

لَهُ مُلْكُ السَّمَاوَاتِ وَالْأَرْضِ



دانشگاه تربیت مدرس
دانشکده مهندسی برق و کامپیوتر

سممه تعالی

تاییدیه اعضای هیات داوران حاضر در جلسه دفاع از پایان نامه کارشناسی ارشد

آقای علیرضا قصبه پایان نامه ۶ واحدی خود را با عنوان کشف انجمن ها در شبکه های

اجتماعی با استفاده از بهینه سازی کندوی زنیور عسل در تاریخ ۱۳۹۳/۱۱/۱۸

ارائه کردند.

اعضای هیات داوران نسخه نهایی این پایان نامه را از نظر فرم و محتوا تایید کرده، پذیرش آنرا

برای اخذ درجه کارشناسی ارشد مهندسی کامپیوترنرم افزار پیشنهاد می کنند.

عضو هیات داوران	نام و نام خانوادگی	رتبه علمی	مضمار
استاد راهنمای	دکتر محمد صنیعی آباده	استادیار	
استاد مشاور	دکتر مهدی آبادی	استادیار	ابراهیم
استاد ناظر	دکتر نصرالله مقدم چرکری	دانشیار	علی
استاد ناظر	دکتر آزاده شاکری	استادیار	سید جواد
مدیر گروه (یا نماینده گروه تخصصی)	دکتر نصرالله مقدم چرکری	دانشیار	

دستورالعمل حق مالکیت مادی و معنوی در مورد نتایج پژوهش‌های علمی دانشگاه تربیت مدرس

مقدمه: با عنایت به سیاست‌های پژوهشی دانشگاه در راستای تحقق عدالت و کرامت انسانها که لازمه شکوفایی علمی و فنی است و رعایت حقوق مادی و معنوی دانشگاه و پژوهشگران، لازم است اعضای هیات علمی، دانشجویان، دانش آموختگان و دیگر همکاران طرح، در مورد نتایج پژوهش‌های علمی که تحت عناوین پایان‌نامه، رساله و طرحهای تحقیقاتی که با هماهنگی دانشگاه انجام شده است، موارد ذیل را رعایت نمایند:

ماده ۱- حقوق مادی و معنوی پایان‌نامه‌ها / رساله‌های مصوب دانشگاه متعلق به دانشگاه است و هرگونه بهره‌برداری از آن باید با ذکر نام دانشگاه و رعایت آئین‌نامه‌ها و دستورالعمل‌های مصوب دانشگاه باشد.

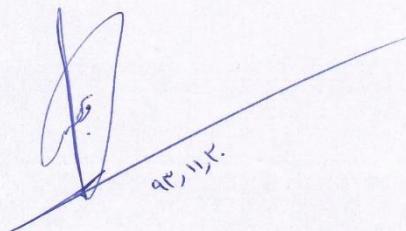
ماده ۲- انتشار مقاله یا مقالات مستخرج از پایان‌نامه / رساله به صورت چاپ در نشریات علمی و یا ارائه در مجتمع علمی باید به نام دانشگاه بوده و استاد راهنما مسئول مکاتبات مقاله باشد. تبصره: در مقالاتی که پس از دانش آموختگی بصورت ترکیبی از اطلاعات جدید و نتایج حاصل از پایان‌نامه / رساله نیز منتشر می‌شود نیز باید نام دانشگاه درج شود.

ماده ۳- انتشار کتاب حاصل از نتایج پایان‌نامه / رساله و تمامی طرح‌های تحقیقاتی دانشگاه باید با مجوز کتبی صادره از طریق حوزه پژوهشی دانشگاه و بر اساس آئین‌نامه‌های مصوب انجام می‌شود.

ماده ۴- ثبت اختراع و تدوین دانش فنی و یا ارائه در جشنواره‌های ملی، منطقه‌ای و بین‌المللی که حاصل نتایج مستخرج از پایان‌نامه / رساله و تمامی طرح‌های تحقیقاتی دانشگاه باید با هماهنگی استاد راهنما یا مجری طرح از طریق حوزه پژوهشی دانشگاه انجام گیرد.

ماده ۵- این دستورالعمل در ۵ ماده و یک تبصره در تاریخ ۱۳۸۴/۴/۲۵ در شورای پژوهشی دانشگاه به تصویب رسیده و از تاریخ تصویب لازم الاجرا است و هرگونه تخلف از مفاد این دستورالعمل، از طریق مراجع قانونی قابل پیگیری می‌شود.

نام و نام خانوادگی علی‌اصغر‌رضیه
امضاء



۱۳۹۰/۰۶/۰۷

آیین نامه چاپ پایان نامه (رساله) های دانشجویان دانشگاه تربیت مدرس

نظر به اینکه چاپ و انتشار پایان نامه (رساله) های تحصیلی دانشجویان دانشگاه تربیت مدرس، مبین بخشی از فعالیتهای علمی - پژوهشی دانشگاه است بنابراین به منظور آگاهی و رعایت حقوق دانشگاه، دانش آموختگان این دانشگاه نسبت به رعایت موارد ذیل معهد می شوند:

ماده ۱: در صورت اقدام به چاپ پایان نامه (رساله) خود، مراتب را قبلاً به طور کتبی به «دفتر نشر آثار علمی» دانشگاه اطلاع دهد.

ماده ۲: در صفحه سوم کتاب (پس از برگ شناسنامه) عبارت ذیل را چاپ کند:
«کتاب حاضر، حاصل پایان نامه کارشناسی ارشد نگارنده در رشته مهندسی کامپیوتر گرایش نرم افزار است که در سال ۱۳۹۳ در دانشکده مهندسی برق و کامپیوuter دانشگاه تربیت مدرس به راهنمایی جناب آقای دکتر محمد

صنیعی آباده، مشاوره جناب آقای دکتر مهدی آبادی از آن دفاع شده است.

ماده ۳: به منظور جبران بخشی از هزینه های انتشارات دانشگاه، تعداد یک درصد شمارگان کتاب (در هر نوبت چاپ) را به «دفتر نشر آثار علمی» دانشگاه اهدا کند. دانشگاه می تواند مازاد نیاز خود را به نفع مرکز نشر در معرض فروش قرار دهد.

ماده ۴: در صورت عدم رعایت ماده ۳، ۵۰٪ بهای شمارگان چاپ شده را به عنوان خسارت به دانشگاه تربیت مدرس، تأديه کند.

ماده ۵: دانشجو تعهد و قبول می کند در صورت خودداری از پرداخت بهای خسارت، دانشگاه می تواند خسارت مذکور را از طریق مراجع قضایی مطالبه و وصول کند؛ به علاوه به دانشگاه حق می دهد به منظور استیفای حقوق خود، از طریق دادگاه، معادل وجه مذکور در ماده ۴ را از محل توقيف کتابهای عرضه شده نگارنده برای فروش، تامین نماید.

ماده ۶: اینجانب علیرضا قصبه دانشجوی رشته مهندسی کامپیوتر گرایش نرم افزار مقطع کارشناسی ارشد تعهد فوق وضمانت اجرایی آن را قبول کرده، به آن ملتزم می شوم.

نام و نام خانوادگی: علیرضا قصبه

تاریخ و امضاء: ۱۳۹۳/۱۱/۲۰





دانشگاه تربیت مدرس

دانشکده مهندسی برق و کامپیوتر

پایان نامه کارشناسی ارشد رشته مهندسی کامپیوتر گرایش نرم افزار

عنوان پایان نامه

کشف انجمن ها در شبکه های اجتماعی با استفاده از بهینه سازی کندوی زنبور عسل

نگارش

علیرضا قصبه

استاد راهنما:

دکتر محمد صنیعی آباده

استاد مشاور:

دکتر مهدی آبادی

۱۳۹۳ بهمن

تعددیم به

پدر بزرگوارم و مادر محربانم

به پاس حیات های بی دینشان

به پاس فداکاری و گذشت بی پیاشان

به پاس سپید کشتن مویشان برای سپید کشتن رویان

به پاس قلب های بزرگشان و صبر و برباری فراوانشان

به پاس عمر کر اقدر و پر بهایشان که به پای امنیت و آسایش و آرامش و کمال ماندشت

پیشکش قدرهای دمقابل دریابی

پاسکزاری:

خدای مهربانم با تعبیری از خودت شروع می‌کنم

اگر تمام دخان روی زمین قلم شوند

اگر تمام دیاهام رکب شوند و پس از آن هفت دیاهی دیگر به وجود آید

اگر تک تک بخطات عمرم را صرف کنم

باز هم نمی‌توانم سکر کی از نعمت‌های تورابه جا آورم.

صمیمانه ترین مشکل‌ها و قدردانی‌ها را تقدیم به جناب آقا‌ی دکتر محمد صنیعی، استاد راهنمایی بزرگوارم می‌کنم. و از ایشان برای حسن

خدمتشان، برای راهنمایی‌های بی‌دین شان، برای صرف وقت فراوانشان و امیدواری‌هایی دلکرم کننده شان نهایت

پاسکزاری را دارم.

از جناب آقا‌ی دکتر مهدی آبادی، استاد مشاور عزیزم به خاطر راهنمایی‌های ارزشمند شان و دلکرمی‌هایشان کمال مشکل و قدردانی را

دارم.

از همه‌ی کسانی که در طول زندیم از معلمین و استادی بزرگوارم، از دوستان همیشه همراهم، از خویشان مهربانم و از تک تک

کسانی که حتی کوچکترین کاری را برای من انجام داده‌اند، تامن در این لحظه توفیق رسیدن به این مرحله از زندگی را داشته باشم، بی

نهایت پاسکزارم.

چکیده

شبکه‌های اجتماعی در واقع نوعی ساختار اجتماعی محسوب می‌شوند که از گره‌های متعددی تشکیل شده‌اند. این گره‌ها می‌توانند افراد حقیقی یا سازمان‌ها باشند. این گره‌ها توسط یک یا چند نوع خاص وابستگی مانند روابط دوستانه، خویشاوندی، همسهری بودن، روابط تجاری، روابط علمی، به یکدیگر متصل هستند. باگسترده شدن اینترنت و وب و همچنین افزایش چشمگیر گوشیهای هوشمند در سال‌های اخیر، شبکه‌های اجتماعی به یکی از ارکان جدایی ناپذیر زندگی ما، حداقل در بعد مجازی تبدیل شده‌اند. تحلیل شبکه‌های اجتماعی در واقع بررسی رابطه میان افراد، شامل بررسی ساختارهای اجتماعی، موقعیت اجتماعی، تحلیل نقش‌ها و موارد دیگر است. یکی از مهمترین ساختارهای اجتماعی انجمن‌ها هستند. انجمن به گروهی از رئوس گفته می‌شود که پیوندهای درونی افراد، بیشتر از پیوندهای آنها با خارج از گروه است.

کشف انجمن‌ها یکی از زمینه‌های تحقیقاتی است که محققان زیادی را در سال‌های اخیر به خود جلب کرده‌اند. پاسخ‌های بسیاری برای این مسئله ارائه شده است ولی هنوز این مسئله به طور رضایتمدانه‌ای حل نشده است. در این پایان‌نامه از ترکیب ایده‌ی خوشه‌بندی کلونی مورچه با بهینه‌سازی کندوی زنبور عسل استفاده شده است. خوشه‌بندی کلونی مورچه که یک جستجوی محلی است، توسط بهینه‌سازی کندوی زنبور عسل که یک راهکار سراسری است هدایت می‌شود. همچنین مدلی برای تخصیص زنبورهای رقصنده پیشنهاد شده است. راهکارهای پیشنهادی در کنار هم، باعث شده اند تا انجمن‌ها به طور دقیق تر و سریع تری کشف شوند. در واقع از زنبورهای رقصنده برای رد و بدل کردن اطلاعات میان گره‌ها، که گره در واقع همان مورچه در الگوریتم خوشه‌بندی کلونی مورچه است، استفاده شده است. نتایج بر روی شبکه‌های واقعی و گراف‌های مصنوعی تولید شده توسط کامپیوتر کارایی روش پیشنهادی را به خوبی نشان می‌دهد، زیرا که روش پیشنهادی توانست مصالحه‌ی خوبی میان دقت الگوریتم و نیز پیچیدگی زمانی برقرار نماید. همچنین الگوریتم برروی مجموعه داده دلفین‌ها به دقت ۱۰۰ درصد دست پیدا کرد. و نیز در آزمون جی ان و لانیچیتی، زمانیکه پارامتر آمیزش از ۵,۰ گذر می‌کند، به دقت‌های بهتری نسبت به سایرین دست پیدا کرده است. همچنین در این پایان‌نامه نشان داده شد که هرچه ابعاد مجموعه داده‌ها بالاتر برود، الگوریتم دقت‌های بهتری را کسب می‌کند.

کلید واژه: شبکه‌های اجتماعی، کشف انجمن‌ها، خوشه‌بندی کلونی مورچه، بهینه‌سازی کندوی زنبور عسل، گراف کاوی

فهرست مطالب

۵ فهرست شکل‌ها
۶ فهرست جدول‌ها
۱ فصل ۱ - کلیات
۲ ۱-۱ پیشگفتار
۴ ۲-۱ تعریف مسئله
۷ ۳-۱ پیچیدگی محاسبات
۹ ۴-۱ اهداف پایان نامه و نوآوری‌ها
۱۰ ۵-۱ ساختار پایان نامه
۱۱ فصل ۲ - مفاهیم پایه
۱۲ ۱-۲ داده کاوی
۱۵ ۲-۲ روش‌های توصیفی یا بدون نظارت
۱۵ ۳-۲ خوشبندی
۱۸ ۴-۲ الگوریتم‌های فرامکاشفه‌ای زیستی
۱۹ ۱-۴-۲ دورنمای برآش
۲۱ ۲-۴-۲ قابلیت‌های پویش و انتفاع
۲۳ ۳-۴-۲ طبقه بندی الگوریتم‌های فرامکاشفه‌ای
۲۴ ۴-۲ الگوریتم‌های مبتنی بر هوش جمعی
۲۵ ۱-۵-۲ روش‌های تقلید-محور
۲۶ ۲-۵-۲ روش‌های علامت محور
۲۹ ۶-۲ بهینه‌سازی کندوی زنبور عسل
۳۳ ۷-۲ الگوهای گرافی در شبکه‌های واقعی
۳۳ ۱-۷-۲ شبکه‌های مقیاس آزاد
۳۴ ۲-۷-۲ پدیده جهان کوچک
۳۵ ۳-۷-۲ انجمن‌ها
۴۱ ۸-۲ شبکه‌های اجتماعی
۴۲ ۹-۲ تاریخچه کوتاهی از شبکه‌های اجتماعی برخط
۴۶ فصل ۳ - پژوهش‌های پیشین
۴۷ ۱-۳ مقدمه
۴۷ ۲-۳ روش‌های سنتی
۴۷ ۱-۲-۳ جزء بندی گراف
۴۹ ۲-۲-۳ خوشبندی سلسله مراتبی

۵۲	- خوشبندی طیفی	۳-۲-۳
۵۳	- الگوریتم‌های تقسیمی	۳-۳
۵۵	- روش‌هایی بر پایه پودمانگی	۴-۳
۵۶	- بهینه‌سازی پودمانگی	۱-۴-۳
۶۱	- روش‌هایی مبتنی بر مکاشفه‌های زیستی	۴-۳
۶۴	- مروری بر روش خوشبندی کلونی مورچه	۶-۳
۶۴	- جمع بندی	۷-۳
۶۶	فصل ۴ - روش پیشنهادی	
۶۷	- توصیف کلی	۱-۴
۷۱	- توصیف جزئیات الگوریتم به تفکیک	۲-۴
۷۱	- محل زندگی	۱-۲-۴
۷۲	- درک محل زندگی	۲-۲-۴
۷۷	- پارامتر آستانه حرکت و تنظیم کردن سازگارانه‌ی آن	۳-۲-۴
۷۸	- استراتژی حرکت	۴-۲-۴
۸۰	- نحوه بروزرسانی زنبورهای رقصنده و جدول رقص	۵-۲-۴
۸۵	- پیچیدگی زمانی الگوریتم	۳-۴
۸۷	- جمع بندی	۴-۴
۸۸	فصل ۵ - نتایج و ارزیابی روش پیشنهادی	
۸۹	- مقدمه	۱-۵
۸۹	- مجموعه داده‌ها	۲-۵
۸۹	- شبکه‌های واقعی	۱-۲-۵
۹۲	- شبکه‌های مصنوعی یا تولیدشده توسط کامپیوتر	۲-۲-۵
۹۴	- زبان مدل کردن گراف	۳-۲-۵
۹۶	- معیارهای ارزیابی	۳-۵
۹۷	- تنظیم پارامترها	۴-۵
۹۸	- بررسی تاثیر پارامتر آستانه حرکت بر الگوریتم	۱-۴-۵
۱۰۱	- ارزیابی و مقایسه روش پیشنهادی با روش‌های دیگر	۵-۵
۱۰۱	- مقایسه میان شبکه‌های واقعی	۱-۵-۵
۱۰۴	- مقایسه با شبکه‌های تولیده شده توسط کامپیوتر	۲-۵-۵
۱۰۹	- کارآیی الگوریتم در ابعاد بالا	۶-۵
۱۱۰	فصل ۶ - نتیجه گیری و پیشنهادات	
۱۱۱	- نتیجه گیری	۶-۱

۱۱۲	-۲- کارهای آتی
۱۱۴	فهرست مراجع
۱۱۹	واژه نامه‌ی فارسی به انگلیسی

فهرست شکل‌ها

صفحه

عنوان

۱-۱	یک گراف ساده با سه انجمن که خط چین از هم جدا شده است.	۷
۱-۲	سلسله مراتب داده تا خرد	۱۲
۱۴	مراحل مختلف داده کاوی	۲-۲
۲۰	نمونه از یک دورنمایی برآش یک بعدی که در آن هدف یافتن بلندترین قله است.	۳-۲
۲۱	نمونه‌ای از دورنمایی برآش در فضای سه بعدی	۴-۲
۲۸	رونده کلی یک الگوریتم علامت-محور	۵-۲
۳۲	شبکه کد ساده الگوریتم بهینه‌سازی کندوی زنبور عسل [۲]	۶-۲
۳۴	نمایش توزیع‌های نرمال و قانون توانی [۳]	۷-۲
۴۲	شبکه اجتماعی متشكل از افراد و ارتباطات آنها	۸-۲
۴۴	رشد قارچ گونه شبکه‌های اجتماعی برخط	۹-۲
۴۷	خط چین نمایش راه حل مسئله دوبخشی کمینه برای گراف ذکر شده است [۱]	۱-۳
۵۱	یک دندروگرام یا درخت سلسله مراتبی. برش افقی یک جزء بندی را که یک معیار خاص را ارضا می‌نماید نشان میدهد [۶]	۲-۳
۷۰	فلوچارت کلی الگوریتم در یک نگام	۴۱
۷۲	محل زندگی مورچه یا ماتریس باید به صورت کروی در نظر گرفته شود	۲-۴
۷۳	نمونه‌ای از یک مورچه در حال پردازش و همسایگی آن. در این مثال محدوده دید افقی و محدوده دید عمودی هردو یک هستند	۳-۴
۷۶	اگر فاصله میان همه مورچه‌ها برابر صفر باشد، در این صورت برآش مورچه سیاه در هردو حالت برابر بک خواهد بود	۴-۴
۷۹	طرح کلی حرکت مارپیچی در محیط زندگی یا همان ماتریس	۵-۴
۸۴	اعداد، درجه یا مرحله‌ی همسایگی مکان‌های همسایه را نسبت به گره i نشان میدهند	۶-۴
۸۶	شبکه کد ساده الگوریتم	۷-۴
۹۰	شبکه باشگاه کارانه زاخاری. مربع یک انجمن و دایره هم انجمن دیگر را نمایش می‌دهد	۱-۵
۹۱	شبکه‌های دلفین‌های دست آموز، دایره نشان دهنده یک انجمن و مربع نشان دهنده انجمن دیگر است	۲-۵
۹۳	یک نمونه از آزمون لایچینتی با 500 گره [۵]	۳-۵
۹۵	مثالی ساده از یک گراف در قالب <i>GML</i>	۴-۵
۹۹	نمایش تاثیر پارامتر آستانه حرکت در شبکه‌های واقعی	۵-۵
۱۰۰	اثر پارامتر آستانه حرکت بر روی شبکه‌های تولیدشده توسط کامپیوترا	۶-۵
۱۰۴	نتایج آزمایشات بر روی آزمون <i>GN</i> . شکل الف از مرجع [۴] و شکل ب راست از [۷] و چپ [۸]	۷-۵
۱۰۵	نتایج بدست آمده از الگوریتم پیشنهادی بر روی آزمون <i>GN</i>	۸-۵

[۴]	نتایج آزمایشات الگوریتم‌های مختلف بر روی آزمون LFR با سایز 1000S, 1000B, 5000S, 5000B	۹-۵
۱۰۶
۱۰۷	نتایج آزمایش الگوریتم CDHHO بر روی شبکه‌های تولید شده توسط روش LFR	۱۰-۵
۱۰۹	۱۱-۵ مقایسه بین دقت‌های کسب شده برای ۵ شبکه با سایز متفاوت و پارامتر آمیزش متفاوت

فهرست جدول‌ها

صفحه

عنوان

۹۸.....	۱-۵ جدول پارامترهای الگوریتم در یک نگاه.....
۱۰۰	۲-۵ پارامترهای تنظیم شده برای الگوریتم.....
۱۰۱	۳-۵ مقایسه روش‌های مختلف با CDHHO بر روی مجموعه داده باشگاه کاراته.....
۱۰۲	۴-۵ مقایسه روش‌های مختلف با CDHHO بر روی مجموعه داده دلفین‌های دست آموز.....
۱۰۳	۵-۵ مقایسه روش‌های مختلف با CDHHO بر روی مجموعه داده فوتیال باشگاه‌های آمریکا.....
۱۰۸	۶-۵ مقایسه پیچیدگی زمانی الگوریتم‌هایی در آزمون LFR روی چهار شبکه شرکت داشته اند.....

فصل ۱- کلیات

۱-۱- پیشگفتار

علم شبکه‌ها، پیشرفت‌های قابل توجهی را در فهم ما از شبکه‌های پیچیده^۱ به ارمنغان آورده است. یکی از پرمعناترین خاصیت نمایش گرافی سیستم‌های حقیقی، ساختار انجمنی^۲ است. گروهی از رئوس که با یالهای زیادی به یکدیگر متصل هستند در حالیکه تعداد یالهای بسیار کمتری، رئوس این گروه را به باقی شبکه متصل می‌نماید، انجمن نامیده می‌شود. به طور منصفانه‌ای یک چنین خوش‌ها یا انجمن‌هایی را می‌توان به عنوان اجزای مستقل یک گراف در نظر گرفت، که همان نقش بافت‌ها یا اعضا را در بدن انسان دارند. کشف انجمن‌ها اهمیت بسیاری در جامعه شناسی، بیولوژی و علوم کامپیوتر و هرجایی که سیستم‌ها را بتوان به صورت گراف مدل کرد دارد. کشف انجمن‌ها از جمله مسائل سخت محسوب می‌شود که هنوز به طور رضایت‌مندانه‌ای حل نشده است، در حالیکه دانشمندان بسیاری در یک دهه گذشته بر روی آن کار و تحقیق انجام داده اند.

تاریخ نظریه گراف تقریباً به سال ۱۷۳۶ باز می‌گردد و از آن موقع به بعد دانش بسیاری درباره گراف و ویژگی‌های ریاضی آن بدست آمد. گراف‌ها در قرن بیست و یکم در نمایش سیستم‌های گوناگونی در حوزه‌های علمی متفاوتی به طور گسترده مورد استفاده قرار گرفتند. شبکه‌های بیولوژیکی^۳، شبکه‌های اجتماعی^۴، شبکه‌های اطلاعاتی^۵ از جمله سیستم‌هایی بودند که به عنوان گراف مورد مطالعه قرار گرفتند، بنابراین تحلیل گراف در فهم ویژگی‌های این سیستم‌ها، بسیار پر اهمیت شد. به عنوان نمونه تحلیل

^۱ Complex Networks

^۲ Community Structure

^۳ Biological Networks

^۴ Social Networks

^۵ Information Networks

شبکه‌های اجتماعی در سال ۱۹۳۰ به عنوان یکی از موضوعات مهم در جامعه شناسی مطرح شد. در سالهای اخیر، انقلاب کامپیوترها، حجم عظیم از داده‌ها و منابع محاسباتی برای پردازش و تحلیل این داده‌ها در اختیار محققین قرار داده است. نیاز به درگیر شدن با چنین شبکه‌های بزرگی نیاز به تغییر مسیری است که در آن گراف‌ها مورد بررسی قرار می‌گرفته‌اند.

نمایش گرافی سیستم‌های حقیقی مثل توری^۱ خیلی با قاعده نیست. آنها اشیایی هستند که در آنها نظم همراه بی نظمی زندگی می‌کند. نمونه گراف‌های بی نظم گراف‌های تصادفی^۲ هستند، که در آن احتمال وجود یک یال میان یک جفت راس با همه جفت راس‌های ممکن برابر است. در گراف تصادفی توزیع یال‌ها میان رئوس، همگن^۳ است. برای مثال توزیع همسایه‌های یک راس یا توزیع درجه، دو جمله‌ای^۴ است بنابراین اکثر راس‌ها درجه یکسان یا شبیهی دارند.

شبکه‌های واقعی گراف تصادفی نیستند. آنها یک عمق زیادی از نظم و سازمان دهنده را آشکار می‌کنند در حالیکه آنها ناهمگونی^۵ بزرگی را نمایش می‌دهند. توزیع درجه در آنها پهن است با یک دنباله‌ای که اغلب از قانون توانی^۶ پیروی می‌کند. بنابراین خیلی از راس‌های با درجه پایین با رئوسی با درجه‌های خیلی بالا همزیستی می‌کنند. علاوه براین توزیع یال نه فقط به طور سراسری که به طور محلی نیز ناهمگون است، که این بدین معناست که تجمع بالایی از یال‌ها میان گروه‌های خاصی از رئوس وجود دارد و یال‌ها متصل کننده این گروه‌ها در بین این گروه‌ها تمرکز و تجمعی کمتری دارند. این ویژگی شبکه‌های اجتماعی

^۱ Lattice

^۲ Random Graphs

^۳ Homogeneous

^۴ Binomial

^۵ Inhomogeneity

^۶ Power Law

“ساختار انجمنی” نامیده می‌شود. انجمن‌ها خوشی یا مازول^۱ نیز نامیده می‌شوند. انجمن‌ها در واقع گروهی از رئوس هستند که احتمالاً ویژگی‌های شبیه‌ی را به اشتراک می‌گذارند و نقش‌های یکسانی در گراف دارند.

۱-۲- تعریف مسئله

تحلیل شبکه‌های اجتماعی^۲ (SNA) در واقع بررسی ارتباطات میان افراد است که شامل تحلیل ساختارهای اجتماعی، موقعیت‌های^۳ اجتماعی، تحلیل نقش‌های^۴ و خیلی چیزهای دیگر است. به طور معمول ارتباط میان افراد یعنی خویشاوندی، دوستی، همسایگی، همکاری، هم‌کلاسی و ... به عنوان یک شبکه یا گراف قابل نمایش است. علوم اجتماعی بدین گونه عمل می‌کردند که با انتشار پرسشنامه‌هایی میان افراد و در خواست جواب از آنها، جزئیات تعاملاتشان با دیگران را استخراج می‌کردند. سپس یک شبکه براساس جواب آنها ساخته می‌شد که گره‌ها نمایانگر افراد و یالها نمایش دهنده‌ی تعاملات میان آنها بودند. این نوع جمع آوری داده تحلیل شبکه‌های اجتماعی سنتی را محدود به مقیاس‌های کوچک می‌کرد. با گسترش اینترنت و وب و پیشرفت فناوری بسیاری از سایت‌های شبکه‌های اجتماعی و رسانه‌های اجتماعی پدیدار گشتند. این پیشرفت‌ها باعث شد مردم به راحتی با یکدیگر در فضای مجازی ارتباط برقرار کنند. این پیشرفت باعث می‌شد که کار تحلیل شبکه‌های اجتماعی در مقیاس‌های بزرگتر نیز آسان شود. شبکه‌های ارتباطی ایمیلی، شبکه‌های پیغام دهی و گفتگو^۵، شبکه‌های موبایلی و شبکه‌های دوستی نمونه‌هایی از این شبکه‌های

^۱ Module

^۲ Social Network Analysis

^۳ Positions

^۴ Roles

^۵ Instant Messaging

اجتماعی مقیاس بزرگ هستند. این شبکه‌های واقعی و بزرگ دارای الگوهایی^۱ هستند که برای SNAها اهمیت زیادی دارند. یکی از پرکاربردترین و مهمترین این الگوها، ساختار انجمنی این شبکه‌ها هستند. افراد یا ایفاکنندگان نقش‌ها^۲ در شبکه‌های اجتماعی گروههایی را تشکیل می‌دهند. بررسی ساختار و توبولوزی شبکه‌ها که منجر به کشف انجمن‌ها می‌شود یکی از وظایف مهم آنهاست.

داده کاوی، پایگاه‌ها و مجموعه‌های حجمی داده‌ها را در پی کشف و استخراج دانش مورد تحلیل و کند و کاوهای ماشینی (و نیمه ماشینی) قرار میدهد. این روش که با حداقل دخالت کاربران همراه است، اطلاعاتی را در اختیار آنها و تحلیلگران قرار میدهد تا براساس آنها تصمیمات مهم و حیاتی را در سازمان مربوطه اتخاذ نمایند. داده کاوی در سالهای اخیر، تأثیرات شگرفی در محیط‌های آکادمیک و صنعتی ایجاد کرده و کاربردهای فراوانی در زمینه‌های مختلف یافته است. به عنوان نمونه می‌توان به کاربردهای تجاری، مدیریت و کشف فریب، پژوهشی، ورزشی، متن کاوی و وب کاوی اشاره نمود.

این فرایند از دو مرحله اصلی تشکیل شده است [۹]. مرحله اول پیش پردازش داده‌ها است که شامل پاکسازی، یکپارچه سازی، انتخاب صفات و تبدیل داده‌ها به قالب مورد استفاده برای داده کاوی می‌باشد. در مرحله دوم، داده‌های بدست آمده از مرحله اول به منظور تشخیص الگو مورد استفاده قرار می‌گیرند که این امر به کمک الگوریتم‌هایی نظریه بندی^۳ و خوشبندی^۴ صورت می‌گیرد. در این مرحله برای کشف الگو یک مدل یادگیری ایجاد می‌شود که این به مدل به مرور زمان و با تکرار فرآیند داده کاوی بهبود می‌یابد. خروجی این مرحله دانش کسب شده است که به کمک ابزارهای موجود قابل نمایش است.

^۱ Patterns

^۲ Actors

^۳ Classification

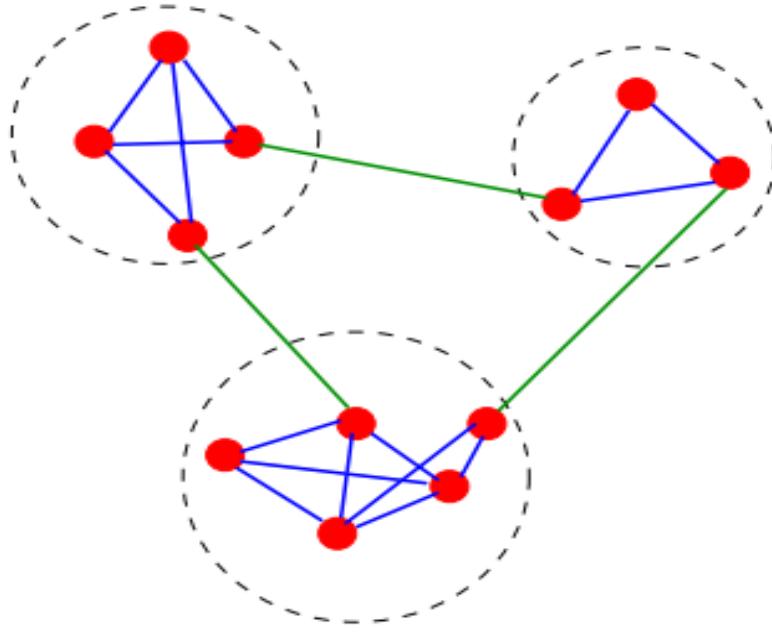
^۴ Clustering

به طور کلی تکنیک‌های داده کاوی به دو گروه با نظارت^۱ و بدون نظارت^۲ تقسیم بندی می‌شوند. اکثر روش‌های داده کاوی روش‌های با نظارت می‌باشند که در این روشها یک متغیر هدف از قبل تعریف شده، وجود دارد. در این روشها نمونه‌های زیادی وجود دارند که مقدار متغیر هدف برای آنها از قبل مشخص می‌باشد، بنابراین الگوریتم می‌تواند به کمک آنها آموزش ببیند و دریابد که متغیرها و صفات توصیف کننده یک نمونه با کدام مقدار متغیر هدف متناظر می‌باشد. اما در روش‌های بدون نظارت متغیر هدفی تعریف نمی‌شود و الگوریتم داده کاوی همبستگی‌ها و ساختارهای بین تمام نمونه‌ها را جستجو می‌کند. از مهمترین روش‌های داده کاوی بدون نظارت، خوشه بندی را می‌توان نام برد.

همانگونه که بیان شد شبکه‌های اجتماعی را می‌توان با یک گراف نمایش داد. و ما می‌خواهیم در این گراف به یافتن الگوهایی به نام انجمن بپردازیم. انجمن به گروهی از رئوس گفته می‌شود که تعداد اتصالات داخلی گروهی آنها بسیار زیادتر از اتصالات مابین گروهی آنهاست. این تعریف کلی انجمن است. بنابراین مسئله کشف انجمن در داده کاوی همان مسئله خوشه‌بندی است. اما تفاوت عمدی و نکته اصلی این است که داده‌ها در اینجا گره‌های یک گراف هستند و باید به ارتباطات میان این گره‌ها در گراف برای خوشه‌بندی توجه شود.

^۱ Supervised

^۲ Unsupervised



۱-۱ یک گراف ساده با سه انجمن که خط چین از هم جدا شده است

۱-۳-۳ - پیچیدگی محاسبات

حجم عظیمی از داده در شبکه‌های واقعی که اکنون در دسترس است، بحث کارآیی الگوریتم‌های خوشبندی را ضروری کرده است. پیچیدگی محاسباتی یک الگوریتم در واقع تخمین میزان منابعی است که الگوریتم برای انجام یک وظیفه نیاز دارد. در این بحث هم گام‌های محاسباتی و هم تعداد واحدهای حافظه‌ای که نیاز است به طور همزمان برای اجرای محاسبات اختصاص داده شود، درگیر می‌شوند. چنین نیازی، به وسیله مقیاس و اندازه سیستمی که در حال مطالعه است بیان می‌شود. که در مورد گراف این اندازه به وسیله تعداد گره‌ها n و/یا تعداد یالهای گراف m نشان داده می‌شود. پیچیدگی محاسباتی یک الگوریتم همیشه قابل محاسبه نیست. در حقیقت این کار گاهی اوقات سخت یا غیرممکن است. که در این موارد حداقل بدست آوردن تخمینی از بدترین حالت پیچیدگی الگوریتم اهمیت دارد. که یعنی بدست آوردن میزان منابع محاسباتی ای که برای اجرای الگوریتم در شرایط غیردلخواه و نامطلوب نیاز است. نماد $O(n^\alpha m^\beta)$ بیانگر این است که زمان محاسبات با توانی از تعداد رؤوس n و تعداد یالها m به ترتیب با نمای α و β رشد می‌کند. نمونه‌های گراف و بی که از میلیون‌ها راس و میلیاردها یال تشکل شده است نمی‌تواند با الگوریتم‌هایی که رشدی سریع تر از $O(n)$ و $O(m)$ دارند، درگیر شوند.

الگوریتم‌هایی با پیچیدگی چندجمله‌ای کلاس P را تشکیل می‌دهند. برای بعضی از مسائل بهینه‌سازی و تصمیم گیری مهم الگوریتم‌های چندجمله‌ای شناخته شده‌ای وجود ندارد. پیدا کردن راه حل‌هایی برای اینگونه مسائل نیازمند یک جستجوی بسیار جامع است که رشد آن بسیار سریع تر از هر تابع چندجمله‌ای از اندازه سیستم است. به طور مثال به طور نمایی رشد می‌کند. مسائلی که راه حل آنها می‌تواند در یک زمان چند جمله‌ای قابل حل باشد، کلاس^۱ NP را تشکیل می‌دهند که شامل کلاس P نیز می‌شود. یک مسئله NP-hard است اگر بتوان راه حل آن را به گونه‌ای تبدیل یا ترجمه کرد که به عنوان راه حلی برای همه مسائل NP باشد. اگرچه همه مسائل NP-hard در کلاس NP قرار نمی‌گیرند. اما اگر متعلق به کلاس NP بود، آنگاه NP-کامل در نظر گرفته می‌شوند.

خیلی از مسائل و الگوریتم‌های خوشبندی و یا مسائل مرتبط با آن NP-hard هستند. در این موارد خیلی بی معنی است که از آن راهکارها برای حل کردنشان استفاده شود، مگر اینکه برای سیستم‌هایی با اندازه کوچک مورد استفاده قرار گیرند. مضاف براین اگر یک الگوریتم پیچیدگی زمانی چندجمله‌ای نیز داشته باشد، هنوز هم برای مسائل با مقیاس بزرگ بسیار آهسته هستند. در اینگونه موارد خیلی معمول است که الگوریتم‌های تخمینی استفاده شود. یعنی روش‌هایی که به یک راه حل دقیق برای مسئله‌ای که در دست هست نمی‌رسند اما یک راه حل تخمینی^۲ را ارائه می‌کنند که مزیت اصلی آن پیچیدگی کمتر است. الگوریتم‌های تخمینی معمولاً غیر قطعی^۳ هستند بدین معنی که برای شرایط اولیه و یا پارامترهای اولیه متفاوت، آنها راه حل‌های متفاوتی را برای یک مسئله ارائه می‌دهند. هدف چنین الگوریتم‌هایی این هست که با یک فاکتور ثابتی، از راه حل بهینه تفاوت یا فاصله داشته باشد. الگوریتم‌های تخمینی به طور معمول در مسائل بهینه‌سازی استفاده می‌شوند که در آن می‌خواهیم مقدار بیشینه یا کمینه یک تابع هزینه را در یک مجموعه بزرگی از پیکربندی سیستم پیدا کنیم.

^۱ Non-deterministic polynomial

^۲ Approximation Solution

^۳ Non deterministic

۴-۱- اهداف پایان نامه و نوآوری‌ها

در این پایان نامه، یک روش و الگوریتم برای کشف انجمن‌های شبکه‌های اجتماعی ارائه شده است. برای این منظور سعی شد دو رویکرد کلی که در مکانیزم‌های جستجو وجود دارند، به طور همزمان مورد توجه قرار داده شود. یعنی هم از روش بهینه‌سازی سراسری^۱ بهره گرفته شد و هم روش‌های جستجوی محلی^۲ را مورد توجه قرار گرفت. برای همین منظور اساس الگوریتم پیشنهادی بر پایه خوشه‌بندی کلونی مورچه^۳ که رهیافتی محلی است بنا گردید و سپس بهینه‌سازی زنبور عسل^۴، که رهیافتی سراسری است با آن ادغام شد. تا هم بتوان بر چالش پیچیدگی الگوریتم غلبه کرد و هم مصالحه^۵ میان آن و دقت روش پیشنهادی را که از اهداف مهم این پژوهش است، برقرار نمود.

از جنبه‌های نوآوری که در این پایان نامه وجود دارند، می‌توان به موارد زیر اشاره کرد:

- ترکیب دو مکانیزم سراسری و محلی خوشه‌بندی کلونی مورچه با بهینه‌سازی کندوی زنبور عسل
- سازگار کردن و تطبیق دادن بهینه‌سازی کندوی زنبور عسل و ارائه مدلی برای به روزرسانی زنبورهای رقصنده در میان مورچه‌ها
- استفاده از تابع برازش جدید

^۱ Global Optimization approach

^۲ Local Search approach

^۳ Ant Colony Clustering

^۴ Honeybee Hive Optimization

^۵ Trade off

۱-۵- ساختار پایان نامه

در فصل اول این پایان نامه که همین فصل جاری محسوب می‌شود، سعی شد به بیان کلیات مساله کشف انجمن‌ها و اهمیت تحلیل شبکه‌های اجتماعی و روش پیشنهادی و جنبه‌های نوآوری پرداخته شود.

اما مطالبی که در فصول آینده مورد بررسی قرار می‌گیرند به شرح زیر است:

- در فصل دوم مفاهیم پایه‌ای داده کاوی و روش‌های بدون نظارت و خوشبندی و الگوریتم فرامکاشفه‌ای علامت محور و به خصوص الگوریتم بهینه‌سازی کندوی زنبور عسل بیان خواهد شد.
- در فصل سوم به تحقیقات صورت گرفته و روش‌های موجود در زمینه کشف انجمن‌ها با ساختار مناسب مربوط به آنها می‌پردازیم.
- فصل چهارم مدل و روش پیشنهادی برای کشف انجمن‌ها در شبکه‌های اجتماعی را به طور کامل معرفی می‌کند.
- فصل پنجم به معرفی مجموعه داده‌ها و معیارهای ارزیابی می‌پردازیم. و در نهایت نتایج بدست آمده‌ی حاصل از اجرای الگوریتم را نشان داده و به مقایسه‌ی این روش با برخی روش‌های موجود همت گماشتیم.
- فصل ششم نیز حاوی خلاصه مطالب پایان نامه، نتیجه گیری و ارائه پیشنهاداتی در زمینه کاری پایان نامه می‌باشد.

فصل ۲ – مفاهیم پایه

۱-۲ - داده کاوی

برای درک کامل معنای داده کاوی می‌بایست ابتدا تعریف درستی از معانی کلمات داده، اطلاعات، دانش و خرد داشته باشیم.

• **داده^۱:** مفهومی پایه‌ای که به دور از هر پردازشی است. مثلا سیمبول، کاراکتر، عدد و یا سیگنال

که معنی خاصی به تنها بی ندارند. مثلا عدد ۷۵

• **اطلاعات^۲:** اگر کنار هر عنصر داده‌ای رشته‌ای هم به عنوان توصیف کننده‌ی معنای آن بیاد،

داده اولیه به اطلاعات تبدیل شده است. به عبارتی داده‌ای درباره داده است. مثلا سن ۷۵

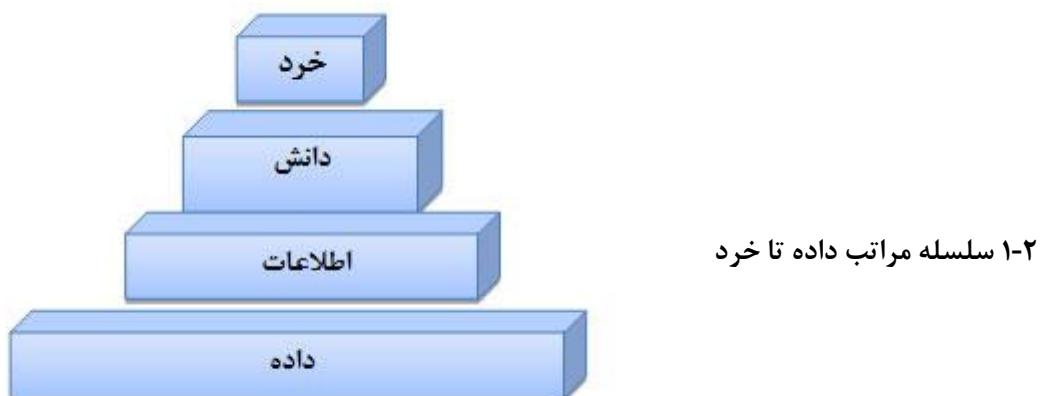
• **دانش^۳:** رابطه میان دو عنصر اطلاعاتی، دانشی را در آن زمینه آشکار می‌کند. به عبارتی

اطلاعاتی درباره اطلاعات است. مثلا: اگر کارمندی به سن ۷۵ سالگی بررسد حتما بازنشسته

شده است.

• **خرد^۴:** بالاترین سطح بینش است که توسط علائم و نمادهای قراردادی تبیین می‌شود. به

عبارة دانشی در باره دانش است.



^۱ Data

^۲ Information

^۳ Knowledge

^۴ Wisdom

شکل ۱-۲۱ در واقع سلسله مراتب ارزشی مفاهیم داده تا خرد را نمایش می‌دهد. که همزمان با افزایش ارزش معنایی مفاهیم، از میزان آن و حجم آنها کاسته می‌شود.

تعاریف زیادی برای داده کاوی موجود است که ما از یک تعریف جامع و کامل استفاده می‌کنیم:

“استخراج خودکار دانش جدید و مفید از منابع داده‌ای حجمی موجود طی یک فرآیند غیر بدیهی مشخص، داده کاوی نامیده می‌شود [۱۰].”

هدف اصلی داده کاوی کشف دانش است. این دانش در واقع نظمی است که در داده‌ها وجود دارد. علم داده کاوی از علوم مختلفی همچون آمار، پایگاه داده، یادگیری ماشین، هوش مصنوعی و شناسایی الگو نشات گرفته است. شاید بتوان از مهمترین نقاط ضعف داده کاوی را وجود داده، صحت داده و کافی بودن ویژگی‌ها نام برد. اما موارد و چالش‌های اصلی که انگیزه استفاده از داده کاوی را به جای روش‌های سنتی تحلیل داده وجود دارند، شامل حجم بودن داده‌ها، ابعاد بالای داده، طبیعت توزیع شده و ناهمگن داده است.

• حجم بالای داده: الگوریتم‌های داده کاوی با تعداد بسیار زیاد از رکوردها سر و کار دارند و

حجم بالایی از آنها را مورد پردازش قرار می‌دهند. حجم رکوردها با ناتوانی روش سنتی در تحلیل آنها و همچنین نمایش هنر داده کاوی رابطه‌ی مستقیم دارد.

• ابعاد بالای داده: منظور از بعد همان ویژگی یا خصیصه است. وجود تعداد بالای ویژگی کار

تحلیل داده‌ها را با مشکل رویرو می‌کند. پیدا کردن نظم‌هایی میان داده‌های با ابعاد بسیار سخت تر خواهد شد.

• طبیعت ناهمگن داده‌ها: ورودی فرآیند داده کاوی انبارهای از داده‌ها با انواع مختلفی از

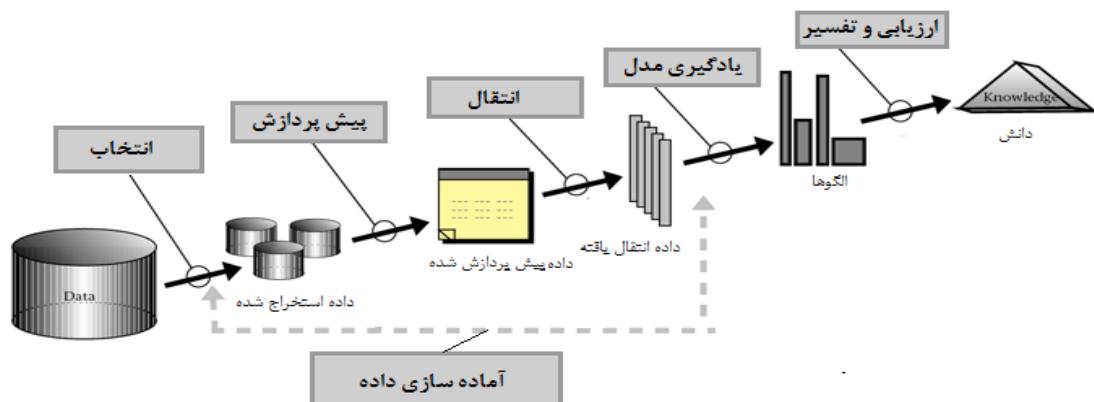
ویژگی‌هاست که هر ویژگی نیز نوع خاص خود و محدوده مقادیر خاص خود را داراست. بعضی

ویژگی‌ها گسسته‌اند، بعضی پیوسته‌اند، بعضی تک مقداری هستند بعضی چند مقداری هستند.

وجود همین تنوع و گوناگونی در ماهیت ویژگی‌ها، کار تحلیل داده را سخت‌تر می‌کند.

علاوه بر چالش‌های اصلی که بدان اشاره شد، موارد دیگر نیز وجود دارند که به آنها چالش‌های ثانویه می‌گویند. که شامل کیفیت داده^۱، عدم مالکیت داده^۲، حفظ حریم شخصی^۳ و داده‌های جریانی^۴ می‌باشد.

فرآیند داده کاوی شامل سه مرحله کلی است: آماده سازی داده، یادگیری مدل، ارزیابی و تفسیر. هدف مرحله اول تامین ورودی مناسب برای مرحله حیاتی یادگیری مدل است. در این مرحله با استفاده از الگوریتم‌های متنوع و با توجه به ماهیت داده نظم‌های مختلف موجود در داده شناسایی و در قالبی مشخص به عنوان دانش نهفته در داده‌ها ارائه می‌شود. مرحله سوم نیز تایید صحت دانش تولید شده است تا بتوان به آن اعتماد کرد.



۲-۲ مراحل مختلف داده کاوی

^۱ Data Quality

^۲ Data Ownership

^۳ Privacy Preservation

^۴ Streaming Data

روش‌های یادگیری مدل را به دو گروه کلی روش‌های پیش‌بینی^۱ و روش‌های توصیفی طبقه بندی می‌کنند که به ترتیب با نام روش‌های با ناظر و روش‌های بدون نظارت نیز شناخته می‌شوند.

از روشن اول برای پیش‌بینی کردن مقدار یک ویژگی مشخص استفاده می‌کنند. دسته بندی، رگرسیون و تشخیص انحراف سه روش یادگیری با ماهیت پیش‌بینی هستند. توضیح بیشتر را در فصل قبل ارائه کردیم و بیش از این نیز با این روش‌ها در این پایان نامه سر و کار نداریم.

۲-۲- روشن‌های توصیفی یا بدون نظارت

این روش‌ها الگوهای قابل توصیفی را پیدا می‌کنند که روابط حاکم بر داده‌ها را بدون در نظر گرفتن هر گونه برچسب یا متغیر خروجی تبیین نمایند. از آنجاییکه که مثال‌هایی که به الگوریتم یادگیری داده می‌شود، بدون برچسب هستند بنابراین هیچ سیگنال جریمه یا پاداشی برای ارزیابی راه حل وجود ندارد. در متون علمی مختلف روش‌های توصیفی را با نام روش‌های بدون ناظر نیز معرفی کرده‌اند. روش‌های کاوش قوانین انجمنی، الگوریتم‌های ترتیبی و خوشبندی سه روش یادگیری مدل در داده کاوی با ماهیت توصیفی هستند.

۳-۲- خوشبندی

همان طور که ذکر شد خوشبندی یکی از روش‌های یادگیری بدون نظارت است. یعنی هیچ خروجی از قبل تعیین شده‌ای برای آن وجود ندارد. عدم وجود برچسب در مسائل خوشبندی سبب می‌شود که هر الگوریتم خوشبندی یک الگوریتم بدون ناظر به حساب آید. همانگونه که می‌دانیم این روش‌ها مراحلی را تحت عنوان آموزش و ارزیابی ندارند و در پایان عملیات نیز مدل ساخته شده (که عملاً همان خوشبندی را ایجاد شده است) به همراه کارآیی آن به عنوان خروجی ارائه می‌شود. در مسائل خوشبندی یک مجموعه رکورد داریم که هر کدام یک مجموعه ویژگی‌ها را دارا هستند. سپس یک معیار مشابه میان آنها تعریف

^۱ Prediction

می‌کنیم. این معیار مشابهت در مسائل مختلف متفاوت هستند. به عنوان نمونه اگر ویژگی‌ها پیوسته باشند می‌توان فاصله اقلیدسی را به عنوان معیار مشابهت در نظر گرفت. از همین رو هر رکورد را می‌توان متناظر با یک نقطه در یک فضای چند بعدی در نظر گرفت. در واقع ابعاد، ویژگی‌های داده‌های ورودی را تشکیل می‌دهند. در خوشبندی، هیچ گونه طبقه یا دسته وجود ندارد که به واقع ویژگی دسته ندارد و تنها بر پایه معیار مشابهت گروه بندی انجام می‌پذیرد. خوشبندی به این شکل انجام می‌شود که رکوردهایی که بیشترین شباهت را به یکدیگر دارند در یک خوش قرار می‌گیرند. که این نیز با توجه به معیار شباهت انتخاب شده انجام می‌شود. بنابراین داده‌های موجود در خوشبندی متفاوت، شباهت کمتری با یکدیگر دارند. خروجی الگوریتم‌های خوشبندی دوباره تحلیل خواهد شد تا در صورت امکان نظمی در خوشبندی آشکار شود. یکی از نکاتی که حائز اهمیت است این است که خوشبندی همیشه بر اساس ویژگی‌های ورودی نمونه‌ها انجام می‌شود. برای مثال در صورت خوشبندی رکوردهای دانشجویان یک دانشکده، هر خوش ممکن است بیانگر رکوردهایی باشد که از جنبه‌های مختلف به یکدیگر شباهت دارند. مثلاً یک وضعیت نشانده‌نده دانشجویان تبل و زرنگ است. ممکن است این دو خوش بیانگر دانشجویان دختر و پسر باشد. ممکن است وضعیتی بیانگر دانشجویان پژوهشند و غیر پژوهشند باشند. وضعیتی ممکن است نشانده‌نده دانشجویان شبانه و روزانه باشد. می‌توان وضعیت‌های دیگری که تعداد خوشبندی نیز متفاوت باشد در نظر گرفت. اما این که کدام از یک از وضعیت‌های احتمالی رخ دهد رابطه و بستگی مستقیمی با ویژگی‌های انتخاب شده برای مقایسه دارد و مستقیماً به الگوریتم خوشبندی مرتبط نیست. وظیفه‌ی ایده آل در همه‌ی الگوریتم‌های خوشبندی کمینه کردن فاصله درون خوشبندی و بیشینه کردن فاصله بین خوشبندی است. زمانی عملکرد خوب یک روش پدیدار می‌شود که تا حد اکثر امکان، خوشبندی را از یکدیگر دورتر کند و داده‌های درون یک خوش حد اکثر شباهت را به هم داشته باشند.

عملیات خوشبندی معمولاً با دو هدف انجام می‌پذیرند:

۱. فهم^۱: بررسی رکوردهای مشابه در مجموعه داده‌ها. به عنوان مثال اسناد مرتبط با هم، گروه‌های

سهام با نوasanات قیمت مشابه

۲. خلاصه سازی^۲: کاهش اندازه‌ی مجموعه داده‌ها بزرگ.

مثال‌هایی از کاربردهای خوشبندی را در زمینه‌های مختلف در ادامه معرفی می‌کنیم:

۱. بازاریابی: خوشبندی مشتریان با توجه به رفتارها و نیازهای آنها از طریق مجموعه ویژگی‌های

مشتریان و نیز خرید آنها

۲. زیست شناسی: خوشبندی حیوانات و گیاهان از روی ویژگی‌های آنها

۳. کتابداری: خوشبندی کتاب‌ها براساس موضوع و دیگر ویژگی‌های موجود در کتابشناسی

۴. نقشه برداری شهری: خوشبندی خانه‌ها براساس نوع و موقعیت جغرافیایی آنها

۵. مطالعات زلزله نگاری: تشخیص مناطق حادثه خیز براساس مشاهدات قبلی

۶. وب: خوشبندی اسناد (مثلا اخبار) و یا خوشبندی مشتریان سایتها.

۷. تشخیص گفتار: تقسیم کردن گفتار بر حسب لحن، گوینده و یا فشرده سازی آنها

۸. تقسیم بندی تصاویر: گروه بندی تصاویر پزشکی و ماهواره‌ای

انواع خوشها را با توجه به شکل نهایی آنها می‌توان به صورت زیر گروه بندی کرد:

• خوشها به خوبی جدا شده^۳

^۱ Understanding

^۲ Summarization

^۳ Well-separated

- خوشه‌های مبتنی بر تمرکز^۱
 - خوشه‌های مجاورتی^۲
 - خوشه‌های تراکمی^۳
 - خوشه‌های مفهومی^۴
 - خوشه‌های مبتنی بر تابع هدف
- توضیحات مربوط به خوشه را در همین نقطه به پایان می‌رسانیم و وارد معرفی مفاهیم دیگری در این پایان نامه خواهیم شد.

۴-۲- الگوریتم‌های فرامکاشفه‌ای زیستی

در بسیاری از مسائل جستجو در دنیای واقعی، نمی‌توان فضای جستجو را به شکل یک درخت نمایش داد. علی‌الخصوص هنگامیکه این فضا بسیار بزرگ و نامنظم باشد. برای همین برای این نوع فضاهای نمی‌توان از هر الگوریتمی استفاده کرد. جستجوی فرامکاشفه‌ای^۵ نام خانواده‌ای از الگوریتم‌های جستجوی آگاهانه^۶ است که در آنها از یک پدیده طبیعی برای کاوش فضای جستجو، الهام گرفته می‌شود. این الگوریتم‌ها با الهام گرفتن از یک رخداد طبیعی این امکان را پیدا می‌کنند که فضای جستجوی بسیار بزرگ طیف وسیعی از مسائل بهینه‌سازی پیچیده را به صورت بسیار هوشمندانه‌ای مورد کاوش قرار دهند. منظور از هوشمندانه در اینجا این است که الگوریتم‌های فرامکاشفه‌ای کل فضای جستجو را به دلیل بزرگی آن پیمایش نمی‌کنند.

^۱ Center-based

^۲ Contiguous

^۳ Density

^۴ Conceptual

^۵ Metaheuristic

^۶ Informed search

به همین دلیل هم هست که آنها ناکامل و البته غیر بهینه هستند. بلکه آنها تنها به پیمایش بخشی از فضا که احتمال وجود یک پاسخ به اندازه‌ی کافی خوب در آن بیشتر است، اکتفا می‌کنند. برخی از الهام‌های طبیعی که بر اساس آنها یک الگوریتم فرامکاشفه‌ای طراحی شده است عبارتند از: تکامل موجودات در طی نسل‌ها، فرآیند سرد شدن یا تبرید در فلزات، زندگی مورچه‌ها در یک کلونی، حرکت گروهی پرندگان و سیستم ایمنی در بدن انسان. در ادامه معرفی مفاهیمی چون دورنمای برازش، قابلیت‌های پویش و انتفاع و نیز دسته‌بندی الگوریتم‌های فرامکاشفه‌ای خواهیم پرداخت.

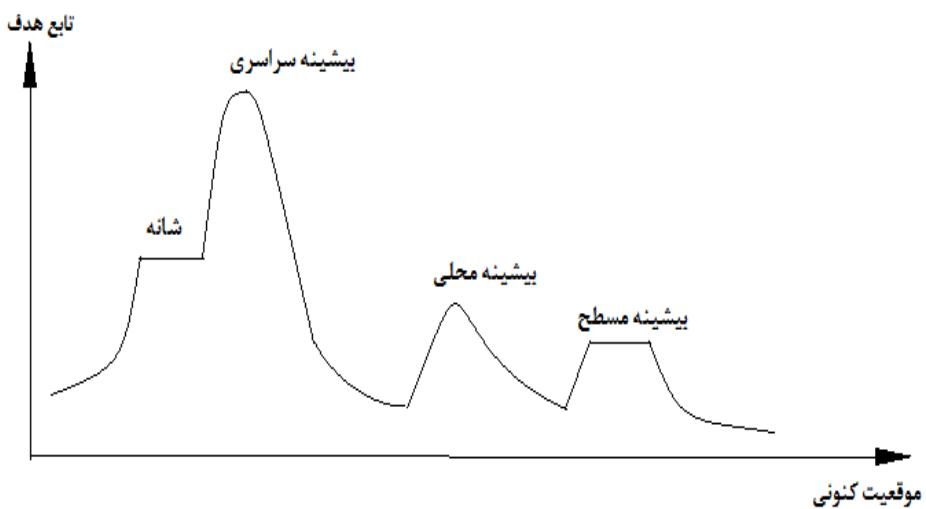
۱-۴-۲ دورنمای برازش

در بسیاری از مسائل دنیای واقعی، فضای جستجو آنچنان بزرگ و نامنظم است که امکان تبیین آن به صورت درخت وجود ندارد و به همین دلیل است که نمی‌توان از الگوریتم‌های جستجوی تحلیلی و ناآگاهانه استفاده کرد. در اینگونه مسائل فضای جستجو را می‌توان به شکل یک سرزمین دانست که جستجوی فرامکاشفه‌ای، با پیمایش هوشمندانه‌ی بخش مهمی از آن، سعی می‌کند تا یک پاسخ به اندازه کافی خوب را بیابند.

سرزمینی که به عنوان فضای جستجوی نامنظم مسائل پیچیده عنوان شد دارای پستی و بلندی‌های متعددی است که از آن با نام دورنمای برازش^۱ یاد می‌شود. پستی‌ها و بلندی‌ها در دورنمای برازش توسط یک تابع برازش^۲ (تابع شایستگی) مشخص می‌گردد. تعریف تابع برازش با توجه به اطلاعات مسئله صورت می‌گیرد. یک دورنمای برازش شامل موقعیت (یا همان پاسخ) و نیز شامل ارتفاع (یا همان مقدار تابع شایستگی) می‌باشد. با توجه اینکه مسئله چه می‌تواند باشد، ما در دورنمای برازش معمولاً به دنبال دو مسئله هستیم: یا یافتن عمیق ترین دره (کمینه سراسری) و یا یافتن بلندترین قله (بیشینه سراسری).

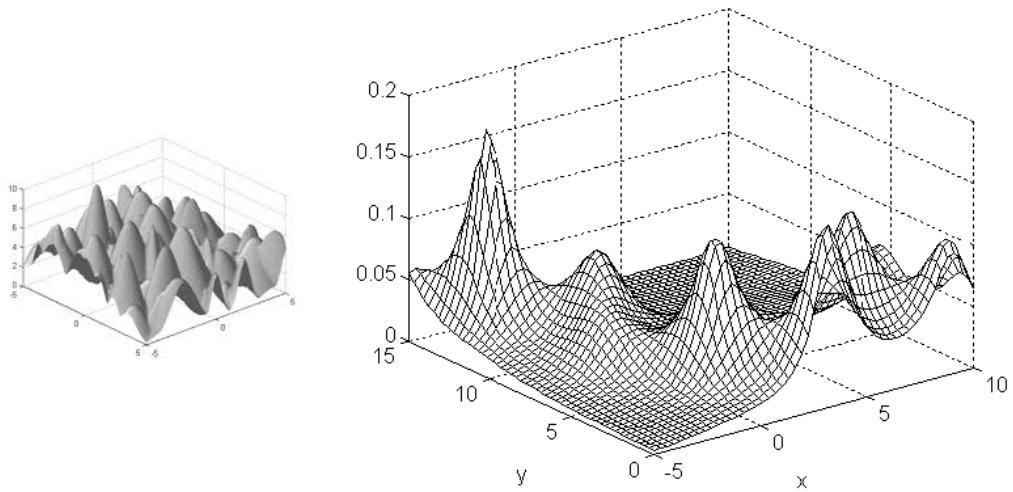
^۱ Fitness Landscape

^۲ Fitness Function



۳-۲ نمونه از یک دورنمای برازش یک بعدی که در آن هدف یافتن بلندترین قله است.

دورنمای برازش ممکن است برای یک مسئله بسیار پیچیده‌تر از شکلی باشد که در ۳-۲ به آن اشاره شده است. در مسائل جستجوی مختلف دورنمایاهای برازش تفاوت‌های چشمگیری با یکدیگر دارند که الگوریتم‌های فرامکاشفه‌ای با قابلیت‌های متفاوتی جهت کاوش موثر را طلب می‌کنند.



۴-۲ نمونه‌ای از دورنمای برآش در فضای سه بعدی

۲-۴-۲ - قابلیت‌های پویش و انتفاع

الگوریتم‌های فرامکاشفه‌ای برای آنکه بتوانند به نحو موثری فضای جستجو را در یک مسئله بهینه‌سازی مورد کاوش قرار دهند، و پاسخ‌های به اندازه کافی خوب را بیابند، می‌بایست به دو خصیصه مهم مجهز باشند. این دو خصیصه را قابلیت پویش^۱ و قابلیت انتفاع^۲ می‌گویند. قابلیت پویش، توانایی الگوریتم فرامکاشفه‌ای در جستجوی آزادانه و بدون هرگونه توجه به دستاوردهای آن در طول فرآیند جستجو است. در مقابل قابلیت انتفاع به میزان توجه الگوریتم به دستاوردهایش در طول فرآیند جستجو گفته می‌شود. بدیهی است که به هر میزان که قابلیت پویش الگوریتم بیشتر باشد، الگوریتم رفتاری تصادفی‌تر و غیرقابل پیش‌بینی‌تر دارد. از طرفی دیگر اگر قابلیت انتفاع الگوریتم بیشتر باشد، الگوریتم رفتاری حساب شده‌تر و محتاطانه‌تر در پیش می‌گیرد. اکثر الگوریتم‌های فرامکاشفه‌ای دارای پارامترهایی هستند که میزان قابلیت‌های پویش و انتفاع را در آنها کنترل می‌کند. به طور مثال اگر دورنمای برآش شامل یک قله باشد، در این فضا باید قابلیت انتفاع یک الگوریتم قوی‌تر باشد و بر عکس اگر مسئله دارای قله‌های متعددی باشد

^۱ Exploration

^۲ Exploitation

برای جلوگیری از افتادن در تله‌ی بیشینه محلی باید قابلیت پویش الگوریتم بیشتر تقویت شود. بسیاری از مسائل دنیای واقعی بسیار پیچیده‌تر از آنچه که ذکر شد هستند. به عبارت دیگر برای یافتن یک پاسخ به قدر کافی خوب برای اکثر مسائل واقعی، می‌بایست یک مصالحه میان قابلیت‌های پویش و انتفاع در این نوع الگوریتم برقرار نماییم. برای تعیین میزان قابلیت‌های پویش و انتفاع در الگوریتم‌های فرامکاشفه‌ای، ابتدا لازم است که ماهیت مسئله را تشخیص دهیم تا بتوانیم مقدار پارامترهای الگوریتم را به درستی تنظیم کنیم.

نکته مهمی در مورد الگوریتم‌های فرامکاشفه‌ای وجود دارد، قضیه ناهار غیر مجانی^۱ است. بر طبق این قضیه فرض می‌کنیم تعدادی رستوران داریم که در هر کدام یک فهرست غذای ثابت و البته با قیمت‌های متفاوت وجود دارد. بدینهی است که برای اینکه یک شخص بتواند بهترین رستوران را برای صرف ناهار خود انتخاب کند، باید به عادات غذایی خود توجه داشته باشد. اگر شخص گوشتخوار است باید به رستوران‌هایی برود که غذای گوشتی ارزانتری دارد و اگر گیاهخوار باشد رستورانی که غذاهای گیاهی ارزانتری دارند بهتر خواهد بود. و اگر شخص همه چیزخوار باشد، به صورت میانگین (در تعداد دفعات بالا) تفاوتی نمی‌کند که به کدام رستوران برود، زیرا به طور میانگن هزینه‌ای که به ازای چندین دفعه صرف ناهار بایستی پرداخت کند، صرف نظر از اینکه به کدام رستوران برود، تفاوتی نخواهد کرد. حال چنانچه هر کدام از الگوریتم‌های فرامکاشفه‌ای را یک رستوران و هر کدام از مسائل را یک غذا و نیز هزینه حل مسئله را معادل قیمت غذا بدانیم، بر اساس قضیه ناهار غیر مجانی، میانگین هزینه حل همه مسائل جستجو توسط هر یک از الگوریتم‌های جستجو مقداری ثابت خواهد بود.

بنابراین با توجه به مطالب ذکر شده واضح است که در طراحی یک الگوریتم فرامکاشفه‌ای برای کاربردی مشخص، مهم آن است که این کار به گونه‌ای انجام شود که عملکرد آن الگوریتم در مقایسه با سایر الگوریتم‌ها برتر باشد. در حل بسیاری از مسائل بهینه‌سازی، بهتر است الگوریتم‌های جستجوی فرامکاشفه‌ای

^۱ No Free Lunch Theorem

در ابتدای فرآیند جستجوی خود در فضای حالت مسئله قابلیت پویش بیشتری داشته باشند و هرچه که به پایان نزدیک تر می‌شوند اندازه‌ی قابلیت انتفاع خود را افزایش دهند.

۴-۳-۲- طبقه‌بندی الگوریتم‌های فرامکاشفه‌ای

در این بخش مختصری در باب طبقه‌بندی الگوریتم‌های فرامکاشفه‌ای از دیدگاه‌های متفاوت و با معیارهای مختلف خواهیم پرداخت. این دیدگاه‌ها عبارتند از:

- **زیستی و غیرزیستی:** بسیاری از الگوریتم‌های فرامکاشفه‌ای از زندگی موجودات طبیعی الهام گرفته شده اند و به عبارت دیگر زیستی هستند در حالیکه برخی اینگونه نیستند مثلاً براساس یک پدیده فیزیکی الهام گرفته شده اند
- **جمعیتی و غیرجمعیتی:** الگوریتم‌های جمعیتی در حین جستجو، یک جمعیت از جواب‌ها را در نظر می‌گیرند در حالیکه الگوریتم‌های غیرجمعیتی (مبتنی بر یک جواب) حین جستجو یک جواب را تغییر می‌دهند.
- **تکاملی و غیرتکاملی:** الگوریتم‌های زیستی که به طور مستقیم از فرامکاشفه بقاء اصلاح که مبتنی بر نظریه داروین است استفاده می‌کنند، تکاملی محسوب می‌شوند. در حالیکه سایر روش‌های جستجوی زیستی به صورت غیر مستقیم به فرامکاشفه بقاء اصلاح داروین مرتبط هستند زیرا عملاً مولود فرآیند تکامل هستند که به آنها زیستی غیرتکاملی گفته می‌شود.
- **با حافظه و بدون حافظه:** برخی الگوریتم‌ها فاقد حافظه اند، بدین معنا که آنها از اطلاعات بدست آمده در طول فرآیند جستجو، استفاده نمی‌کنند (مثل تبرید شبیه سازی شده). اما برخی دیگر از حافظه استفاده می‌کنند.
- **احتمالی و قطعی:** برخی الگوریتم‌های فرامکاشفه‌ای با برخی قوانین احتمالی در حین جستجو دست به تصمیم گیری می‌زنند، در حالیکه برخی دیگر با تصمیمات قطعی مسئله را پیش می‌برند.

شاید مهمترین دسته میان طبقه بندی‌های ذکر شده، الگوریتم‌های زیستی باشند. که از آن میان نیز دو دسته تکاملی و مبتنی بر هوش جمعی قابل اهمیت هستند.

۲-۵- الگوریتم‌های مبتنی بر هوش جمعی

طراحی یک سیستم هوشمند که از قابلیت‌های تطبیق شوندگی، توزیع شدگی و انعطاف پذیری برخوردار باشد و همچنین توانایی بالایی را در حل مسائل مختلف داشته باشد، سیستم‌هایی که به سیستم‌های مبتنی بر هوش جمعی^۱ معروف هستند، به طور عمده از رفتار موجودات اجتماعی الهام می‌گیرند. بدین صورت که با بررسی هوش جمعی و روش‌های ارتباطی که موجودات اجتماعی برای رسیدن به اهدافشان استفاده می‌کنند، سعی در شناخت مکانیزم‌های جستجو و ارتباطی می‌نمایند. پس از کشف این مکانیزم، سعی می‌گردد مدلی ارائه شود که با پیروی از این رفتارها بتواند مسئله خاصی را حل کند.

در اصطلاح به گروه موجودات ازدحام^۲ گفته می‌شود. یک ازدحام را می‌توانیم گروهی از عامل‌های (عموماً متحرک) که با یکدیگر از طریق فعالیت در محیط‌های محلی شان (به صورت مستقیم یا غیر مستقیم) ارتباط برقرار می‌کنند، تعریف کنیم. هوش ازدحامی به رفتار حل مسئله‌ای گفته می‌شود که از تعامل چنین موجوداتی پدیدار می‌گردد و محاسبات هوش ازدحامی به همه مدل‌های الگوریتمی مبتنی بر چنین رفتاری گفته می‌شود. هوش ازدحامی به هوش جمعی نیز معروف است. مطالعات بر روی رفتار اجتماعی حیوانات و حشرات منجر به تولید تعدادی مدل‌های محاسباتی هوش جمعی شده است. موجوداتی که مطالعات محاسباتی مبتنی بر هوش جمعی از آنها الهام گرفته شده اند، شامل مورچه‌ها، زنبورها، موریانه‌ها، عنکبوت‌ها، ماهی‌ها، دسته‌های پرنده‌گان و ... هستند. در این ازدحام‌ها موجودات ساختارهای ساده‌ای دارند،

^۱ Swarm Intelligence

^۲ Swarm

اما رفتارهای جمعی آنها پیچیده است. رفتار پیچیده یک ازدحام در نتیجه الگوهای ارتباطی بین موجودات یک ازدحام در طول زمان است. این رفتار پیچیده، ویژگی هیچ موجود به تنها یی نبوده و معمولاً به آسانی از رفتار ساده موجودات قابل پیش بینی و نتیجه گیری نمی باشد. این پدیده را ظهور^۱ می گویند.

هوش جمعی به صورت غیر خطی از رفتار موجودات یک ازدحام پدیدار می گردد. یک اتصال محکم بین موجودات و رفتار جمعی وجود دارد. رفتار جمعی موجودات که رفتار ازدحام را شکل می دهد و از طرف دیگر، رفتار ازدحام بر روی شرایطی که تحت آن هر موجود فعالیت می کند تاثیر می گذارد. این فعالیتها ممکن است محیط را تغییر دهند و بنابراین ممکن است رفتار موجودات و همسایگانش تغییر کند، که دوباره ممکن است منجر به تغییر رفتار جمعی ازدحام شود. پس مهمترین بخش هوش جمعی، تعامل و همکاری است. تعامل بین موجودات به بهبود دانش تجربی درباره محیط کمک می کند. تعامل می تواند به صورت مستقیم (از طریق ارتباط فیزیکی، یا از طریق ورودی های ادراکی بینایی، شنوایی یا شیمیایی) باشد و یا می تواند به صورت غیرمستقیم (از طریق تغییر در محیط) باشد. اصطلاح استیگمرجی^۲ اشاره به نوع ارتباط غیر مستقیم بین موجودات دارد.

الگوریتم های جستجوی مبتنی بر هوش جمعی، به دو دسته کلی علامت-محور و تقلید محور تقسیم شده اند [۲].

۱-۵-۲ - روش های تقلید-محور

دسته ای از الگوریتم های مبتنی بر هوش جمعی، از یک مفهوم سرعت، اینرسی و یا تمایل حرکتی در فرآیند جستجوی خود بهره می برند. در این روشها، ارتباط میان موجودات در جمعیت به صورت مستقیم می باشد و هیچ حافظه مشترکی وجود ندارد. در روش های مربوط به این دسته معمولاً هر موجود یک حافظه دارد که در آن معمولاً بهترین مکان یافته شده توسط خود (پاسخ برازنده محلی) و بهترین موجودات

^۱ Emergence

^۲ Stigmergy

جمعیت (پاسخ برازنده سراسری) را نگهداری می کنند. موجود ذکر شده معمولاً تمایلی را به سمت بهترین پاسخهای محلی و سراسری دارد. از آنجا که موجودات در این دسته از الگوریتمها، تمایلی به سمت بهترین پاسخهای یافته شده توسط خودشان و دیگر موجودات دارند، یعنی رفتاری تقلید گونه دارند، به آنها روش های تقلید محور گفته می شود.

۲-۵-۲ روشهای علامت محور

دسته ای از الگوریتمها مبتنی بر هوش جمعی، از یک حافظه محیطی مشترک برای برقراری ارتباط غیرمستقیم میان موجودات بهره می برند. این گونه الگوریتمها که فاقد هرگونه رفتار تقلید گونه هستند، علامت-محور نامیده می شوند. منظور از علامت-محور هدایت غیر مستقیمی است که این موجودات با درج علامت در حافظه محیطی مشترکشان برای یکدیگر به وجود می آورند. به طور کلی آنچه که اهمیت دارد عدم وجود یک رهبر و یا واحد کنترل مرکزی برای ایجاد هماهنگی در بخش های مختلف سیستم می باشد. اصطلاحاً به آنها سیستم های خود سازمانده^۱ گفته می شود.

۲-۵-۱-۱ مراحل یک الگوریتم علامت-محور

- مرحله ۱- مقداردهی اولیه به حافظه علامت-محور: در این مرحله حافظه علامت-محور مقدار دهی اولیه می شود. این مقداردهی اولیه برای جامعه مورچگان شامل تعیین فرومون اولیه بر روی هر کدام از یال های گراف مسیر و یا در مورد جامعه زنبوران عسل شامل تعیین تعداد زنبورهای رقصende در هر کدام از درایه های ماتریس اولیه حافظه علامت-محور است. چگونگی تعیین مقدار اولیه نیز به نوع مسئله بستگی دارد.

- مرحله ۲- ساخت یک پاسخ: در این مرحله، یک پاسخ جدید، توسط یک موجود علامت-محور براساس احتمالی که مبتنی بر حافظه علامت محور است، تولید می شود. بدیهی است که احتمال

^۱ Self-organized System

گفته شده، ساخت پاسخ را به سویی هدایت می کند که حافظه علامت-محور راستای آن را نمایش می دهد.

• مرحله ۳- برازش پاسخ جدید: در این مرحله، برازش پاسخ جدید ساخته شده در مرحله قبل

با توجه به تابع برازش محاسبه خواهد شد.

• مرحله ۴- به هنگام سازی محلی حافظه علامت-محور: در این مرحله، حافظه علامت-محور

به منظور جلوگیری از تولید مجدد یک پاسخ توسط سایر موجودات علامت-محور به روز می شود.

منظور از این به هنگام سازی، کم کردن احتمال انتخاب اجزای پاسخ جدیدی است که در حلقه

جاری توسط موجود علامت-محور ساخته شده است. به این ترتیب در دورهای بعدی که سایر

موجودات علامت-محور قصد ساخت پاسخ خود را دارند، مکانهای جدیدی را در فضای جستجو

مورد پیمایش قرار می دهند. که در سیستم مورچه با تبخیر فرومون و در سیستم زنبورهای عسل

با کاهش تعداد زنبورهای رقمنده صورت می گیرد.

• مرحله ۵- تکرار برای کل جامعه علامت-محور: در این مرحله بررسی می شود که آیا به تعداد

کل موجودات علامت-محور، مراحل ۲ الی ۴ تکرار شده است یا خیر. در صورت مثبت بودن جواب

این سوال، مرحله ۶ اجرا می شود.

• مرحله ۶- به هنگام سازی سراسری حافظه علامت-محور: در این مرحله دو عملیات انجام

می شود. ابتدا حافظه علامت-محور تحت فرآیند گذر زمان قرار گرفته و دستخوش فراموشی

می شود. که در سیستم زنبورهای عسل با اتمام رفتار زنبورهای رقمنده قدیمی و در سیستم مورچه

با تبخیر فرومون انجام می گیرد. عملیات بعدی که صورت می گیرد، یادگیری برازش مکانهای طی

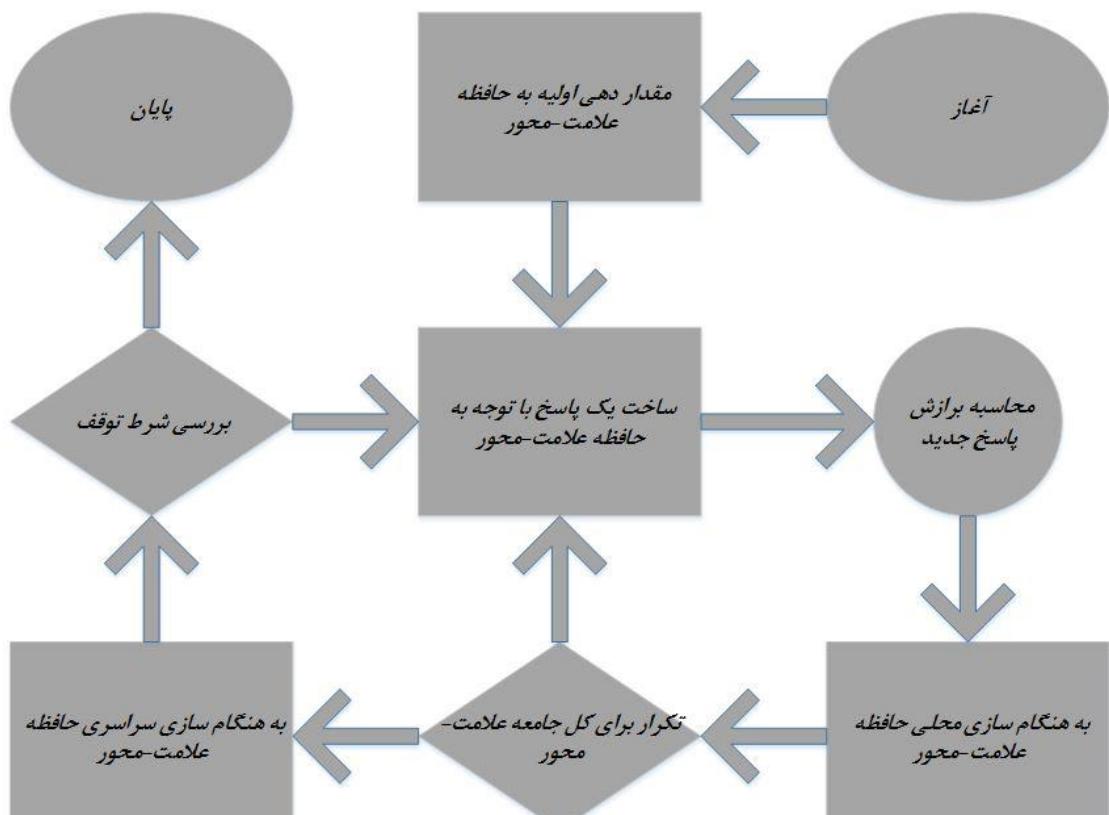
شده در فضای جستجو توسط حافظه علامت-محور است. عملیات مذبور با ترشح فرومون بر روی

مسیرها و یا درایههای مربوط به همه یا برخی پاسخهای ساخته شده (بر اساس میزان برازش آنها)

انجام می شود. و در سیستم زنبورها با تخصیص تعداد مشخصی از زنبورهای رقمنده به هر درایه

حافظه علامت-محور پیاده سازی می شود.

- مرحله ۷- بررسی شرط توقف: در این مرحله، در مورد ادامه فرآیند جستجوی الگوریتم تصمیم گیری می‌شود. در صورت ارضاء شرط توقف، فرآیند جستجوی الگوریتم متوقف شده و بهترین پاسخ‌های یافته شده به عنوان حاصل جستجوی الگوریتم علامت-محور در خروجی ارائه می‌گردد.



با توجه به مطالب گفته شده، فرآیند جستجوی فرامکالشفه‌ای در الگوریتم‌های علامت-محور تحت تاثیر مولفه‌های زیر می‌باشد:

- چگونگی طراحی حافظه علامت-محور
- طراحی تابعی که با آن برازش پاسخ‌ها را ارزیابی کنیم.
- چگونگی مقداردهی اولیه به حافظه علامت-محور.
- الگوریتم مربوط به بروزرسانی محلی و سراسری حافظه علامت-محور

• تعیین شرط توقف

در همین نقطه بحث خود در ارتباط با الگوریتم‌های فرامکاشفه‌ای را پایان می‌دهیم. زیرا که قدر لزوم این پایان نامه مفاهیم مورد نیاز شرح و بسط داده شد. در بخش بعدی کمی بیشتر درباره الگوریتم فرامکاشفه‌ای بهینه‌سازی کندوی زنبور عسل توضیح داده خواهد شد.

۶-۲- بهینه‌سازی کندوی زنبور عسل

اکثر فعالیت‌های تحقیقی انجام شده در زمینه هوش جمعی حشرات به مورچه‌ها و قوانینی که آنها در مسیریابی خود استفاده می‌کنند، صورت گرفته است. در مقابل فعالیت‌های انجام شده در زمینه هوش جمعی سایر حشرات از جمله زنبورهای عسل محدود بوده است.

زنبورهای عسل فعالیت شهدیابی^۱ خود را جهت تولید عسل به صورت یک امر اجتماعی سازمان دهی می‌کنند. این حشرات در فعالیت شهد یابی خود، از یک سیستم ارتباطی پیچیده استفاده می‌کنند. زنبورهای عسل در این سیستم ارتباطی به یک شکل خاص و بدیع، فاصله و کیفیت منابع غذایی را به اطلاع دیگر زنبورانی که قصد جستجوی شهد گل دارند می‌رسانند. ساختار زندگی زنبورهای عسل نشان می‌دهد که آنها از زبان ارتباطی خاصی تحت عنوان زبان رقص^۲ جهت برقراری ارتباط استفاده می‌کنند [۱۱].

هنگامی که هریک از زنبورهای عسل از سفر خود برای جستجوی شهد گل باز می‌گردد، در مسیر خود و در جمع آوری شهد خود مسلمًا به منابع موجود و غنی از شهد گل‌ها برخورده است. عوامل مختلفی می‌تواند مشخص کننده مناسب بودن یا نبودن یک منبع گل باشند. کیفیت گل و فاصله‌ای که از منبع آن دارد میزان مناسب بودن گل را مشخص می‌کند. علاوه بر عوامل ذکر شده، زنبورها نیز باید به نحوی

^۱ Foraging

^۲ Dance Language

جهت منبع مورد نظر را نیز به اطلاع زنبورهای بعدی برسانند. تمام این فعالیتهای نام برده شده توسط رفتاری به نام رقص پیچشی^۱ توسط زنبور در مکان خاصی از کندو به نام سالن رقص^۲ انجام می‌پذیرد. این رقص که شکلی از برقراری ارتباط غیر مستقیم میان زنبورها در کندو است (که ویژگی لازم برای یک الگوریتم علامت-محور است)، شامل اطلاعاتی از فاصله، جهت و کیفیت منابع غذایی است. تعداد چرخش (پیچش) نمایانگر فاصله و مدت زمان چرخش نشان دهنده کیفیت منابع غذایی است [۱۱].

نحوه برقراری ارتباط غیر مستقیم زنبورها به این صورت است که زنبور بازگشته از مسیر، وقتی به کندو می‌رسد در سالن رقص مدت زمان خاصی را با توجه به کیفیت منابع یافته شده، در راستای آن منبع می‌رقصد. این رقص با توجه به اهمیت منبع یاد شده زمان متفاوتی خواهد داشت. هر زمان که یکی از زنبورهای جدید قصد یافتن شهد را داشته باشد، پیش از هر کاری ابتدا به صورت تصادفی به سالن رقص نگاه می‌کند. این نگاه کردن به سالن رقص به معنای تصمیم گیری و انتخاب از بین رقصنده‌های موجود نمی‌باشد. چون احتمالاً در یک زمان واحد بیش از یک زنبور در سالن مشغول رقصیدن است، زنبوری که به سالن نگاه می‌کند با مشاهده اولین زنبور رقصنده به سوی جهتی که او اشاره می‌کند حرکت خواهد کرد. بدیهی است که در این فرآیند منابعی که از مطبوعیت بیشتری برخوردار باشند، تعداد رقص بیشتری در جهت آنها و به مدت طولانی‌تری صورت می‌گیرد. به همین دلیل منابع ذکر شده در طول فرآیند تصادفی مطرح شده، از شанс بیشتری برای انتخاب شدن برخوردار خواهند بود.

بهینه‌سازی کلونی زنبور عسل^۳ (HHO) یک جستجوگر فرامکاشفه‌ای است که براساس رفتار شهیدیابی زنبورهای عسل و رقص پیچشی آنها در کندو، طراحی شده است. ذکر این نکته در اینجا ضروریست که زنبورهای عسل، از تعداد دیگر انواع رقص نیز جهت اطلاع رسانی در زمینه‌های مختلف دیگر استفاده

^۱ Waggle Dance

^۲ Dance Floor

^۳ Honeybee Hive Optimization

می‌کنند. لیکن در طراحی الگوریتم بهینه‌سازی کندوی زنبور عسل، تنها از رقص پیچشی به عنوان زبان ارتباطی غیر مستقیم زنبورها در فرآیند شهدیابی آنها استفاده شده است.

بهینه‌سازی کلونی زنبور عسل اولین بار در سال ۲۰۰۸ توسط صنیعی آباده ارائه شد [۱۲]. شبه کد این الگوریتم در شکل ۶-۲ ارائه شده است. این الگوریتم از نظر ساختاری شباهت بسیاری با بهینه‌سازی کلونی مورچگان دارد. مهمترین تفاوتی که باعث جدا شدن منطق رفتاری بهینه‌سازی کلونی زنبور عسل از بهینه‌سازی کلونی مورچه می‌شود استفاده از یک جدول رقص به جای جدول فرومونی، به عنوان رسانه ارتباطی غیرمستقیم، است. استفاده از جدول رقص به جای جدول فرومونی منجر به رفتار گستته‌تر (پویشی‌تر) می‌شود. خصوصیات مهم الگوریتم بهینه‌سازی کلونی زنبور عسل عبارتند از:

۱. در شبه کد الگوریتم دقیقاً قبل از ورود به حلقه while، تعداد زنبورهای رقصندۀ اولیه برای هر

درایه در جدول رقص به تعداد δ_0 زنبور رقصندۀ در نظر گرفته می‌شود.

۲. تابع ConstructSolution در الگوریتم با توجه به یک احتمال مبتنی بر جدول رقص، مسیری را

به ازای هر زنبور می‌سازد. این تابع از رابطه زیر بدست می‌آید:

$$P_k(r, s) = \begin{cases} \frac{[\delta(r, s)]^\alpha \cdot [\eta(r, s)]^\beta}{\sum_{u \in J_k(r)} [\delta(r, u)]^\alpha \cdot [\eta(r, u)]^\beta} & \text{if } s \in J_k(r) \\ 0 & \text{otherwise} \end{cases} \quad (1-2)$$

در این رابطه داریم $P_k(r, s)$ احتمال حرکت زنبور K از گره r به گره s ($\delta(r, s)$ تعداد زنبورهای رقصندۀ برای یالی که گره‌های r و s را به یکدیگر متصل می‌کند (مقدار موجود در درایه (r, s) در جدول رقص)، $\eta(r, s)$ برآشحرکت از گره r به گره s که این برآش از دانش پس زمینه مربوط به مساله بدست می‌آید. α ضریب مکافه‌ای برای تنظیم میزان توجه به اطلاعات موجود در جدول رقص است و β ضریب مکافه‌ای برای میزان توجه به اطلاعات مساله است. ($J_k(r)$ مجموعه گره‌های پیموده شده توسط زنبور K است به طوریکه که گره r آخرین گره پیموده شده باشد).

۳. برای به هنگام سازی جدول رقص و تعداد زنبورهای رقصندۀ موجود در آن از رابطه‌ی زیر استفاده

می‌شود:

$$\delta(r,s) = \delta(r,s) + \Delta\delta(r,s) \quad (2-2)$$

که $\Delta\delta(r,s)$ نیز می‌تواند از رابطه زیر بدست بیاید:

$$\Delta\delta(r,s) = \begin{cases} Fitness(s_k) & \text{if } (r,s) \in S_k \\ 0 & \text{otherwise} \end{cases} \quad (3-2)$$

Function HHO(problem) **returns** a state that is local maximum

Input: Population_{size}, Problem_{size}, δ_0 , α , β

Output: S_{best}

$S_{best} \leftarrow \text{CreateHeuristicSolution(Problem}_{size}\text{);}$

$\text{DanceTable} \leftarrow \text{InitializeDanceTable}(\delta_0);$

while $\neg \text{StopCondition}()$ **do**

for $k=1$ to Population_{size} **do**

$S_k \leftarrow \text{ConstructSolution(DanceTable, Problem}_{size}, \alpha, \beta\text{);}$

if $\text{Fitness}(S_k) > \text{Fitness}(S_{best})$ **then**

$S_{best} \leftarrow S_k;$

end

end

for $k=1$ to Population_{size} **do**

$\text{UpdateDanceTable(DanceTable, } S_k\text{);}$

end

end

return $S_{best};$

[۲] ۶-۲ شبیه کد ساده الگوریتم بهینه‌سازی کندوی زنبور عسل

۷-۲- الگوهای گرافی در شبکه‌های واقعی

بیشتر شبکه‌های واقعی الگوهای رایجی را به اشتراک می‌گذارند. در میان این الگوها خواص شناخته شده‌ای هستند: توزیع مقیاس آزاد^۱، اثر جهان کوچک^۲ و ساختار انجمانی^۳.

۷-۲-۱- شبکه‌های مقیاس آزاد:

خیلی از آمارها در شبکه‌های واقعی از "مقیاس" معمول یا نمونه برخوردار هستند. یعنی یک مقداری که باقی نمونه‌های اندازه گیری شده در اطراف آن جمع می‌شوند. برای مثال قد همه‌ی افراد ایران یا سرعت ماشین‌ها در بزرگراه‌ها. اما درجه یک گره در شبکه‌های واقعی از توزیع قانون توانی^۴ (یا پارتو یا زیفان^۵) پیروی می‌کند. متغیر تصادفی x از توزیع قانون توانی پیروی می‌کند اگر:

$$P(x) = Cx^{-\alpha}, x \geq x_{min}, \alpha > 1 \quad (4-2)$$

شکل ۷-۲ (الف) و (ب) به ترتیب توزیع نرمال و توزیع توانی که به آن مقیاس آزاد نیز می‌گویند، نمایش می‌دهد. برخلاف توزیع نرمال که یک مرکز دارد، توزیع قانون توانی خیلی اریب گونه است. برای توزیع نرمال به ندرت پیش می‌آید که اتفاقی رخ دهد که از میانگین انحراف زیادی داشته باشد اما در توزیع مقیاس آزاد، آن دم یا دنباله اجازه می‌دهد که طولانی‌تر شود. بنابراین خیلی معمول است که در یک شبکه اجتماعی تعدادی از گره‌ها، درجه‌ی خیلی بالایی داشته باشند در حالیکه تعداد قابل توجهی از گره‌ها، درجه پایینی دارند.

^۱ Scale-free Distribution

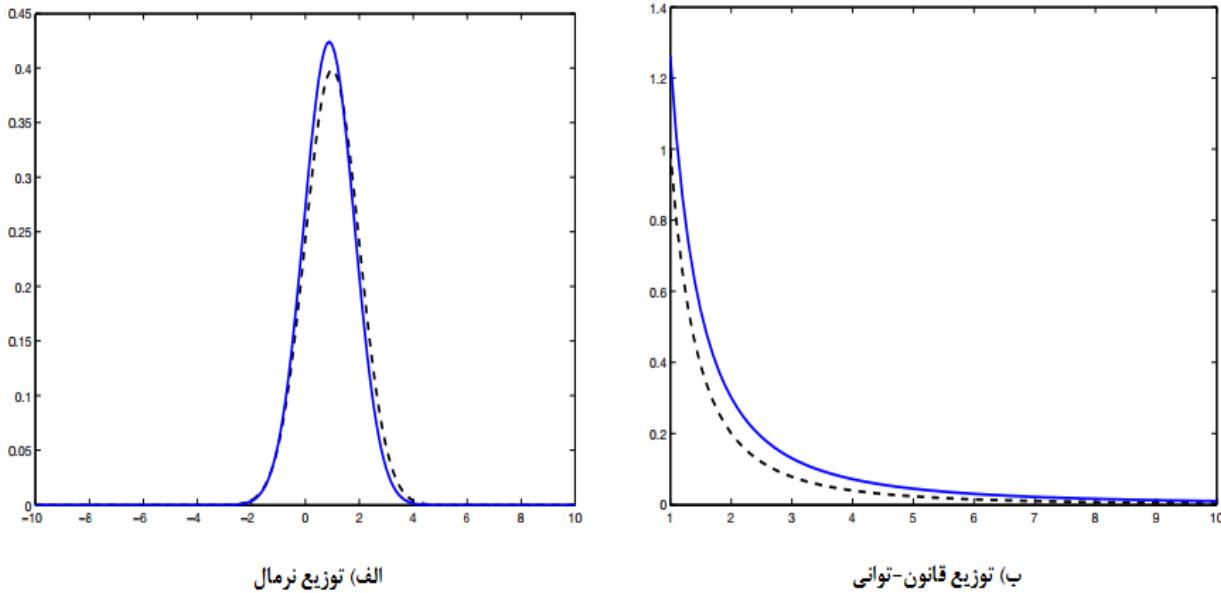
^۲ Small-world effect

^۳ Community Structure

^۴ Power-law distribution

^۵ Zipfian or Pareto

در کنار درجه گره چند معیار آماری دیگر نیز از توزیع قانون توانی پیروی می‌کنند. بزرگترین مقدار ویژه ماتریس همسایگی که از شبکه بدست می‌آید، اندازه اجزای متصل در شبکه، اندازه انتشار اطلاعات در شبکه و چگالی یک شبکه در حال رشد مثال‌هایی از خواصی هستند که از این توزیع پیروی می‌کنند.



[۳] ۷-۲ نمایش توزیع‌های نرمال و قانون توانی

۷-۴-۲-۲ پدیده جهان‌کوچک

تراورس و میلگرام یک آزمایش مشهوری را برای بررسی متوسط طول مسیر برای شبکه‌های اجتماعی مردم آمریکا انجام دادند [۱۳]. آزمایش فرستادن نامه‌های زنجیره‌ای از یک شخص به آشنایانش، برای رساندن نامه به یک فرد هدف در شهر دیگری بود. که در نهایت ۶۴ نامه رسید و متوسط طول مسیر تقریباً مابین ۵,۵ تا ۶ بود. که بعداً "شش درجه جدایی"^۱ نامیده شد. که این نتیجه بعدها در شبکه پیام رسانی در بین بیش از ۱۸۰ میلیون فرد تایید شد که متوسط طول دو فرد، ۶,۶ بود [۱۴].

^۱ Six Degrees of Separation

۳-۷-۲ - انجمن‌ها

مهم ترین مشکل در کشف انجمن‌ها یا خوشبندی گراف عدم وجود یک تعريف قابل اندازه گیری از انجمن است. هیچ تعريفی که مورد قبول همگان باشد یا در جهان یکتا باشد برای آن وجود ندارد. در حقیقت یک تعريف، به سیستم خاصی که در دست هست یا کاربردی که مد نظر هست بستگی دارد. باید تعداد یال‌ها "درون" یک انجمن نسبت به لبه‌هایی که رئوس آن انجمن را به باقی گراف متصل می‌کند زیادتر باشد. این جمله اخیر در واقع مرجع و راهنمای اکثر تعاریفی است که از انجمن ارائه شده است. اما خیلی از دستورالعمل‌های جایگزینی وجود دارد که سازگار با این تعريف است. علاوه براین در بعضی موارد آنها به صورت الگوریتمی تعريف می‌شوند. به طور مثال آنها خروجی نهایی یک الگوریتم خواهند بود بدون اینکه تعريف دقیق قبلی از انجمن در آن ارائه شده باشد.

با زیرگراف C از گراف G شروع می‌کنیم که اندازه‌ی آنها به ترتیب برابر است با $n_C = n$ و $|C| = n$. درجه داخلی و خارجی راس $v \in C$ را به ترتیب k_v^{ext} و k_v^{int} می‌نامیم. که اولی تعداد لبه‌هایی که رئوس C را به v وصل می‌کند و دومی آن را به باقی گراف متصل می‌نماید. اگر $g_C = 0$ باشد یعنی که هیچ همسایه‌ای خارج از زیر گراف C برای آن وجود ندارد درنتیجه به نظر C یک خوشه مناسب برای v است. اما اگر $g_C = 0$ باشد یعنی که این راس اصلاً به C متصل نیست و بهتر است آن را به خوشه‌ی دیگری تخصیص دهیم. درجه داخلی C (k_C^{int}) در واقع مجموع درجات داخلی رئوسش هست. همینطور درجه خارجی C (k_C^{ext}) مجموع درجات خارجی همه‌ی راس‌هایش است. بنابراین درجه کل C (K^C) برابر است با:

$$K^C = k_C^{int} + k_C^{ext} \quad (5-2)$$

می‌توان چگالی درون-خوشه‌ای ($\delta_{int}(C)$) زیرگراف C را به عنوان نرخی از تعداد یال‌های درونی C نسبت به کل تعداد یال‌های ممکن در C در نظر گرفت.

$$\delta_{int} = \frac{\# \text{ internal edges of } C}{n_c(n_c-1)/2} \quad (6-2)$$

به طور مشابه چگالی بین-خواهی ($\delta_{ext}(C)$) به عنوان نرخی از تعداد یالهایی که رئوس C را به باقی گراف متصل می‌کند نسبت به تعداد تمام یالهای ممکن که C را می‌تواند به باقی گراف متصل نماید.

$$\delta_{ext} = \frac{\# \text{ inter cluster edges of } C}{n_c(n - n_c)} \quad (7-2)$$

به طور ضمنی یا صریح هدف اکثر الگوریتم‌های خوشبندی گراف، یافتن مصالحه‌ای میان ($\delta_{int}(C)$) بزرگ و ($\delta_{ext}(C)$) کوچک است. به طور مثال یک راه ساده برای انجام این کار بیشینه کردن مقدار $\delta_{ext}(C)$ بر روی همه جزء بندی‌های ممکن روی گراف g است [۱۵].

معمولًا سه دسته تعريف وجود دارد: تعريف محلی، تعريف سراسری و تعريفی بر پایه شباهت رئوس.

۷-۳-۱- تعريف محلی

چهار نوع قاعده برای این کار وجود دارد: تقابل کامل^۱، قابلیت رسیدن^۲، درجه راس، مقایسه میان چسبندگی^۳ داخلی و خارجی. انجمن‌های قابل تطبیق با این تعريف بیشتر زیرگراف‌های بیشینه‌ای هستند که با اضافه کردن یک راس یا حذف کردن یک یال آن خصوصیتی که بر اساس آن، انجمن به وجود آمده است، از بین می‌رود.

انجمن‌های اجتماعی را می‌توان به طور سختگیرانه‌ای اینگونه تعريف کرد که یک زیرگروهی که همه اعضای آن با یکدیگر دوست هستند (قابل کامل). که در گراف به این تعريف کلیکو^۴ می‌گویند. پیدا کردن کلیکوها در یک گراف یک مسئله NP-کامل است. این تعريف خیلی محدود کننده است مثلاً اگر گراف کاملی پیدا شود که تنها یک یال آن حذف شده باشد، آن را به عنوان انجمن در نظر نمی‌گیرد. برای همین تعريف

^۱ Complete mutuality

^۲ Reachability

^۳ Cohesion

^۴ Clique

دیگری که شبیه کلیکو بودند به وجود آمد. باید از امکان قابلیت رسیدن برای تعاریف جدید استفاده میشد. n -clique بزرگترین زیرگرافی است که در آن فاصله هرجفت راس نباید بیشتر از n باشد [۱۶]. که این نیز محدودیتهایی داشت بنابراین n -club و n -clan پیشنهاد شدند [۱۷]. در واقع n -clique بی است که قطر آن نباید بیشتر از n باشد. n -club هم بزرگترین زیرگرافی است که قطر آن n است. قطر یک گراف در واقع بزرگترین کوتاهترین مسیرها در آن گراف است. ضابطه‌ی دیگر زیرگراف‌های چسبنده‌ای است که براساس همسایگی رئوسش به وجود می‌آید. k -plex بزرگترین زیرگرافی که در آن هر راس با همه رئوس دیگر همسایه است به جز حداکثر k تا از آنها یا k -core بزرگترین زیرگرافی است که در آن هر راس با حداقل k راس دیگر همسایه است. همانقدر که یک زیرگراف می‌تواند چسبنده باشد به همان سختی می‌تواند یک انجمن باشد اگر یک چسبندگی قوی میان آن زیرگراف با باقی گراف وجود داشته باشد. بنابراین مقایسه میان چسبندگی داخلی و خارجی یک زیرگراف می‌تواند اهمیت داشته باشد. LS -set به زیرگرافی می‌گویند که درجه داخلی هر راس آن بزرگتر از درجه خارجی آن باشد. یا تعریف دیگر براساس مفهومی به نام اتصال لبه‌ای^۱ ارائه شده است. اتصال لبه‌ای یک جفت راس در یک گراف در واقع کمترین تعداد یالهایی که می‌بایست از گراف حذف شوند تا این دو راس از هم جدا بشوند. مجموعه لمبدا^۲ زیرگرافی که هرجفت راس آن اتصال لبه‌ای بیشتری نسبت به اتصال لبه‌ای آن جفت راس‌هایی دارد، که یک راس در آن زیرگراف است و راس دیگر در آن زیرگراف نیست. تعریف جالب دیگر چگالی نسبی زیرگراف C ($\rho(c)$) است. که در واقع نسبت میان درجه داخلی زیرگراف به کل درجه آن است. یافتن زیرگراف‌هایی که با سایز مشخص و اینکه چگالی نسبی آنها از یک آستانه داده شده بیشتر باشد یک مسئله NP-کامل است [۱۸].

^۱ Edge Connectivity

^۲ Lambda

۲-۳-۷-۲ - تعاریف سراسری

انجمن‌ها می‌توانند با این دیدگاه که به گراف به چشم یک کل نگریسته شود تعریف شوند. برای همین معیارهای سراسری برای تشخیص انجمن‌ها پیشنهاد شده است. در بیشتر موارد آنها تعاریف غیرمستقیمی هستند، که در یک الگوریتم چند خصوصیت سراسری گراف بکار گرفته می‌شود تا در نهایت انجمن‌ها را به عنوان خروجی تحویل دهد. یک دسته از تعاریف مناسب بر پایه این ایده به وجود آمدند که یک گراف ساختار انجمنی دارد اگر آن گراف متفاوت از یک گراف تصادفی باشد. در یک گراف تصادفی احتمال اینکه هر دو راس از آن باهم همسایه باشد و این احتمال در کل گراف برابر باشد، انتظار نداریم که ساختار انجمنی وجود داشته باشد، چون به نظر نمی‌رسد که یک گروه خاص از رئوس، در این نوع گراف گرد هم بیایند. بنابراین می‌توان یک مدل پوچ^۱ را تعریف کرد. یعنی گرافی که در بعضی از ویژگی‌های ساختاری با گراف اصلی تطابق می‌کند اما از طرف دیگر، یک گراف تصادفی است. این مدل پوچ به اصطلاح برای مقایسه به کار گرفته می‌شود تا بتوان دریافت یک گرافی که در حال مطالعه آن هستیم ساختار انجمنی دارد یا خیر. محبوب ترین مدل پوچ توسط نیومن^۲ و گیروان^۳ پیشنهاد شده است. و در واقع نسخه تصادفی شده‌ی گراف اصلی است که در آن یالها به طور تصادفی به اصطلاح دوباره سیم‌بندی^۴ می‌شوند منتها تحت یک محدودیت و آن محدودیت این است که درجه مورد انتظار یک راس باید با درجه آن راس در گراف اصلی مطابقت کند [۶]. و این مدل پوچ مفهوم اساسی هست که پشت تعریف پودمانگی^۵ قرار گرفته است. تابعی

^۱ Null model

^۲ Newman

^۳ Girvan

^۴ Rewired

^۵ Modularity

که میزان خوب بودن تقسیم بندی گراف به خوشهاش را ارزیابی می‌کند. که درباره پودمانگی در بخش‌های بعدی توضیحات تفضیلی ارائه خواهد شد.

۳-۷-۲- تعاریفی بر پایه شباهت رئوس

خیلی طبیعی است که فرض کنیم که یک انجمان یک گروه از راس‌های شبیه به هم است. می‌توان شباهت هر جفت راس را با در نظر گرفتن یک خوصیت مرجع که می‌تواند محلی یا سراسری باشد محاسبه کرد، بدون توجه به اینکه آیا این دو راس به هم متصل هستند یا خیر. معیارهای شباهت در واقع پایه و اساس روش‌های سنتی مانند خوشبندی سلسله مراتبی و خوشبندی جزء‌بندی و خوشبندی طیفی^۱ است. اگر بتوان رئوس یک گراف را به یک فضای n -بعدی اقلیدسی متناظر کرد، می‌توان از "فاصله"^۲ به عنوان یک معیار شباهت (در واقع یک معیار بی شباهتی^۳) استفاده کرد. فرض شود که دو نقطه‌ی داده‌ای $B = (b_1, b_2, \dots, b_n)$ و $A = (a_1, a_2, \dots, a_n)$ استفاده کرد. مثلاً فاصله اقلیدسی نرم- L_2 :

$$d_{AB}^E = \sqrt{\sum_{k=1}^n (a_k - b_k)^2} \quad (8-2)$$

یا فاصله منهتن^۴، نرم- L_1 :

$$d_{AB}^M = \sum_{k=1}^n |a_k - b_k| \quad (9-2)$$

یا معیار محبوب طیفی که شباهت کسینوسی نام دارد، بدین صورت تعریف می‌شود:

$$\rho_{AB} = \arccos \frac{a \cdot b}{\sqrt{\sum_{k=1}^n a_k^2} \sqrt{\sum_{k=1}^n b_k^2}} \quad (10-2)$$

^۱ Spectral

^۲ Dissimilarity

^۳ Manhattan

جاییکه $a.b$ ضرب داخلی دو بردار A و B است و متغیر ρ_{AB} بین $[0, \pi]$ است [۱۸].

اما اگر نتوان گراف را به یک فضای اقلیدسی متناظر کرد، شباهت باید از روابط همسایگی میان رئوس استنتاج شود. یک تعریف ممکن فاصله میان رئوس می‌تواند به صورت زیر تعریف شود [۱۹]:

$$d_{ij} = \sqrt{\sum_{k \neq ij} (A_{ik} - A_{jk})^2} \quad (11-2)$$

که A ماتریس همسایگی است. در واقع این یک معیار نامشابهت است که برپایه مفهوم معادل ساختاری^۱ تعریف شده است [۲۰]. در بخش‌های بعدی نیز در این مورد بیشتر توضیح داده خواهد شد.

یک معیار می‌تواند میزان "همپوشانی"^۲ بین همسایه‌های دو راس i و j که به ترتیب با $\Gamma(i)$ و $\Gamma(j)$ نمایش داده می‌شود، باشد. در واقع نسبت میان اشتراک و اجتماع میان همسایه‌ها:

$$w_{ij} = \frac{|\Gamma(i) \cap \Gamma(j)|}{|\Gamma(i) \cup \Gamma(j)|} \quad (12-2)$$

یک دسته دیگر از معیارهای شباهت که فقط به آن اشاره‌ای میکنیم بر پایه خصوصیات گام‌های تصادفی^۳ تعریف می‌شوند. یکی از این خصوصیات زمان مهاجرت (یا مسافت) بین هر دو راس است که در واقع متوسط تعداد گام‌هایی است که یک راه رونده‌ی تصادفی از هر راسی آغاز می‌کند تا به راس دیگر برسد و به راس آغازین برگردد. که در [۲۱] زمان مسافت به عنوان یک معیار نامشابه استفاده شد. هرچه زمان مسافت بیشتر شود، آن دو راس کمتر به هم شبیه هستند.

^۱ Structural Equivalence

^۲ Overlap

^۳ Random Walk

۸-۲- شبکه‌های اجتماعی

یک شبکه اجتماعی یک ساختار اجتماعی است که از یک مجموعه از بازیگران اجتماعی (که می‌تواند افراد یا سازمان‌ها باشند) و نیز یک مجموعه از پیوندهای دو تایی میان این بازیگران تشکیل شده است. دورنمای شبکه‌های اجتماعی، مجموعه‌ای از روش‌ها را برای تحلیل ساختار همه‌ی موجودیت‌های اجتماعی فراهم می‌کند. همچنین به همان خوبی، نظریات گوناگونی که الگوهای مشاهده شده در این ساختارها را توصیف می‌کنند، فراهم می‌کنند [۱۹]. بررسی این ساختارها از تحلیل شبکه اجتماعی استفاده می‌کند تا الگوهای محلی و سراسری را شناسایی کند، موجودیت‌های اثرگذار را معین کند و پویایی^۱ و نحوه حرکت شبکه‌ها را شرح دهد.

شبکه‌های اجتماعی و تحلیل آنها یک زمینه‌ی کاری دانشگاهی بین رشته‌ای است که از روانشناسی اجتماعی، جامعه‌شناسی، آمار و نظریه گراف پدیدار شده است. جرج سیمل^۲ خیلی زود تئوری‌های ساختاری در جامعه‌شناسی را ارائه کرد که بر روی حرکت سه تایی‌ها و وابستگی‌های شبکه‌ای از گروه‌ها تاکید می‌کرد . جیکوب مورنو^۳ اولین سوسیوگرام^۴ را در سال ۱۹۳۰ توسعه داد. یک سوسیوگرام یک نمایش گرافی از لینک‌های اجتماعی است که یک شخص دارد. در واقع یک نوع کشیدن گراف است که ساختار روابط بین شخصی را در یک وضعیت گروهی ترسیم می‌کند [۲۲]. این روش‌ها به طور ریاضی از ۱۹۵۰ رسمیت پیدا کرد و نظریه‌ها و متدی‌های شبکه اجتماعی در علوم رفتاری و اجتماعی تا ۱۹۸۰ رواج پیدا کرد [۲۳].

^۱ Dynamics

^۲ Georg Simmel

^۳ Jacob Moreno

^۴ Sociogram



۸-۲ شبکه اجتماعی متشكل از افراد و ارتباطات آنها

۹-۲- تاریخچه کوتاهی از شبکه‌های اجتماعی برخط

در برنامه‌های شبکه‌های اجتماعی، هر کاربر به طور معمول توسط یک نمایه^۱ تعریف می‌شود، که همراه با کاربردهایی برای جستجو و جمع‌آوری تماس‌ها یا اتصال‌ها در یک لیست تماس است. برای هر تماسی که برقرار می‌شود، دو طرف به طور متقابل باید بپذیرند که این "پیوند" ایجاد شود. کاربردهای دیگر نظیر چت، آلبوم عکس و دیوار که در آن کاربر می‌تواند پیام‌ها و محتوای خود را منتشر کند، که برای همه افراد موجود در لیست تماس پخش خواهد شد. برنامه‌های آنلاین نظیر بازی‌ها به کاربر این امکان را می‌دهد که با دیگر کاربران سهیم شود، رقابت یا همکاری کند.

یک شبکه‌ی اجتماعی آنلاین می‌تواند به عنوان یک برنامه کاربردی کامپیوتر شناخته شود، که به ایجاد و تعریف روابط اجتماعی میان مردم بر پایه‌ی آشنایی، علاقمندی‌های کلی، فعالیت‌ها، علاقمندی‌های حرفه‌ای، خانواده و ... کمک می‌کند.

^۱ Profile

بعضی از شبکه‌های اجتماعی آنلاین محبوب اینترنت فیسبوک^۱، توئیتر^۲، لینکداین^۳، گوگل پلاس^۴، مای اسپیس^۵ و ... است که محدوده تعداد کاربران در سال ۲۰۱۱ از ۸۰۰ میلیون برای فیسبوک تا ۶۱ میلیون برای مای اپیس بود. کشورهای متفاوت برنامه‌های مخصوص خود را دارند که در داخل کشورشان بسیار محبوب هستند. به طور مثال در چین، رن رن^۶ (معادل فیسبوک) به طور حدودی ۱۶۰ میلیون کاربر ثبت نام شده دارد که ۳۱ میلیون آن فعال هستند. ویبو^۷ (یک برنامه میکروبلاگ اجتماعی شبیه به توئیتر) بیش از ۳۰۰ میلیون کاربر ثبت نام شده دارد. اسپانیا دارای توئنتی^۸ است، فایوهای^۹ در آمریکای جنوبی محبوبیت دارد، اورکات^{۱۰} (هنگ و برزیل)، استادی وی زد^{۱۱} (آلمان)، و اسکای راک^{۱۲} (فرانسه) از جمله‌ی آنها می‌باشد.

^۱ Facebook

^۲ Twitter

^۳ LinkedIn

^۴ Google+

^۵ MySpace

^۶ RenRen

^۷ Weibo

^۸ Tuenti

^۹ 5Hi

^{۱۰} Orkut

^{۱۱} StudiVZ

^{۱۲} Skyrock

بعضی برنامه‌ها برای اشتراک عکس هستند مانند فلیکر^۱ و پیکاسا^۲، یا برای اشتراک موزیک و ویدئو مانند یوتیوب^۳ و اسپاتیفای^۴.

اوایل ۱۹۶۰ بعضی از اولین سرویس‌های شبکه‌ای آنلاین مانند یوزنت^۵، آرپین^۶، و BSS، آمریکا آنلاین^۷ و کامپوسرو^۸ بودند که ویژگی‌های شبکه‌های اجتماعی ابتدایی را نمایش می‌دهند.



۹-۲ رشد قارچ گونه شبکه‌های اجتماعی برخط

^۱ Flickr

^۲ Picasa

^۳ Youtube

^۴ Spotify

^۵ Usenet

^۶ ARPANE

^۷ America Online

^۸ CompuServe

با ظهور وب در ۱۹۹۴، جئوسیتیز^۱ از اولین برنامه‌هایی بود که از این محیط جدید بهره گرفت تا به تعاملات مردم از طریق اتاق‌های چت، کمک کند. سپس در ۱۹۹۷، سیکس دیگریز^۲ قابلیت‌های شبکه‌های اجتماعی آنلاین معاصر خود را برای مدیریت "نمایه‌های کاربر" و "لیست‌های دوستی"، ترکیب کرد. از دیگر برنامه‌های قابل ذکر فرنندستر^۳ در سال ۲۰۰۲ و به دنبال آن مای اسپیس و لینکداین در ۲۰۰۳ بودند. در این سالها بود که انفجار اطلاعات باعث شکوفایی قارچ گونه وب سایت‌های شبکه‌های اجتماعی در اینترنت شد. در سال ۲۰۰۴، فیسبوک راه اندازی شد و تا ۲۰۰۹ بزرگترین سایت شبکه‌ی اجتماعی آنلاین شد. می‌توانیم اعلام کنیم که یک دوره بزرگی از رشد برنامه‌های OSN در خلال سال‌های ۲۰۰۵ تا ۲۰۱۰ آشکار شد.

این در حالیست که ما نگاهی گذرا به سیر شبکه‌های اجتماعی با رویکرد وب داشتیم. بعد از پیدایش گوشی‌های هوشمند، برنامه متنوعی نظریر واپر^۴، تانگو^۵، واتس آپ^۶، اینستاگرام^۷ و خیلی برنامه‌های دیگر که گستره‌ی بسیار زیادی از کاربران را تحت تسلط خود درآورده اند، به وجود آمدند [۲۴].

^۱ Geocities

^۲ SixDegrees

^۳ Friendseter

^۴ Viber

^۵ Tango

^۶ Whatsup

^۷ Instagram

فصل ۳ - پژوهش‌های پیشین

۱-۳- مقدمه

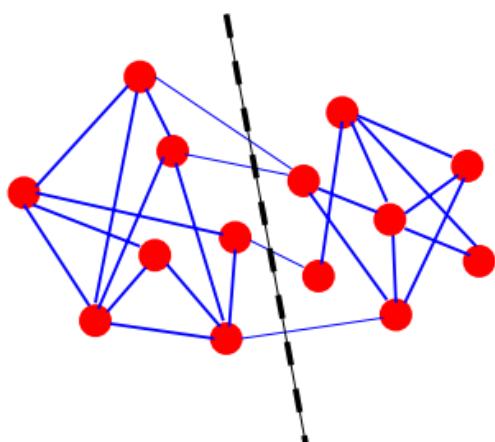
همان طور که قبلا هم عنوان مسئله کشف انجمن‌ها در واقع یک مسئله خوش‌بندی است اما نه خوش‌بندی داده معمولی، در واقع داده این مسئله یک داده‌ی خاص به نام گراف است. پس کشف انجمن‌ها نیز در واقع به نوعی خوش‌بندی گراف است. برای همین از دیدگاهی می‌توان یک دسته از الگوریتم‌ها را معرفی می‌کنیم که به آنها روش‌های سنتی گفته می‌شود. دیدگاه‌های غیرسنتی نیز پس از ارائه یک قاعده برای گراف‌های شیکه‌ی اجتماعی وارد دوره جدیدی شد. در ادامه روند کلی تحقیقات را مورد بررسی قرار می‌دهیم.

۲-۳- روش‌های سنتی

۱-۲-۳- جزء بندی گراف

مسئله جزء بندی^۱ گراف در واقع دسته بندی کردن رئوس در g گروه با اندازه از پیش تعیین شده است. به گونه‌ای که تعداد یالهایی که بین گروه‌ها قرار می‌گیرد کمینه باشد. تعداد لبه‌هایی که میان گروه‌ها قرار می‌گیرد “سایز برش^۲” نام دارد. شکل ۱-۳ نمایش راه حل مسئله برای یک گراف با چهارده راس که به $g=2$ و خوش‌هایی با اندازه برابر را نمایش می‌دهد.

۱-۳ خط چین نمایش راه حل مسئله دوبخشی
کمینه برای گراف ذکر شده است [۱]



^۱ Partitioning

^۲ Cut size

مشخص کردن تعداد خوشه‌های جزء بندی ضروری است. اگر کسی تحمیل کند که کار جزء بندی فقط با کمینه اندازه برش صورت گیرد و تعداد خوشه‌ها مهم نباشد، نتیجه این می‌شود که تمام رئوس در یک خوشه قرار می‌گیرند و اندازه برش به صفر می‌رسد. مشخص شدن اندازه نیز ضروری است در غیر این صورت یک راه حل ناجالب برای این کار این است که گره‌های با کمترین درجه را از باقی گراف جدا کنیم. یکی از اولین روش‌هایی که پیشنهاد شد و هنوز نیز به طور مکرر در ترکیب با سایر روش‌ها استفاده می‌شود الگوریتم کرنینگهان-لین^۱ است [۲۵]. انگیزه نویسنده‌گان در واقع جزء بندی کردن مدارهای الکتریکی رو صفحات مدار بود: گره‌هایی که روی مدارهای متفاوت بودند نیاز داشتند که به هم‌دیگر با کمترین اتصال ممکن پیوند داده شوند. رویه‌ای که یک تابع سود Q را بهینه می‌کرد که این تابع نمایشگر اختلاف بین تعداد یال‌های داخل یک ماثول و یال‌هایی است که میان آنها قرار می‌گیرد. نقطه شروع در واقع تقسیم گراف به دو خوشه با اندازه از پیش تعیین شده است. سپس زیرمجموعه‌هایی که شامل تعداد مساوی از گره‌ها هستند، بین دو گروه جا به جا می‌شوند به گونه‌ای که Q به سمت بیشینه خود حرکت کند و برای جلوگیری از افتادن در تله بیشینه محلی گاهی انتخاب‌ها به گونه‌ای است که Q را کاهش دهد. پیچیدگی زمانی الگوریتم $O(n^2 \log n)$ است که n به طور معمول تعداد رئوس است. بعدها این الگوریتم برای اینکه جزء بندی‌هایی با هر تعداد خوشه دلخواه ارائه دهد گسترش یافت [۲۶].

تکنیک محبوب دیگر در واقع روش دوبخشی طیفی است که برپایه خصوصیات طیفی ماتریس لاپلاس است. خوشبندی طیفی در بخش‌های بعد بیشتر مورد بررسی قرار می‌گیرد. ما بیشتر روی بخش تقسیمی آن تمرکز خواهیم کرد.

هر جزء بندی از یک گراف n راسی به دو گروه می‌تواند توسط یک بردار شاخص s نمایش داده شود که عناصر آن $s_1 + s_2 = 1$ است اگر راس متعلق به گروه یک باشد و -1 است اگر آن راس متعلق به دیگری باشد. اندازه برش یک جزء بندی گراف به دو گروه از رابطه‌ی زیر بدست می‌آید:

^۱ Kernighan-Lin

$$R = \frac{1}{4} S^T L S \quad (1-3)$$

که L در واقع ماتریس لاپلاس است و S^T ترانهاده بردار S است. بردار S را می‌توان به صورت $\sum_i a_i v_i$ نوشت که v_i ($i=1 \dots n$) بردارهای ویژه لاپلاس هستند. اگر S به طور مناسب نرمالسازی شود در این

صورت:

$$R = \sum a_i^2 \lambda_i \quad (2-3)$$

که a_i مقادیر ویژه مطابق با بردارهای ویژه v_i هستند. کمینه کردن R در واقع کمینه کردن رابطه جمع در سمت راست معادله (2-3) است.

نظریه شناخته شده بیشینه-جريان کمینه-برش^۱ فورد و فالکرسون^۲، کمینه برش را بین هر دو راس s و t از گراف قرار داد [۲۷]. یعنی هر زیر مجموعه کمینه‌ای از یالها، که حذف آنها به طور توبولوژیکی s را از t جدا می‌کند، بیشترین جريان را که از s به t در طول گراف می‌توان انتقال داد را، حمل می‌کند. که در واقع در این مفهوم یالها نقش لوله‌های آب با یک ظرفیت مشخص را بازی می‌کنند و گره‌ها نقاط وصل لوله‌ها هستند. در واقع این نظریه برای تعیین برشهای کمینه از جريان بیشینه در الگوریتم‌های خوشبندی استفاده می‌شود. آقای فلیک^۳ و سایرین با استفاده از جريان‌های بیشینه به کشف انجمن‌ها در یک مجموعه داده وب پرداختند [۲۸].

۳-۲-۲- خوشبندی سلسه مراتبی

یک گراف ممکن است ساختار سلسه مراتبی داشته باشد، یعنی ممکن است چند مرحله از گروهی از رئوس را به نمایش بگذارد، خوشهایی کوچک در دل خوشه بزرگ که آنها نیز به نوبه‌ی خود در خوشه‌های بزرگتری قرار می‌گیرند. بعضی از شبکه‌های اجتماعی می‌تواند مثال خوبی برای این موضوع باشند. که در

^۱ Max-flow min-cut

^۲ Ford and Fulkerson

^۳ Flake

اینگونه موارد از الگوریتم‌های خوشبندی سلسله مراتبی استفاده می‌شود یعنی تکنیک‌هایی که ساختارهای چندمرحله‌ای گراف را آشکار می‌کنند.

نقطه آغاز روش خوشبندی سلسله مراتبی تعریف یک معیار شباهت میان رئوس است. بعد از انتخاب، باید این معیار را برای هر جفت راس از گراف بدون توجه به اینکه آیا به هم متصل هستند یا خیر، محاسبه نمود. که در آخر این فرآیند یک ماتریس شباهت $n \times n$ وجود دارد. روش سلسله مراتبی به تشخیص گروه‌هایی از رئوس با شباهت زیاد کمک می‌کند و شامل دو دسته کلی هستند:

۱. الگوریتم‌های تجمیعی، که در آن خوشه‌ها در یک فرآیند تکرار پذیر، اگر به اندازه کافی مشابه

باشند با یکدیگر ادغام می‌شوند.

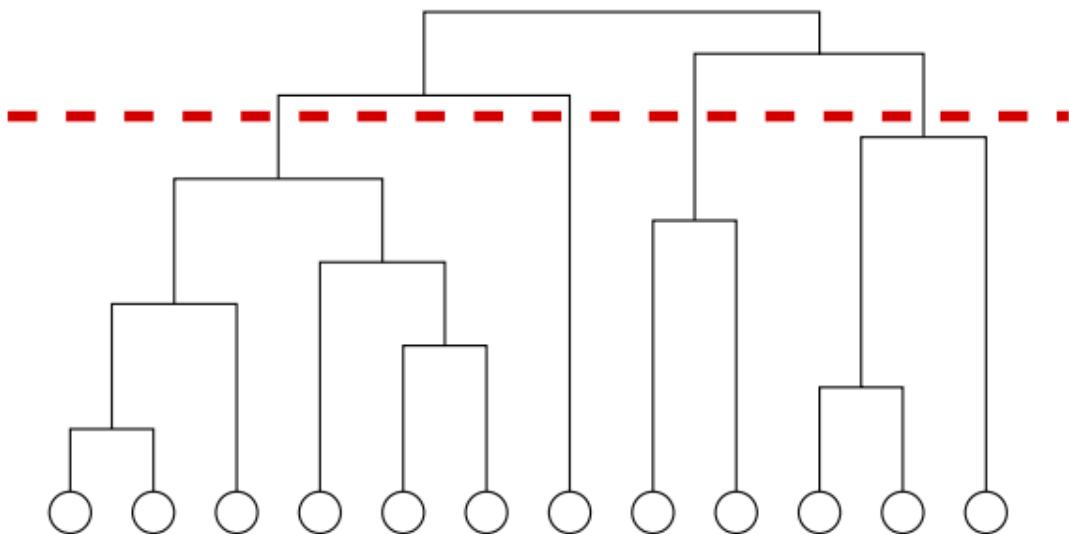
۲. الگوریتم‌های تقسیمی، که در یک فرآیند تکرار پذیر خوشه‌ها به وسیله حذف کردن لبه (یال)

میان رئوسی که شباهت کمی به هم دارند، شکسته می‌شوند.

دو کلاسی که به دو فرآیند معکوس مراجعه می‌کنند. الگوریتم‌های تجمیعی پایین به بالا هستند، یعنی از رئوس که در ابتدا به عنوان خوشه‌های جدا در نظر گرفته می‌شوند شروع کرده تا در نهایت به یک گراف به عنوان یک خوشی یکتا ختم شود. اما الگوریتم‌های تقسیمی بالا به پایین هستند و در جهت عکس مسیر قبلی حرکت می‌کنند.

در الگوریتم‌های تجمیعی از آنجاییکه خوشه‌ها برپایه یک معیار شباهت با یکدیگر ادغام می‌شوند، تعریف یک معیار یا تخمینی که شباهت میان خوشه‌ها را بدون توجه به ماتریس شباهت به ما دهد، بسیار ضروریست. در خوشبندی پیوندی یکتا شباهت میان دو گروه، کوچکترین عنصر X_{ij} (در ماتریس شباهت) است که i در یک گروه و j در گروه دیگر قرار دارد. در مقابل در خوشبندی پیوندی کامل، بزرگترین عنصر X_{ij} در دو گروه متفاوت انتخاب می‌شود و در خوشبندی پیوندی میانگین متوسط عناصر را باید محاسبه و انتخاب کرد.

شکل ۲-۳ می‌تواند این فرآیند به وسیله دندروگرام^۱ بهتر نمایش دهد. گاهی اوقات شروط توقفی تحمیل می‌شود که یک جزء بندی یا گروهی از جزء بندی‌هایی که یک معیار خاص را ارضا میکنند انتخاب کنیم.



۲-۳ یک دندروگرام یا درخت سلسله مراتبی. برش افقی یک جزء بندی را که یک معیار خاص را ارضا می‌نماید نشان میدهد [۶]

خوشه‌بندی سلسله مراتبی این مزیت که در آن نیازی به دانشی درباره اندازه و تعداد خوشه‌ها ندارد، را دارد. اگرچه راهی را هم برای اینکه تمایز میان جزء بندی‌هایی که توسط این رویه بدست آمده قائل شود، و انتخاب آنها که بهتر ساختار انجمنی را نمایش میدهند، مشخص نمی‌کند. علاوه بر این رئوس یک انجمن ممکن است به درستی دسته بندی نشوند حتی در بعضی از موارد که رئوس نقش مرکز بودن را در خوشه خود ایفا میکنند [۲۹]، این مشکل بروز می‌کند. مشکل دیگر هم این است که رئوسی با یک همسایه، معمولاً به عنوان یک خوشه جدا در نظر گرفته می‌شوند. و در نهایت هم یک ضعف الگوریتم‌های تجمیعی این است که به راحتی مقیاس پذیر نیستند. پیچیدگی محاسبات برای پیوند یکتا $O(n^2)$ و برای پیوند کامل و پیوند میانگین از $O(n^2 \log n)$ است.

^۱ Dendrogram

۳-۲-۳ - خوشبندی طیفی

فرض کنیم یک مجموعه از n شی x_1, x_2, \dots, x_n با یک تابع شباهت دوگانی S داریم که متقارن و نامنفی هستند، یعنی $S(x_i, x_j) = S(x_j, x_i) \geq 0, \forall i, j = 1, \dots, n$. خوشبندی طیفی شامل همه روش‌ها و تکنیک‌هایی است که این مجموعه را توسط بردارهای ویژه یک سری ماتریس به خوشبندی طیفی در جزء بندی می‌کند. این ماتریس‌ها مثلاً می‌توانند S یا ماتریس‌هایی که از آن مشتق می‌شود باشند. به ویژه این اشیا می‌توانند نقاطی از بعضی فضاهای متری^۱ باشند یا رئوس یک گراف باشند. خوشبندی طیفی در واقع از انتقال مجموعه اشیای اولیه به مجموعه نقاطی از یک فضایی که ابعاد آن، عناصر بردارهای ویژه $k\text{-means}$ هستند، که این مجموعه نقاط نیز به نوبه خود توسط روش‌های استانداری مثل خوشبندی تقسیم بندی می‌شوند. دلیل اینکه چرا از ابتدا با وجود اشیا و تابع شباهت، عمل خوشبندی صورت نمی‌گیرد این هست که تغییری که در نمایش توسط بردارهای ویژه استنتاج می‌شود، خصوصیات خوشبندی مجموعه داده اولیه را بیشتر نمایان می‌کند. که در واقع خوشبندی طیفی قادر خواهد بود نقاط داده‌ای را خیلی بهتر از زمانی که الگوریتم مثل $k\text{-means}$ مستقیماً اعمال شود، جدا کند.

خوشبندی طیفی نیازمند محاسبه اولین k بردار ویژه ماتریس لایپلاس است. اگر گراف بزرگ باشد محاسبه دقیق همه بردارهای ویژه کاری غیر ممکن است زیرا پیچیدگی محاسباتی آن $O(n^3)$ است. خوشبختانه روش‌های تخمینی مثل روش قدرت یا تکنیک‌های زیرفضای کریلو^۲ مانند روش لنکزووس^۳ به وجود آمدند [۱۸]. که سرعت این روش‌ها وابسته به گپ‌های ویژه $|\lambda_{k+1} - \lambda_k|$ که به ترتیب k -امین و $(k+1)$ -امین کوچکترین مقدار ویژه ماتریس هستند. هرچه گپ‌های ویژه بزرگ‌تر باشند سرعت همگرایی بیشتر است.

^۱ Metric

^۲ Krylov

^۳ Lanczos

۳-۳- الگوریتم‌های تقسیمی

یک راه ساده برای شناسایی انجمن‌ها، کشف یالهایی است که رئوس انجمن‌های مختلف را به یکدیگر وصل کرده‌اند و سپس با حذف کردن آنها، خوشه‌ها از یکدیگر جدا می‌شوند. این فلسفه‌ی الگوریتم‌های تقسیمی^۱ است. نکته مهم پیدا کردن خصوصیتی برای یالهای بین انجمنی است، که منجر به کشف آنها شود. الگوریتم‌های تقسیمی مفهوم ذاتی خاصی را نسبت به روش‌های سنتی معرفی نمی‌کنند، آنها در واقع همان خوشه‌بندی سلسله مراتبی را بروی گراف اجرا می‌کنند. تفاوت اصلی که با خوشه‌بندی سلسله مراتبی تقسیمی دارد، این است که در اینجا یالهای بین خوشه‌ای به جای یالهای میان رئوس با شباهت کم حذف می‌شوند و تضمینی وجود ندارد که یالهای بین خوشه‌ای، رئوس با شباهت کم را به هم متصل کرده باشند. در بعضی موارد ممکن است به جای یالهای تکی، رئوس (با تمام یالهای همسایه اش) یا تمام یک زیرگراف، حذف شوند. در الگوریتم‌های خوشه‌بندی سلسله مراتبی خیلی مرسوم است که برای نمایش جزء بندی‌ها از دندروگرام استفاده شود.

یکی از محبوب‌ترین الگوریتم‌ها توسط نیومن و گیرون پیشنهاد شده است [۳۰]. این روش از نظر تاریخی اهمیت دارد چرا که یک ناحیه جدید را در زمینه کشف انجمن‌ها گشوده است. یالها براساس میزان معیاری به نام مرکزیت یال^۲ انتخاب می‌شوند که در واقع اهمیت نسبی یالها را بر اساس چند خصوصیت و یا یک فرآیندی که روی کل گراف اجرا می‌شود، تخمین می‌زنند. قدم‌های الگوریتم در ادامه آمده است:

۱. محاسبه مرکزیت تمام یالها

۲. حذف یالهایی با بالاترین مرکزیت

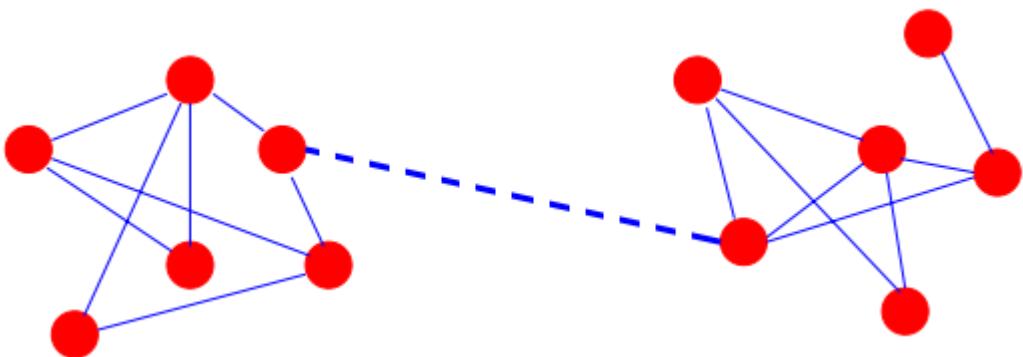
^۱ Divisive

^۲ Edge Centrality

۳. محاسبه مجدد مرکزیت تمام یالها

۴. تکرار این فرآیند از گام دوم

نیومن و گیروان ببروی مفهوم مابینی^۱ تمرکز کردند. متغیری که در واقع تعداد تکرارهای یک یال در یک فرآیند را توصیف می‌کند. که آنها سه تعریف جایگزین را مطرح کردند: مابینی یال کوتاهترین مسیر^۲، مابینی یال گام-تصادفی^۳ و مابینی جریان فعلی^۴. محاسبه مابینی میان همه یالها در یک گراف کم پشت که با تکنیک‌هایی بر پایه جستجوی اول سطح است از $O(mn)$ و $O(n^2)$ است [۳۱, ۳۲].



شکل ۱-۳ مقدار مابینی یال برای یالهایی که انجمن‌ها را به یکدیگر متصل می‌کنند، بیشتر است. در شکل یال خط چین مابینی بیشتری نسبت به بقیه یالها دارد [۱]

یک روش دیگر برای کشف یالهای بین خوش‌های مرتبه با وجود دور^۵‌هاست، که در واقع همان مسیرهای نامتقاطع بسته در گراف است. انجمن‌ها توسط چگالی زیاد یالها مشخص می‌شوند، پس منطقی است که

^۱ Betweenness

^۲ Geodesic

^۳ Random-walk

^۴ Current-flow

^۵ Cycle

انتظار داشته باشیم چنین یالهایی، دورها را تشکیل دهنند. در مقابل بسیار سخت است که یالهای بین انجمن‌ها قسمتی از دورها را تشکیل دهنند. براساس این ایده شهودی، رادیکچی^۱ یک معیار جدید به نام ضریب خوشبندی یالی^۲ را پیشنهاد کرد، که اندازه پایین این معیار برای یک یال، بین انجمنی بودن آن یال را بسیار محتمل می‌کند [۳۳].

یک معیار مرکزیت دیگر برای یالها مرکزیت اطلاعات است. که این نیز بر پایه مفهوم کارآیی است [۳۴]، که در واقع تخمین می‌زند که چقدر اطلاعات به راحتی می‌تواند در گراف، بر طبق طول کوتاهترین مسیرهای بین رؤوس به گردش در بیاید. کارآیی یک شبکه بر طبق میانگین معکوس فاصله‌ها میان همه جفت راس‌های گراف تعریف می‌شود. وقتی رؤوس به یکدیگر نزدیک باشند، کارآیی شبکه بالاست. مرکزیت اطلاعات یک یال در واقع از انحراف نسبی کارآیی گراف زمانی که آن یال حذف شود بدست می‌آید. براساس الگوریتم فورتوناتو^۳ یالهایی که مرکزیت اطلاعات پایینی دارند حذف خواهند شد [۳۵].

۴-۳- روشهایی بر پایه پودمانگی

پودمانگی O نیومن و گیروان در اصل برای یک ضابطه توقف برای الگوریتم نیومن و گیروان معرفی شد. اما به سرعت به یک جزء اساسی الگوریتم‌های خوشبندی تبدیل شد. پودمانگی یکی از پراستفاده‌ترین و محبوب‌ترین معیارهای کیفیت است. که در واقع یکی از اولین تلاش‌ها برای فهم اصول اولیه مسائل خوشبندی گراف را نمایش می‌دهد. ما در این بخش به معرفی روشهای و تکنیک‌هایی که به طور مستقیم یا غیر مستقیم نیازمند پودمانگی هستند، می‌پردازیم.

^۱ Radicchi

^۲ The edge clustering coefficient

^۳ Fortunato

۳-۴-۱- بهینه‌سازی پودمانگی

با این فرض که مقدارهای زیاد پودمانگی نشان دهنده جزء بندی‌های خوب است، بنابراین جزء بندی که با مقدار بیشینه تطابق دارد، باید بهترین جزء بندی یا حداقل یکی از جزء بندی‌های خیلی خوب است. این بیشترین انگیزش برای بیشینه سازی پودمانگی است که در واقع محبوب ترین کلاس از روش‌های کشف انجمن‌ها محسوب می‌شود. بهینه‌سازی کامل پودمانگی غیر ممکن است، زیرا تعداد راههایی که می‌توان یک گراف را جزء بندی کرد بسیار زیاد است حتی اگر گراف کوچک باشد. در کنار اینکه بیشینه درست دور از دسترس است، ثابت شده است که مسئله بهینه‌سازی پودمانگی یک مسئله NP-کامل است [۳۶].

۳-۱-۱-۱- روش حریصانه

اولین الگوریتمی سعی در بیشینه کردن پودمانگی کرد، الگوریتم حریصانه نیومن بود [۳۷]. این الگوریتم یک روش خوشبندی سلسله مراتبی تجمیعی است که گروهی از رئوس برای تشکیل انجمن‌های بزرگ به طور صحیحی به یکدیگر می‌پیوندند به گونه‌ای که بعد از ادغام پودمانگی افزایش پیدا کند. این الگوریتم با n خوشی که شامل یک راس تنها می‌باشد آغاز می‌شود. يالها در ابتدا هیچ حضوری ندارند و آنها یکی یکی در طول این رویه اضافه می‌شوند. اما در هر صورت پودمانگی جزء بندی‌هایی که در طول رویه جستجو می‌شوند از همه‌ی توپولوژی گراف محاسبه می‌شود. اضافه کردن یک یال به مجموعه رئوس نامتصل گراف، تعداد گروه‌ها را از n به $n-1$ کاهش میدهد، بنابراین یک جزء بندی جدید از گراف بدست می‌آید. این یال باید به گونه‌ای انتخاب شود که این جزء بندی، بیشترین افزایش (یا کمترین کاهش) پودمانگی را نسبت به پیکربندی قبلی داشته باشد. سپس همه يالهای دیگر براساس همین اصل اضافه می‌شوند. اگر اضافه کردن يالی، جزء بندی را تغییری نداد در واقع یعنی متعلق به یکی از خوشبندی‌هایی است که قبلاً تشکیل شده است، بنابراین پودمانگی تغییری نخواهد کرد. تعداد جزء بندی‌هایی که در طول این رویه پیدا می‌شود n تا هستند که تعداد خوشبندی‌های هر کدام از آنها از 1 تا n متفاوت است. بزرگترین مقدار پودمانگی در این زیر مجموعه از جزء بندی‌ها، در واقع تخمین پودمانگی هست که به وسیله الگوریتم داده می‌شود. در هر تکرار نیاز به محاسبه تفاوت پودمانگی ΔQ بی هست که از ادغام دو انجمن از جزء بندی در حال اجرا بدست می‌آید.

سپس می‌توان بهترین ادغام را انتخاب کرد. ادغام انجمن‌هایی که با هیچ یالی به هم متصل نیستند منجر به افزایش پودمانگی خواهد شد، بنابراین نیاز هست که متصل بودن هر جفت از انجمن به وسیله یالها بررسی شود، که این نیز نمی‌تواند بیشتر از $O(m)$ باشد. بعد از تصمیم گرفتن درباره اینکه کدام انجمن‌ها باید ادغام شوند، نیاز هست که ماتریس e_{ij} به روزرسانی شود. این ماتریس در واقع بیان کننده نسبت یالهایی است که خوشی i را به خوشی j در جزء بندی در حال اجرا متصل می‌کند. که انجام این عمل نیز در بدترین حالت از $O(n)$ است. بنابراین می‌توان گفت که پیچیدگی زمانی الگوریتم از $O(m + n)$ است یا برای یک گراف خلوت از n^2 است. آقای کلاست^۱ و همکارانش در [۳۸] تمرکز خود را بر روی بروزرسانی ماتریس e_{ij} قرار دادند. آنها معتقد بودند که در زمان خلوتی ماتریس همسایگی این فرآیند تعداد زیادی از عملیات‌های نامفید انجام میدهد. عملیات‌هایی که می‌توانستند به طور خیلی کارآتری با استفاده از ساختمان داده‌های مخصوص ماتریس خلوت، مثل کوه-بیشینه^۲ که داده‌ها در یک درخت دودویی بازسازماندهی می‌شوند، صورت گیرد که پیچیدگی الگوریتم از $O(md \log n)$ است که d عمق دنдрوغرامی است که جزء بندیهایی که در طول اجرای الگوریتم بدست می‌آید را توصیف می‌کند.

روش‌های بهینه‌سازی پودمانگی حریصانه به سرعت به سمت تشکیل انجمن با مصرف کردن انجمن‌های کوچک گرایش دارند. که اغلب مقدارهای ضعیف بیشینه پودمانگی را کسب می‌کنند. دنوں^۳ و همکارانش پیشنهاد کردند که تغییرات پودمانگی ΔQ را که بوسیله ادغام دو انجمن بدست می‌آمد را نرمالیزه شود [۳۹]. یک راه برای جلوگیری از تشکیل انجمن‌های بزرگ به وسیله شوئتز^۴ و کافلیچ^۵ پیشنهاد شد. در این راه به جای اینکه در هر تکرار یک جفت انجمن را برای ادغام کردن انتخاب کنیم، چند جفت انجمن انتخاب می‌شوند [۴۰]. اگر الگوریتم به جای شروع از رؤس تنها از چند پیکربندی منطقی واسطه شروع

^۱ Clauset

^۲ max-heap

^۳ Danon

^۴ Schuetz

^۵ Caflisch

کند، دقت بهینه‌سازی‌های حریصانه در الگوریتم سلسله مراتبی به طور قابل ملاحظه‌ای بهبود پیدا می‌کند

. [۴۱]

۲-۱-۴-۳ - تبرید شبیه سازی شده

تبرید شبیه سازی شده^۱ [۴۲] یک روش احتمالی است که برای بهینه‌سازی سراسری مسائل متخلصی به کار گرفته شده است. این الگوریتم شامل اجرای یک جستجو در فضای حالت‌های ممکن، یافتن بهینه‌ی سراسری یک تابع F و در نهایت دادن بیشینه مقدار آن است. انتقال‌ها از یک حالت به حالت دیگر با احتمال $\exp(\beta\Delta F)$ یک نوع معکوس دما بعد از هر تکرار رخ میدهد. در این فرمول ΔF در واقع کاهش تابع F و β یک شاخص غیر قطعی نویز است. این نویز ریسک به دام افتادن در بهینه‌ی محلی را کاهش می‌دهد. در بعضی مراحل سیستم به یک حالت پایدار همگرا می‌شود که می‌تواند به طور دلخواه‌های یک تقریب خوب از بیشینه F باشد. که این نیز به تعداد حالاتی که جستجو می‌شود و نیز اینکه β چقدر به آهستگی تغییر می‌کند، بستگی دارد. تبرید شبیه سازی شده برای بهینه‌سازی پودمانگی به کار گرفته شده است. در پیاده سازی استاندار [۴۳] دو نوع "جا به جایی" در نظر گرفته شده است. جا به جایی محلی که یک راس تنها از یک خوشه به خوشه‌ای دیگر به طور تصادقی جا به جا می‌شود. جا به جایی سراسری که شامل ادغام‌ها و شکستن انجمان‌هاست. که شکستن‌ها از چند راه متفاوت می‌توانند اجرا شود. در برنامه‌های عملی در واقع n^2 جا به جایی محلی همراه با n جا به جایی سراسری در یک تکرار صورت می‌گیرد. در این روش این پتانسیل وجود دارد که به مقدار بیشینه بسیار نزدیک شد اما این روش آهسته‌ای محسوب می‌شود. پیچیدگی واقعی الگوریتم را نمی‌توان تخمین زد، زیرا به طور سرسختانه نه فقط به سایز و اندازه گراف، که به پارامترهایی که برای بهینه‌سازی انتخاب می‌شوند نیز بستگی دارد مانند دمای اولیه و عامل سردی. که بیشتر این روش برای گراف‌های کوچک مناسب می‌باشد.

^۱ Simulated annealing

۳-۱-۴-۳ - بهینه‌سازی اکسترمال

بهینه‌سازی اکسترمال^۱ (EO) یک روش جستجوی مکاشفه‌ای است که توسط بوچر^۲ و پرکوس^۳ پیشنهاد شده است [۴۴] تا بتواند دقت قابل مقایسه‌ای را با روش تبرید شبیه سازی شده داشته باشد اما دستاورد قابل توجهی در زمان اجرا بدست آورد. این روش بر پایه بهینه کردن متغیرهای محلی است. که این متغیر در واقع مشارکت هر واحد سیستم را برای یک تابع سراسری در دستِ مطالعه، شرح می‌دهد. این روش برای بهینه‌سازی پودمانگی توسط داج^۴ و آرناز^۵ بکار گرفته شد [۴۵]. در واقع پودمانگی را می‌توان به صورت یک جمع بر روی رئوسمش نوشت: که پودمانگی محلی یک راس در واقع مقدار عبارت مطابق در آن جمع است. یک مقدار برازش برای هر راس از تقسیم پودمانگی محلی آن راس به درجه اش محاسبه می‌شود. یک راه این هست که از یک جزء بندی گراف که در حقیقت دو گروه مساوی از رئوس هستند کار آغاز شود. در هر تکرار، یک راس با پایین ترین برازش به خوشی دیگر منتقل می‌شود. این جا به جایی جزء بندی را تغییر می‌دهد در نتیجه نیاز هست که برازش خیلی از رئوس دوباره محاسبه شود. این فرآیند آنقدر ادامه پیدا می‌کند که پودمانگی سراسری Q دیگر به وسیله این رویه بهبود پیدا نکند. برخلاف روش‌هایی مثل جزء بندی بندی گراف که تعداد انجمن‌ها مشخص بود، در این روش این تعداد به وسیله خود الگوریتم بدست می‌آید. بعد از دو قسمتی کردن گراف، هر قسمت به عنوان یک گراف برای خود محسوب می‌شود و فرآیند برای آنها نیز ادامه پیدا می‌کند، این کار تا زمانی که Q افزایش پیدا می‌کند ادامه می‌یابد. همانطور که شرح داده شد این رویه به صورت قطعی از یک جزء بندی اولیه پیشرفت می‌کند و همین طور به صورت سیستماتیک یک راس با برازش کم را جا به جا می‌نماید، بنابراین احتمال اینکه در یک بهینه محلی گرفتار شود، زیاد است. با معرفی انتخاب‌های احتمالی نتایج بهتری کسب شد. در این روش رئوس بر اساس برازش

^۱ External

^۲ Boettcher

^۳ Percus

^۴ Duch

^۵ Arenas

خود رتبه بندی می‌شود و راس با رتبه q با احتمال $p(q) \sim q^{-\tau}$ برداشته می‌شود. در مورد این احتمال میتوان در مقاله [۴۴] بیشتر دانست. الگوریتم تخمین خوبی از بیشینه پودمانگی بدست آورد و همچنین نتایج خوبی را بر روی محک نیومن و گیروان بدست آورد. رتبه بندی مقادیر برازش از $O(n \log n)$ است، که می‌تواند به $O(n)$ کاهش پیدا کند اگر ساختمان داده‌ی آن را به هیپ تغییر بدهیم. پیدا کردن راسی که باید جا به جا شود نیز با یک جستجوی دودویی از $O(\log n)$ انجام پذیر است. در نهایت نیز تعداد گام‌هایی که قرار است این نکته را تایید کند که بیشینه پودمانگی بهبود پیدا می‌کند یا خیر، از $O(n)$ است. بنابراین پیچیدگی زمانی کل این روش از $(n^2 \log n)$ است. نتیجه‌ای که گرفته شد این بود که EO یک مصالحه‌ی خوب میان دقت و سرعت برقرار می‌کند، اگرچه دو قسمتی کردن بازگشتی منجر به نتایج ضعیفی بر روی شبکه‌های بزرگ با انجمن‌های زیاد می‌شود.

۴-۱-۴-۳ - بهینه‌سازی طیفی

پودمانگی می‌تواند به وسیله مقادیر ویژه یا بردارهای ویژه یک ماتریس مخصوص بهینه شود. ماتریس پودمانگی \mathbf{B} که عناصر آن به صورت زیر تعریف می‌شوند:

$$B_{ij} = A_{ij} - \frac{k_i k_j}{2m} \quad (3-3)$$

که توضیحات مربوط به پارامترها در معرفی پودمانگی آورده شده است. اگر s برداری باشد که نمایش دهنده هر جزء بندی از یک گراف با دو خوشه X و Y باشد آنگاه $s_i = 1$ اگر راس i متعلق به خوشه X باشد و $s_i = -1$ اگر راس i متعلق به خوشه Y باشد. بنابراین پودمانگی می‌تواند به صورت زیر نوشته شود:

$$\begin{aligned} Q &= \frac{1}{2m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta(c_i, c_j) \\ &= \frac{1}{4m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) (s_i s_j + 1) = \frac{1}{4m} \sum_{ij} B_{ij} s_i s_j \\ &= \frac{1}{4m} s^T B s \end{aligned} \quad (4-3)$$

عبارت آخری ضرب استاندارد ماتریس‌ها را نمایش می‌دهد. بردار s می‌تواند براساس بردارهای ویژه ماتریس پودمانگی B ، u_i تجزیه شود. که $a_i = u_i^T s$ و $s = \sum_i a_i u_i$. با داخل کردن معادل s در فرمول (۴-۳)

رابطه‌ی زیر بدست می‌آید:

$$Q = \frac{1}{4m} \sum_i a_i u_i^T B \sum_j a_j u_j = \frac{1}{4m} \sum_{i=1}^n (u_i^T \cdot s)^2 \beta_i \quad (5-3)$$

که β_i در واقع مقدار ویژه‌ی B که با بردار ویژه‌ی u_i مطابقت دارد. فرمول (۵-۳) با فرمول (۲-۳) که برای سایز برش در حل مسئله جزء بندی کردن گراف بود، تناظر و تشابه دارد. یک راه می‌تواند بهینه‌سازی پودمانگی در الگوریتم‌های جزء بندی بندی که در قسمت ۱-۲-۳ به آن اشاره شد، از طریق دو قسمتی کردن طیفی باشد. که به جای ماتریس لaplas از ماتریس پودمانگی استفاده می‌شود [۴۶، ۴۷].

۳-۵- روش‌هایی مبتنی بر مکاشفه‌های زیستی

این بسیار ارزشمند است که اشاره کنیم در سالهای اخیر گرایشاتی به سمت استفاده از مکاشفه‌های زیستی و هوش تجمعی برای کشف انجمن‌ها ایجاد شده است. برای همین ما در این بخش به معرفی برخی مقالات و کارهایی که در این زمینه صورت گرفته است می‌پردازیم.

پیزووتی^۱ یک الگوریتم ژنتیک را برای حل مسئله کشف انجمن‌ها پیشنهاد کرد. این الگوریتم یک تابع بازش ساده اما کارآ را که به تشخیص گروههایی از گره‌های متصل به هم چگال منجر می‌شود (که این گروه‌ها بین خود از اتصالات کمتری برخوردار بودند)، بهینه می‌کند. این متد روش عملگرهای انحراف را بهبود داد تا بتواند همبستگی واقعی میان گره‌ها را مورد توجه قرار دهد. بنابراین فضای جستجوی راه حل را به طور معقولانه کاهش می‌دهد [۴۸].

^۱ Pizutti

سادی^۱ در مقاله‌ی خود [۴۹] اشاره می‌کند که با توجه به اینکه خیلی از روش‌های سنتی و تحلیل شبکه‌های اجتماعی در شبکه‌های مقیاس بزرگ، کارآیی خود را از دست می‌دهند. برای همین، روشی برای کاهش کل گراف، به گره‌های کلیکو شده با استفاده از تکنیک‌های بهینه‌سازی کلونی مورچه ارائه شد. نشان داده شد که می‌توان گراف اصلی را به یک اندازه قابل مدیریتی کاهش داد که در آن انجمن‌ها را بر پایه کلیکوها می‌توان در نظر گرفت. پس از کاهش اندازه گراف، آنها از متدهای سنتی برای کشف انجمن‌های شبکه‌های اجتماعی استفاده کردند. بلافاصله نویسنده‌گان روش خود را بهبود بخشیدند. آنها با استفاده از روش نمونه برداری توب برفی^۲، زیرگراف‌هایی را توانستند تولید کنند، سپس تکنیک یافتن کلیکوها بر اساس بهینه‌سازی کلونی مورچه را بر روی هر کدام از زیرگراف‌ها به طور موازی اجرا کردند [۵۰].

لیو^۳ و سایرین یک الگوریتم کشف انجمن‌های ارتباطی را بر پایه مدل خوشبندی مورچه پیشنهاد کردند [۵۱]. در این روش حرکت‌ها به کار گرفته شدند، عملگرهای برداشت و انداختن برای انجام خوشبندی گره‌ها در یک شبکه ایمیل صورت گرفت. همچنین روشی برای اندازه گیری ارتباط خویشاوندی پیشنهاد شد که چند دیدگاه را به طور همزمان برای محاسبه خویشاوندی در نظر می‌گرفت.

ژانگ^۴ و سایرین با در نظر گرفتن معیار پودمانگی به عنوان تابع هدف، یک روش انجمن کاوی در شبکه‌های اجتماعی پویا را پیشنهاد دادند. در این روش از مرکزهای خوشبندی آغازین، بروزرسانی فرومون^۵ و یک تابع مکاشفه‌ای که الگوریتم را برای کسب راه حل خوشبندی راهنمایی می‌کند، استفاده شد [۵۲].

جين^۶ و همکارانش یک الگوریتم بهینه‌سازی مورچه که ACOMRW نام داشت را پیشنهاد کردند [۵۳]. ایده‌ی اصلی این الگوریتم قوی کردن پیوندهای درون-انجمانی و ضعیف کردن پیوندهای بین-انجمانی بود

^۱ Sadi

^۲ Snowball

^۳ Liu

^۴ Zhang

^۵ Pheromone

^۶ Jin

که به طور تکراری و رو به جلویی انجام می‌شد. در هر تکراری یک مدل پیاده روى تصادفی مارکف به وسیله مورچه‌ها به عنوان یک قانون مکاشفه‌ای به کار گرفته می‌شد. همه‌ی راه حل‌های محلی مورچه‌ها از طریق اثر کلی خوشبندی به یک راه حل سراسری مجتمع می‌شد، که بعد از برای بروزرسانی ماتریس فرمونی مورد استفاده قرار می‌گرفت. خوشبختانه این همگرایی به یک راه حل، ساختار انجمنی زیرین شبکه‌های پیچیده را به طور واضحی آشکار می‌کرد.

در سال ۲۰۱۳ آقای ویگاند^۱ به همراه همکارانش فرآیند کشف انجمن‌ها را به عنوان یک مسئله بهینه‌سازی چند هدفه (MOP) برای بررسی ساختارهای انجمنی شبکه‌های اجتماعی پیشنهاد کردند [۵۴]. آنها برای غلبه بر محدودیت‌های مسئله کشف انجمن‌ها یک الگوریتم بهینه‌سازی چند هدفه‌ی جدید را که بر پایه الگوریتم بهبودیافته کرم شب تاب است، پیشنهاد دادند. بنابراین یک مجموعه راه حل (بهینه پارتو^۲) بدست آمد. همچنین در این الگوریتم یک روش تنظیم پارامتر بر اساس مکانیزم بی نظمی^۳ بکار گرفته شد. ایده دیگری که برای بهبود کارآیی الگوریتم در نظر گرفته شد استفاده از یک جهش احتمالی خود-تنظیم جدید بود.

همچنین در ۲۰۱۳ یک روش تشخیص انجمن‌ها بر اساس پودمانگی و یک الگوریتم ژنتیک بهبودیافته (MIGA) توسط جین و همکارانش ارائه شد [۸]. MIGA پودمانگی را به عنوان تابع هدف در نظر گرفت تا بتواند الگوریتم را ساده‌تر نماید. در ضمن برای اینکه بتواند الگوریتم را هدفمند تر کند و نیز برای بهبود دقت و پایداری کشف انجمن، از یک مجموعه اطلاعات قبلی (مثلاً تعداد ساختارهای انجمنی) استفاده می‌کند. همچنین در MIGA از تبرید شبیه سازی شده برای جستجوی محلی استفاده می‌شود که با تنظیم پارامترها، می‌توان توانایی آن را در جستجوهای محلی بیش از پیش بهبود بخشید.

^۱ Wigand

^۲ Pareto

^۳ Chaotic

۳-۶- مرواری بر روشن خوشبندی کلونی مورچه

یکی از علی که مروری اجمالی بر این عنوان را انجام میدهیم این است که ایده‌ی اصلی الگوریتم برپایه چنین خوشبندی هست. مورچه‌ها علاوه بر رفتار غذایابی خود که منجر به ابداع و کشف الگوریتم بهینه‌سازی کلونی مورچه (ACO) گشت، رفتار دیگری نیز از خود بروز می‌دهند که اساس خوشبندی کلونی مورچه است. مورچه‌ها در محیط زندگی خود عادت به تشکیل گروه‌هایی دارند، این گروه‌ها براساس شباهت عادات، تشکیل می‌شوند. یعنی مورچه‌های با عادات شبیه کنار یکدیگرند اما مورچه‌های با عادات متفاوت از یکدیگر دور می‌شوند. در میان بسیاری از تکنیک‌های خوشبندی که از طبیعت الهام گرفته، خوشبندی کلونی مورچه بیشتر مورد توجه بوده است. الگوریتم‌های بر پایه جمعیت به عنوان جایگزینی ضروری برای الگوریتم‌های خوشبندی معمول مثل K-Means هستند. این الگوریتم‌ها توانایی تولید هزینه پایین، تسریع راه حل‌های صحیح منطقی برای مسائل پیچیده‌ی خوشبندی را دارا هستند. به طور مثال چن^۱ و همکارانش یک مدل خواب مورچه^۲ (ASM) را پیشنهاد کردند. در ASM هر داده‌ای توسط یک عامل نمایش داده می‌شود که محل زندگی این عامل یک محیط دوبعدی است. همچنین آنها یک روش قابل سازگار را در [۵۵] نمایش دادند، که در واقع رفتار کلونی‌های مورچه‌های جمعیت دوست^۳ را تقلید می‌کنند. سپس آنها کار خود را گسترش دادند و یک مدل حرکت مورچه (AM) را ارائه کردند [۵۶]. نتایج نشان می‌داد که این روش، راه حل مناسبی برای مسایل خوشبندی با مقیاس بالا و پیچیده هستند.

۷-۳- جمع بندی

در این فصل روش‌های کشف انجمان‌ها و روند تحقیقات در سالهای متمادی و دهه‌های اخیر به طور خلاصه‌ای مورد بررسی قرار گرفت. در این فصل سعی شد هم نگاهی اجمالی به روش‌های سنتی که خود

^۱ Chen

^۲ Ant Sleeping Model

^۳ Gregarious

شامل جزء بندی بندی گراف، خوشبندی سلسله مراتبی و خوشبندی طیفی می‌شود، انداخته شود. و هم روش‌های جدیدتر و غیر سنتی‌تری مثل روش‌های مبتنی بر بهینه‌سازی پویمانگی که این روش‌ها نیز خود به انواع دسته‌ها تقسیم می‌شوند. روش‌های حریصانه، تبرید شبیه سازی شده، بهینه‌سازی اکسترمال و بهینه‌سازی طیفی از جمله زیر شاخه‌های این روش محسوب می‌شوند. همچنین سعی شد روند الگوریتم‌های مبتنی بر هوش تجمعی در سالهای اخیر مورد بررسی قرار گیرد تا اهمیت این الگوریتم‌ها و استفاده‌ی آنها در مسئله‌ی کشف انجمن‌ها نشان داده شود.

در نهایت نیز به معرفی خوشبندی که برپایه رفتار مورچه‌ها هستند پرداختیم، چون همان طور که ذکر شد این نوع خوشبندی اساس کار الگوریتم و روش پیشنهاد ما قرار گرفته است. دقت منطقی، سرعت و در عین حال هزینه پایین از مزیت‌های این نوع خوشبندی محسوب می‌شود.

با توجه به آنچه در بالا ذکر شد به نظر میرسد انتخاب روشی مبتنی بر هوش جمعی و الهام گرفته شده از طبیعت که در سالهای اخیر گرایش به آن زیاد شده است و می‌تواند ما را به نتایج بهتری برساند، عاقلانه و منطقی بوده است و ما نیز همین رویه را در این پایان نامه دنبال کردہ‌ایم.

فصل ۴ - روش پیشنهادی

۴-۱- توصیف کلی

همانطور که در فصل‌های پیشین نیز به آن اشاره شده است، مسئله کشف انجمن‌ها یک مسئله خوشبندی است. اما نه خوشبندی داده‌های معمولی و نه حتی در یک فضای اقلیدسی آسان، بلکه خوشبندی گره‌های یک گراف در اینجا مطرح است. در این نوع ساختار داده‌ای به خوش، انجمن گفته می‌شود. مازول یا گروه نیز از دیگر اسامی هستند که مورد استفاده می‌باشند. در فصول قبل اشاره شده است که هیچ تعریف جهانی برای انجمن وجود ندارد. تنها تعریفی که به طور مشترکی برای همگان قابل قبول است، این تعریف است که یک انجمن به گروهی از گره‌های گراف گفته می‌شود که تعداد پیوندهای درون آن گروه بسیار بیشتر و چگال‌تر از تعداد پیوندهای میان گره‌های همان گروه با دیگر گروه‌هاست. اما این تعریف گنگی است و برای حل این مساله نیاز است، تعریفی رسمی‌تر ارائه گردد و به طور ریاضی مسئله کشف انجمن‌ها را بیان شود. که این تعریف یک دیدگاه از بین دیدگاه‌های موجود است.

برای اینکه بتوان با شبکه‌های اجتماعی در الگوریتم‌ها کار کرد در ابتدا باید یک مدل برای آن در نظر گرفت. مدلی که می‌توان با آن شبکه‌های اجتماعی را نمایش داد، گراف است. اما گراف‌ها نیز برای خود انواع و اقسامی دارند و نمی‌توان برای آن هر گرافی را به کار برد. در این مسئله شبکه‌های اجتماعی به صورت یک گراف غیر جهت دار $G(V, E)$ در نظر گرفته شده است. V در واقع مجموعه‌ی رؤوس (افراد در شبکه‌های اجتماعی) و E مجموعه یالها (پیوندهای میان افراد در شبکه‌های اجتماعی) است. مسئله خوشبندی گراف یا همان کشف انجمن‌ها بیشتر بر روی این موضوع تمرکز می‌کند که گراف G را به یک مجموعه از زیر گراف‌های چگال $G_k, G_1, G_2, G_3, \dots$ طوری تقسیم نماید که شرط زیر را دارا باشد:

- $\bigcup_{1 \leq i \leq k} G_i = G$
- $\bigcap_{1 \leq i \leq k} G_i = \emptyset$

حال اگر یک جزء بندی که ویژگی‌های بالا را ارضا می‌کند و دارای این خاصیت است که پیوندهای درون انجمنی آن چگال و پیوندهای بین انجمنی آن خلوت است، می‌توان این جزء بندی را، یک خوشبندی "خوش تعریف" نامید. در این بخش ما یک الگوریتم جستجوی سراسری را براساس خوشبندی کلونی

مورچه‌ی سازگار برای استخراج انجمن‌ها توسعه دادیم که با استفاده از یک تابع برازش نسبتاً جدید محیط محلی خود را درک می‌کند و یک مدل اختصاص زنبورهای رقصنده را برای تبادل اطلاعات بکار می‌گیرد.

در حقیقت مطالعه و بررسی رفتار مورچه‌های واقعی به طور گستره‌های باعث ایجاد و انگیزش و الهام بخش توسعه‌های زیاد در الگوریتم بهینه‌سازی کلونی مورچه (ACO) و الگوریتم خوشه‌بندی کلونی مورچه (ACC) شده است. زیست شناسان کشف کرده اند که مورچه‌ها به طور طبیعی نزدیک یکدیگر زندگی کرده و سکنی می‌گزینند، بنابراین آنها از یکدیگر مراقبت کرده و به طور جمعی در برابر تهاجمات خارجی ایستادگی می‌کنند. مورچه‌های با عادات شبیه بسیار نزدیک یکدیگر زندگی می‌کنند در حالیکه مورچه‌های با عادات متفاوت به طور نسبی دور از یکدیگر زندگی می‌کنند. بر پایه‌ی چنین رفتار زیستی، مورچه‌ها به طور طبیعی راحت‌ترین محیط را، با جمع شدن شبیه‌ها کنار یکدیگر و دفع کردن متفاوت‌ها از هم، کسب می‌کنند.

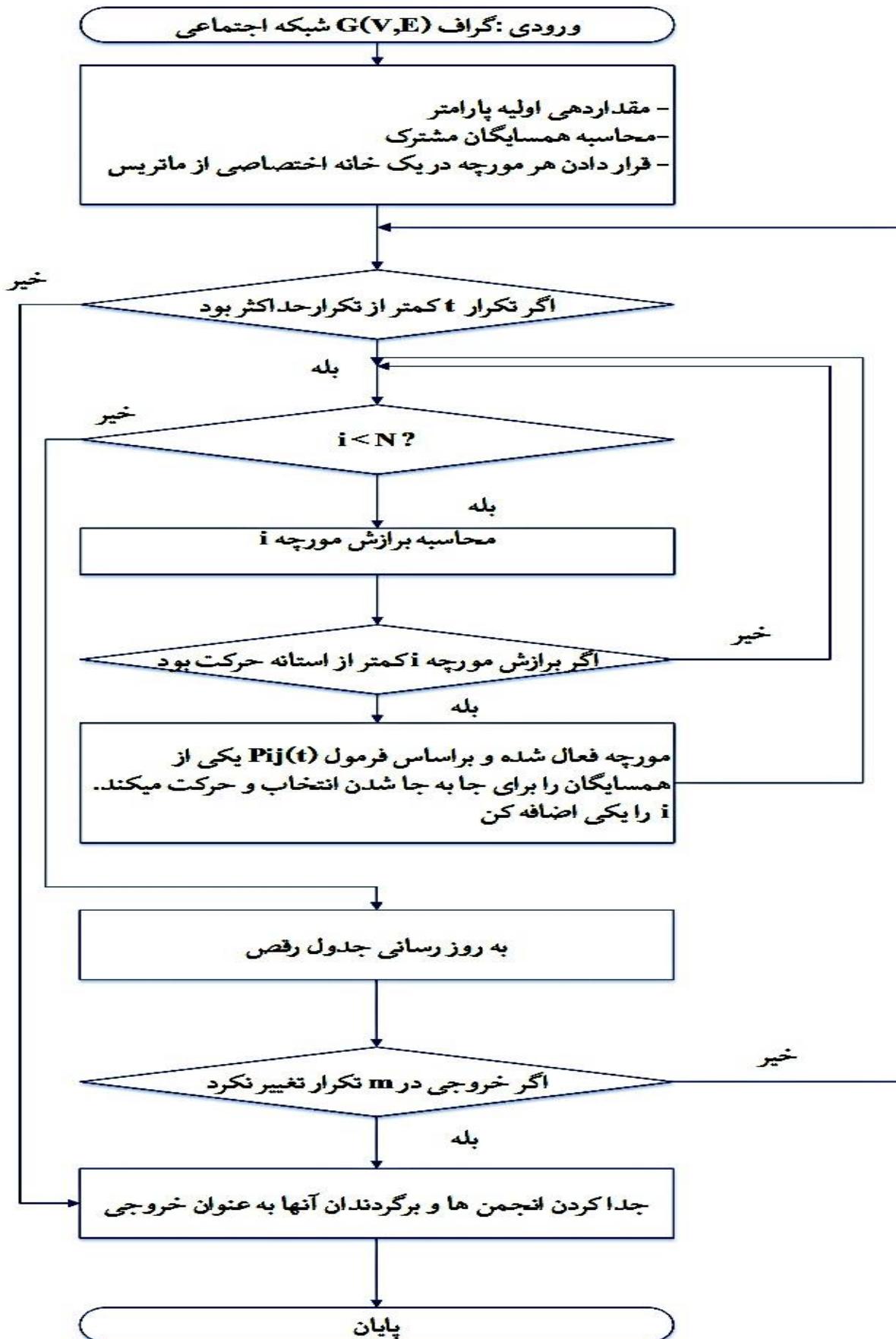
مدل حرکت مورچه (AM) که در فصل قبل کمی در باره آن بحث شد، معمول ترین روشی است که با آن این رفتار مورچه‌ها را که به جستجوی راحت‌ترین مکان برای اقامت در محیط زندگی می‌پردازنند، شبیه سازی می‌کند. بر اساس چنین ایده‌ای، مورچه‌ها و محیط زندگی آنها به شکلی که شرح داده می‌شود، مدل شدند. همه‌ی مورچه‌ها در یک شبکه^۱ دو بعدی (یا همان ماتریس دو بعدی) زندگی می‌کنند. هر مورچه نیز در واقع یک عامل ساده است که یک گره در شبکه اجتماعی را نمایش می‌دهد. در ابتدا مورچه‌ها به طور تصادفی در محل‌های متفاوتی از ماتریس مجازی (محیط زندگی) قرار می‌گیرند. در طول پیشرفت فرآیند، هر مورچه در هر تکرار با استفاده از یک تابع برازش، درک می‌کند و می‌فهمد که آیا باید به یک مکان جدید در محیط زندگی خود برود یا اینکه در محل فعلی خود باقی بماند. این یعنی اینکه اگر یک مکان راحت‌تر برای مورچه وجود داشته باشد، مورچه فعال شده و به مکان جدید خود می‌رود و در غیر این صورت به خواب خود ادامه می‌دهد و در آن مکان باقی می‌ماند. در هر تکرار ممکن است مورچه‌های زیاد یا کمی، حرکت داشته باشند، اما همین حرکت‌های کلونی مورچه است که یک خوشه‌بندی را در هر

^۱ Grid

تکرار شکل می‌دهد، یعنی در هر تکرار یک راه حل ارائه می‌دهد که ممکن است خوب یا بد باشد. پس باید کیفیت این نتیجه‌ی بدست آمده ارزیابی شود. این ارزیابی برای بروزرسانی تعداد زنبورهای رقصدۀ‌ی اختصاص یافته به هر گره بکار گرفته می‌شود. این فرآیند آنقدر تکرار می‌شود تا اینکه همه‌ی مورچه‌ها بتوانند برای خودشان راحت‌ترین مکان ممکن را یافته و در آنجا اقامت کنند. بعد از آن، ملاحظه خواهد شد که ساختار انجمنی پنهان در شبکه‌های اجتماعی و در واقع گراف آنها در ماتریس یا همان محیط زندگی آشکار خواهد شد.

شكل ۱-۴ فلوچارت کلی الگوریتم را نمایش می‌دهد. که در آن می‌توان توصیف خوبی از کارکرد الگوریتم را ارائه کرد. البته برخی مفاهیمی مثل احتمال حرکت هنوز به طور کامل توضیح داده نشده است. اما با داشتن یک دید کلی از آن، می‌توان جزئیات الگوریتم را که در زیر بخش بعدی به تفضیل گفته خواهد شد، آسانتر و قابل فهم تر کند.

لازم به ذکر است که در فلوچارت ذکر شده، t نشان دهنده تکرار فعلی است، N تعداد گره‌های شبکه اجتماعی است و به نوعی تعداد مورچه‌های موجود را هم نشان می‌دهد، چرا که هر مورچه نمایشگر یک گره در شبکه اجتماعی است. i شماره مورچه‌ی در حال پردازش است. π در واقع شماره مورچه‌ی دیگری است که باید احتمال حرکت مورچه i در تکرار t به سمت آن بدست بیاید. m تعداد تکرارهایی است که اگر نتایج خوشبندی تغییری نکند، یعنی الگوریتم همگرای به یک راه حل شده است و متوقف خواهد شد. در ضمن ماتریس در این فلوچارت، همان محل زندگی مورچه‌هاست.



۴-الفواچارت کلی الگوریتم در یک نگاه

۴-۲- توصیف جزییات الگوریتم به تفکیک

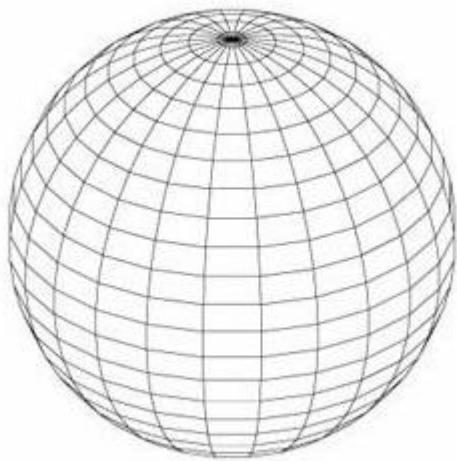
در این بخش ما وارد جزییات بیشتری از الگوریتم پیشنهادی می‌شویم و سعی می‌کنیم که بخش‌های مختلف این روش را به طور موثر و کاملی شرح دهیم.

۴-۱- محل زندگی

هر مورچه (همان گره در گراف اصلی شبکه‌ی اجتماعی) در یک ماتریس دو بعدی قرار می‌گیرد. مورچه‌ها به طور انحصاری یک مکان را اشغال می‌کنند که در ابتدای الگوریتم این امر به صورت تصادفی رخ می‌دهد. انحصاری بودن مکان، بدین معناست که دو یا چند مورچه به طور همزمان نمی‌توانند در یک مکان باشند. به عبارت دیگر اگر تعداد مورچه‌ها یا گره‌های شبکه اجتماعی n باشد، ما در محیط زندگی همیشه n مکان داریم که اشغال شده است و باقی مکان‌ها خالی از سکنه است. ابعاد این محیط زندگی یا ماتریس را می‌توان از رابطه‌ی زیر بدست آورد:

$$N = 10 \times \lfloor \sqrt{n} \rfloor \quad (1-4)$$

N در واقع اندازه یک بعد ماتریس است. بنابراین ابعاد ماتریس $N \times N$ است. این میزان به طور تجربی کسب شده است و با توجه به آزمایشات مختلف انجام شده، این مقدار برگزیده شد. اگر محیط زندگی مورچه بسیار کوچک انتخاب شود که درنتیجه الگوریتم به خوبی قادر نخواهد بود تا انجمن‌های شبکه‌ی اجتماعی را کشف کند، در واقع مورچه‌ها به یکدیگر نزدیک شده و جداسازی آنها برای الگوریتم ممکن نخواهد بود. از طرف دیگر اگر ابعاد آن را بسیار بزرگ در نظر بگیریم، فضا و زمان بیشتری را از دست خواهیم داد. نکته‌ی بسیار مهم دیگری که باید در مورد محیط زندگی مجازی و در واقع ماتریس در نظر گرفت این است که این محیط حتما باید به صورت کروی در نظر گرفته شود. علت این امر را در زیربخش‌های بعدی که بحث کمی واضح‌تر شده است، بیان خواهد شد.



۲-۴ محل زندگی مورچه یا ماتریس باید به صورت کروی در نظر گرفته شود

۴-۲-۲-۴ درک محل زندگی

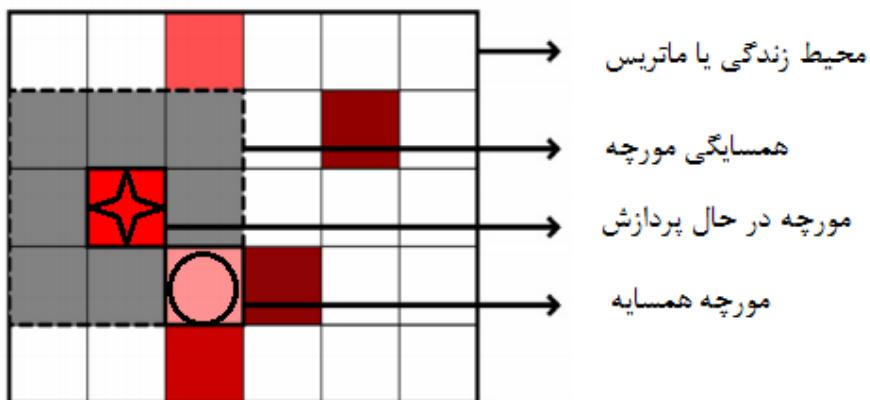
این نکته را مطرح کردیم که هر مورچه‌ای نیاز دارد که بداند مکانی که در آن قرار گرفته است آیا برایش راحت است یا خیر، آیا باید از مکان فعلی به سراغ مکان بهتری برود یا اینکه در همین مکان باقی بماند. برای این منظور از یک تابع برازش و یک پارامتر آستانه حرکت استفاده می‌شود. یعنی عامل یا مورچه با استفاده از این تابع محیط محلی خود را درک و تفسیر می‌کند، خروجی این تابع یک عدد است که در مقایسه با عدد دیگر معنا پیدا می‌کند و آن عدد پارامتر آستانه حرکت است. بنابراین اگر برازش مورچه کمتر از پارامتر آستانه حرکت باشد، بدین معنی است که او در مکان نامناسبی قرار دارد و باید فعال شده و به دنبال مکان بهتری بگردد. اما اگر مقدار برازش از پارامتر آستانه حرکت بیشتر شد، بدین معنی است که مورچه در مکان راحتی قرار دارد و نیازی به حرکت ندارد. قبل از آنکه به معرفی تابع برازش بپردازیم نیاز است که کمی درباره محیط محلی یا همسایگی یک مکان صحبت کنیم.

۴-۲-۱-۱- تعریف همسایگی در محیط زندگی مجازی (ماتریس دو بعدی)

هر مورچه برای اینکه تابع برازش خود را حساب کند و به تبع آن بتواند به این سوال پاسخ دهد که آیا مکان فعلی اش، جای مناسبی برای اوست یا خیر، ناچار است به اطراف خود نگاه کند. یعنی باید پاسخ را در همسایگی خود جستجو نماید. همسایگی در یک ماتریس با توجه به دو پارامتر محدوده دید افقی و محدوده دید عمودی تعریف می شود. این دو پارامتر قابل تنظیم هستند. معنای این دو پارامتر این است که از مکان فعلی مورچه چقدر به طور افقی به چپ یا راست و چقدر به طور عمودی به بالا یا پایین، می تواند به عنوان همسایه اش توجه کند. در واقع یک مستطیل که مرکز آن مکان فعلی عامل یا مورچه است و ابعاد آن از رابطه‌ی زیر بدست می آید:

$$(2 \times \text{horizontalVision} + 1) \times (2 \times \text{verticalVision} + 1) \quad (2-4)$$

شکل ۳-۴ می تواند نمونه‌ای از این همسایگی را نمایش دهد:



۳-۴ نمونه ای از یک مورچه در حال پردازش و همسایگی آن. در این مثال محدوده دید افقی و محدوده دید عمودی هردو یک هستند.

بعد از تعریف همسایگی، زمان آن رسیده است که به تعریف تابع برازش بپردازیم.

۴-۲-۲-۲- تابع برازش

برای ایجاد تناظر و ارتباط میان درک یک مورچه از محیط اطراف خودش با الگوریتم نیاز به یک میزان یا مقدار یا عدد هست. این عدد توسط یک تابع که به آن تابع برازش می‌گویند بدست می‌آید. خروجی که توسط یک تابع برازش داده می‌شود در زمان‌های مختلف و شرایط محیطی متفاوت می‌تواند تغییر کند. از این رو برای محاسبه برازش مورچه n در تکرار t باید از رابطه زیر استفاده کنیم:

$$f_i(t) = \frac{1}{|A|} \sum_{j \in A} \frac{1}{1 + d_{ij}} \quad (3-4)$$

در این فرمول A در واقع همسایه‌های مورچه i محسوب می‌شوند. $|A|$ تعداد همسایه‌ها را نشان می‌دهد. d_{ij} نیز نشان دهنده فاصله میان دو گره i و j در گراف اصلی شبکه‌های اجتماعی هستند. در ادامه بیشتر با این معیار فاصله آشنا خواهیم شد.

فاصله میان گره‌ها

اگر اشیایی که باید خوشه‌بندی شوند، گره‌های یک گراف یا شبکه باشند، شباهت میان گره‌ها با اصطلاحی به نام معادل ساختاری^۱ قابل تعریف می‌شود. معادل ساختاری یکی از مفاهیم کلیدی در شبکه‌های اجتماعی محسوب می‌شود که توسط لورین و وايت^۲ در سال ۱۹۷۱ معرفی و تعریف شده است. اساساً دو گره از گراف با یکدیگر معادل ساختاری هستند اگر آنها الگوهای ارتباطی مشابهی را با دیگر گره‌های گراف داشته باشند [۲۰]. برای مثال دو فرد از یک شبکه اجتماعی دوستی، اگر دوستان یکسانی داشته باشند با همدیگر معادل ساختاری هستند. راههای مختلفی برای محاسبه معادل ساختاری وجود دارد، و چنین معیارهایی برای شناسایی گروه‌هایی از گره‌ها که به طور ساختاری مشابه یکدیگر هستند و با گره‌های سایر گروه‌ها

^۱ Structural equivalence

^۲ Lorrain and White

تفاوت دارند، استفاده می‌شوند. استفاده از فاصله اقلیدسی به عنوان یک معیار از معادل ساختاری بودن، در ابتدا توسط برت^۱ معرفی شد. این فاصله میان دو گره i و j که در واقع فاصله میان ردیف‌ها (یا ستون‌ها) i و j از ماتریس همسایگی است، از رابطه‌ی زیر بدست می‌آید [۵۷]:

$$d_{ij} = \sqrt{\sum_{k=1, k \neq i, j}^n (a_{ik} - a_{kj})^2} \quad (4-4)$$

که درایه موجود در ماتریس همسایگی است. اگر دو گره i و j به طور دقیق معادل ساختاری یکدیگر باشند، آنگاه عناصر مطابق موجود در ماتریس همسایگی آنها باید یکسان باشد و $d_{ij} = 0$ است. این فاصله اقلیدسی همه‌ی معیارهای فاصله را دارد. یعنی:

- $d_{ij} \geq 0, \forall i, j$
- $d_{ii} = 0, \forall i$
- $d_{ij} = d_{ji}, \forall i, j$

ماتریس همسایگی

این خیلی معمول است که یک نمایش ماتریسی از یک گراف را مورد توجه قرار دهیم. یک گراف (V, E) می‌تواند به طور کامل توسط یک ماتریس توصیف شود. این ماتریس، ماتریس همسایگی^۲ یا اتصال^۳ نامیده می‌شود. این ماتریس که معمولاً با حرف A نمایش داده می‌شود به شکل یک مربع است که ابعاد آن $V \times V$ است. درایه موجود در این ماتریس a_{ij} ($i, j = 1, 2, 3, \dots, N$) نشانده‌نه وضعيت و ارتباط میان دو گره i و j است. اگر بین دو گره i و j پیوند و یا یالی وجود داشته باشد آنگاه درایه a_{ij} برابر ۱ است و اگر هیچ یالی بین آن دو وجود نداشته باشد a_{ij} برابر صفر است. بنابراین ماتریس همسایگی ماتریسی از صفر و یک‌هاست. از نکات مهم این ماتریس این است که در گراف‌های بدون جهت، متقارن هستند.

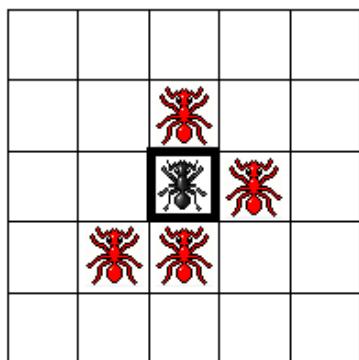
بهبودی بر فرمول برآش

^۱ Burt

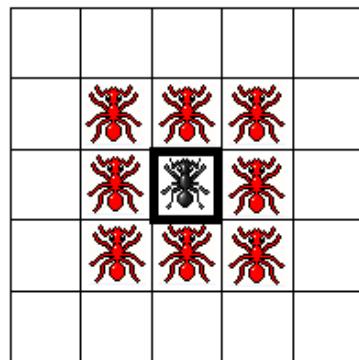
^۲ Adjacency

^۳ Connectivity

از آنجاییکه با فرمول (۳-۴) بیشتر آشنا شدیم، می‌توان ایراد کوچک آن را بیان کرد و آن را کمی بهبود بخشد. دو حالت را متصور شوید که در یکی یک مورچه، چهار همسایه داشته باشد و در حالت دیگر هشت همسایه وجود داشته باشد. فرض دیگری که مطرح می‌شود این است که فاصله میان همه مورچه‌ها را برابر با صفر بدانیم. در این صورت هم در حالت اول و هم در حالت دوم میزان برآش برابر با یک است. این در حالی است که حالت دوم برای مورچه حالت مطلوب تری است.



ب



الف

۴-۴ اگر فاصله میان همه مورچه‌ها برابر صفر باشد، در این صورت برآش مورچه سیاه در هردو حالت برابر یک خواهد بود

از این رو ما فرمول (۳-۴) را به شکل زیر تغییر می‌دهیم تا این خطای فرمول رفع گردد.

$$f_i(t) = \frac{1}{(2 \times horizontalVision + 1) \times (2 \times verticalVision + 1)} \sum_{j \in A} \frac{1}{1 + d_{ij}} \quad (5-4)$$

یعنی به جای اینکه مجموع عبارات بخش بر تعداد همسایه‌ها شود، بخش بر تعداد مکان‌های همسایه می‌شود. که با این فرمول برآش در شکل ۴-۴ ب برابر با 5^0 و در شکل ۴-۴ الف برابر با ۱ است. با توجه به مطالب قبلی می‌توان نتیجه گرفت که هرچه تعداد گره‌هایی که به یک گره نزدیکترند، بیشتر باشد آنگاه برآش آن گره بیشتر شده و نهایتاً دارای مکان بهتری خواهد بود.

۴-۲-۳- پارامتر آستانه حرکت و تنظیم کردن سازگارانه‌ی آن

پارامتر آستانه حرکت یکی از مهم ترین پارامترهایی است که در این الگوریتم وجود دارد. این نکته بیان شد که تابع برازش به تنها ی نمی‌تواند مشخص کننده این باشد که آیا یک مورچه در مکان مناسبی قرار دارد یا خیر؟ خروجی تابع برازش صرفاً یک عدد است که در مقایسه با عدد دیگری می‌تواند معنا دقیق‌تری پیدا کند. به بیان دقیق‌تر اگر در تکرار t برای مورچه i داشته باشیم که $f_i(t) < MT$ ، آنگاه مورچه فعال می‌شود و بر اساس استراتژی حرکتی که وجود دارد به دنبال مکان‌های بهتری می‌گردد. MT نام پارامتر آستانه حرکت^۱ است و در واقع مخفف واژه انگلیسی آن است. در غیر انصورت مورچه به خوابیدن خود ادامه داده و در مکان خود باقی می‌ماند. در واقع حرکت مورچه‌ها به این پارامتر وابسته است. اگر مقدار این پارامتر بسیار کوچک باشد، در نتیجه گره‌ها بیشتر احساس راحتی کرده و در نتیجه کمتر حرکت خواهند کرد، و به دنبال آن الگوریتم خیلی زود به یک راه حل نامناسب همگرا خواهد شد. و بالعکس اگر این پارامتر بسیار بزرگ شود، مورچه‌ها به طور دائم از مکان خود احساس ناخرسنده کرده و اغلب در حال حرکت خواهند بود و به تبع آن الگوریتم به راه حلی همگرا نخواهد شد. بنابراین سعی شد این الگوریتم به طور خود-سازگارانه‌ای به تنظیم این پارامتر در طول پیشرفت فرآیند بپردازد.

ایده‌ی اصلی به این صورت است که اگر نتیجه خوشبندی بهتر شد، مقدار آستانه حرکت کاهش داده می‌شود، تا مورچه‌ها کمتر حرکت کنند و نتایج خوب بیشتر حفظ شود. در مقابل اگر نتیجه خوشبندی بدتر شد، مقدار پارامتر را افزایش می‌دهیم تا مورچه‌ها را به حرکت وا داریم و محدوده‌ی جستجو را افزایش دهیم. قانون تنظیم این پارامتر به شکل زیر تعریف می‌شود:

$$MT = \begin{cases} (1 + \varepsilon) \times MT & \text{if } Q(t) < Q(t - 1) \\ (1 - \varepsilon) \times MT & \text{if } Q(t) > Q(t - 1) \end{cases} \quad (6-4)$$

^۱ Moving Threshold

ع ضریب ثابتی است که برای کنترل و تنظیم آستانه حرکت مورد استفاده قرار می‌گیرد. و $Q(t)$ میزان کیفیت راه حل در تکرار t را نشان می‌دهد. که در بخش‌های بعدی درباره آن صحبت خواهد شد. در اینجا کاملاً مشخص است که چگونه از یک معیار سراسری برای یادگیری و راهنمایی الگوریتم استفاده شده است.

۴-۲-۴ - استراتژی حرکت

اگر در هر تکرار، عامل یا مورچه حس کند که مکان راحت تری برایش وجود دارد، فعال شده و تلاش می‌کند تا به مکان جدیدی در ماتریس برود. در الگوریتم خوشبندی کلونی مورچه که معرفی شد، نحوه حرکت مورچه‌ها بدین گونه است که از همسایه‌های خالی خود یکی را به صورت تصادفی انتخاب کرده و بدان جا می‌رود. که می‌توان نحوه حرکت را کمی تغییر داد. اولین گام این است که به جای انتخاب یک خانه خالی در همسایگی گره در ماتریس، مکان جدید را از همسایگی همسایگانش در گراف اصلی (V, E) انتخاب نماییم. یعنی در ابتدایکی از همسایه‌های خود در گراف اصلی را انتخاب کرده و سپس به یک مکان خالی در همسایگی او می‌رویم. این انتخاب با توجه به یک احتمال صورت می‌گیرد.

فرض می‌کنیم که i و j دو همسایه در گراف شبکه اجتماعی ما باشند، $p_{ij}(t)$ احتمال انتخاب گره j توسط گره i به عنوان گره هدف را نمایش می‌دهد:

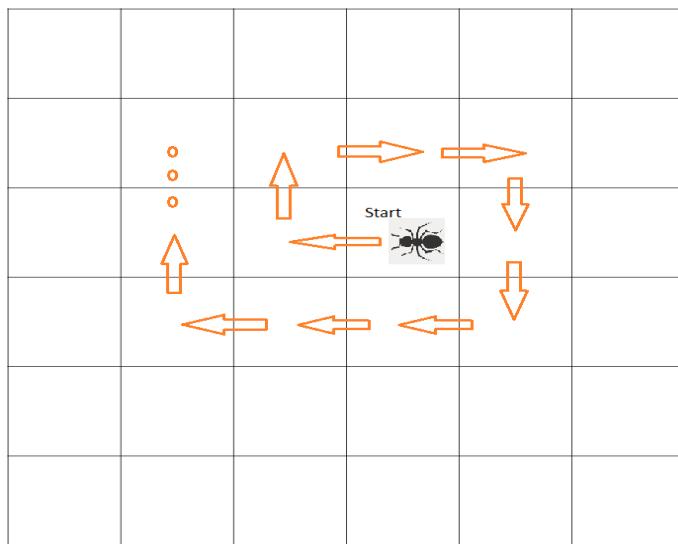
$$p_{ij}(t) = \begin{cases} \frac{[\tau_j(t)]^\alpha \cdot [\eta_j]^\beta}{\sum_{k \in \text{Neighbor}(i)} [\tau_k(t)]^\alpha \cdot [\eta_k]^\beta} & j \in \text{Neighbor}(i) \\ 0 & \text{otherwise} \end{cases} \quad (7-4)$$

اینجا جایی است که بهینه‌سازی زیبور عسل وارد شده و نقشی را در این الگوریتم به عهده می‌گیرد. در رابطه‌ی (7-4)، $\tau_j(t)$ نشان دهنده‌ی تعداد زیبورهای رقصدنده‌ای است که به گره j در تکرار t اختصاص داده شده است، که چگونگی تغییرات آن در زیر بخش بعدی بحث خواهد شد. یعنی درایه τ_j -ام جدول رقص را نشان میدهد. $(\tau_j(t))$ در واقع نشانه یادگیری الگوریتم است که از مراحل قبلی بدست آمده است. پارامترهای α و β نیز اهمیت نسبی تعداد رقصدنده‌ها را نسبت به فاکتور مکاشفه نشان می‌دهد.

η_j در واقع اطلاعات ما از مساله و همان مکاشفه‌ای است که در الگوریتم بهینه‌سازی کندوی زنبور عسل و سایر الگوریتم‌های مکاشفه‌ای به چشم می‌خورد. مقدار η_j از رابطه‌ی زیر بدست می‌آید:

$$\eta_j = \begin{cases} commonNeighbor(i,j) & \text{if } \exists l \in Neighbor(i), commonNeighbor(i,l) \neq 0 \\ \frac{\theta}{d_{ij}} & \text{if } \forall l \in Neighbor(i), commonNeighbor(i,l) = 0 \end{cases} \quad (8-4)$$

$commonNeighbor(i,j)$ در واقع تعداد همسایه‌های مشترک گره i و گره j می‌باشد. d_{ij} فاصله‌ی دو گره از همدیگر است و θ در واقع یک ضریب ثابت است که در ابتدای مساله تعیین می‌شود و باید به نحوی باشد که انحراف و تغییرات مقدار مکاشفه‌ها خیلی زیاد نباشد. معنای رابطه‌ی (8-4) این است که اگر در میان همسایه‌های i در گراف اصلی، گره‌ای وجود داشته باشد که i با آن همسایه مشترک داشته باشد، مکاشفه ما تعداد همسایه‌های مشترک i و j خواهد بود. اما اگر i با هیچ یک از همسایگان گرافی اش، همسایه مشترک نداشته باشد باید از قسمت دوم رابطه استفاده شود که با فاصله‌ی دو گره نسبت عکس دارد. بعد از محاسبه فرمول احتمال، بر طبق روش حریصانه مورچه، گره‌ای که بیشترین احتمال را دارد انتخاب کرده و به یافتن مکان خالی اطراف آن می‌گردد. ضمناً ما برای یافتن مکان‌های خالی اطراف یک گره از یک روش پیمایش مارپیچی استفاده کردیم. به طوریکه که ابتدا همسایه‌های درجه یک در ماتریس را نگاه می‌کنیم، اگر مکان خالی وجود نداشته باشد به سراغ همسایگان درجه دو می‌رویم و همین طور



5-4 طرح کلی حرکت مارپیچی در
محیط زندگی، یا همان ماتریس

ادامه می‌دهیم تا نهایتاً به خانه‌های خالی دست پیدا میکنیم. نحوه حرکت مارپیچی از مرکز، که گره هدف است، شروع می‌شود تا مکان‌های خالی اطراف گره را پیدا کند.

اهمیت کروی بودن

حال که برخی جزیئات الگوریتم بیان شده و برخی نقاط تاریک آن روشن شده است، زمان آن فرا رسیده است تا علت کروی بودن محیط زندگی یا ماتریس بیان شود. پاسخ ساده است: برای اینکه همه مکان‌های ماتریس از اهمیت یکسان و یکتایی برخوردار شوند. همانطور که قبلاً عنوان شده ماتریس، ابعادی مربع شکل دارد، و اگر بخواهیم با این محیط به صورت مربع نگاه کنیم، آنگاه موقعیت مکان‌های نزدیک اضلاع و گوشه‌های این ماتریس، نسبت به دیگر نقاط ماتریس متفاوت می‌شود، چرا که تعداد همسایگان آنها متفاوت خواهد بود، و همچنین تعداد مکان‌های خالی اطراف آنها متفاوت از بقیه می‌توانست باشد. برای جلوگیری از این امر، ماتریس به صورت کروی در نظر گرفته شده است. کره بودن به معنای حقیقی آن نیست بلکه بدین معناست که به طور مثال همسایه‌ی قبلی ستون یک، ستون n -ام است و یا همسایه بعدی ردیف n -ام، ردیف یک است. این موضوع مهم باید هم در محاسبه برازش گره‌ها و هم در نحوه حرکت به مکان جدید و نیز در هنگام استخراج انجمن‌ها مورد توجه قرار گیرد.

۴-۵-۲- نحوه بروزرسانی زنبورهای رقصنده و جدول رقص

یکی از مهمترین بخش‌های الگوریتم که در واقع ایده هوش جمعی را وارد الگوریتم می‌کند، به روز رسانی جدول رقص و تعداد رقصنده‌های است. نویسنده‌گان در مقاله [۵۸] مطرح کردند که به طور کلی، هرچه تعداد گره‌های بیشتری در اطراف محل زندگی یک گره، در ماتریس جمع شوند، آن گره می‌تواند گره‌های بیشتری را به خود جذب نماید. که این نیز به نوبه‌ی خود بدین معناست که رقصنده‌های بیشتری به آن گره اختصاص پیدا می‌کند. و گره‌های موجود در همسایگی‌های نزدیک تاثیر بیشتری نسبت به گره‌های موجود در همسایگی‌های دورتر خواهند داشت. از طرف دیگر زنبورهای رقصنده بعد از مدتی که در سالن رقص مشغول رقصیدن می‌شوند، از آنجا خارج می‌شوند. ضمن اینکه با مصرف مواد غذایی که زنبورهای رقصنده به آن اشاره می‌کنند، با گذشت زمان از تعداد زنبورهای رقصنده مختص آن کاسته می‌شود. در واقع این نکات به

قابلیت پویش و انتفاع الگوریتم اشاره می‌کنند. برای شبیه سازی این پدیده ما یک روش اختصاص زنبورهای رقصنده و به تبع آن یک فرمول به روزرسانی را معرفی میکنیم. ایده‌ی پایه‌ای این بود که بتوان که اثر زنبورهای رقصنده را در میان مکان‌های نزدیک در محیط زندگی (ماتریس) مورد توجه قرار داد تا بتوان خوشه‌بندی کلونی مورچه را به سمت یک راه حل قابل دستیابی و مناسب راهنمایی کرد. به عبارتی نوعی اثرباری اتصالی میان مکان‌های همسایه در تخصیص زنبورهای رقصنده وجود دارد. دو مکان با فاصله‌ی نزدیک تر تاثیر قوی‌تری بر روی همدیگر می‌گذارند و بر عکس دو مکان با فاصله دورتر اثر کمتری بر روی هم خواهند گذاشت.

بعد از اینکه در هر تکرار یک راه حل بدست آمد، زنبورهای رقصنده اختصاص یافته به گره‌ها باید بروزرسانی شوند. الگوریتم به طور ویژه‌ای در سایه سود و منفعت یا همان بهبود کیفیت راه حل به بروزرسانی تعداد رقصنده‌های همه گره‌ها خواهد پرداخت. فرمول یا رابطه‌ای که برای بروزرسانی در نظر گرفته شده است، به صورت زیر است:

$$\tau_i(t+1) = \omega \times \tau_i(t) + \Delta\tau_i(t, t-1) \quad (9-4)$$

$\tau_i(t+1)$ در واقع تعداد رقصنده‌های اختصاص یافته به گره i در تکرار $t+1$ را نمایش می‌دهد. برای اینکه بتوانیم در طول زمان از تعداد رقصنده‌های اختصاص یافته به یک گره (در واقع شبیه سازی کاهش منابع غذایی و رقصنده در گذر زمان) بکاهیم، از ω به عنوان ضریب کاهش تعداد رقصنده‌ها استفاده کرده‌ایم. که این ضریب کاهش تعداد رقصنده‌ها عددی بین صفر و یک است. که از راه تجربه و آزمایش می‌توان مقدار مناسب آن را بدست آورد. و در نهایت $(1 - \Delta\tau_i(t, t-1))$ افزایش تعداد رقصنده‌هایی است که دیگر گره‌ها به گرهی i اختصاص می‌دهند. که این افزایش به چند چیز بستگی دارد که مهمترین آن تغییراتی است که در محیط زندگی مورچه‌ها (ماتریس) رخ می‌دهد. دیگر اینکه آیا حرکت و جا به جایی مورچه، حرکت خوبی بوده است یا خیر، به عبارت دیگر آیا مورچه به مکان بهتری رفته است یا به مکان بدتری منتقل شده است. دست آخر نیز گفتیم که به روزرسانی بر پایه منفعت و سودی است که الگوریتم کسب کرده، یعنی این افزایش به تغییرات کیفیت راه حل بدست آمده در پایان هر تکرار بستگی دارد.

حرکت خوب یا بد؟

گفتیم که افزایش تعداد زنبورهای رقصنده بستگی به این دارد که آیا مورچه i به مکان بهتری رفته است یا خیر؟ که باید به این سوال قبل از نشان دادن چگونگی افزایش پاسخ داد. برای تشخیص این خوب بودن حرکت یا بد بودن آن باید به سراغ تغییرات تابع برازش برویم. یعنی اگر $\Delta f_i(t) > 0$ باشد می‌توان نتیجه گرفت که گره i به مکان بهتری (نسبت به مکان قبلی) رفته است و بالعکس اگر $\Delta f_i(t) < 0$ باشد این معنی را می‌دهد که گره i به مکان نامناسبتری (نسبت به مکان قبلی) منتقل شده است. برای محاسبه $\Delta f_i(t)$ از رابطه‌ی زیر استفاده می‌کنیم:

$$\Delta f_i(t) = f_i(t) - f_i(t-1) \quad (10-4)$$

تعیین کیفیت راه حل

بعد از اینکه الگوریتم یک دور کامل اجرا شد، نحوه‌ی کنار هم آمدن مورچه‌ها، یک خوشبندی را تشکیل می‌دهد، یعنی یک راه حل کسب شده است. برای تعیین کیفیت این راه حل روش‌های مختلفی پیشنهاد شده است. به طور مثال متوسط برازش همه‌ی گره‌ها، به عنوان کیفیت مطرح شده است [۵۸]. اما در آزمایشات به این نتیجه رسیده شد که این معیار خوبی برای نشان دادن کیفیت خوشبندی نمی‌تواند باشد، حتی در مواردی که خوشبندی به درستی انجام می‌شود، مقدار آن کمتر از راه حل‌های دیگر بود. بنابراین ما به سراغ معیار دیگری رفتیم. محبوب ترین معیارهایی که در کشف انجمن‌ها وجود دارد پودمانگی نیومن و گیروان است [۶] که به تفضیل به معرفی آن در فصل قبل پرداخته شده است. ما از تعریف معادل پودمانگی

استفاده کردیم:

$$Q = \sum_{c=1}^{n_c} \left[\frac{l_c}{m} - \left(\frac{d_c}{2m} \right)^2 \right] \quad (11-4)$$

که در این رابطه n_c تعداد انجمن‌ها را نمایش میدهد، l_c تعداد یالهایی که رئوس انجمن c را به هم متصل کرده است را نشان میدهد، d_c در واقع مجموع درجات همه رئوس موجود در c است. و در نهایت m نیز تعداد کل یال‌های موجود در گراف را ارائه می‌کند.

بنابراین اگر بخواهیم بفهمیم که آیا الگوریتم در این تکرار سود کرده است یا ضرر، به عبارت دیگر راه حل تکاملی بهتری پیدا کرده است یا خیر، باید تغییرات پودمانگی را مورد بررسی و تفسیر قرار دهیم. اگر $\Delta Q(t) > 0$ باشد، یعنی به راه حل بهتری رسیده‌ایم و بالعکس اگر $\Delta Q(t) < 0$ باشد بدین معنی است که نتیجه خوشبندی بدتر شده است. $\Delta Q(t)$ را که به عنوان یک تاثیر تکاملی در نظر گرفت، به صورت زیر قابل توصیف است:

$$\Delta Q(t) = Q(t) - Q(t-1) \quad (12-4)$$

با مقدماتی که در بالا ذکر شد و بر اساس تغییراتی که به صورت محلی و سراسری در محیط زندگی مورچه‌ها (ماتریس) رخ می‌دهد، میزان افزایش رقصدنه‌ها، $\Delta \tau_i(t, t-1)$ را می‌توان از رابطه‌ی زیر بدست آورد:

$$\Delta \tau_i(t, t-1) = \begin{cases} \sum_{j \in A} \omega \times \tau_j \gg_i(t) + C.P & \text{if } \Delta f_i(t) > 0 \text{ and } \Delta Q(t) > 0 \\ \max\left(\sum_{j \in A} \omega \times \tau_j \gg_i(t) - C.p, 0\right) & \text{if } \Delta f_i(t) < 0 \text{ and } \Delta Q(t) < 0 \\ \sum_{j \in A} \omega \times \tau_j \gg_i(t) & \text{otherwise} \end{cases} \quad (13-4)$$

که در این رابطه A مجموعه همسایه‌های گره i در محیط زندگی را نمایش می‌دهد. $\tau_j \gg_i(t)$ تعداد رقصدنه‌هایی است که گره j به گره i اضافه می‌کند، که در ادامه توضیحات تفضیلی درباره آن داده خواهد شد. اما $C.P$ در یک پارامتر کنترلی برای میزان افزایش یا کاهش رقصدنه‌هاست.

اثر گره‌های دیگر در افزایش تعداد رقصدنه‌های یک گره

گفتیم که $\tau_j \gg_i(t)$ تعداد رقصدنه‌هایی است که گره j به گره i اضافه می‌کند، این همان تاثیری است که گفته شد مکان نزدیک برهم می‌گذارند. این افزایش براساس رابطه‌ی زیر بدست می‌آید:

$$\tau_{j \gg i}(t) = \frac{1}{2^{NeighboringLevel(j)}} \tau_j(t) \quad (14-4)$$

یا درجه یا مرحله‌ی همسایگی گره j ، این را نشان می‌دهد که این گره در چه فاصله یا شعاعی از گره i قرار دارد. بر روی شکل ۶-۴ می‌توان این موضوع را به خوبی درک کرد.

			:			
	2	2	2	2	2	
	2	1	1	1	2	
...	2	1	i	1	2	...
	2	1	1	1	2	
	2	2	2	2	2	
			:			

۶-۴ اعداد، درجه یا مرحله‌ی همسایگی مکان‌های همسایه را نسبت به گره i نشان میدهند.

یکی از نقاطی که ایده هوش جمعی وارد الگوریتم می‌شود، همینجاست. در توصیف رابطه‌ی (۱۳-۴)

این نکته بسیار شفاف و البته حائز اهمیت است که افزایش تعداد زنبورهای رقصدنده در واقع مصالحه‌ای

میان ارزیابی سراسری یک راه حل و برازش محلی یک گره است. سه حالت موجود که بیان می‌کند:

۱. زمانی که کیفیت کلی راه حل بهتر شده باشد و مورچه به مکان بهتری رفته باشد، تعداد رقصدنده‌های

آن را افزایش می‌دهیم.

۲. زمانی که کیفیت کلی راه حل بدتر شده باشد و نیز گره به مکان بدی نسبت مکان قبلی خود رفته

باشد، ما تعداد زنبورهای رقصدنده را به طور ملایمی افزایش می‌دهیم.

۳. در غیر اینصورت فقط تغییراتی که در محیط زندگی مورچه (ماتریس) رخ داده است را بازتاب

می‌شود.

در حقیقت این استراتژی اثر یک راه حل خوب را قوی‌تر می‌کند، در حالیکه تاثیر یک راه حل بد، تضعیف

می‌شود و این به طور کامل ایده هوش جمعی را پیاده سازی می‌کند.

۴-۳- پیچیدگی زمانی الگوریتم

شیه کد ساده‌ای از الگوریتم را در شکل ۷-۴ می‌بینید. همان طور که مشخص است الگوریتم از چند فاز تشکیل شده است. فاز اول مرحله مقدار دهی اولیه است که به محاسبه همسایه‌های مشترک می‌پردازد، و گره‌ها (مورچه) را در یک خانه‌ی ماتریس به طور تصادفی قرار می‌دهد. فاز دوم مرحله خوشبندی است، که در آن هر گره شروع به درک اطراف محل زندگی خود می‌کند که اگر در مکان مناسبی نبود، شروع به انتخاب یکی از همسایگان گرافی با توجه به تعداد زنبورهای رقصنده‌ها و مکاشفه‌ها می‌کند. بعد از آن براساس خوب بودن حرکت و همچنین کیفیت خوشبندی، تعداد رقصنده‌های هر گره به روزرسانی می‌شود. سپس نوبت به بروز کردن پارامتر آستانه حرکت می‌رسد. نهایتاً در فاز سوم نیز انجمان‌ها کشف و استخراج می‌شوند.

از مقدمه‌ی بالا برای تحلیل پیچیدگی زمانی الگوریتم استفاده می‌شود. ابتدا فرض می‌کنیم که بیشینه درجه رئوس در گراف شبکه اجتماعی d_{max} است. نکته‌ی مهمی که در پیاده سازی باید رعایت می‌شد تا بتوان پیچیدگی زمانی را کاهش داد، استفاده همزمان از ساختمان داده‌های لیست پیوندی و آرایه برای ماتریس همسایگی بود. با توجه به مقدمه‌ای که ذکر شد، فاز اول الگوریتم، محاسبه تعداد همسایه‌های مشترک، از $O(d_{max}^2 \cdot N)$ است. نکته مهم این است که همسایه‌های مشترک فقط برای گره‌هایی که با هم همسایه‌ی گرافی هستند، محاسبه می‌شود و نه برای تمامی گره‌ها. در فاز دوم نیز فرض می‌کنیم که تعداد همسایه‌های محل زندگی (ماتریس) $neighbor_{max}$ است. که به ترتیب بخش یک فاز دو، شامل درک و حرکت $(d_{max} + d_{max} \cdot neighbor_{max}) \cdot N$ است، بخش دوم شامل به روزرسانی رقصنده‌هاست، نیز برای تعیین خوشبندی هر گره‌ای به گره‌های همسایه خود نگاه می‌کند و شماره انجمان آنها را به خود اختصاص می‌دهد که این نیز $O(neighbor_{max} \cdot N)$ است. بنابراین پیچیدگی زمانی کل الگوریتم بدین صورت است:

$$O(d_{max}^2 \cdot N) + O(T_{max} \cdot (d_{max} + d_{max} \cdot neighbor_{max} + 1) \cdot N)$$

اما با توجه به اینکه شبکه‌های اجتماعی دارای ویژگی جهان-کوچک هستند و نیز جزو شبکه‌های مقیاس-آزاد محسوب می‌شوند، پس $d_{max} \ll N$ است. همچنین با توجه به آزمایشات ما، معمولاً در ماتریس همسایگی مرحله اول و نهایتاً در گراف‌های بزرگ همسایگی مرحله دو را نگاه می‌کنیم بنابراین میباشد که این نیز بسیار کمتر از N است. علاوه بر این در آزمایشات ما حداقل ۲۴ حداکثر $neighbor_{max}$ تعداد دورهای تکرار الگوریتم از T_{max} تجاوز نکرده است، بنابراین الگوریتم پیشنهادی با تخمین از پیچیدگی زمانی $O(d_{max}^2 \cdot N)$ است که نسبت به $O(N^2)$ بسیار بهتر است و کارآتر عمل می‌کند.

Algorithm : CDHHO

Input: Graph $G(V, E)$ of Social Network

① Initialization

initialize all parameters $T_{max}, movingThreshold(0), \tau(0), \varepsilon, \alpha, \beta$

N: colony size (number of nodes)

common neighbors

each node (ant) must be put in a cell of matrix

For $t = 1$ to T_{max} do{

② Clustering phase

i. for $i = 1$ to N do{

compute Distances to Grid Neighbors

compute fitness $f_i(t)$

if $f_i(t) < movingThreshold$ then{

ant activated and must be go to another place

compute transfer probability $p_{ij}(t)$ for each neighbor in graph

select one neighbor (node k) and go to empty cell around k in grid}}

ii. for $i = 1$ to N do

update dance table (number of dancers assigns to a node)

iii. updating movingThreshold parameter

③ Result

if the solution not change in m iterations then

return each separated community of social network graph

۴-۴- جمع بندی

در این فصل سعی شد روشی را که برای کشف انجمن‌ها از شبکه‌های اجتماعی طراحی شده است، به طور واضحی ارائه دهیم. گفتیم که مسئله کشف انجمن‌ها در واقع نوعی خوشه‌بندی است اما خوشه‌بندی داده‌های معمولی نیست بلکه داده‌ها در این مسئله گره و یالهای یک گراف هستند. در این روش از خوشه‌بندی کلونی مورچه به عنوان اساس الگوریتم استفاده شد، که این روش یک نوع جستجوی محلی برای داده‌های معمولی بود. در ابتدا باید این روش را با داده گرافی سازگار می‌شد، که این کار در سایه تعاریفی مثل معادل ساختاری و فاصله اقلیدسی که در گراف مطرح می‌شود و همچنین تعریف تابع برازشی بر اساس آن صورت گرفت.

همچنین ما از یک روش سازگارانه برای تعیین پارامتر آستانه حرکت که در الگوریتم خوشه‌بندی مورچه وجود دارد، استفاده کردیم، تا از بازخورد سراسری الگوریتم بتوانیم نتایج خوب را حفظ کرده و نتایج بد را تاحد ممکن به دست فراموشی بسپاریم.

اما همان طور که قبلا ذکر شده است برای بهبود دقت این روش جستجوی محلی کافی نبود. بنابراین سعی شد این روش را با یک روش جستجوی سراسری ادغام و ترکیب کنیم. تا بتوان از مزایا هر دو نوع جستجو برای بهبود کارآیی الگوریتم بهره جست. برای جستجوی سراسری نیز از الگوریتم بهینه‌سازی کندوی زنبور عسل استفاده شد، تا بتوان از قدرت این الگوریتم در راستای جستجوی سراسری استفاده کرد و کارآیی و دقت الگوریتم را بهبود بخشید.

فصل ۵-

نتایج و ارزیابی روش پیشنهادی

۱-۵ - مقدمه

در این فصل به ارزیابی و مقایسه روش پیشنهادی در این پایان نامه پرداخته می‌شود. همچنین در این فصل معیارهای ارزیابی در زمینه کشف انجمن‌ها معرفی می‌شوند، مجموعه داده‌هایی که در این پایان نامه مورد استفاده قرار گرفته است، بررسی می‌شوند. پارامتر و مقداردهی آنها در الگوریتم مورد کند و کاو قرار خواهد گرفت. و در نهایت نیز روش پیشنهادی در این پایان نامه با دیگر روش‌ها مقایسه خواهد شد.

۲-۵ - مجموعه داده‌ها

برای اینکه بتوان کارآیی الگوریتم مورد تحلیل و اندازه گیری قرار داد، هم از مجموعه داده‌های شبکه‌های اجتماعی واقعی^۱ و هم از شبکه‌های تولید شده توسط کامپیوتر^۲ استفاده شد.

۱-۲-۵ - شبکه‌های واقعی

در ابتدای الگوریتم پیشنهادی بر روی سه مجموعه داده‌ای که به طور گستردگی مورد استفاده الگوریتم‌ها قرار می‌گیرد و همچنین ساختار انجمنی آنها مشخص است، اعمال شد. این سه مجموعه داده از شهرت خوبی در زمینه و ناحیه کاری کشف انجمن‌ها برخودار هستند. این سه عبارتند از: شبکه باشگاه کاراته^۳ [۵۹]، شبکه دلفین‌های دست آموز^۴ [۶۰]، شبکه فوتبال دانشگاهی آمریکا^۵ [۳۰].

^۱ Real-world network

^۲ Computer generated network

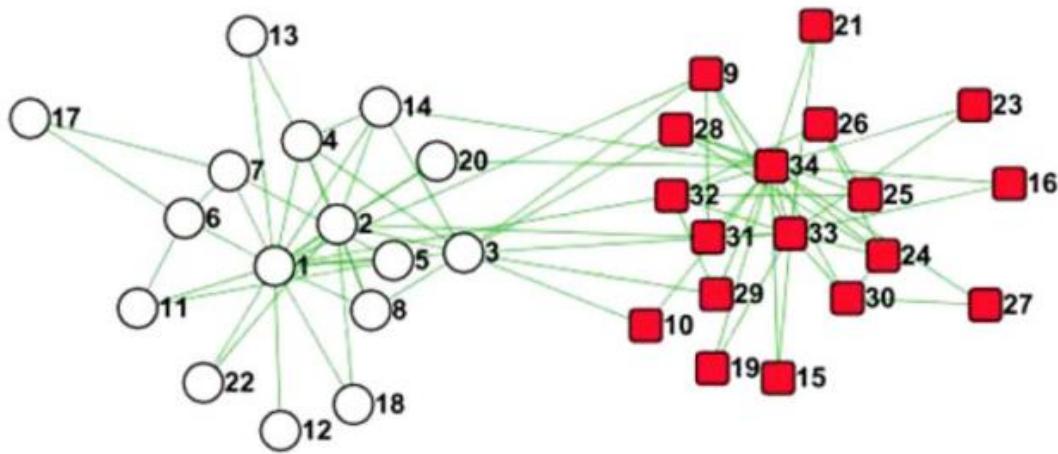
^۳ Karate Club Network

^۴ Bottlenose Dolphin Network

^۵ American College football Network

۱-۱-۲-۵ - مجموعه داده باشگاه کاراته

مجموعه داده باشگاه کاراته توسط زاخاری^۱ ساخته شده است. این یک شبکه‌ی دوستی مشتمل بر ۳۴ عضو یک باشگاه کاراته است. در واقع اعضا را می‌توان گره‌های این شبکه در نظر گرفت. در این صورت روابط دوستی میان اعضا را می‌توان با یال‌های گراف نشان داد. که تعداد آنها نیز ۷۸ یال است. بر اساس مباحث مربوط به رهبری و ریبیس و مرئوسی، باشگاه به دو گروه مجزا شکسته شد. این شبکه ساده همانطور گفته شد در اکثر ارزیابی الگوریتم‌ها مورد استفاده قرار می‌گیرد.



۱-۵ شبکه باشگاه کاراته زاخاری. مریع یک انجمن و دایره هم انجمن دیگر را نمایش می‌دهد.

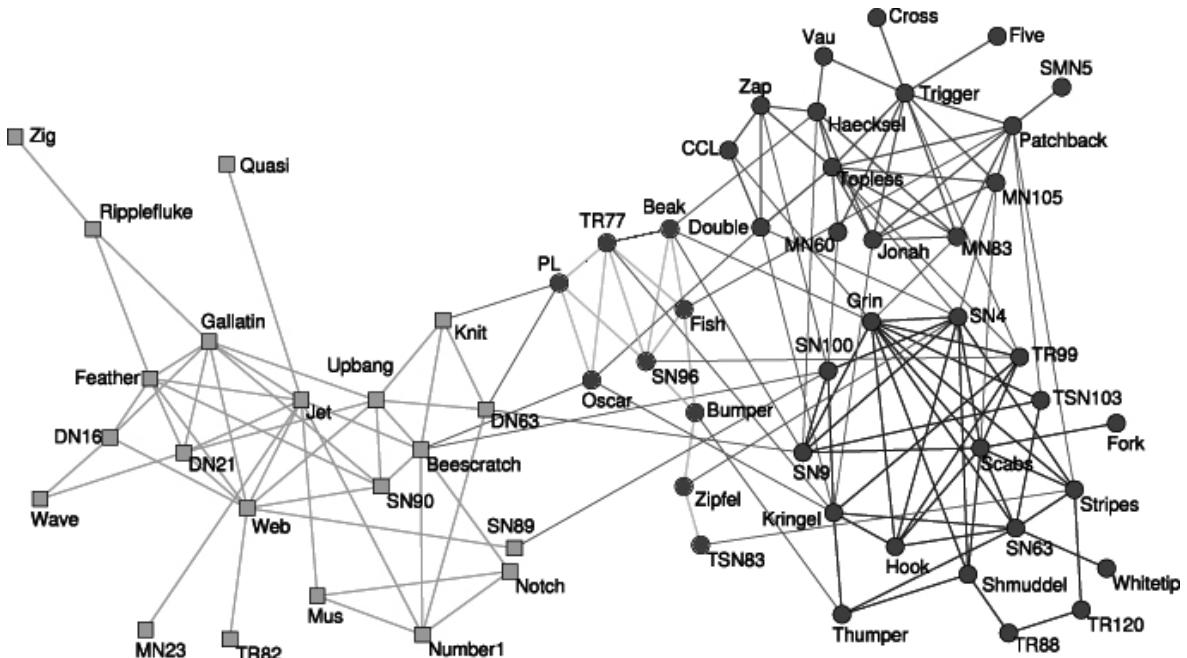
۱-۱-۲-۶ - مجموعه داده شبکه دلفین‌های دست آموز

مجموعه داده شبکه دلفین‌های دست آموز یک انجمن از دلفین‌های دست آموز است که در نیوزلند زندگی می‌کنند. این شبکه از مشاهده‌ی رفتار دلفین‌ها در طول هفت سال از زندگی آنها توسط فردی به نام لوسيو^۲ ساخته شده است. ۶۲ دلفین در این شبکه وجود دارد که گره‌های گراف را تشکیل می‌دهند. تعداد یال‌های موجود در گراف ۱۵۹ است. این یال بین اعضا بر این اساس به وجود آمدند که دو عضو بیش

^۱ Zachary

^۲ Lusseau

از حد مورد انتظار شناسی با یکدیگر باشند، به وجود آمدند. حد مورد انتظار شناسی یعنی در طول ساعت زندگی دو دلفین ممکن است به طور اتفاقی در کنار هم قرار بگیرند که این که زمان و حد مشخصی داشته است. این شبکه به طور طبیعی به دو گروه بزرگ شکسته شد. که در واقع به گروه دخترها و گروه پسرها تقسیم شدند.



۲-۵ شبکه های دلفین های دست آموز، دایره نشان دهنده یک انجمان و مربع نشان دهنده انجمن دیگر است

۳-۱-۲-۵- مجموعه داده شبکه فوتبال دانشگاهی آمریکا

مجموعه داده فوتبال دانشگاهی آمریکا، در واقع متعلق به فوتبال دانشگاهی ایالات متحده آمریکاست. این شبکه در واقع بازی های فوتبال در بخش IA را در طول فصل ۲۰۰۰ این بازی ها به نمایش می گذارد. گره ها در این شبکه نشان دهنده تیم های شرکت کننده است و یال ها، مسابقات معمول بین این تیم ها در آن فصل را نمایش می دهد. شبکه به ۱۲ کنفرانس (انجمان ها) شکسته شد. تیم ها به طور میانگین ۴ مسابقه بین کنفرانسی و ۷ مسابقه درون کنفرانسی انجام دادند. بنابراین تیم ها به انجام مسابقات درون کنفرانسی گرایش داشتند. این شبکه مشتمل بر ۱۱۵ گره و ۶۱۶ یال بود که به ۱۲ گروه تقسیم شد. البته بعد از بررسی هایی که انجام دادیم، متوجه شدیم این مجموعه داده دارای سه یال تکراری درون خود است. که در واقع تعداد یال های آن به ۶۱۳ گره می رسد.

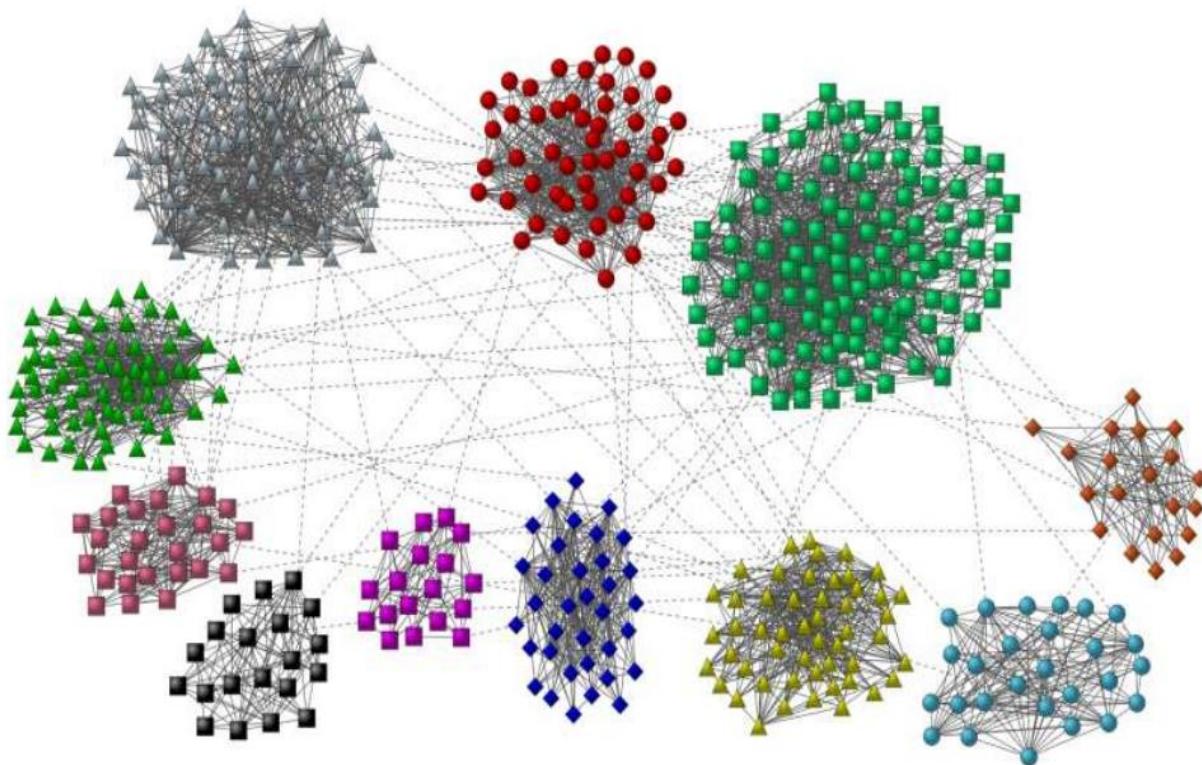
۵-۲-۲- شبکه‌های مصنوعی یا تولیدشده توسط کامپیوتر

یکی از نکات مهمی که درباره الگوریتم‌های کشف انجمنی وجود دارد، بحث آزمایش کردن آنهاست، یعنی یک الگوریتم در مقایسه با دیگران چقدر خوب است؟، بحثی است که هنوز باز مانده است و نیاز به کار دارد. بعضی آزمایش‌های استاندارد ساده با یک ساختار انجمنی پیش ساخته وجود دارد که الگوریتم‌ها می‌توانند آنها را بازیابی کنند. اما این گراف‌ها با شبکه‌های واقعی فاصله دارند. برای تولید شبکه‌های تولید شده توسط کامپیوتر، یکی از مشهورترین آزمون‌ها، متند کلاسیک نیومون و گیروان است [۳۰]. در این آزمون شبکه‌ها، دارای ۱۲۸ گره بودند که به چهار گروه ۳۲ تایی تقسیم می‌شدند که متوسط درجه رئوس نیز ۱۶ است. گره‌ها تمایل داشتند که بیشتر با همگروهی‌های خود ارتباط داشته باشند تا بیرون. یک پارامتر k_{out} نمایش دهنده درصد لینکهایی است که یک گره با اعضای خارج انجمن به اشتراک می‌گذارد. الگوریتم‌های خوب معمولاً باید بتوانند از پس این آزمایش بریباشند. اما این آزمون دارای معایبی است که آن را بر می‌نماید:

شماریم:

- همه گره‌های شبکه دارای درجه‌ی یکسانی هستند
- سایز و اندازه همه انجمن‌ها برابر است
- شبکه کوچک است

دو ایراد اولی نشان می‌دهد که آزمون GN نمی‌تواند نماینده خوبی برای شبکه‌های دنیای واقعی باشد. زیرا که شبکه‌های واقعی از خصوصیات بارزشان توزیع درجه همگن گره‌های گراف است که از توزیع قانون- توانی پیروی می‌کند. و اینکه اندازه انجمن‌ها نیز برابر باشد خیلی صحیح نیست. به طور منصفانه‌ای می‌شود گفت که اندازه انجمن‌های گراف را نیز می‌توان با توزیع قانون- توانی تخمین زد. و ایراد سوم نیز خیلی روشن است زیرا بسیاری از الگوریتم ممکن است این آزمایش سربلند بیرون بیایند اما برای اندازه‌های بزرگتر شاید این اتفاق نیافتدند.



۳-۵ یک نمونه از آزمون لانیچینتی با ۵۰۰ گره [۵]

از همین رو آقای لانیچینتی^۱ و فورتوناتو در سال ۲۰۰۸ متد جدیدی را ارائه کردند [۵]. برای تولید یک شبکه توسط این متد پنج پارامتر به طور ضروری باید مشخص شوند. $(N, k, \gamma, \beta, \mu)$ پارامترهای این روش جدید هستند. در این روش N تعداد گره‌های گراف را نشان میدهد، k متوسط درجه‌ی هر راس را ارائه می‌کند، γ نمای توزیع درجه‌های گراف است، β نمای توزیع سایز انجمنهاست و توزیعی که هردوی آنها از آن پیروی می‌کنند، توزیع قانون-توانی است و درنهایت μ نیز پارامتر آمیزش^۲ است که بیانگر این است که یک گره، $(\mu - 1)$ درصد از لینکهایش به گره‌های داخل همان انجمنی که متعلق به آن است، متصل است و μ درصد لینکهایش به باقی شبکه متصل است. شبکه‌ها در چهار سایز مختلف یعنی $N = 128, 300, 500, 1000$ تولید شدند، در میان شبکه‌های تولیدی ما فقط پارامتر آمیزش را متغیر قرار دادیم

^۱ Lancichinetti

^۲ Mixing

و باقی پارامترها ثابت باقی ماندند. که μ با تصاعد 10^0 تغییر می‌کند تا شبکه‌هایی با توپولوژی‌های مختلف داشته باشیم.

۳-۲-۵ - زبان مدل کردن گراف

ما برای تبدیل مجموعه داده‌های انتخابی به ورودی‌های مناسب، یک پارسرا ساختیم تا بتواند زبان مدل کردن گراف^۱ (*GML*) را درک و تبدیل کند. زبان مدل کردن گراف یک قالب فایل بر پایه کدهای آسکی^۲ است که یک گراف را توصیف می‌کند. همچنین به زبان فرا گراف نیز شناخته می‌شود. یک مثال ساده از قالب *GML* در شکل ۴-۵ آمده است.

در واقع تعداد برنامه‌های متفاوتی که با گراف کار می‌کنند بسیار زیاد است. و هر کدام از آنها ممکن است از قالب فایل خود استفاده کنند. در نتیجه تبادل داده میان برنامه‌های مختلف غیرممکن خواهد شد. وظایف ساده‌ای مثل تبادل داده، تولید نتایج به صورت خارجی. بنابراین دانشگاه پاسائو^۳ آلمان شروع به توسعه این نوع قالب فایل کرد. که توضیحات بیشتر آن در [۶۱] آمده است.

^۱ Graph Modeling Language

^۲ ASCII

^۳ Passau

```

graph [
    comment "This is a sample graph"
    directed 1
    id 42
    label "Hello, I am a graph"
    node [
        id 1
        label "node 1"
        thisIsASampleAttribute 42
    ]
    node [
        id 2
        label "node 2"
        thisIsASampleAttribute 43
    ]
    node [
        id 3
        label "node 3"
        thisIsASampleAttribute 44
    ]
    edge [
        source 1
        target 2
        label "Edge from node 1 to node 2"
    ]
    edge [
        source 2
        target 3
        label "Edge from node 2 to node 3"
    ]
    edge [
        source 3
        target 1
        label "Edge from node 3 to node 1"
    ]
]

```

۴-۵ مثالی ساده از یک گراف در قالب GML

۳-۵- معیارهای ارزیابی

برای ارزیابی نیازمند ضابطه‌هایی هستیم که بتوان براساس آن میزان دقت و کیفیت خوشبندی را محاسبه کنیم. معیارها می‌تواند داخلی باشد مثل پودمانگی، یا چگالی اتصالات داخلی گروه‌ها نسبت به اتصالات خارجی. که از پودمانگی برای هدایت الگوریتم به سمت پاسخ مطلوب استفاده شده است. از آنجاییکه که آزمایشات ما بر روی گراف‌هایی است که ساختار انجمنی آنها مشخص است، ما از معیارهای خارجی نیز استفاده کرده‌ایم. نسبت گره‌هایی که درست خوشبندی شده اند^۱ [۳۷] و اطلاعات متقابل نرمال شده^۲ [۶۲] دو معیار خارجی برای نشان دادن دقت الگوریتم هستند. $FVCC$ در واقع بیانگر میزان دقت خوشبندی است و NMI میزان شباهت انجمن‌های کشف شده را با انجمن‌های درست نشان می‌دهد. تعریف دقیق‌تر را در ادامه آمده است:

- $FVCC$ نسبت گره‌هایی که درست خوشبندی شده اند: در اولین تست‌هایی که برروی آزمون‌های نیومن و گیروان صورت گرفت، محققان از این معیار که باز توسط خود نیومن و گیروان پیشنهاد شده بود، استفاده کردند. فرمول دقیقی برای مشخص کردن این معیار نیست اما تعریف واضحی وجود دارد. یک راس به درستی خوشبندی شده است اگر در همان خوشه‌ای قرار گرفته باشد که حداقل نیمی (۵۰ درصد) از الگوهای طبیعی شبیه به خودش در آن خوشه قرار داشته باشند. اگر در جزء بندی که بدست می‌آید، خوشه‌ای موجود باشد که از ادغام دو یا سه گروه از گروه‌های طبیعی آن به وجود آمده باشد، تمامی رئوس آن خوشه این طور در نظر گرفته می‌شوند که به درستی خوشبندی نشده اند. سپس تعداد راس‌هایی که به درستی خوشبندی شده اند به کل تعداد رئوس گراف تقسیم می‌شوند تا عددی میان ۰ و ۱ بدست بیاید.

^۱ Fraction of Vertices Classified Correctly

^۲ Normalized Mutual Information

• **NMI** اطلاعات متقابل نرمال: در واقع یک معیار خارجی است که میزان شباهت دو جزء

بندی واقعی و کشف شده را تخمین می‌زند. اطلاعات تکمیلی در ارتباط با این معیار در [۶۲]

است. آمده

$$NMI = \frac{-2 \sum_{i=1}^{C_A} \sum_{j=1}^{C_B} C_{ij} \times \log\left(\frac{N \cdot C_{ij}}{C_i \cdot C_j}\right)}{\sum_{i=1}^{C_A} C_i \times \log\left(\frac{C_i}{N}\right) + \sum_{j=1}^{C_B} C_j \times \log\left(\frac{C_j}{N}\right)} \quad (1-5)$$

یک جزء بندی در واقع یک تقسیم بندی گراف به خوش‌های است. فرض کنیم که دو جزء بندی A و

B داده شده است. برای محاسبه فرمول بالا ابتدا لازم است که ماتریس درهم ریختگی C را محاسبه

کرده باشیم. هر درایه این ماتریس C_{ij} ، تعداد گره‌های مشترک انجمن i از جزء بندی A با انجمن

j از جزء بندی B دارد را نشان می‌دهد. ($C_A(C_B)$ تعداد انجمن‌ها در جزء بندی (B) را نشان

می‌دهند و $(C_i(C_j)$ جمع همه‌ی عناصر ماتریس C در ردیف i (ستون j) است. میزان NMI بین

صفر و یک است. هرچه این میزان بیشتر باشد در واقع دو جزء بندی بیشتر به هم شبیه هستند،

یعنی اگر $A = B$ باشد آنگاه $NMI = 1$ است. و بالعکس اگر دو جزء بندی کاملاً متفاوت از هم

باشند $NMI = 0$ است.

۴-۵- تنظیم پارامترها

در این بخش پارامترهای الگوریتم ارائه شده در فصل قبل را به منظور رسیدن به کارآیی و دقت بالاتر بهینه

و تنظیم می‌کنیم. جدول ۱-۵ لیست تمامی پارامترهایی که در الگوریتم وجود دارد و می‌تواند توسط کاربر

تنظیم شود به همراه معنی مختصر آنها ارائه کرده است. مقدار تمامی پارامترها به وسیله حدس و آزمایش

و خطابه دست آمده است. و نیز تاثیر برخی پارامترهای مهم بر روی نتیجه نهایی الگوریتم را نمایش

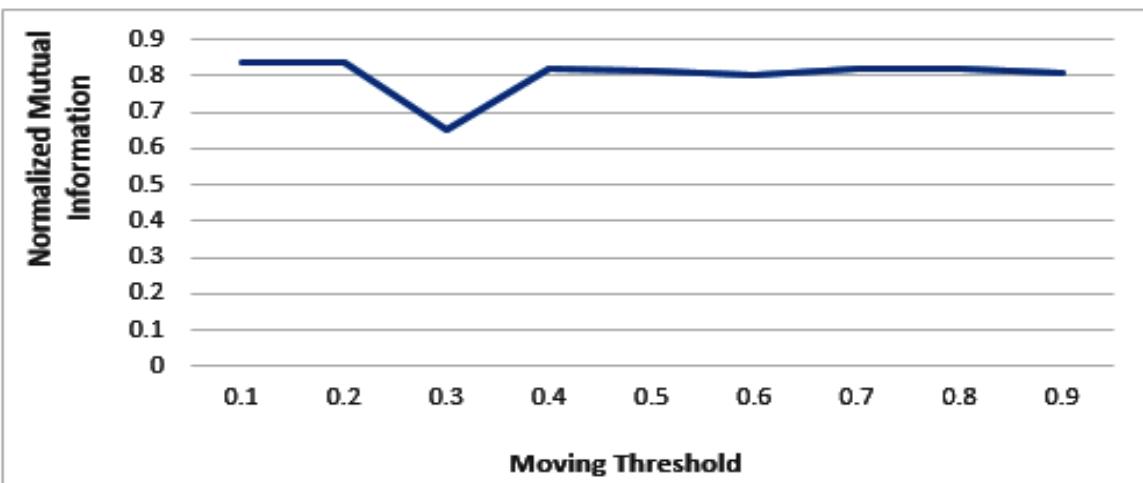
خواهیم داد.

۱-۵ جدول پارامترهای الگوریتم در یک نگاه

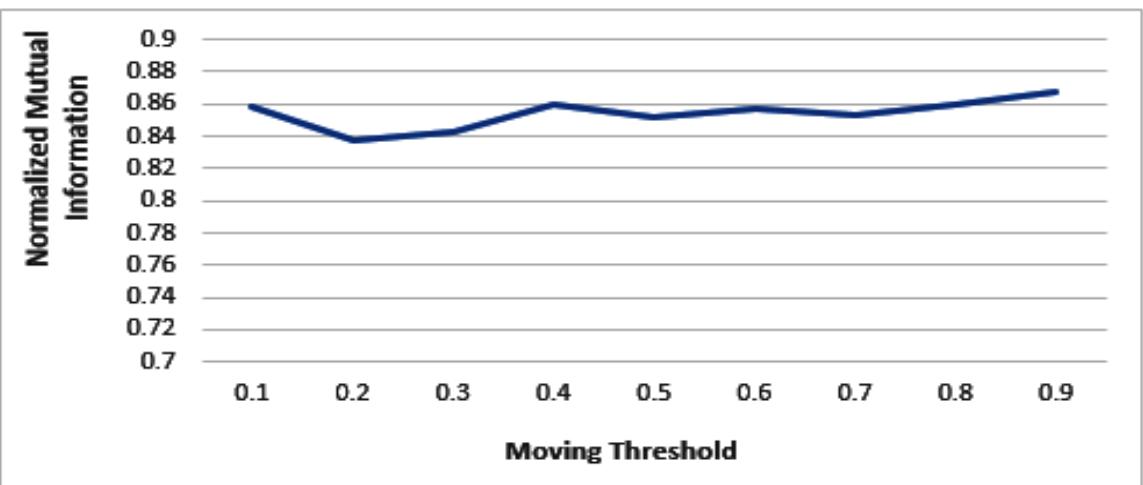
iterationNumberLimitation	حداکثر تکرارهای الگوریتم
colonySize	تعداد گرههای گراف
initialNumberOfDancers	تعداد زنبورهای رقصدنده اولیه
movingThreshold	پارامتر آستانه حرکت
ϵ	ضریب ثابتی است برای کنترل و تنظیم آستانه حرکت
C.P	پارامتر کنترل برای افزایش/کاهش تعداد زنبورهای رقصدنده
dancerCountWeight (α)	نمای توان در فرمول احتمالی برای فاکتور تعداد رقصدندهها
heuristicInformationWeight (β)	نمای توان در فرمول احتمالی برای فاکتور مکافهفه الگوریتم
horizontalVisionLimitation	محدوده دید در افق
verticalVisionLimitation	محدوده دید عمودی
dancerDecreaseCoefficient(ω)	ضریب کاهش تعداد زنبورهای رقصدنده در گذر زمان
θ	ضریب ثابتی است برای محاسبه مکافهفه در فرمول احتمال

۱-۶-۵ - بررسی تأثیر پارامتر آستانه حرکت بر الگوریتم

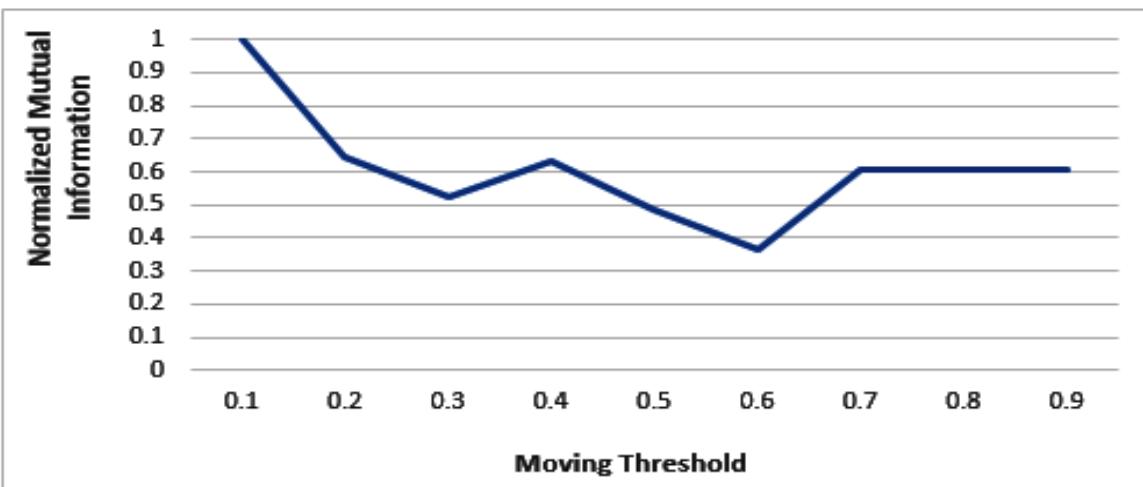
همانطور که گفتیم این پارامتر یکی از پارامترهای مهم الگوریتم است و به نوعی الگوریتم به آن حساسیت نشان می‌دهد، حساسیت هم بدان معنا نیست که اگر تغییر کوچکی در مقدار آن صورت گرفت، دقت هم به سرعت تغییر کند، بلکه بدان معناست که مقدار اولیه‌ی مناسب برای این پارامتر به میزان زیادی به بهبود دقت الگوریتم کمک خواهد کرد.



الف - اثر پارامتر آستانه حرکت در مجموعه داده کاراته



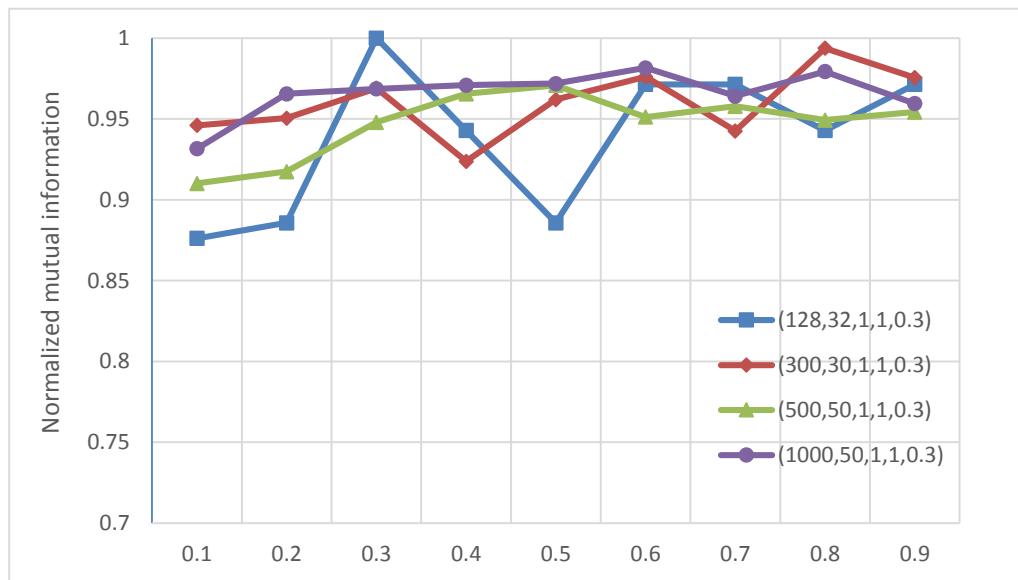
ب - اثر پارامتر آستانه حرکت در مجموعه داده فوتبال



ج - اثر پارامتر آستانه حرکت در مجموعه داده دلفین ها

۵-۵ نمایش تاثیر پارامتر آستانه حرکت در شبکه های واقعی

در واقع هدف نمودارهای بالا بررسی تاثیر پارامتر آستانه حرکت بر روی شبکه‌های مختلف با سایزهای متفاوت و پارامتر آمیزش متفاوت هست. همان طور که نمودارها نشان می‌دهد مساله ما به این پارامتر وابستگی و حساسیت دارد. از آنجاییکه که شبکه‌های مختلف دارای توپولوژی متفاوتی نسبت به یکدیگرند، بنابراین پارامتر آستانه حرکت مناسب برای هر شبکه متفاوت است. همچنین ما این تاثیر را بر روی شبکه‌های تولید شده توسط کامپیوتر نیز بر روی نمودار ترسیم کردی‌ایم.



۶-۵ اثر پارامتر آستانه حرکت بر روی شبکه‌های تولیدشده توسط کامپیوتر

بعد از انجام آزمایش و خطاهایی مقدار هریک از پارامترهای دیگر نیز بدست آمد. جدول ۲-۵ مقادیر مربوط به پارامترهای مختلف را نمایش می‌دهد.

۲-۵ پارامترهای تنظیم شده برای الگوریتم

initialNumberOfDancers	5
iteraionNumberLimitaion	100
ϵ	0.1
C.P	2
dancerCountWeight (α)	1
heuristicInformationWeight (β)	2
horizontalVision	1
verticalVision	1
dancerDecreasementCoefficient	0.6
θ	10

۵-۵- ارزیابی و مقایسه روش پیشنهادی با روش‌های دیگر

در ابتدا مقایسه خود را بر روی مجموعه داده‌های شبکه‌های واقعی نشان می‌دهیم و سپس به سراغ مجموعه داده تولید شده توسط کامپیوتر خواهیم رفت.

۱-۵-۵- مقایسه میان شبکه‌های واقعی

جداول زیر مقایسه میان روش پیشنهادی در این پایان نامه یعنی *CDHHO* را با بعضی از روش‌های موجود دیگر نمایش می‌دهد. سعی شده است از مراجعی استفاده شود که حتی الامکان از مجموعه داده مشابهی استفاده کرده باشند که در این رساله یعنی مجموعه داده باشگاه کاراته، مجموعه داده دلفین‌های دست آموز و مجموعه داده فوتبال باشگاهی آمریکا.

۳-۵ مقایسه روش‌های مختلف با *CDHHO* بر روی مجموعه داده باشگاه کاراته

Method		NMI (%)	FVCC (%)	Num. of Com.
GN	[۳۰]	57.98	97.06	5
FN	[۳۷]	69.25	97.06	3
CNM	[۳۸]	66.75	-	-
FEC	[۶۳]	69.49	97.06	3
RB	[۶۴]	97.8	-	-
Blondel	[۶۵]	98.84	-	-
FA	[۶۶]	97.85	-	-
RN	[۶۷]	98.75	-	-
MOGA-Net	[۶۸]	99.86	-	-
iMeme-Net	[۷]	100	100	2
ACC-FP	[۵۸]	95.6	100	2.3
MIGA	[۸]	100	-	2
EFA	[۵۴]	99.83	-	-
<i>CDHHO</i>		83.7	97	2

۴-۵ مقایسه روش‌های مختلف با **CDHHO** بر روی مجموعه داده دلفین‌های دست آموز

Method		NMI (%)	FVCC (%)	Num. of Com.
GN	[۳۰]	44.17	98.39	13
FN	[۳۷]	50.89	96.77	5
CNM	[۳۸]	57.38	-	-
FEC	[۶۳]	52.93	96.77	4
RB	[۶۴]	95.55	-	-
Blondel	[۶۵]	95.43	-	-
FA	[۶۶]	95.52	-	-
RN	[۶۷]	95.79	-	-
MOGA-Net	[۶۸]	95.96	-	-
iMeme-Net	[۷]	100	100	2
ACC-FP	[۵۸]	88.46	98.39	2.6
MIGA	[۸]	81	-	2
EFA	[۵۴]	95.85	-	-
CDHHO		100	100	2

الگوریتم **CDHHO** در مورد مجموعه داده کاراته، می‌توان گفت به نسبت عملکرد خوبی نداشته است، اما برای دفاع از عملکرد آن بیشتر توضیح می‌دهیم. مجموعه داده کاراته به دو انجمن ۱۶ و ۱۸ عضوی تقسیم می‌شود، این در حالی است که الگوریتم پیشنهادی ما دو انجمن ۱۷ عضوی را تشخیص می‌دهد، تنها گره شماره ۱۰ را که با یک پیوند به انجمن شماره یک و با پیوندی دیگر به انجمن دیگر متصل است که الگوریتم آن را در انجمن مخالف تشخیص داده است. طبیعت معیار *NMI* در مجموعه داده‌های کوچک اینگونه نشان می‌دهد که با تشخیص یک گره اشتباه اندازه آن به ۸۳,۷٪ می‌رسد. این در حالی است که معیار FVCC میزان ۹۷٪ را نمایش می‌دهد یعنی تنها ۳٪ از گره‌ها را درست تشخیص نداده است.

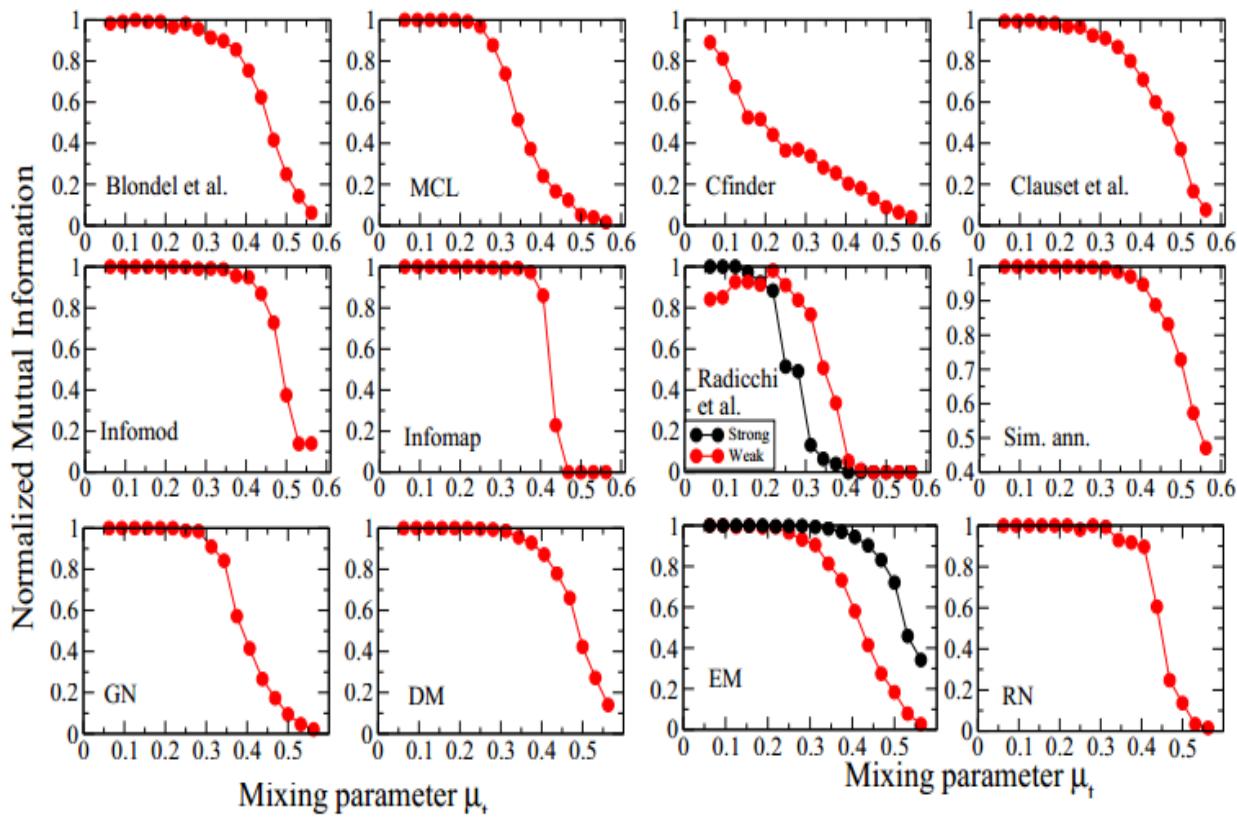
۵-۵ مقایسه روش‌های مختلف با **CDHHO** بر روی مجموعه داده فوتیال باشگاه‌های آمریکا

Method		NMI (%)	FVCC (%)	Num. of Com.
GN	[۳۰]	87.89	83.48	10
FN	[۳۷]	75.71	63.48	7
CNM	[۳۸]	73.4	-	-
FEC	[۶۳]	80.27	77.39	10
RB	[۶۴]	79.22	-	-
Blondel	[۶۵]	79.62	-	-
FA	[۶۶]	79.38	-	-
RN	[۶۷]	79.12	-	-
MOGA-Net	[۶۸]	77.8	-	-
iMeme-Net	[۷]	86.2	94.78	12
ACC-FP	[۵۸]	87.03	86.96	12
MIGA	[۸]	91	-	12
EFA	[۵۴]	79.73	-	-
CDHHO		86.69	82.95	12.1

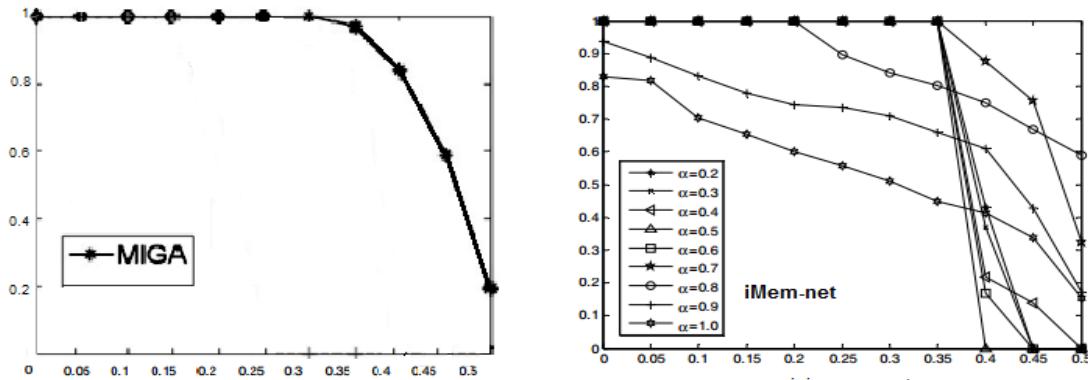
البته الگوریتم *CDHHO* در مورد مجموعه داده دلفین‌ها توانسته است به درستی همه گره‌ها را خوشه‌بندی کند و نمره کامل را بگیرد. همچنین این الگوریتم در مورد مجموعه داده فوتیال باشگاه‌های آمریکا نیز عملکرد قابل قبولی از خود ارائه داده است و در مقایسه با بیشتر الگویتمندان اختلاف دقیق خوبی را کسب کرده است.

۵-۵-۲ - مقایسه با شبکه‌های تولیده شده توسط کامپیوتر

در ابتدا تست خود را با آزمون GN آغاز می‌کنیم و نمودار بدست آمده از آزمایشات خود را در کنار دیگر نمودارها نمایش می‌دهیم تا بتوان مقایسه قابل قبولی را در این زمینه ارائه کرد.

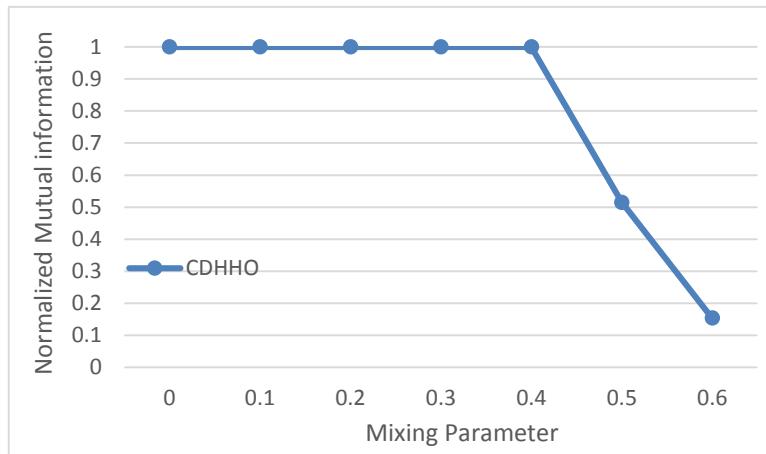


(الف)



۷-۵ نتایج آزمایشات بر روی آزمون GN. شکل الف از مرجع [۴] و شکل ب راست از [۷] و چپ [۸]

آزمون GN در واقع یک شبکه‌ی ۱۲۸ گره‌ای است که به چهار گروه ۳۲ تایی تقسیم می‌شود. متوسط درجه هر راس ۱۶ است، که با پارامتر آمیزش متفاوت مورد آزمایش قرار گرفته است. شکل ۵ نتایج دیگر الگوریتم‌ها را نمایش می‌دهد. اما شکل ۸-۵ نتیجه الگوریتم $CDHHO$ را بر روی آزمون GN نمایش می‌دهد.

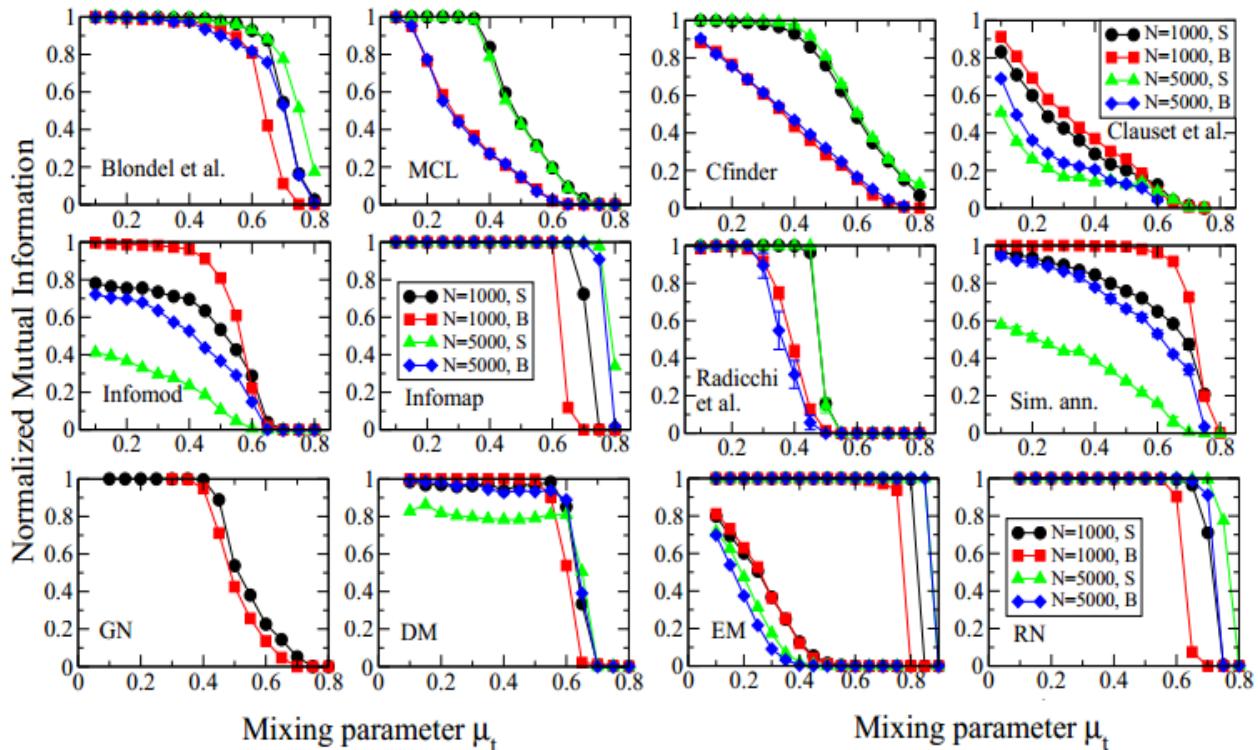


۸-۵ نتایج بدست آمده از الگوریتم پیشنهادی بر روی آزمون GN

همان طور که ملاحظه می‌شود هر چه پارامتر آمیزش به ۰,۵ نزدیک می‌شود، کار الگوریتم برای تشخیص درست انجمن‌ها سخت‌تر می‌شود. چون بیش از نیمی از پیوندها در واقع به خارج از انجمن تعلق پیدا می‌کنند.

همانگونه که در شکل مشخص است الگوریتم ما توانسته است در زمانیکه پارامتر آمیزش ۰,۵ است به دقیقه ۱۵,۴٪ بررسد و در ۰,۶ به دقیقه ۱۵,۵٪ بررسد. این در حالیست که بسیاری از الگوریتم که نشان داده شده اند در ۰,۵ دقیقه شان به صفر یا نزدیک آن می‌رسد. الگوریتم Miga که در سال ۲۰۱۳ به پیشنهاد شده است، به دقیقه ۲۰,۲٪ رسیده است و الگوریتم iMem-net که یک مقاله کنفرانسی در ۲۰۱۲ است توانسته است با تغییر یکی از پارامترهای الگوریتم که بسیار بدان وابسته است در اعداد مختلف به نتیجه خوبی دست پیدا کند. و هیچکدام از این دو نیز برای ۰,۶ گزارشی ارائه نکرده اند.

برای اینکه بتوانیم از صحت قدرت الگوریتم اطمینان کسب کنیم، آزمون ۱۲۸ گرهای GN چندان مشخص کننده‌ی خوبی نخواهد بود. بنابراین به سراغ شبکه‌های با سایز بزرگتری رفتیم که دیگران نیز با روش LFR آنها را تولید کرده‌اند و نتایج را گزارش کرده‌اند. چهار شبکه مختلف در نظر گرفته شده است.

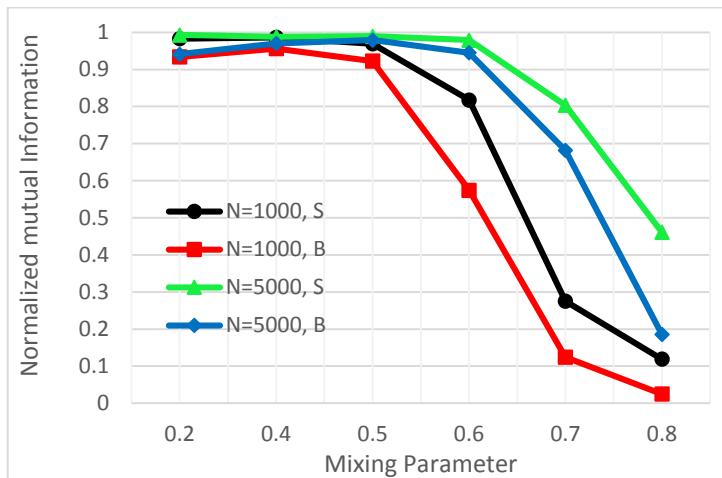


۹-۵ نتایج آزمایشات الگوریتم‌های مختلف بر روی آزمون LFR با سایز [۴] 1000S, 1000B, 5000S, 5000B

مشخصات شبکه‌ها بدین شرح است که تعداد گرهها ۱۰۰۰ و ۵۰۰۰ است. نمای توزیع درجه ۲- و نمای توزیع درجه ۱- است. اما تفاوت ^۱S و ^۲B در این است که سایز انجمان‌ها در اولی بین ۱۰ تا ۵۰ است ولی در دومی بین ۲۰ تا ۱۰۰ گره است. نتیجه آزمایش الگوریتم CDHHO را در شکل ۱۰-۵ نمایش داده‌ایم تا بتوان با نتایج سایر الگوریتم‌ها مقایسه نمود.

^۱ Small

^۲ Big



۱۰-۵ نتایج آزمایش الگوریتم CDHHO بر روی شبکه های تولید شده توسط روش LFR

از روی اشکال قابل تحلیل است که اکثر الگوریتم ها زمانیکه به پارامتر α نزدیک می شوند، به شدت کارآیی خود را از دست میدهند و دقتشان بر روی هر چهار شبکه به صفر نزدیک می شود. در بعضی الگوریتم ها نیز این اتفاق در پارامتر $\alpha = 0.7$ نیز رخ می دهد. اما الگوریتم پیشنهادی ما بر روی شبکه S5000 ۵۰۰۰S توانسته است دقت ۴۶,۱٪ را کسب کند. در ضمن بر روی شبکه با سایز ۱۰۰۰۰ نیز هیچکدام از الگوریتم ها در پارامتر آمیزش $\alpha = 0.8$ دقت بیش از صفر کسب کنند اما الگوریتم در این مقطع نیز توانسته است دقت های ۱۱٪ برای ۱۰۰۰S کسب کند.

اما می توان از لحاظ پیچیدگی زمانی نیز نگاهی به الگوریتم های بالا و مقایسه آن با الگوریتم خود انداخت، که در جایگاه خود برای این پایان نامه ارزشمند است. در واقع جدول ۶-۵ که از مرجع [۱۸] گرفته شده است، توسط فورتوناتو برای دوازده الگوریتم بالا ارائه شده است.

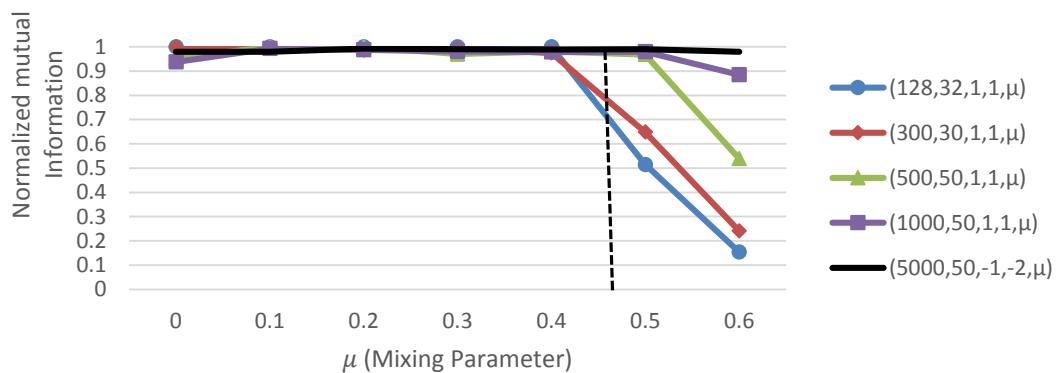
۶-۵ مقایسه پیچیدگی زمانی الگوریتم‌هایی در آزمون LFR روی چهار شبکه شرکت داشته اند.

method	Order	Ref
GN	$O(nm^2)$	[۳۰, ۶]
Clauset et al.	$O(m \log^2 n)$	[۳۸]
Blondel et al.	$O(m)$	[۶۵]
Sim. Ann.	<i>Parameter dependent</i>	[۶۹, ۴۳]
Radicchi et al.	$O(m^4/n^2)$	[۳۳]
Cfinder	$O(\exp(n))$	[۷۰]
MCL	$O(nk^2), k < n$	[۷۱]
Infomod	<i>Parameter dependent</i>	[۷۲]
Infomap	$O(m)$	[۶۴]
DM	$O(n^3)$	[۷۴, ۷۳]
EM	<i>Parameter dependent</i>	[۷۵]
RM	$O(m^\beta \log n), \beta \sim 1.3$	[۶۷]

این در حالیست که الگوریتم پیشنهادی ما از لحاظ پیچیدگی زمانی نیز برتری خوبی نسبت به خیلی از الگوریتم‌های بالا دارد. پیچیدگی الگوریتم از $O(d_{max}^2 n)$ است که $n \ll d_{max}$ است.

۶-۵- کارآیی الگوریتم در ابعاد بالا

بعد از مقایسه‌هایی که صورت گرفت. نمودار ۱۱-۵ ارائه می‌شود. در واقع آزمایش دقیق نسبت به افزایش پارامتر آمیزش برای پنج شبکه با سایز ۱۲۸ و ۳۰۰ و ۵۰۰ و ۱۰۰۰ و ۵۰۰۰ گره است.



۱۱-۵ مقایسه بین دقتهای کسب شده برای ۵ شبکه با سایز متفاوت و پارامتر آمیزش متفاوت

در تحلیل نمودار بالا باید گفت تا زمانیکه پارامتر آمیزش از ۰,۵ کمتر است، اوضاع برای سایزهای متفاوت یکسان است و دقتهای خوبی نیز کسب می‌شود. اما زمانیکه پارامتر آمیزش به ۰,۵ میرسد و از آن میگذرد یعنی نسبت پیوندهای خارج از انجمان به پیوندهای داخل انجمان پیشی می‌گیرد و تopolوژی شبکه متفاوت از قبل می‌شود، الگوریتم‌ها به سختی می‌توانند انجمان‌های درست را تشخیص بدهند. نکته‌ای که از نمودار بالا قابل برداشت است این هست که الگوریتم ما هرچه که سایز شبکه بیشتر می‌شود، انجمان‌ها را به درستی تشخیص می‌دهد به طوریکه برای شبکه با سایز ۵۰۰۰ در پارامتر آمیزش ۰,۹۷، دقیق ۰,۹۷ را کسب کرده است. که این نشان از قدرت الگوریتم در ابعاد بالا دارد.

فصل ٦ -

نتیجه گیری و پیشنهادات

۶-۱- نتیجه گیری

هدف این پایان نامه ارائه روشی برای تشخیص انجمنهای شبکه‌های اجتماعی است. تحلیل شبکه‌های اجتماعی در واقع بررسی رابطه میان افراد، کشف الگو و مجھولاتی از شبکه است. یکی از الگوهای شبکه‌های اجتماعی ساختار انجمنی است. یک انجمن به گروهی از رئوس (گره‌های گراف) گفته می‌شود که ارتباطاتش (یال‌ها یا پیوندها یا اتصالاتش) با دیگر اعضای آن گروه بیشتر از ارتباطاتش با اعضای گروه‌های دیگر باشد. این تعریفی است که در میان تعاریف زیاد و غیر واحد از انجمن میان همگان پذیرفته شده است. تحقیقات در زمینه کشف انجمن‌ها هنوز یکی از زمینه‌های مهم تحقیقاتی است. شناخت ساختار گراف شبکه‌ها مخصوصاً در شبکه‌های اجتماعی اطلاعات مفید و مهمی را از آن شبکه در اختیار می‌گذارد و در درک گره‌های مجھول کمک می‌کند. در بعضی موارد میتوان انجمن‌ها را خلاصه مفیدی از کل شبکه دانست. می‌توان دریافت که سمت و سوی علایق اشخاص و گروه‌ها به کدام سمت است تا بتوان برای دستیابی به آن برنامه ریزی کرد. حتی می‌توان پا را فراتر گذاشت و گرایشات و علایق انجمن‌ها و افراد را تغییر داد. یکی از دلایلی که هنوز تحقیقات در این زمینه ادامه دارد بحث هزینه محاسباتی الگوریتم و دقت آنها علی الخصوص در شبکه‌های بزرگ است. این مسئله در واقع همان مسئله خوشبندی در داده کاوی محسوب می‌شود که به جای خوشبندی داده‌های معمولی، باید گره‌های یک گراف خوشبندی شوند. در این پایان نامه سعی شد با استفاده از ترکیب و ادغام الگوریتم خوشبندی کلونی مورچه (که یک جستجوی محلی محسوب می‌شود) با الگوریتم بهینه‌سازی کندوی زنبور عسل (که یک بهینه‌سازی سراسری محسوب می‌شود) هم هزینه محاسباتی را کاهش دهیم و همزمان دقت الگوریتم را بالاتر ببریم. درک محیط محلی توسط مورچه با استفاده از تابع برازشی نسبتاً جدید، تصمیم گیری برای حرکت، یافتن نقاط بهتر بر اساس مکافه‌ها و تعداد زنبورهای رقصدنده اختصاص یافته به گره‌ها (مورچه‌ها)، ارزیابی کیفیت خوشبندی و نهایتاً بروزرسانی جدول رقص براساس کیفیت حرکت و کیفیت خوشبندی از ویژگی‌های الگوریتم پیشنهادی است. الگوریتم $CDHHO$ بر روی سه مجموعه داده واقعی و یک مجموعه از شبکه‌های تولید شده توسط کامپیوتر که در اندازه‌های متفاوت با ویژگی و توپولوژی‌های مختلف اعمال

شدن. آزمون‌های GN و LFR که از مهمترین آزمون‌ها در این زمینه هستند مورد استفاده قرار گرفتند. نتایج بدست آمده از الگوریتم $CDHHO$ نشان دهنده‌ی کارآیی و دقیقیت الگوریتم در کشف انجمن‌هاست. علی‌الخصوص کارآیی الگوریتم در شبکه‌های با مقیاس بزرگ نشان داده شد. این الگوریتم توانست مصالحه‌ی بسیار خوبی بین دقیقیت کشف انجمن‌ها و نیز پیچیدگی زمانی الگوریتم، که از اهداف مهم این پایان نامه محسوب می‌شود، برقرار کند. از دستاوردهای این پایان نامه می‌توان به ارسال دو مقاله، که یکی کنفرانسی و دیگری برای مجلات *ISI* است اشاره نمود:

- **Ants attack to social networks for reaching communities under leadership of Bees (*ISI Journal*)**

- کشف انجمن‌ها در شبکه‌های اجتماعی با ترکیب خوشبندی کلونی مورچه و بهینه‌سازی کندوی زنبور عسل (بیست و سومین کنفرانس مهندسی برق شریف)

۶-۲- کارهای آتی

می‌توان در ادامه‌ی راه و پیشرفت مسیر تحقیقات در این زمینه پیشنهادات زیر را مطرح کرد:

- ارائه یک روش ترکیبی^۱ با دیگر روش‌های موجود، که ضمن پایین نگه داشتن پیچیدگی محاسبات، بتوان نتایج بدست آمده از هر دو الگوریتم را ترکیب کرد و بدین صورت دقیق و ضریب اطمینان به الگوریتم را افزایش داد.

- ارائه نسخه موازی الگوریتم $CDHHO$ ، برای کاهش زمان اجرای الگوریتم و تسريع در محاسبات به خصوص در محاسبه فواصل میان گره‌ها و همسایگان مشترک
- ارائه روشی برای تنظیم و بهینه‌سازی خودکار پارامترهای مسئله
- بررسی قابل استفاده بودن این روش برای دیگر شبکه‌های پیچیده که الگوهایی نظیر شبکه‌های اجتماعی را دارند، مانند شبکه‌های بیولوژیکی، شبکه‌های وب، شبکه‌های موبایل و ...

^۱ Hybrid

- ارائه یک روش برای محاسبه مقدار اولیه پارامتر آستانه حرکت مناسب برای هر مجموعه داده
- در طول تحقیقات نیز، نیاز به یک معیار که کیفیت داخلی شناسایی انجمان‌ها و خوشبندی در گراف را نمایش دهد و همزمان این معیار در تناظر با معیارهای خارجی نظیر اطلاعات متقابل نرمال شده باشد، حس شد. هرچند پویمانگی به عنوان یک معیار محبوب در این زمینه هست، اما محدودیت‌هایی نیز دارد.
- سعی در استفاده و تغییر و سازگار کردن این روش برای استفاده در شبکه‌ها و گراف‌های وزن دار

فهرست مراجع

1. Fortunato, S. and C. Castellano, *Community Structure in Graphs*. 2007.
2. Mohammad Saniee Abadeh, Z.J.A., *Evolutionary Algorithms and Biological Computing*. 2013: Niaz Danesh Insurance, ISBN 978-600-6481-44-9.
3. Clauset, A., C.R. Shalizi, and M.E. Newman, *Power-law distributions in empirical data*. SIAM review, 2009. **51**(4): p. 661-703.
4. Lancichinetti, A. and S. Fortunato, *Community detection algorithms: A comparative analysis*. Physical Review E, 2009. **80**(5): p. 056117.
5. Lancichinetti, A., S. Fortunato, and F. Radicchi, *Benchmark graphs for testing community detection algorithms*. Physical Review E, 2008. **78**(4): p. 046110.
6. Newman, M.E.J. and M. Girvan, *Finding and evaluating community structure in networks*. Physical Review E, 2004. **69**(2): p. 026113.
7. Maoguo, G., et al. *An improved memetic algorithm for community detection in complex networks*. in *Evolutionary Computation (CEC), 2012 IEEE Congress on*. 2012.
8. Shang, R., et al., *Community detection based on modularity and an improved genetic algorithm*. Physica A: Statistical Mechanics and its Applications, 2013. **392**(5): p. 1215-1231.
9. Fayyad, U.M., et al., *Advances in Knowledge Discovery and Data Mining*. National Conference on Artificial Intelligence. 1996.
10. Mohammad Saniee Abadeh, S.M., Mohadese Taherpour, *Practical Data Mining*. first ed. 2012: Niaz Danesh.
11. Frisch, K.v., *The Dance Language and Orientation of Bees*. 1967, Harvard University Press. Cambridge.
12. Abadeh, M.S., *A Hybrid Evolutionary Fuzzy System for Intrusion Detection in Computer Networks*. 2008, Unpublished doctoral dissertation, Sharif University of Technology.
13. Travers, J. and S. Milgram, *An Experimental Study of the Small World Problem*. Sociometry, 1969. **32**(4): p. 425--443.
14. Leskovec, J. and E. Horvitz, *Planetary-scale views on a large instant-messaging network*, in *Proceedings of the 17th international conference on World Wide Web*. 2008, ACM: Beijing, China. p. 915-924.
15. Mancoridis, S., et al., *Using Automatic Clustering to Produce High-Level System Organizations of Source Code*, in *Proceedings of the 6th International Workshop on Program Comprehension*. 1998, IEEE Computer Society. p. 45.
16. Luce, R.D., *Connectivity and generalized cliques in sociometric group structure*. Psychometrika, 1950. **15**(2): p. 169-190.

17. Mokken, R., *Cliques, clubs and clans*. Quality and Quantity, 1979. **13**(2): p. 161-173.
18. Fortunato, S., *Community detection in graphs*. Physics Reports, 2010. **486**(3–5): p. 75-174.
19. Wasserman, S. and K. Faust, *Social network analysis : methods and applications*. 1994, Cambridge; New York: Cambridge University Press.
20. Lorrain, F. and H.C. White, *Structural equivalence of individuals in social networks*. The Journal of Mathematical Sociology, 1971. **1**(1): p. 49-80.
21. Fouss, F., et al., *Random-Walk Computation of Similarities between Nodes of a Graph with Application to Collaborative Recommendation*. IEEE Trans. on Knowl. and Data Eng., 2007. **19**(3): p. 355-369.
22. Moreno, J.L. and H.H. Jennings, *Who shall survive?: A new approach to the problem of human interrelations*. 1934: Nervous and mental disease publishing co.
23. Freeman, L.C., *The Development of Social Network Analysis: A Study in the Sociology of Science*. 2004: Empirical Press, Vancouver.
24. Nettleton, D.F., *Data mining of social networks represented as graphs*. Computer Science Review, 2013. **7**(0): p. 1-34.
25. Kernighan, B.W. and S. Lin, *An Efficient Heuristic Procedure for Partitioning Graphs*. Bell System Technical Journal, 1970. **49**(2): p. 291-307.
26. Suaris, P.R. and G. Kedem, *An algorithm for quadrisection and its application to standard cell placement*. Circuits and Systems, IEEE Transactions on, 1988. **35**(3): p. 294-303.
27. L.R. Ford, D.R.F., *A simple algorithm for finding maximal network flows and an application to the Hitchcock problem*. CANADIAN JOURNAL OF MATHEMATICS, 1957: p. 210-218.
28. Flake, G.W., et al., *Self-organization and identification of web communities*. Computer, 2002. **35**(3): p. 66-70.
29. Newman, M.E.J., *Detecting community structure in networks*. The European Physical Journal B - Condensed Matter and Complex Systems, 2004. **38**(2): p. 321-330.
30. Girvan, M. and M.E.J. Newman, *Community structure in social and biological networks*. Proceedings of the National Academy of Sciences, 2002. **99**(12): p. 7821-7826.
31. Zhou, T., J.-G. Liu, and B.-H. Wang, *Notes on the Algorithm for Calculating Betweenness*. Chinese Physics Letters, 2006. **23**(8): p. 2327.
32. Brandes, U., *A Faster Algorithm for Betweenness Centrality*. Journal of Mathematical Sociology, 2001. **25**: p. 163--177.
33. Radicchi, F., et al., *Defining and identifying communities in networks*. Proceedings of the National Academy of Sciences of the United States of America, 2004. **101**(9): p. 2658-2663.
34. Latora, V. and M. Marchiori, *Efficient Behavior of Small-World Networks*. Physical Review Letters, 2001. **87**(19): p. 198701.

35. Fortunato, S., V. Latora, and M. Marchiori, *Method to find community structures based on information centrality*. Physical Review E, 2004. **70**(5): p. 056104.
36. Brandes, U., et al., *On modularity-np-completeness and beyond*. 2006.
37. Newman, M.E.J., *Fast algorithm for detecting community structure in networks*. Physical Review E, 2004. **69**(6): p. 066133.
38. Clauset, A., M.E.J. Newman, and C. Moore, *Finding community structure in very large networks*. Physical Review E, 2004. **70**(6): p. 066111.
39. Danon, L., A. Díaz-Guilera, and A. Arenas, *The effect of size heterogeneity on community identification in complex networks*. Journal of Statistical Mechanics: Theory and Experiment, 2006. **2006**(11): p. P11010.
40. Schuetz, P. and A. Caflisch, *Multistep greedy algorithm identifies community structure in real-world and computer-generated networks*. Physical Review E (Statistical, Nonlinear, and Soft Matter Physics), 2008. **78**(2): p. 026112.
41. Du, H., et al., *An algorithm for detecting community structure of social networks based on prior knowledge and modularity*. Complexity, 2007. **12**(3): p. 53-60.
42. Kirkpatrick, S., C.D. Gelatt, and M.P. Vecchi, *Optimization by simulated annealing*. science, 1983. **220**(4598): p. 671-680.
43. Guimera, R. and L.A.N. Amaral, *Functional cartography of complex metabolic networks*. Nature, 2005. **433**(7028): p. 895-900.
44. Boettcher, S. and A.G. Percus, *Optimization with extremal dynamics*. Phys. Rev. Lett., 2001. **86**: p. 5211–5214.
45. Duch, J. and A. Arenas, *Community detection in complex networks using extremal optimization*. Physical Review E, 2005. **72**(2): p. 027104.
46. Newman, M.E.J., *Modularity and community structure in networks*. Proceedings of the National Academy of Sciences, 2006. **103**(23): p. 8577-8582.
47. Newman, M.E.J., *Finding community structure in networks using the eigenvectors of matrices*. Physical Review E, 2006. **74**(3): p. 036104.
48. Pizzuti, C., *GA-Net: A Genetic Algorithm for Community Detection in Social Networks*, in *Parallel Problem Solving from Nature – PPSN X*, G. Rudolph, et al., Editors. 2008, Springer Berlin Heidelberg. p. 1081-1090.
49. Sadi, S., S. Etaner-Uyar, and S. Gündüz-Öğüdücü, *Community Detection Using Ant Colony Optimization Techniques*. 15th International Conference on Soft Computing, MENDEL 2009, 2009: p. 206-213.
50. Sadi, S. and A.S. Uyar. *An efficient community detection method using parallel clique-finding ants*. in *Evolutionary Computation (CEC), 2010 IEEE Congress on*. 2010.
51. Yan, L., et al. *Finding Closely Communicating Community Based on Ant Colony Clustering Model*. in *Artificial Intelligence and Computational Intelligence (AICI), 2010 International Conference on*. 2010.

52. Zhang, N. and Z. Wang. *Community mining in dynamic social networks--Clustering based improved ant colony algorithm*. in *Computer Science & Education (ICCSE), 2011 6th International Conference on*. 2011.
53. Jin, D., et al., *Ant Colony Optimization with Markov Random Walk for Community Detection in Graphs*, in *Advances in Knowledge Discovery and Data Mining*, J. Huang, L. Cao, and J. Srivastava, Editors. 2011, Springer Berlin Heidelberg. p. 123-134.
54. Amiri, B., et al., *Community Detection in Complex Networks: Multi-objective Enhanced Firefly Algorithm*. Know.-Based Syst., 2013. **46**: p. 1-11.
55. Ling, C., X. Xiao-Hua, and C. Yi-Xin. *An adaptive ant colony clustering algorithm*. in *Machine Learning and Cybernetics, 2004. Proceedings of 2004 International Conference on*. 2004.
56. Xu, X.H. and L. Chen, *An adaptive ant clustering algorithm*. Journal of Software, 2006. **17**: p. 1884–1889.
57. Burt, R.S., *Positions in Networks*. Social Forces, 1976. **55**: p. 93-122.
58. Ji, J, *Ant colony clustering with fitness perception and pheromone diffusion for community detection in complex networks*. Physica A: Statistical Mechanics and its Applications, 2013. **392**(15): p. 3260-3272.
59. Zachary, W.W., *An information flow model for conflict and fission in small groups*. Journal of anthropological research, 1977: p. 452--473.
60. Lusseau, D., et al., *The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations*. Behavioral Ecology and Sociobiology, 2003. **54**(4): p. 396-405.
61. *Graph Modeling Language*. Available from: <http://www.fim.uni-passau.de/index.php?id=17297&L=1>.
62. Danon, L., et al., *Comparing community structure identification*. Journal of Statistical Mechanics: Theory and Experiment, 2005. **2005**(9): p. P09008-09008.
63. Yang, B., W. Cheung, and J. Liu, *Community Mining from Signed Social Networks*. IEEE Trans. on Knowl. and Data Eng., 2007. **19**(10): p. 1333-1348.
64. Rosvall, M. and C.T. Bergstrom, *Maps of random walks on complex networks reveal community structure*. Proceedings of the National Academy of Sciences, 2008. **105**(4): p. 1118-1123.
65. Blondel, V.D., et al., *Fast unfolding of communities in large networks*. Journal of Statistical Mechanics: Theory and Experiment, 2008. **2008**(10): p. P10008.
66. Yang, X.-S., *Firefly Algorithms for Multimodal Optimization*, in *Stochastic Algorithms: Foundations and Applications*, O. Watanabe and T. Zeugmann, Editors. 2009, Springer Berlin Heidelberg. p. 169-178.
67. Ronhovde, P. and Z. Nussinov, *Multiresolution community detection for megascale networks by information-based replica correlations*. Physical Review E, 2009. **80**(1): p. 016109.
68. Pizzuti, C. *A Multi-objective Genetic Algorithm for Community Detection in Networks*. in *Tools with Artificial Intelligence, 2009. ICTAI '09. 21st International Conference on*. 2009.

69. Guimera, R., M. Sales-Pardo, and L.A.N. Amaral, *Modularity from fluctuations in random graphs and complex networks*. Physical Review E, 2004. **70**(2): p. 025101.
70. Palla, G., et al., *Uncovering the overlapping community structure of complex networks in nature and society*. Nature, 2005. **435**(7043): p. 814-818.
71. van Dongen, S.M., *Graph Clustering by Flow Simulation*. 2000, University of Utrecht, The Netherlands.
72. Rosvall, M. and C.T. Bergstrom, *An information-theoretic framework for resolving community structure in complex networks*. Proceedings of the National Academy of Sciences, 2007. **104**(18): p. 7327-7331.
73. Donetti, L. and M.A. Munoz, *Detecting network communities: a new systematic and efficient algorithm*. J. Stat. Mech., 2004. **P10012**.
74. Donetti, L. and M.A. Munoz, *Improved spectral algorithm for the detection of network communities*. Modeling Cooperative Behavior in the Social Sciences, 2005. **779**: p. 104-107.
75. Newman, M.E. and E.A. Leicht, *Mixture models and exploratory analysis in networks*. Proceedings of the National Academy of Sciences, 2007. **104**(23): p. 9564-9569.

واژه نامه‌ی فارسی به انگلیسی

Connectivity	اتصال
Edge Connectivity	اتصال لبه‌ای
Small-world effect	اثر جهان کوچک
Swarm	ازدحام
ASCII	اسکی
Normalized Mutual Information	اطلاعات متقابل نرمال شده
External	اکسترمال
Pattern	الگو
Exploitation	انتفاع
Moving Threshold	آستانه حرکت
Mixing	آمیزش
Supervised	با ناظارت
Unsupervised	بدون ناظارت
Big	بزرگ
Well-separated	به خوبی جدا شده
Honeybee Hive Optimization	بهینه‌سازی کندوی زنبور عسل
Boettcher	بوچر
Dissimilarity	بی شباهتی
Chaotic	بی نظمی
Max-flow min-cut	بیشینه-جریان کمینه برش
Pareto	پارتو
Passau	پاسائو
Percus	پرکوس
Dynamics	پویایی
Exploration	پویش
Pizutti	پیزووتی
Prediction	پیش‌بینی

Fitness Function	تابع برازش
Tango	タンゴ
Simulated annealing	تبیرید شبیه سازی شده
Social Network Analysis	تحلیل شبکه های اجتماعی
Density	تراکمی
Complete mutuality	تقابل کامل
Divisive	تقسیمی
Partitioning	تقسیمی
Snowball	توب برفی
Power-law distribution	توزيع قانون توانی
Scale-free Distribution	توزيع مقیاس آزاد
GeorgSimmel	جرج سیمل
Current-flow	جريان فعلی
Informed search	جستجوی آگاهانه
Gregarious	جمعيت دوست
Jacob Moreno	جيکوب مورنو
Jin	جين
Cohesion	چسبندگی
Non-deterministic polynomial	چندجمله ای غیرقطعی
Privacy Preservation	حفظ حریم شخصی
Wisdom	خرد
Summarization	خلاصه سازی
Clustering	خوشبندی
Ant Colony Clustering	خوشبندی کلونی مورچه
Duch	داج
Data	داده
Streaming Data	داده های جریانی
Knowledge	دانش
Six Degrees of Separation	درجه شش جدایی
Classification	دسته بندی
Dendrogram	دندروگرام

Danon	دنون
Binomial	دو جمله‌ای
Rewired	دوباره سیم بندی کردن
Cycle	دور
Fitness Landscape	دورنمای برازش
Radicchi	رادیکچی
Approximation Solution	راه حل تخمینی
Waggle Dance	رقص پیچشی
Global Optimization approach	روش‌های بهینه‌سازی سراسری
Local Search approach	روش‌های جستجوی محلی
Zachary	زاخاری
Dance Language	زبان رقص
Graph Modeling Language	زبان مدل کردن گراف
Ziphan or Pareto	زیفان یا پارتو
Zhang	ژانگ
Community Structure	ساختار انجمنی
Sadi	садی
Dance Floor	سالن رقص
Cut size	سایز برش
Sociogram	سوسيوگرام
Self-organized System	سيستم خود سازمان ده
Grid	شبکه
Lattice	شبکه
Karate Club Network	شبکه باشگاه کاراته
Bottlenose Dolphin Network	شبکه دلفین‌های دست آموز
American College football Network	شبکه فوتبال دانشگاهی آمریکا
Social Networks	شبکه‌های اجتماعی
Information Networks	شبکه‌های اطلاعاتی
Biological Netowrks	شبکه‌های بیولوژیکی
Complex Networks	شبکه‌های پیچیده
Instant Messaging	شبکه‌های پیغام دهی و گفت و گو

Computer generated network	شبکه‌های تولیدشده توسط کامپیوتر (مصنوعی)
Real-world network	شبکه‌های واقعی
Foraging	شهدیابی
Schuetz	شوئتر
The edg eclustering coefficient	ضریب خوشبندی یالی
Spectral	طیفی
Emergence	ظهور
Non deterministic	غیر قطعی
Metaheuristic	فرامکاشفه‌ای
Pheromone	فرومون
Flake	فلیک
Understanding	فهم
Fortunato	فورتوناتو
Ford and Fulkerson	فورد و فالکرسون
Reachability	قابلیت رسیدن
Power Law	قانون توانی
Cafisch	کافلیچ
Kerninghan-Lin	کرنینگهان-لین
Clauset	کلاست
Clique	کلیکو
Geodesic	کوتاهترین مسیر بین دو گره در گراف
Small	کوچک
max-heap	کوه-بیشینه
Data Quality	کیفیت داده
Random Walk	گام تصادفی
Random Graphs	گراف‌های تصادفی
Girvan	گیروان
Lancichinetti	لاچینتی
Lambda	لمبدا
Lanczos	لنکزوس
Lorrain and White	لورین و وايت

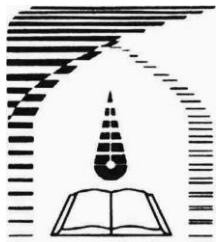
Lusseau	لوسیو
Liu	لیو
Betweenness	مابینی
Module	ماژول
Modularity	پودمانگی
Data Ownership	مالکیت داده
Center-based	مبتنی بر مرکز
Contiguous	مجاورتی
Null model	مدل پوج
Ant Sleeping Model	مدل خواب مورچه
Edge Centrality	مرکزیت یال
Tradeoff	مصالحه
Structural Equivalence	معادل ساختاری
Conceptual	مفهومی
Positions	موقعیت‌های اجتماعی
Inhomogeneity	ناهمگونی
Fraction of Vertices Classified Correctly	نسبت گره‌هایی که درست خوشبندی شده اند
No Free Lunch Theorem	نظریه ناهار غیر مجانی
Actor	نقش
Role	نقش
Profile	نمایه
Newman	نیومن
Overlap	همپوشان
Adjacency	همسایگی
Homogeneous	همگن
Swarm Intelligence	هوش تجمعی
Whatsup	واتس آپ
Wigand	ویگاند

Abstract

Social networks are kind of social structure that include number of vertices. This vertices can be individuals or organizations. These are connected together with one or more type of dependencies such as friendships, relatives, business relationships, scientific relationships. As the internet and web are extend and also great grows in using smart phones in recent years, social networks become one of the imitable elements of our lives at least in the virtual aspect. Social network analysis is the study of relationships between individuals including study of social structures, social positions, role analysis and other items. One of the important social structures is community. Group of vertices that number of internal links are more than external links is called community.

The problem of community detection has been attracted many scientists in last decade. Good solutions proposed for it, but this problem are not solved satisfiably yet. In this Thesis was tried to use combination of ant colony clustering idea and honey bee hive optimization. Ant colony clustering which is local search solution was guiding by honey bee hive optimization which is a global approach. Also one model proposed for assigning dancer bees. These proposed methods have caused to detect communities more accurately and faster. Actually dancer bees have been used for exchanging information among nodes, in fact node is the same ant in ant colony clustering. Experimental results on real-world networks and artificial graphs generated by computers show performance of algorithm. Because of the proposed method can make good tradeoff between accuracy and time complexity of algorithm. Also algorithm can obtain 100% accuracy in dolphin's dataset. Also it can obtain better accuracy than others in GN and Lancichinetti benchmark when mixing parameter passes from 0.5. Then in this Thesis has been shown that proposed method can obtain better result as the datasets become larger.

Keywords: *Social Networks, community detection, ant colony clustering, honeybee hive optimization, graph mining*



Tarbiat Modares University
Faculty of Electrical and Computer Engineering

Thesis for the Degree of Master of Science (M.Sc.) in computer Engineering,
software engineering

Thesis Title

**Community detection in social networks with honey bee hive
optimization**

By:

Alireza Ghasabeh

Supervisor:

Dr. Mohammad Saniee Abadeh

Advisor:

Dr. Mahdi Abadi

February 2015