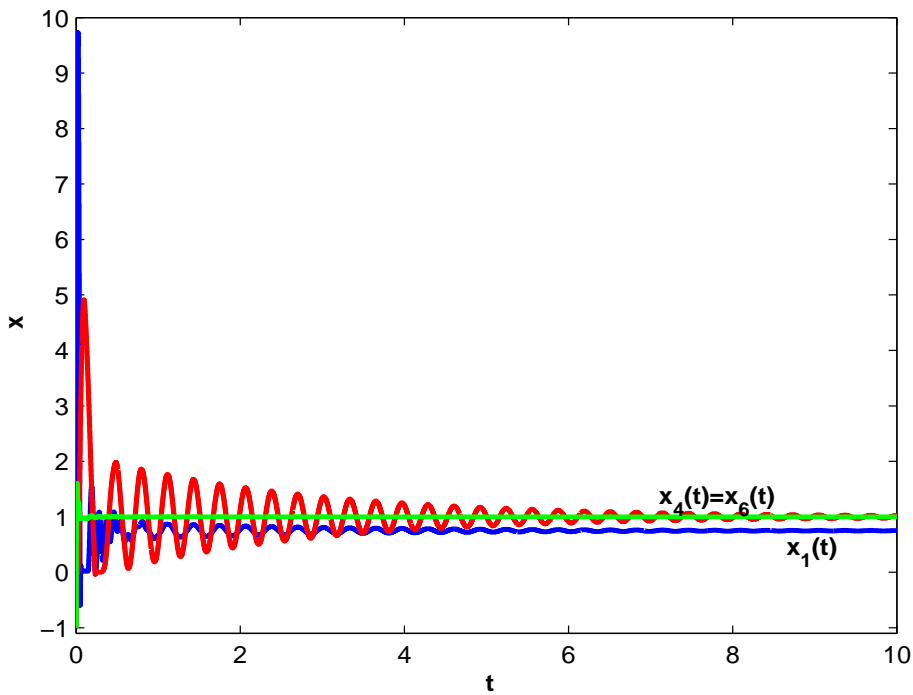
$$\begin{aligned} \min \quad & -\frac{3}{4}x_4 + 2x_5 - \frac{1}{4}x_6 + 6x_7 \\ \text{s.t.} \quad & \begin{cases} x_1 + \frac{1}{4}x_4 - 8x_5 - x_6 + 9x_7 = 0, \\ x_2 + \frac{1}{4}x_4 - 12x_5 - \frac{1}{4}x_6 + 3x_7 = 0, \\ x_3 + x_6 - 1 = 0, \\ x \geq 0. \end{cases} \end{aligned}$$

این مثال یک مسئله برنامه ریزی خطی است که وقتی با الگوریتم سیمپلکس حل شده، در آن دور ظاهر شده است. نتایج شبیه سازی نشان می دهد که مسیرهای مدل ارائه شده به جواب بهینه منحصر به فرد $x^* = (0/75, 0, 0, 1, 0, 1, 0)^T$ همگرا هستند. شکل‌های ۴.۴ و ۵.۴ نشان می دهند که مسیرهای مدل شبکه عصبی (۷.۴) بانقطه شروع $x_0 = (1, -1, 1, -1, 1, -1, 1)^T$ به جواب بهینه مسئله همگرا هستند. برای مقایسه مسئله بالا با استفاده از شبکه عصبی فریزو همکاران (۶.۴) که در شکل ۶.۴ نشان داده شده است حل می کنیم. روشن است که این مدل با $x_0 = (1, -1, 1, -1, 1, -1, 1)^T$ پایدار نیست.

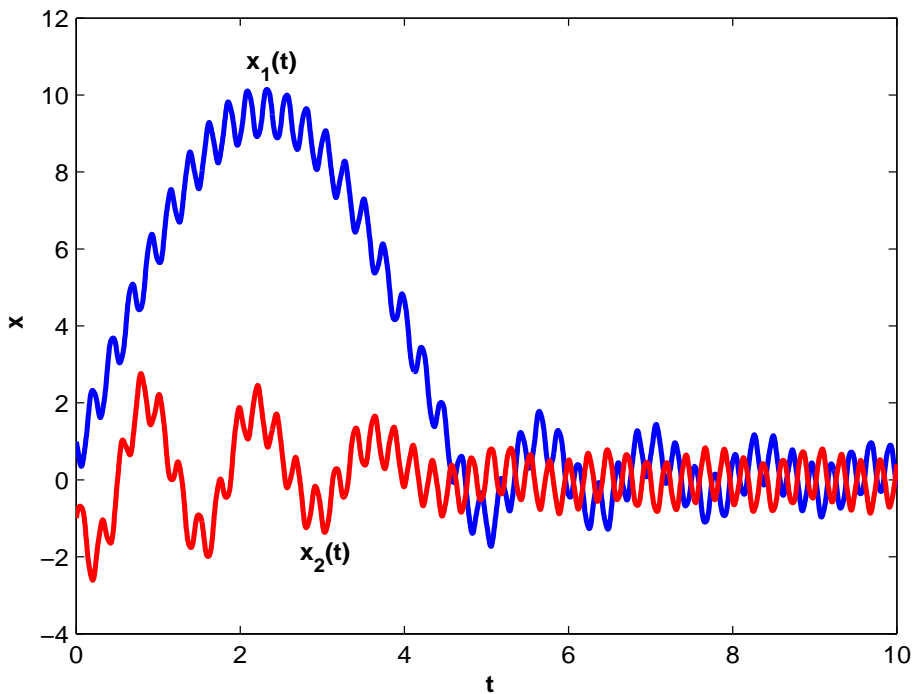
مسئله بالا همچنین با استفاده از مدل شبکه عصبی (۷.۲) با $x_0 = (1, -1, 1, -1, 1, -1, 1)^T$ حل شده است. از شکل ۷.۴ درمی یابیم که مدل ارائه شده در (۷.۲) برای حل این مسئله برنامه ریزی خطی مناسب نیست.



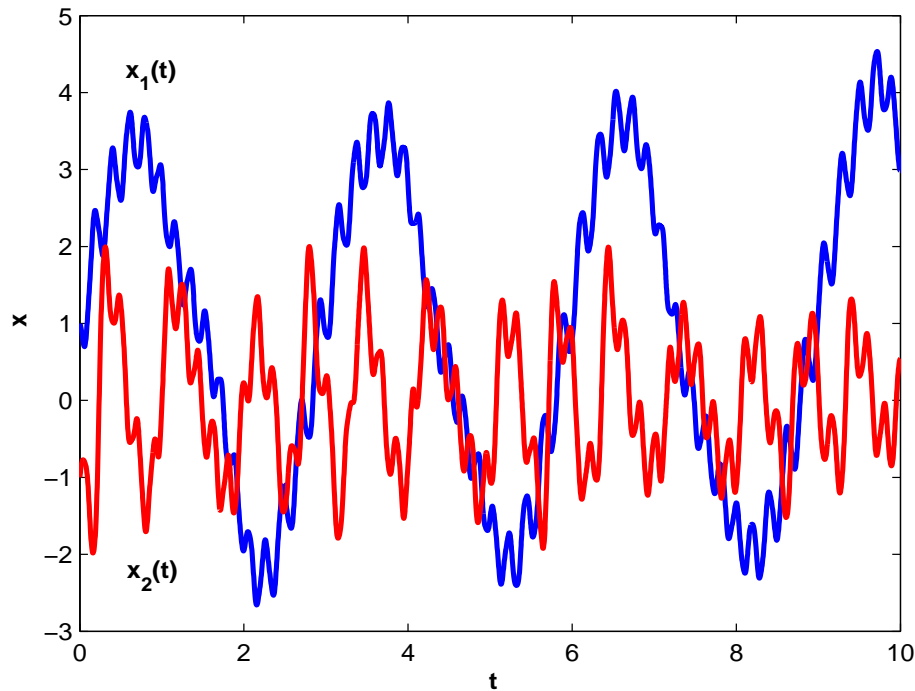
شکل ۴.۴: رفتارهای گذرای $x_2(t), x_3(t), x_5(t), x_7(t)$ در شبکه عصبی (۷.۴) بانقطه شروع $x_0 = (1, -1, 1, -1, 1, -1, 1)^T$ در مثال ۳.۴.۴.



شکل ۵.۴: رفتارهای گذرای $x_1(t)$, $x_4(t)$, $x_6(t)$ در شبکه عصبی (۷.۴) بانقطه شروع در $x_0 = (1, -1, 1, -1, 1, -1, 1)^T$ مثال ۳.۴.۴.



شکل ۶.۴: رفتارواگرایی شبکه عصبی (۶.۴) بانقطه شروع در $x_0 = (1, -1, 1, -1, 1, -1, 1)^T$ مثال ۳.۴.۴.



شکل ۷.۴: رفتار و آگرای شبکه عصبی (۷.۲) با نقطه شروع $x_0 = (1, -1, 1, -1, 1, -1, 1)^T$ در مثال ۳.۴.۴.

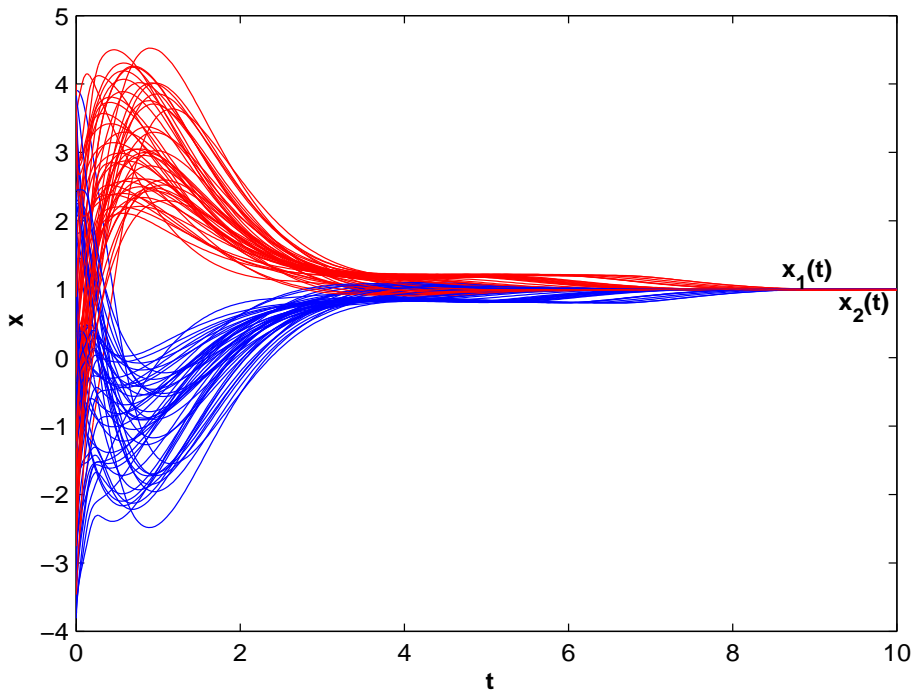
مثال ۴.۴.۴ ([۲۳]). مسئله برنامه ریزی غیرخطی زیر را در نظر بگیرید:

$$\begin{aligned} \min \quad & \frac{1}{4}(x_1 + x_2)^4 - 16x_2 \\ \text{s.t.} \quad & \begin{cases} -x_1 + x_2 \leq 0, \\ -3 \leq x_i \leq 3, \quad i = 1, 2. \end{cases} \end{aligned}$$

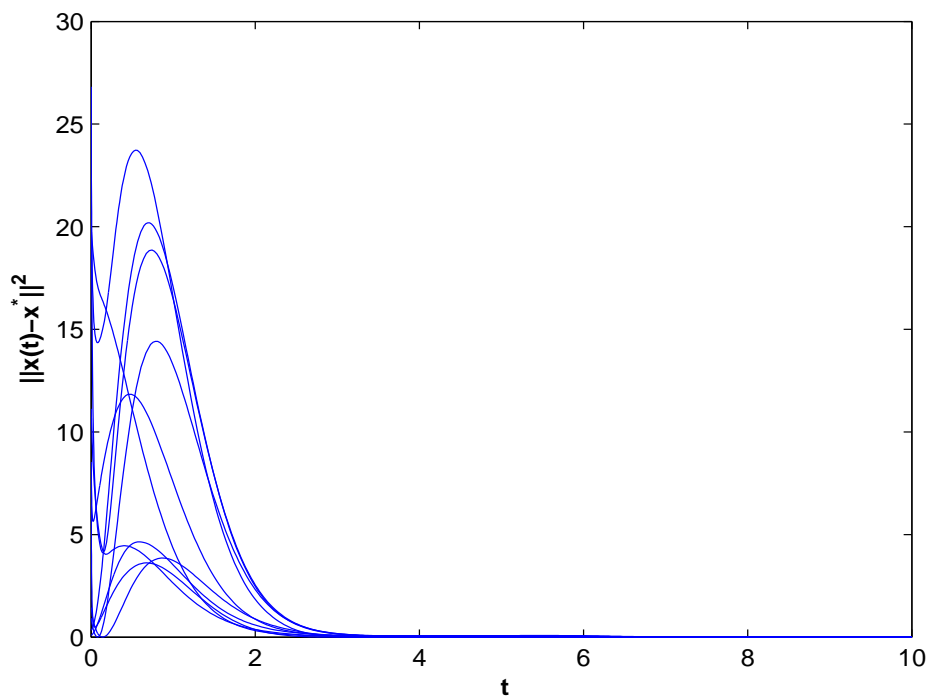
جواب بهینه این مسئله $x^* = (1, 1)^T$ است. از شبکه عصبی (۷.۴) برای حل این برنامه ریزی ریاضی استفاده می‌کنیم. نتایج شبیه‌سازی نشان می‌دهد که مسیر شبکه عصبی (۷.۴) با هر نقطه شروع همواره به $y^* = (x^{*T}, u^{*T})^T$ همگراست. برای مثال شکل ۸.۴ رفتار گذرای $x(t)$ را بر اساس شبکه عصبی (۷.۴) با ۵۰ نقطه شروع تصادفی نشان می‌دهد. شکل ۹.۴ خطای نرم l_2 بین x و x^* را با ۱۰ نقطه شروع تصادفی نشان می‌دهد. برای مقایسه، مسئله بالا با استفاده از سیستم دینامیکی فریزو همکاران (۶.۴) در مثال ۴ در [۲۳] حل شده است که از آن درمی‌یابیم مسیر شبکه عصبی (۶.۴) پایدار نیست. علاوه بر این، مسئله بالا در مثال ۴ در [۲۳] با استفاده از شبکه عصبی کندی-چاو

$$\frac{dx}{dt} = -(\nabla f(x) + \gamma[\nabla g(x)g^+(x) + \nabla h(x)h(x)])$$

حل شده است. از آن درمی‌یابیم که شبکه عصبی کندی-چاو به نقطه تعادل $(1 + \frac{\gamma}{\gamma}, 1 - \frac{\gamma}{\gamma})$ همگراست و هنگامی که γ بزرگ است، می‌تواند به عنوان جواب تقریبی مسئله بالا در نظر گرفته شود اما این نقطه تعادل شدنی نیست.



شکل ۸.۴: رفتارهای منحنی $x(t)$ در شبکه عصبی (۷.۴) با ۵۰ نقطه شروع مختلف در مثال ۴.۴.۴.



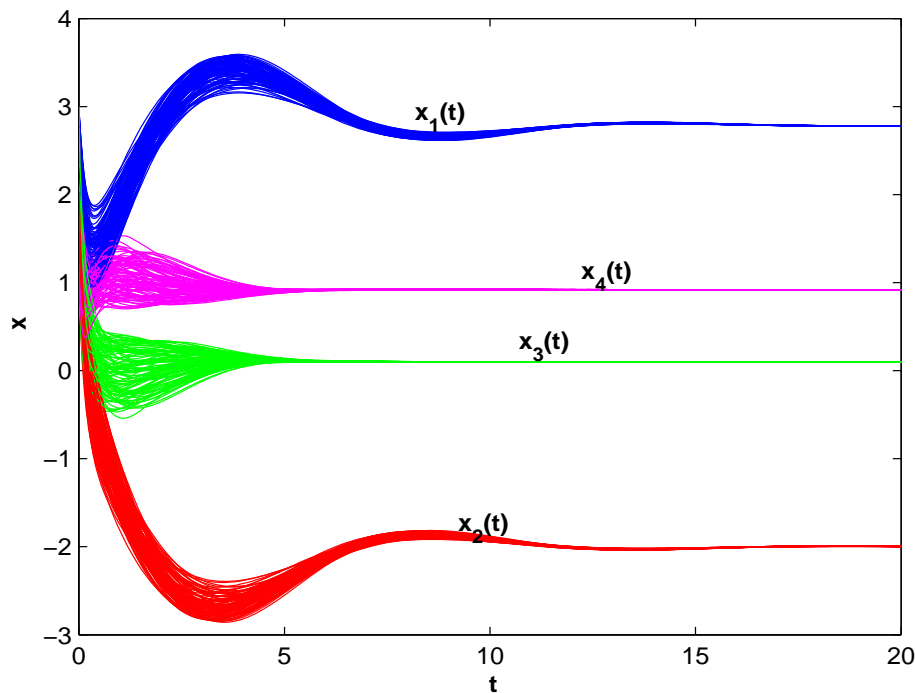
شکل ۹.۴: رفتار همگرایی $\|x(t) - x^*\|_2$ در مثال ۴.۴.۴.

مثال ۵.۴.۴ ([۳۴]). مسئله برنامه ریزی غیرخطی زیر را در نظر بگیرید:

$$\min \frac{1}{4}(x_1^4 + x_2^4) + 2(x_3^2 + x_4^2) - \ln(x_1 x_4) + 3x_1 x_2 + 4x_3 x_4 - 2x_1 - 3x_4$$

$$\text{s.t.} \begin{cases} -x_1 + x_2 + x_3 + x_4 \leq 2, \\ x_1 + x_2 - x_4 \leq 12, \\ 2x_1 - x_2 + x_3 \leq 18, \\ 2x_1 + x_2 - x_3 + x_4 \geq 2, \\ 0 \leq x_1 \leq 10, \\ -2 \leq x_2 \leq 10, \\ 0/1 \leq x_3 \leq 10, \\ 0/1 \leq x_4 \leq 10. \end{cases}$$

این مسئله فقط یک جواب $x^* = (2/786, -2, 0/1, 0/9213)$ دارد. شکل ۱۰.۴ همگرایی مسیر $x(t)$ را با ۱۰۰ نقطه شروع تصادفی نشان می‌دهد.



شکل ۱۰.۴: رفتارهای منحنی $x(t)$ در شبکه عصبی (۷.۴) با ۱۰۰ نقطه شروع تصادفی در مثال ۵.۴.۴.

۵.۴ نتیجه‌گیری

در این پایان‌نامه ابتدا مروری بر شبکه‌های عصبی داشته و با استفاده از یک مدل شبکه عصبی کارا به حل مسائل برنامه‌ریزی غیرخطی پرداختیم. همچنین پایداری و همگرایی سراسری به جواب بهینه را مورد بررسی قرار دادیم. از جمله مزیت‌های مدل ارائه شده می‌توان به موارد زیر اشاره کرد:

- حل مسأله رگرسیون
- حل مسأله ربات چند انگشتی
- حل مسأله پردازش تصویر
- حل مسائل بهینه‌سازی سبد سهام
- استفاده از مدل پیشنهادی برای حل مسأله طبقه بندی
- تعمیم مدل پیشنهادی برای داده‌های رگرسیون با ابعاد بالاتر
- حل مسأله رگرسیون ریج^۳ با مدل ارائه شده
- مدل ذکر شده همگرایی تضمین شده دارد
- این مدل نیاز به هیچ‌گونه پارامتر جریمه ندارد
- با وجود سادگی مدل، در رده بسیاری از مسائل بهینه‌سازی محدب قابل استفاده است
- این مدل هیچ‌گونه وابستگی به نقطه شروع نداشته و این نقطه می‌تواند خارج از ناحیه شدنی و نقطه‌ای تصادفی باشد
- این مدل می‌تواند قادر به حل مسائل وابسته به زمان و مسائل با ابعاد بزرگ باشد
- قابل توجه است که مدل‌های (۷.۲)، (۹.۳)، (۱۰.۳) و (۶.۴) مسائل برنامه ریزی خطی را حل نمی‌کنند اما مدل (۷.۴) این مسائل را حل می‌کند.

^۳Ridge Regression

پوست آ

کد مثال‌های فصل ۴

آ.۱ مثال (۱.۴.۴)

```
function xdot=e25(t,x);
xdot=8*[-(2*(x(1)-x(2))^3+x(1)+x(3)+2*x(1)*max(x(4)+x(1)^2+x(2)^4-x(3),0)
+(-4+2*x(1))*max(x(5)+(2-x(1))^2+(2-x(2))^2-x(3),0)
-2*exp(-x(1)+x(2))*max(x(6)+2*exp(-x(1)+x(2))-x(3),0)
+(2*x(1)-2)*max(x(7)+x(1)^2+x(2)^2-2*x(1)+x(2)-4,0)
-max(x(8)-x(1)-2,0)+max(x(9)+x(1)-2,0)];
-(2*(x(1)-x(2))^3+x(2)+x(3)+4*x(2)^3*max(x(4)+x(1)^2+x(2)^4-x(3),0)
+(-4+2*x(2))*max(x(5)+(2-x(1))^2+(2-x(2))^2-x(3),0)
+2*exp(-x(1)+x(2))*max(x(6)+2*exp(-x(1)+x(2))-x(3),0)
+(2*x(2)+1)*max(x(7)+x(1)^2+x(2)^2-2*x(1)+x(2)-4,0)
-max(x(10)-x(2)-2,0)+max(x(11)+x(2)-2,0)];
-(x(1)+x(2)+2*x(3)-max(x(4)+x(1)^2+x(2)^4-x(3),0)
-max(x(5)+(2-x(1))^2+(2-x(2))^2-x(3),0)-max(x(6)+2*exp(-x(1)+x(2))-x(3),0)
-max(x(12)-x(3),0));

max(x(4)+x(1)^2+x(2)^4-x(3),0)-x(4);
max(x(5)+(2-x(1))^2+(2-x(2))^2-x(3),0)-x(5);
max(x(6)+2*exp(-x(1)+x(2))-x(3),0)-x(6);
max(x(7)+x(1)^2+x(2)^2-2*x(1)+x(2)-4,0)-x(7);
```

```
max(x(8)-x(1)-2,0)-x(8);
max(x(9)+x(1)-2,0)-x(9);
max(x(10)-x(2)-2,0)-x(10);
max(x(11)+x(2)-2,0)-x(11);
max(x(12)-x(3),0)-x(12)];
end
```

```
[t,x]=ode45('e25',[0 10],a);
a=unifrnd(-w,w,[1,12]);
plot(t,x(:,1),'r');
hold on;
plot(t,x(:,2),'b');
hold on;
plot(t,x(:,3),'g');
hold on;
text(3,1.3,'\bf x_1(t)','FontSize',10);
text(3,0.75,'\bf x_2(t)','FontSize',10);
text(3,1.8,'\bf x_3(t)','FontSize',10);
xlabel('\bf t')
ylabel('\bf x')
```

مراجع

- [۱] آذر. ضمیری. شبکه‌های عصبی مصنوعی و بهینه‌سازی. پایان‌نامه کارشناسی‌ارشد، دانشکده علوم ریاضی، دانشگاه فردوسی مشهد، آبان ۱۳۸۷.
- [۲] محمد. مقدس. حل رده‌ای از مسائل بهینه‌سازی غیرخطی با استفاده از شبکه‌های عصبی. پایان‌نامه کارشناسی‌ارشد، دانشکده علوم ریاضی، دانشگاه فردوسی مشهد، تابستان ۱۳۹۲.
- [۳] محمد. مقدس، سهراب. عفتی و محمد. اسحاق‌نژاد. یک شبکه عصبی برای حل مسائل بهینه‌سازی غیرخطی نامحدب. در ششمین کنفرانس بین‌المللی انجمن ایرانی تحقیق در عملیات، تهران، ایران، اردیبهشت ۱۳۹۲.
- [4] Agrawal SK, Fabien BC. Optimization of dynamic systems. Netherlands: Kluwer Academic Publishers; 1999.
- [5] A. Bouzerdoum and T. R. Pattison. Neural network for quadratic optimization with bound constraints. IEEE Trans. Neural Netw., 4(2): 293–304, (1993).
- [6] A. Nazemi, S. Effati. An application of a merit function for solving convex programming problems, (2013) 212–221.
- [7] Bazaraa, M.S., Jarvis, J.J., Sherali, H.D., 1990. Linear Programming and Network Flows, second ed. Wiley, New York.
- [8] Bazaraa, M.S., Sherali, H.D., Shetty, C.M., 1993. Nonlinear Programming—Theory and Algorithms, 2nd ed. Wiley, New York.
- [9] B. Widrow. Generalization and information storage in networks of adaline ‘neurons’ in Self Organizing Systems, pp. 435–461. Spartan Books, (1959).
- [10] B. Widrow and M. E. Hoff. Adapting switching circuits. in 1960 ire wescon Convention Record, no. 4, pp. 96–104, New York, NY, (1960). ire.
- [11] B. Widrow and M. A. Lehr. 30 years of adaptive neural networks: perceptron, madaline, and backpropagation. Proc. IEEE, 78(9): 1415–1442, (1990).
- [12] B. Widrow and R. Winter. Neural nets for adaptive filtering and adaptive pattern recognition. Computer, 21(3): 25–39, (1988).

-
- [13] Chen, K. Z., Leung, Y., Leung, K. S., Gao, X. B.: A Neural Network for Solving Nonlinear Programming Problem, *Neural Comput. and Applica.*, 11, (2002), 103-111
- [14] C.Y. Maa and M. A. Schanblatt. A two-phase optimization neural network. *IEEE Trans. Neural Netw.*, 3(6): 1003–1009, (1992).
- [15] D. Luenberger and Y. Ye. *Linear and Nonlinear Programming*. Springer, 3rd ed., (2008).
- [16] D. O. Hebb. *The Organization of Behavior: A Neuropsychological Theory*. John Wiley Sons, Inc., New York, (1949).
- [17] D. E. Rumelhart, G. E. Hinton and R. J. Williams. Learning representations by backpropagating errors. *Nature*, 323: 533–536, (1986). 1370–1379.
- [18] Effati S, Ghomashi A, Nazemi AR. Application of projection neural network in solving convex programming problems. *Appl Math Comput* 2007;188:1103–14.
- [19] Effati S, Nazemi AR. Neural network models and its application for solving linear and quadratic programming problems. *Appl Math Comput* 2006;172:305–31.
- [20] Friesz, T. L., Bernstein, D. H., Mehta, N. J., Tobin, R. L., Ganjizadeh, S.: Day-today Dynamic Network Disequilibria and Idealized Traveler Information Systems, *Opera. Res.*, 42, (1994), 1120-1136
- [21] F. Rosenblatt. *Neurocomputing: foundations of research*. chap. The perception: a probabilistic model for information storage and organization in the brain, pp. 89–114. MIT Press, Cambridge, MA, USA, (1988).
- [22] F. Rosenblatt. *Principles of neurodynamics: perceptrons and the theory of brain mechanisms*. Report (Cornell Aeronautical Laboratory). Spartan Books, (1962).
- [23] Gao, X., 2004. A novel neural network for nonlinear convex programming. *IEEE Trans. Neural Networks* 15, 613–621.
- [24] Gass, S. I. (1969). *Linear Programming* (third ed.). New York: McGraw-Hill.
- [25] H. K. Khalil. *Nonlinear Systems*. Prentice Hall PTR, 3rd ed., (2002).
- [26] Huang YC. A novel method to handle inequality constraints for convex programming neural network. *Neural Process Lett* 2002;16:17–27.
- [27] J. J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proc. Nat. Acad. Sci.*, 79(8): 2554–2558, (1982).

- [28] J. J. Hopfield and D. W. Tank. “neural” computation of decisions in optimization problems. *Biological Cybernetics*, 52(3): 141–152, (1985).
- [29] J. M. Ortega and W. C. Rheinboldt. *Iterative Solution of Nonlinear Equations in Several Variables*. Classics in Applied Mathematics. Society for Industrial and Applied Mathematics, (2000).
- [30] J. Zabczyk. *Mathematical control theory: an introduction*. Systems control. Birkhäuser, (1992).
- [31] Kennedy, M. P., Chua, L. O.: *Neural Networks for Nonlinear Programming*, *IEEE Trans. Circuits Syst.*, 35, (1988), 554-562
- [32] Kinderlehrer, D., Stampcchia, G.: *An Introduction to Variational Inequalities and Their Applications*, New York: Academic, 1980
- [33] L. Perko. *Differential Equations and Dynamical Systems*. Springer, 3rd ed., (2001).
- [34] Malek, A., Hosseinipour-Mahani, N., Ezazipour, S., 2010. Efficient recurrent neural network model for the solution of general nonlinear optimization problems. *Optim. Methods Software* 25, 1–18.
- [35] M. L. Minsky and S. Papert. *Perceptrons: An Introduction to Computational Geometry*. MitPress, (1972).
- [36] M. Moghaddas and S. Effati. A novel recurrent neural network based on ncp function for solving convex nonlinear optimization problems. in *The 5th Iranian Conference on Applied Mathematics*, Hamadan, Iran, September (2013). Bu-Ali Sina University.
- [37] M.S. Bazaraa, H.D. Sherali and C. M. Shetty. *Nonlinear Programming*. John Wiley Sons, Inc., 3rd ed. , (2005).
- [38] P. Werbos. *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*. Ph.D. thesis, Harvard University, Cambridge, MA, (1974).
- [39] Rodriguez-Vazquez, A., Dominguez-Castro, R., Rueda, A., Huertas, J.L., Sanchez-Sinencio, E., 1990. Nonlinear switched-capacitor neural networks for optimization problems. *IEEE Trans. Circuits Syst.* 37, 384–397.
- [40] S. Zhang and A.G. Constantinides. Lagrange programming neural networks. *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, 39(7): 441–452, (1992).
- [41] S. Effati, and M. Baymani. A new nonlinear neural network for solving convex nonlinear programming problems. *Appl. Math. and Comput.*, 168(2): 1370 – 1379, (2005).

- [42] S. Effati, A. Ghomashi and M. Abbasi. A novel recurrent neural network for solving mlcps and its application to linear and quadratic programming. *Asia-Pacific Journal of Operational Research*, 28(4): 523–541, November (2011).
- [43] Tank, D. W., Hopfield, J. J.: Simple ‘Neural’ Optimization Network: An A/D Converter, Signal Decision Circuit and A Linear Programming Circuit, *IEEE Trans. Circuits Syst.*, 33, (1986), 533-541
- [44] W. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *Bull. Math. Biophys.*, 5(4): 115–133, (1943).
- [45] Xia Y, Wang J. A general methodology for designing globally convergent optimization neural networks. *IEEE Trans Neural Netw* 1998;9:1331–43.
- [46] Xia Y, Wang J. A general projection neural network for solving monotone variational inequality and related optimization problems. *IEEE Trans Neural Netw* 2004;15:318–28.
- [47] Xia Y, Wang J. A recurrent neural network for solving linear projection equations. *Neural Netw* 2000;13:337–50.
- [48] Xia Y, Feng G. A new neural network for solving nonlinear projection equations. *Neural Netw* 2007;20:577–89.
- [49] Xia Y, Leng H, Wang J. A projection neural network and its application to constrained optimization problems. *IEEE Trans Circuits Syst* 2002;49:447–58.
- [50] Xia Y, Wang J. A recurrent neural network for nonlinear convex optimization subject to nonlinear inequality constraints. *IEEE Trans Circuits Syst* 2004;51:447–58.
- [51] Yang Y, Cao J. A feedback neural network for solving convex constraint optimization problems. *Appl Math Comput* 2008;201:340–50.
- [52] Yang Y, Xu X. The projection neural network for solving convex nonlinear programming. In: Huang D-S, Heutte L, Loog M, editors. *ICIC 2007, LNAI*, vol. 4682. Heidelberg: Springer-Verlag; 2007. p. 174–81.
- [53] Y. Xia, G. Feng and J. Wang. A novel recurrent neural network for solving nonlinear optimization problems with inequality constraints. *IEEE Trans. Neural Netw.*, 19(8): 1340–1353, (2008).
- [54] Y. Xia and J. Wang. A recurrent neural network for solving nonlinear convex programs subject to linear constraints. *IEEE Trans. Neural Netw.*, 16(2): 379–386, (2005).

-
- [55] Sun, J., Chen, J.-S., Ko, C.-H. (2012). Neural networks for solving second-order cone constrained variational inequality problem. *Computational Optimization and Applications*, 51, 623–648.
- [56] Sun, J., Zhang, L. (2009). A globally convergent method based on Fischer–Burmeister operators for solving second-order cone constrained variational inequality problems. *Computers and Mathematics with Applications*, 58, 1936–1946.

واژه‌نامه فارسی به انگلیسی

LaSalle Principle of Invariance	اصل تغییرناپذیری لسال
Axon	آکسون
Strictly	اکیداً
Simplex Algorithm	الگوریتم سیمپلکس
Integration	انتگرال گیری
Vector	بردار
Programming	برنامه ریزی
Optimal	بهینه
Optimization	بهینه سازی
Unbounded	بی کران
Postsynaptic Terminal	پایانه بعدازسیناپسی
Presynaptic Terminal	پایانه قبل ازسیناپسی
Stable	پایدار
Stability	پایداری
Stability in the Sense of Lyapunov	پایداری به مفهوم لیاپانوف
Globally Asymptotically Stable	پایدارمجبانی سراسری
Back Propagation	پس انتشار
Feed Forward	پیش خورد
Continuous	پیوسته
Lipschitz Continuous	پیوسته لیپ شیتز
Locally Lipschitz Continuous	پیوسته لیپ شیتز محلی
Function	تابع
Activation Function	تابع فعال سازی
Firing	تحریک شدن
Random	تصادفی

Projection	تصویر
Equilibrium	تبادل
Invariant	تغییرناپذیر
Approximate	تقریبی
Penalty	جریمه
Solution	جواب
Linear	خطی
Dendrite	دندریت
Dual	دوگان
Dynamic	دینامیکی
Differential	دیفرانسیل
Behavior	رفتار
Mathematical	ریاضی
Subsequence	زیر دنباله
Jacobian	ژاکوبین
Globally	سراسری
Switched-Capacitor	سوئیچ-خازن
System	سیستم
Synapse	سیناپس
Network	شبکه
Feedback (Recurrent) Neural Network	شبکه های عصبی بازگشتی
Artificial Neural Networks	شبکه های عصبی مصنوعی
Simulation	شبیه سازی
Feasible	شدنی
Second-Order Sufficient Conditions	شرایط کافی مرتبه دوم
Radially	شعاعی
Synaptic Cleft	شکاف سیناپسی
Zero	صفر
Multipliers	ضرایب
Neural	عصبی
Projection Operator	عملگر تصویر
Adaptive Linear Elements	عنصر خطی تطبیقی

Nonlinear	غیرخطی
Phase	فاز
Transient	گذرا
Lagrange	لاگرانژ
Hessian Matrix	ماتریس هسین
Example	مثال
Variable	متغیر
Finite	متناهی
Asymptotically	مجانبی
Set	مجموعه
Invariant Set	مجموعه تغییرناپذیر
Level Set	مجموعه سطح
Convex	محدب
Constraints	محدودیت‌ها
Trajectory	مسیر
Derivative	مشق
Differentiable	مشق‌پذیر
Equation	معادله
Ordinary	معمولی
Positive definite	معین مثبت
Value	مقدار
Unique	منحصربه‌فرد
Negative	منفی
Mean	میانگین
Region	ناحیه
Nonconvex	نامحدب
Inequality	نامساوی
Neuron	نرون
Infeasible	نشدنی
Initial point	نقطه شروع
Strict Local Minimum Point	نقطه مینیمم محلی اکید
Exponentially	نمایی

Diagram	نمودار
Positive Semidefinite	نیمه معین مثبت
Convergent	همگرا
Converge	همگرا بودن
Convergence	همگرایی
Globally Convergent	همگرای سراسری
Monotone	یکنوا

واژه‌نامه انگلیسی به فارسی

Activation Function	تابع فعال سازی
Adaptive Linear Elements	عنصر خطی تطبیقی
Approximate	تقریبی
Artificial Neural Networks	شبکه های عصبی مصنوعی
Asymptotically	مجانبی
Axon	آکسون
Back Propagation	پس انتشار
Behavior	رفتار
Constraints	محدودیت ها
Continuous	پیوسته
Convex	محدب
Converge	همگرایی
Convergent	همگرا
Convergence	همگرایی
Dendrite	دندریت
Derivative	مشتق
Diagram	نمودار
Differential	دیفرانسیل
Differentiable	مشتق پذیر
Dual	دوگان
Dynamic	دینامیکی
Equation	معادله
Equilibrium	تعادل
Example	مثال
Exponentially	نمایی

Feasible	شدنی
Feedback (Recurrent) Neural Network	شبکه های عصبی بازگشتی
Feed Forward	پیش خورد
Finite	متناهی
Firing	تحریک شدن
Function	تابع
Globally	سراسری
Globally Convergent	همگرای سراسری
Globally Asymptotically Stable	پایدارمجانبی سراسری
Hessian Matrix	ماتریس هسین
Inequality	نامساوی
Infeasible	نشدنی
Initial point	نقطه شروع
Integration	انتگرال گیری
Invariant	تغییرناپذیر
Invariant Set	مجموعه تغییرناپذیر
Jacobian	ژاکوبین
Lagrange	لاگرانژ
LaSalle Principle of Invariance	اصل تغییرناپذیری لسال
Level Set	مجموعه سطح
Linear	خطی
Lipschitz Continuous	پیوسته لیپ شیتز
Locally Lipschitz Continuous	پیوسته لیپ شیتز محلی
Mathematical	ریاضی
Mean	میانگین
Monotone	یکنوا
Multipliers	ضرایب
Negative	منفی
Network	شبکه
Neural	عصبی
Neuron	نرون
Nonconvex	نامحدب

Nonlinear	غیرخطی
Optimal	بهینه
Optimization	بهینه سازی
Ordinary	معمولی
Penalty	جریمه
Phase	فاز
Positive definite	معین مثبت
Positive Semidefinite	نیمه معین مثبت
Postsynaptic Terminal	پایانه بعدازسیناپسی
Presynaptic Terminal	پایانه قبل ازسیناپسی
Programming	برنامه ریزی
Projection	تصویر
Projection Operator	عملگر تصویر
Radially	شعاعی
Random	تصادفی
Region	ناحیه
Second-Order Sufficient Conditions	شرایط کافی مرتبه دوم
Set	مجموعه
Simplex Algorithm	الگوریتم سیمپلکس
Simulation	شبیه سازی
Solution	جواب
Stable	پایدار
Stability	پایداری
Stability in the Sense of Lyapunov	پایداری به مفهوم لیاپانوف
Strict Local Minimum Point	نقطه مینیمم محلی اکید
Strictly	اکیداً
Subsequence	زیر دنباله
Switched-Capacitor	سوئیچ-خازن
Synapse	سیناپس
Synaptic Cleft	شکاف سیناپسی
System	سیستم
Trajectory	مسیر

Transient.....	گذرا
Unbounded.....	بی کران
Unique.....	منحصربه فرد
Value.....	مقدار
Variable.....	متغیر
Vector.....	بردار
Zero.....	صفر

Aabstract

A wide variety of scientific and engineering problems can be formulated as nonlinear optimization problems (NOPs). One promising approach to solve the NOPs with high dimension is to employ artificial neural networks. In this thesis we present two neural network models for solving NOPs. The first model can solve convex nonlinear programming problems (CNPPs) with the constraints of equality and inequality. The second model is derived based on projection function for solving nonconvex nonlinear programming problems (NCNPPs). In addition, we also analyze the existence and the convergence of the trajectory, and the stability properties of the neural networks models. The validity and efficiency of the proposed neural networks are demonstrated by using numerical examples.

keywords: Convex optimization, Neural network, KKT conditions, Stability



Shahrood University
Faculty Of Mathematical Sciences

Dissertation Submitted in Partial
Fulfillment of The Requirements For The
Degree of Master of Science in
Applied Mathematics

**An application of a projection operator and
dynamic systems for solving a class of
nonlinear programming problems**

Supervisor
Dr. Alireza Nazemi

by
Hasan Farokhi

2015